UTRECHT UNIVERSITY

MASTER THESIS

Is Generative AI Mature Enough for Maturity Models? Insights from a Comparative Analysis

Author:
Mischa VAN EK

1st Supervisor:
Dr. Slinger JANSEN

2nd Supervisor:
Drs. Nico BRAND

A thesis submitted in fulfillment of the requirements
for the degree of Master in Science

in the

Business Informatics
Department of Information and Computing Sciences

August 18, 2024

"If you don't know where you are, a map won't help."

Watts S. Humphrey

UTRECHT UNIVERSITY

Abstract

Faculty Name
Department of Information and Computing Sciences

Master in Science

Is Generative AI Mature Enough for Maturity Models? Insights from a Comparative Analysis

by Mischa VAN EK

Maturity models (MMs) serve as a basis to understand the improvement of quality. As an assessment tool, current capabilities are able to be recognized. With this, paths to higher levels, that yield better outcome, are made available for users. However, these MMs face challenges. Three of these challenges are considered in this study. The first challenge is market fluctuations, where MMs become outdated. Second, is the difficulty of finding an appropriate MM (assuming that an appropriate model even exists). Last, the creation process of an MM is, in general, significantly time and effort consuming. With the advent of generative-AI (GenAI), there seems to be potential in solving these problems. Since, in just an instant, GenAI can form an MM. This MM includes all the latest information known and is personalized, based on the prompt that has been given. This research sets out to discover the potential role that GenAI could play in the life cycle of an MM. To ground this, a literature review and comparative analysis were done. 17 interviews were conducted, where two selected human-created MMs were evaluated in contrast to two AI-generated variants of these models. All the models were compared in terms of quality. This study gives reasons to believe that AI-generated MM are on the same level, or even better, than human created ones. Additionally, evidence is shown that GenAI has a plurality of potential roles in the life cycle of an MM.

Acknowledgements

Every journey ends. I am here at the end to write down who helped me to reach the end of my journey. Firstly, I would like to thank my two supervisors. Whenever I felt lost, they helped me to get a sense of the best direction to take. Next, I would like to thank all the participants. At the beginning I was worried about getting enough responses. However, with the help of some interviewees, who put me in touch with additional participants, this worry quickly faded. I am absolutely grateful, as this research would have been much paler without the participation of all the interviewees. I would also like to thank Watts S. Humphrey for inspiring me. When I started my research, my excitement was largely driven by my interest in Generative AI. However, as I read, watched and learned (from the material that Watts S. Humphrey left in his legacy) my excitement quickly shifted to Maturity Models. This excitement led me to think more critically about the implications of my research. Finally, I would like to thank the artists of the music I listened to while writing this thesis. Without their music, I would not have been able to concentrate on completing this thesis. As the writing of my acknowledgements comes to an end, so does my journey. I hope that the result of my journey, which can be read in the following pages, will light the fires of inspiration for those who are here to embark on their own journey.

Contents

List of Figures

List of Tables

List of Abbreviations

AI	**A**rtificial **I**ntelligence
CMM	**C**apability **M**aturity **M**odel
CMMI	**C**apability **M**aturity **M**odel **I**ntegration
DevOps	**Dev**elopment **Op**eration**s**
DoD	**D**epartment **o**f **D**efence
FAMM	**F**ocus **A**rea **M**aturity **M**odel
GenAI	**Gen**erative **A**rtifical **I**ntelligence
GPT	**G**enerative **P**retrained **T**ransformer
LLM	**L**arge **L**anguage **M**odel
MM	**M**aturity **M**odel
SEI	**S**oftware **E**ngineering **I**nstitute
SPM	**S**oftware **P**roduct **M**anagement
SPO	**S**oftware **P**roduct **O**rganization
SRQ	**S**ub **R**esearch **Q**uestion
UU	**U**niversity **U**trecht

Chapter 1

Introduction

Disclaimer: This thesis used chatGPT as a generative AI tool to gain insight and inspiration. Most of the tables and graphs were generated using ChatGPT. However, the final words were written entirely by the author. For a few grammar and language mistakes, DeepL was used. The AI tools were used ethically to ensure that original ideas were expressed with academic integrity.

1.1 Background

Maturity models (MM) are among the most widely used frameworks, in software quality, for assessing how well a process is managed. The most notable is the Capability Maturity Model (CMM) (Paulk et al., 1991) , which pioneered and served as the basis for many other MMs (Wendler, 2012). New MMs have been created to cover different domains (mainly software development (Adekunle et al., 2022)). Critics of MMs have questioned whether an MM has any empirical validation (Bach, 1994). Despite this misconceived criticism, it applies to many of the MMs developed later (Adekunle et al., 2022). For development and validation, there is a call to action among MM developers who need insight into how good an MM is (Hillegersberg, 2019) (Lasrado, Vatrapu, and Andersen, 2015). To date, MMs have typically been created by:

- Gathering information on the subject matter of a domain;
- Assessing, with the help of expert advice, which maturity level the key practices are in;
- Observing, through a case study, what an MM user might think.

Once an MM has been created, it needs to be maintained. Since, the ever-changing information landscape can render previous models obsolete (Normann Andersen et al., 2020). MM development has already proven to be a time-consuming activity. Given that, new MMs are often created by a student who spends their entire Master or PhD thesis developing them (Sanchez-Puchol and Pastor-Collado, 2017).

Recent technological developments, in Generative Artificial Intelligence (GenAI), have led to the possibility of accelerating the process of developing an MM. The increased interest in GenAI is mainly due to the tool 'ChatGPT', which can serve as an easily accessible support tool to increase productivity (Brynjolfsson, Li, and Raymond, 2023). The main purpose of this paper is to evaluate whether the development of an MM can be partially taken over by a GenAI. A comparative analysis will be conducted. Professionals, who have experience with MMs, will evaluate both an MM developed by a human and an MM generated by an AI. The evaluation will be based on design criteria. The MMs are both on the same topic and the participant is *not* told which MM was created by a human or an AI to avoid a biased view.

1.2 Problem Statement

> "if you don't continue to think about it and work at it, it will degrade. The number one reason is that the world changes. Competition changes. You can't stand still."
>
> *Bill Curtis*

The process of developing and applying MMs, in assessing organizational or technological maturity, is time-consuming and resource-intensive. This presents a significant challenge in rapidly evolving fields, where current MMs may not fully capture emerging trends or domains. Letting AI generate MMs, that align with the topic where maturity needs to be evaluated, would speed up applying the model. However, it is not certain whether replacing the traditional MM, with an AI-generated MM, would perform nearly as well or even better. A major drawback of sticking to traditional MMs is that it limits the user to the pre-built models that already exist. There is an exception (although outdated), a tool which could find the right model for a user. To reach the right model, the tool let the user go through a series of questions (Van Looy et al., 2013). However, the tool was only created for one domain: business process management.

Overall, current problems with the use of MMs include:

- **Difficulty of choice:**
  There is a huge amount of MMs created. Each of these MMs has its own topic or situation where it would be most appropriate to use it. Finding the right MM, for a specific situation, can therefore be a challenge. There are all-encompassing MMs (e.g. CMMI) that are designed to cover most, if not all, of the required maturity measurements within an organisation. However, the use of these models comes with the burden of having to generalise. In some cases a more specialised model, designed for a specific topic or situation, might be more beneficial. However, the problem with that is the difficulty of finding an existing appropriate model.
- **Outdated MMs:**
  Once a suitable MM has been found and selected, the question arises as to whether the MM is still up to date with the rapid market changes. MMs face the burden of accurately projecting what the current best practices (based on maturity) of their topics are. Creators of MMs, who do not maintain their MM every year, face the risk of their created MM quickly becoming outdated.
- **Development effort to create an MM:**
  Despite the plethora of MMs that have been created, there is not an MM for every topic. Typically, the creation of a new MM starts with the identification of a need or opportunity. The development cycle for an MM then begins. It often takes considerable time and effort to complete the cycle. Once an MM is completed, it would also need to be maintained. Where the maintenance is yet another burden of time and effort.

With the recent breakthrough in GenAI, there is now the ability to produce text that could be formatted in the form of an MM. The MM can also be about any topic the user wants (Achiam et al., 2023). This ability is possibly a huge opportunity for MMs, by breaking the current challenge(s).

1.3 Research Objective

The aim of the paper is to understand what role GenAI could possibly have in MMs. In an attempt to understand this, the MM community was asked: ”What role do you envision GenAI could have in MMs?”. Asking for what they envision, is meant to uncover potential roles. Besides that, a potential role of GenAI, which could solve the problems stated in the previous section, was thought of by the author. The potential role would be to let AI generate an MM. An AI-generated MM is able to be personalized on the situation, which would contain current day market practices (based on the GenAI training set). It could also alleviate the burden of a long development time. To understand the potential, a comparison between an AI-generated MM and a human created MM was done. If the comparison shows that the AI-generated MM performs equal or significantly better, then we have reasons to believe that there are roles for GenAI in MM development. The comparison, of how AI-generated MMs currently fair against human created MMs, has not been done yet. Therefore, this is a research objective within the present paper.

1.4 Research Questions

AI has now the capability to completely take over (or assist in) the process of creating an MM. However, it is unknown whether AI-generated/assisted MMs can perform equal or better than human created MMs. To understand this, the following main research question (MRQ) was formulated:

MRQ: What role can Generative AI play in the life cycle of maturity models?

The main research question aims to assess two things. First, it assesses whether GenAI is suitable for developing MMs. Second, it assesses whether there are other possible applications for using GenAI for MMs. In order to answer ther MRQ, these three sub-research questions (SRQs) were formulated:

SRQ 1: What are the design criteria for evaluating a maturity model?

SRQ 2: How do AI-generated generated maturity models compare to traditional human created models in terms of the design criteria?

SRQ 3: What expectations does the maturity model community envision for GenAI in the usage and development of maturity models?

SQ2 was formulated to figure out, whether AI even has potential. Because, if an AI-generated MM can perform (at least) equal to a human created MM, then GenAI will have potential for the MM life cycle. To evaluate MMs, design criteria are needed. To figure out those criteria, SQ1 was formulated. With the answer to these questions SQ3 can be worked on. In order to figure out the specific role(s) GenAI could have. With the answers to all of the SRQs, the answer to the MRQ can be given.

1.5 Thesis Outline

This paper is divided into seven chapters. The first chapter gives a brief overview of the background related to this study. Furthermore, the problem statement, research objective and research questions are stated. Chapter 2 reviews the relevant literature, in order to set the context of the research. In Chapter 3 the research method is discussed. Here a breakdown of the used type of research is given. Additionally, the procedure which the research followed is shown. Chapter 4 describes the design of an MM life cycle, the framework of a CustomGPT and how an AI-generated FAMM was created with the CustomGPT. Chapter 5 presents and further analyses the results of the interviews. Chapter 7 reflects on the limitations and validity of the research. Finally, 8 answers the research questions stated in sub-chapter 1.4.

Chapter 2

Literature Review

This chapter describes all the relevant literature within the scope of the research. The first section gives a brief overview of why MMs were created. The second section examines how MMs has been adapted over time. The third section explains what GenAI is and how it might be used.

2.1 Origins of Maturity Models

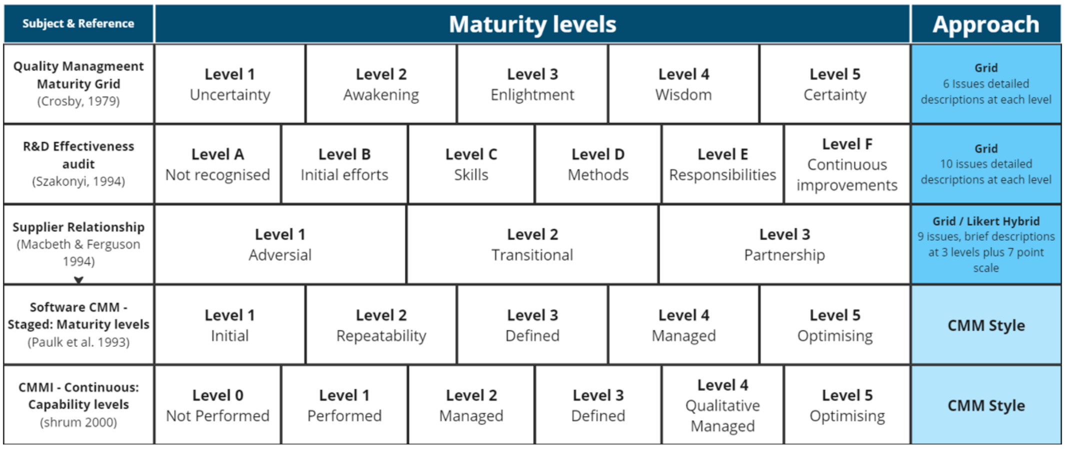
For the research, it is essential to understand the use and development of MMs. The following section provides a brief overview of why MMs were created and how their use has changed over time.

2.1.1 CMM - Where it started

The origins of MMs started in the late 1980s, with one MM in particular that served as a springboard for MM recognition and development. This MM was called the Capability Maturity Model (CMM). The reason for the creation, of the CMM, can be traced back to concerns in the 1980s about the quality and delivery of software for military projects. The US recognised the need for a method to systematically assess the capabilities of its software contractors. Large, complex projects were underway that often suffered from delays, budget overruns and quality problems. In response, the Department of Defence planned to fund the establishment of the Software Engineering Institute (SEI) at Carnegie Mellon University in 1984 (Defense and Druffel, 1982). This research was led by Watts Humphrey, considered the ′father of software quality′. In 1988 professor Humphrey wrote a paper entitled ′Software Process: A Maturity Framework", which described the five levels of process maturity(Humphrey, 1988). The paper also drew on previous empirical research, showing that higher maturity yields favourable results (Humphrey and Kitson, 1987).

A year later, Watts Humphrey extended that work. Humphrey further integrated the ′process maturity framework′ in his book ′Managing the Software Process′ (Humphrey, 1989). After four years of experience, using the methods developed by the SEI, the CMM was developed to recommend best practices in key process areas. The purpose was to improve software development (Paulk et al., 1991). In the study, Bill Curtis et al. included expected improvement goals (such as time, cost, etc.), as the process becomes more mature. The aforementioned CMM follows a progressive five leveled structure. From first to last these levels are: Initial, Repeatable, Defined, Managed and Optimise.

The word maturity itself refers to the progression of aging. While the CMM ages, it would pass through intermediate stages. Each stage serves as a stepping stone for the next stage in a progressive manner. The stages represent maturity levels. Variations in the typologies of maturity levels exist and have been applied to different models, as shown in Figure 2.1 (Fraser, Moultrie, and Gregory, 2002). Nevertheless, the CMM has been one of the most widely used frameworks for developing new MMs over the years (Wendler, 2012).



Subject & Reference	Maturity levels						Approach
Quality Managmeent Maturity Grid (Crosby, 1979)	Level 1 Uncertainty	Level 2 Awakening	Level 3 Enlightment	Level 4 Wisdom	Level 5 Certainty		Grid 6 Issues detailed descriptions at each level
R&D Effectiveness audit (Szakonyi, 1994)	Level A Not recognised	Level B Initial efforts	Level C Skills	Level D Methods	Level E Responsibilities	Level F Continuous improvements	Grid 10 issues detailed descriptions at each level
Supplier Relationship (Macbeth & Ferguson 1994)	Level 1 Adversial	Level 2 Transitional	Level 3 Partnership				Grid / Likert Hybrid 9 issues, brief descriptions at 3 levels plus 7 point scale
Software CMM - Staged: Maturity levels (Paulk et al. 1993)	Level 1 Initial	Level 2 Repeatability	Level 3 Defined	Level 4 Managed	Level 5 Optimising		CMM Style
CMMI - Continuous: Capability levels (shrum 2000)	Level 0 Not Performed	Level 1 Performed	Level 2 Managed	Level 3 Defined	Level 4 Qualitative Managed	Level 5 Optimising	CMM Style

FIGURE 2.1: Evolution and variation of maturity levels: Different names and approaches

2.1.2 CMMI - Expansion of the foundation

Based on the first CMM new models, that follow the CMM style, have emerged. Examples of these model are the System Engineering CMM (Bate et al., 1995), Capability Maturity Model for Software (Paulk et al., 1991) or the Integrated Product Development CMM ((ACQUISITION and DC, 1996). However, the emergence of these new models created confusion and inefficiency within the user community. It was especially the case for organisations that had to comply with multiple models at the same time. As a result of the confusion, Capability Maturity model Integration (CMMI) was developed in 2000. The CMMI integrates the CMM styled models stated above into a single comprehensive framework (Team, 2002). In figure 2.2, the framework is shown with the associated characteristics per level. The CMMI, which replaced the CMM, aimed to be more flexible and adaptable to improve performance in different domains, not just software development. Unlike the CMM, which has a ’staged representation’, CMMI also offers ’continuous representation’ in a later updated version (Chrissis, Konrad, and Shrum, 2011). The continuous representation allows processes, in certain areas, to improve and mature over time. This also allows to get to a higher maturity level in a specific area. Instead of needing to go through predefined sets of process areas, which CMM uses with its staged representation. A manual has been created for CMMI, based on the industry’s accumulated experience, to set specific and general goals for each process area.

As knowledge, on how to manage the software process, expanded and changed throughout time so did the CMMI. New versions have been created to reflect modern practices in the software development industry. Notable changes included version 1.3 of CMMI in 2010 (Team et al., 2010), which incorporated agile methodology into the enhancements to high maturity practices. At the time of writing, the latest version of CMMI, V3.0, was released in 2023 (CMMI Institute, 2023). In the new version, the main focus is on integrating DevOps and cybersecurity practices into the model. The CMMI is not intended to replace modern methodology. Instead it is intended as a complementary framework for organisations to identify, within process

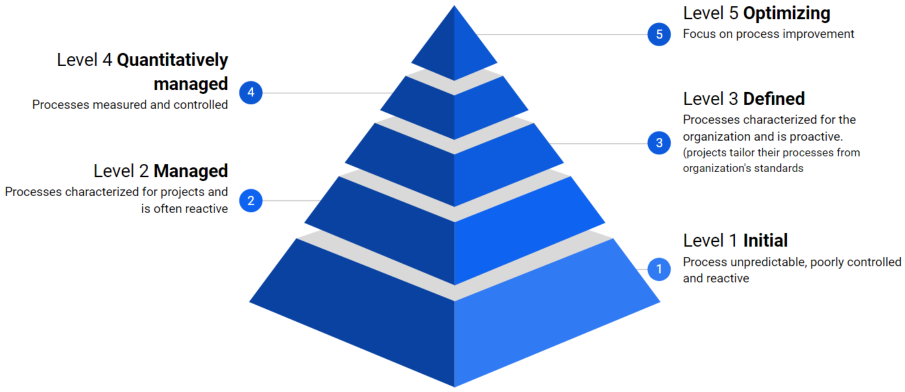

Level 5 Optimizing
Focus on process improvement
5
Level 4 Quantitatively managed
Processes measured and controlled
4
Level 3 Defined
Processes characterized for the organization and is proactive. (projects tailor their processes from organization's standards
3
Level 2 Managed
Processes characterized for projects and is often reactive
2
Level 1 Initial
Process unpredictable, poorly controlled and reactive
1

FIGURE 2.2: Characteristics of Capability Maturity Model Integration

areas, gaps in their capabilities. Once these gaps have been identified, the organisation can decide whether time and resources should be devoted to filling them. So that a process matures and therefore works better (Goldenson and Gibson, 2003).

2.2 Ecosystem of Maturity Models

The research required the selection of an MM. In this sub chapter, the first step was taken in reviewing the literature on the variety of MMs. From the variety of methodologies, one approach was selected, the Focus Area Maturity Model (FAMM), which was further described below.

2.2.1 New maturity models - Creation on top of the foundation

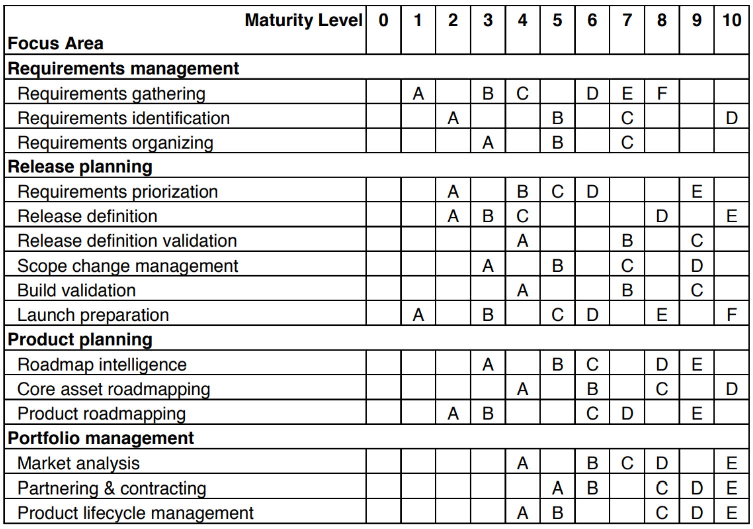
A growing number of MMs have been created. Those models attempted to capture the maturity of specific domains. It can also be said that there is a growth trend in the publication of new articles over the years (Wendler, 2012). Within the research on MMs, most articles show that the CMM(I) style is included for model application and model validation. Only a small proportion follows a model development methodology directly. The majority follows their own/other methodologies. Roy Wendler's paper "The maturity of MM research: A systematic study" is recommended and recognised by the MM community as the paper to read for a better understanding of MM research (Wendler, 2012). In his paper, Roy Wendler describes how the application of new MMs is mainly related to software development and engineering. However, new MMs also show overlap within the public sector and project management.

A more detailed overview of MM development can be found in Lester et al (Lasrado, Vatrapu, and Andersen, 2015), who conducted a literature review. Lester et al. claims that the growth in MM development, by organizations, has been driven by the certification culture. In terms of the methodology for developing models, a structure was created around four dimensions: conceptual, qualitative, quantitative and practical. The paper also included design factors such as representation, construction and evaluation for the design and development of MMs. Just as Wendlers pointed out (Wendler, 2012), Lester et al. highlights the importance of empirical validation for MMs.

2.2.2 Focus area maturity models - Further transformation of the foundation

In 2010, a new framework was developed for modelling an MM. The framework was created to provide more flexibility for developing MMs. This was an alternative approach to capturing maturity (Steenbergen et al., 2010). It is called the 'Focus Area Maturity Model' (FAMM), which allows different areas to be mature, while still recognising the interdependencies between components. The framework allows practices to be placed on a maturity scale, typically from 1 to 10, to indicate how mature key practices are considered to be. The approach itself has been validated by consulting experts, such as product managers, for their views on the applicability of the model (Steenbergen et al., 2013). The FAMM is a framework that, as a branch of maturity research, has created its own ecosystem of MMs. The weaknesses of the CMM for developing MMs are mitigated by using the FAMM framework (Steenbergen et al., 2013).

The timelines of FAMM publications shows that only a few per year get published. Most of the new FAMMS, that were created and published, either originate from a master thesis or from a journal (Sanchez-Puchol and Pastor-Collado, 2017). In a comparative analysis of areas of interest, Sanchez-Puchol et al. found that most publications in FAMM research were of Dutch origin (Sanchez-Puchol and Pastor-Collado, 2017). This finding can be attributed to the influence of the original framework of the FAMM. Since, that framework was created by Dutch researchers. A notable FAMM that was created, was the Software Product Management Maturity Model (SPM-MM) (Weerd, Bekkers, and Brinkkemper, 2010) (Shown in figure 2.3. The SPM-MM combines the capabilities of Software Product Management into the focus areas: Requirements Management, Release Planning, Product Roadmapping and Portfolio Management. Each capability is then treated as independent, with specific improvement actions linked and placed on a numerical maturity scale.



Maturity Level Focus Area	0	1	2	3	4	5	6	7	8	9	10
Requirements management											
Requirements gathering		A		B	C		D	E	F		
Requirements identification			A			B		C			D
Requirements organizing				A		B		C			
Release planning											
Requirements priorization			A		B	C	D			E	
Release definition			A	B	C				D		E
Release definition validation					A			B		C	
Scope change management				A		B		C		D	
Build validation					A			B		C	
Launch preparation		A		B		C	D		E		F
Product planning											
Roadmap intelligence				A		B	C		D	E	
Core asset roadmapping					A		B		C		D
Product roadmapping			A	B			C	D		E	
Portfolio management											
Market analysis					A		B	C	D		E
Partnering & contracting						A	B		C	D	E
Product lifecycle management					A	B			C	D	E

FIGURE 2.3: Focus area Maturity model - Software product management (Weerd, Bekkers, and Brinkkemper, 2010)

2.2.3 Maturity development - Creation of maturity models

A framework for developing MMs has already been explored. De Bruin et al. evaluated several creations of MMs and derived phases of development from them (De

Bruin et al., 2005). Other researchers followed and created their own framework for developing an MM (Mettler and Rohner, 2009) (Maier, Moultrie, and Clarkson, 2009). For the FAMM, a design, for the development of a FAMM, was created. That design looked at previously created frameworks. From that they deduced that creating an MM has these four phases (Steenbergen et al., 2010).

1. Scoping
2. Designing a model
3. Developing a tool
4. Implementation

The creation of maturity starts from 1 and goes through all the phases till 4. These phases were further extended to include individual steps to create a FAMM. Once the FAMM has been created, it will be distributed electronically. Added to that, the industry will be able to gain insight into how it has worked as an assessment tool through case studies. In turn it can provide new insights to iteratively improve the FAMM. The final step in the development of a FAMM is the communication of the results, which can take the form of a publication or presentation of the model at a conference (Steenbergen et al., 2010).

2.2.4 Maturity panic - Looking at the pile of maturity models

Many MMs have been created, but when analysed, not many follow the same methodological development path. The representation of the model structure often varies. On top of that, many MMs have not been validated with the same rigor. Poppelbußet al. argue that MMs can be considered scientifically undetermined because of the vagueness around the logic of maturity (Becker et al., 2010).

There is also a barrier to the use of MMs. First, choosing the right MM is a difficult exercise. A paper from Amy van Looy et al. developed a solution for this, by the use of a questionnaire which has 14 questions. Once the questionnaire is filled in, an answer will be given. The answer recommends three 'Business Process Maturity Models' (BPMM) from a pool of 69 BPMM's (Van Looy et al., 2013). It is a useful tool to reduce the difficulty, of choosing the right model, for an industry expert. However, it is difficult to maintain, since new MMs are created over time. Other difficulties were researched in a survey that sought to answer why MMs were not used in small organizations. The first difficulty was that the participants, of the survey, said that applying an MM was too expensive. The rationale was that applying an MM would take a significant amount of time, which the organization would not be able to afford. The second difficulty given, was that small businesses in general do not want to adopt an MM (Staples et al., 2007). The reason is that MMs are mainly created for the software development industry (Wendler, 2012). Thus organizations whichever fall out of this domain, were deemed to be irrelevant to adopt such an MM.

Although new MMs are being created and frameworks for their development are available, Jos van Hillersberg argues that maturity modelling, in general, has not yet reached a high level of maturity (Hillegersberg, 2019). The article calls into question how mature the creation of an MM is. Despite existing frameworks, the constructs of MMs are varied, which gives an unclear sense of the quality of an MM. Few MMs have been empirically validated. If we look at the validation stage for FAMMs, it is often based on data from expert reviews and case studies. In those reviews, the validation of an MM is done only by a subjective examination. (Lasrado, Vatrapu, and Andersen, 2015).

2.3 Generative AI

For this thesis, GenAI was used. Before we go into detail how GenAI was used, it is first important to understand more about GenAI. The basics of GenAI is explained below.

2.3.1 Large Language Model basics - Generate text on whatever topic asked about

A Large Language Model (LLM) is a form of GenAI that can generate new textual data. Interest in GenAI, both academically and publicly, has skyrocketed since the launch of ChatGPT (an LLM that allows a user to chat with an intelligent chatbot) in 2022 (Zhao et al., 2023). ChatGPT used the GPT3 model. GPT stands for Generative Pre-Trained Transformer. Transformer refers to the transformer architecture used from a google paper called 'Attention is all you need' (Vaswani et al., 2017). A revolutionary paper that introduced an algorithm (shown in figure 2.4), which significantly reduces model training time on large amounts of Tokens. The term tokens stand for broken up words into smaller units of text. For example, the sentence: "What are Maturity models?" could be broken up into tokens as: "Wha", "t ar", "e Matu", etc. The algorithm is quite complex and will therefore not be further explained in this literature review. OpenAI, the research lab for ChatGPT, applied the Transformers architecture and trained on a very large corpus of textual data. The ability to make sense of the data is achieved by using parameters, in order to find the relevant patterns. The number of parameters has increased with each new GPT release. For comparison, GPT1 started with 117 million parameters and GPT4, openAI's latest model at the time of writing (currently in 2024), has been trained on 1.7 trillion tokens (Zhao et al., 2023). The innovation that OpenAI introduced with ChatGPT was, making it easy for a user to interact with an LLM. A user can communicate with the model by giving it a prompt, to then afterwards get AI-generated information in return. A prompt can be any form of text which the AI uses on a previously trained model. The model predicts the next words, often to answer a question posed by the person's prompt.

2.3.2 Influence - Improvements and caution of generated text

New companies entered the fray to market their own LLM (Zhao et al., 2023). As new LLMs were released to the public, it was important to understand which LLM performed best. Benchmarks were introduced that asked questions to the LLM. The answers on these questions measured how many answers were given correctly. Companies wanted their model to perform best compared to other LLMs on the market. As one of the competitors, Google claims that their most recent LLM is the first model to outperform human experts on the benchmark questions (Gemini Team, 2023). Despite the achievement of training a model that outperforms human experts, we should be careful with letting an LLM alone make our decisions. Because, an LLM can generate a fabricated answer and present it as fact. In fields such as law and medicine, the consequences of using false information can be enormous (Zhang and Kamel Boulos, 2023). As a result, GenAI is still not used in fields that

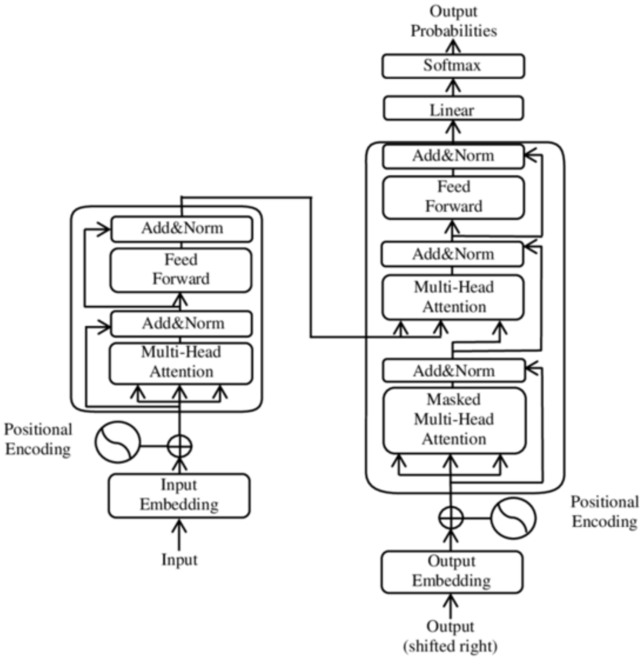


FIGURE 2.4: Transformer algorithm: a type of neural network architecture

require error-free information, even though the decision making is generally better than that of a human expert.

OpenAI has claimed that chatGPT is not intended to replace humans, but rather a tool to increase productivity. A study conducted for call centres using GenAI tools shows that productivity does indeed increase when the tool is used (Achiam et al., 2023).

2.3.3 Prompt engineering - Getting the right generated text

With a users prompt there are two ways to get the best and desired response in an LLM. First, choosing an LLM that performs best on the benchmarks. Second, the user can influence the LLM through a method called prompt engineering. Prompt engineering is done by the user to help the GenAI better interpret the information in providing a response. The user does this by structuring their prompt. An example, of prompt engineering, is asking the model to assume a persona (White et al., 2023). Subsequent responses will then follow a style that is more in line with the user's preferences and therefore provide better answers. Another example of prompt engineering is showing what 'the response', of the LLM, should be. Then the model knows what structural output it can follow.

The GPT Store was created, by OpenAI, to provide a place where users can share and find custom versions of GPT (OpenAI, 2024). These custom versions use the same model, but have applied prompt engineering techniques to provide answers that fit a particular domain. Effective prompt engineering has been made much easier with the GPT Marketplace. With the GPT Marketplace, OpenAI provides a GPT builder that can create custom GPT's. These are configured through instructions, within a chat, on how the custom GPT should behave. Additional configurations are also possible when building a GPT. Such as, assigning skills or uploading files as additional knowledge resources.

Nevertheless, it remains difficult to control the output of an LLM. Each time a GPT is asked to produce an output, a different computational path is followed. If the user asks an LLM the same question again, it may give a new answer instead of the previous answer to the same question (Achiam et al., 2023).

2.3.4 Generative applications - Not only text but much more

Text is not the only form of AI-generated output. Images, video and audio are also generated by AI (Zhang and Kamel Boulos, 2023). Models such as OpenAI's DALLe offer the possibility of a text-to-image prompt. The prompt then generates an image based on the user's input (Ramesh et al., 2021). Similarly, in audio, models are being created that can imitate human voices for a particular genre of music (Dhariwal et al., 2020).

These AI-generated capabilities are being integrated into various applications. These applications raise the question of how the user experience is affected by the use of AI-generated content. In a study of a GenAI storytelling application, the immersion of a good user experience may be broken (Antony and Huang, 2023). This was because the user perceives the content to be AI-generated. More sophisticated reasoning has also been tested, namely for business process modelling. Out of this study came the conclusion that getting AI to generate a business process model, from a work instruction document, has not been successful yet (Santos, 2023). They concluded that because of the current limited reasoning capabilities of GenAI.

Despite the limitations of GenAI, there have been positive results from letting it take over tasks. GenAI has shown over the years that it can reason better when:

- more parameters are added;
- algorithms are improved;
- quality of input data is taken into account.

As the models improve, difficult tasks, which are not possible with current models, may be exploited or completely replaced by GenAI in the near future (Zhao et al., 2023).

Chapter 3

Research Methods

In this chapter, the research method is explained. The first section 3.1 starts with describing which specific type of research methodology was used. Following that, section 3.2 gives an explanation of the requirements and procedure for conducting the interviews. Lastly, section 3.3 addresses the part of the research which had to be conducted through a literature review. The main research question is further broken down into SRQs. Table 3.1 shows each method on how the SRQs were answered.

	Grounded Theory	Qualitative surveys	Literature review
SQ1			X
SQ2		X	
SQ3	X	X	X

TABLE 3.1: A breakdown of each method used to answer the SRQ's

The primary objective of this research is to assess whether GenAI could have a role within the life cycle of an MM. Hence, the following research question was formulated: "What role can Generative AI have in the life cycle of maturity models?".

Hevner's 'Information System Research Framework' was used and filled in (as shown in figure 3.1 (Hevner, Chatterjee, and Iivari, 2010)) to understand the research context. Hevner's framework is based on design science research (Peffers et al., 2007). The left side of the figure represents the relevance cycle, while the right side of the figure represents the rigour cycle. In the centre of the model is the design cycle (Hevner, 2007).

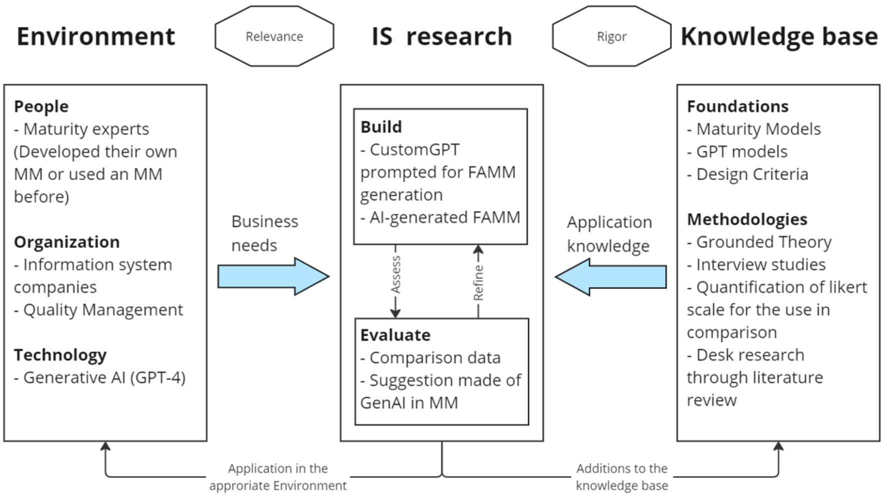

Environment
Relevance
IS research
Rigor
Knowledge base
People
- Maturity experts (Developed their own MM or used an MM before)
Organization
- Information system companies
- Quality Management
Technology
- Generative AI (GPT-4)
Business needs
Build
- CustomGPT prompted for FAMM generation
- AI-generated FAMM
Assess
Refine
Evaluate
- Comparison data
- Suggestion made of GenAI in MM
Application knowledge
Foundations
- Maturity Models
- GPT models
- Design Criteria
Methodologies
- Grounded Theory
- Interview studies
- Quantification of likert scale for the use in comparison
- Desk research through literature review
Application in the approriate Environment
Additions to the knowledge base

FIGURE 3.1: Information Systems research framework Hevner, 2007, as applied to this research.

3.1 Mixed Methods

To understand whether GenAI can play a role in the life cycle, a mixed methods approach should be used. The methods used are a quantitative method (comparing averages among the collected ratings) and two qualitative method (grounded theory and descriptive approach). The quantitative method is needed to measure the difference in performance between a GenAI MM vs a human developed MM. Whereas the qualitative methods are needed to provide a broad understanding. They provide insight in the role(s) GenAI can play in the development and application of MMs. Gathering only quantitative data would leave the rationale for the outcomes unexplained. On the contrary, if research relied only on qualitative methods, the position of both development methods would remain subjectively vague.

When the methods are taken together, Quality can be measured. This is needed to understand how an AI-generated FAMM performs, in comparison to a human created FAMM. To measure the quality of an MM design criteria are needed. To select those design criteria a literature review was conducted. These methods are combined by measuring the AI-generated & human created MMs performance. The performance is based on the participants (who have previous affiliations with MMs) quality evaluation of the MMs .

The mixed method has the disadvantage that only a small sample of participants can be approached for the data collection. Since, it takes a lot of time to conduct an interview that covers both the quantitative and qualitative sides. In total, 17 interviews were conducted. Of the 17 interviews, only 1 interview did not include an evaluation. No additional factors of the participants were considered within the study. For example, whether the occupation of the participants might influence the results.

3.2 Interview

This section describes how the interviews were prepared and conducted. First, it is explained which participants are qualified for an interview. Secondly, a selection was made of the FAMMs to be used for comparison. Each existing FAMM was collected and filtered for structural composition, relevance and preference. Thirdly, the design criteria used to evaluate the FAMMs were selected. Last, is a brief description of how an interview was structured.

3.2.1 Selecting Experts

The research design for this study, needs to have an environment that has organization and relevant people. In the Hevner model (in figure 3.1) that can be seen. To ensure that the interview creates relevant insights, purposive sampling is used for selecting participants. For purposive sampling, a set of requirements were formulated to establish the eligibility of a participant for an interview. These requirements, such as the participant is already acquainted with an MM, will minimize risks. For the selection of participants, demographic as age, gender or specific experience with an MM will not be taken into account. The only requirements were:

1 - The participants must have had previous experience of developing an MM. Or have used an MM in an organisation or for research.

2 - The participants should be available for at least one hour [1].

Participants were identified by contacting previous authors of MMs, searching people who had affiliations with MMs and networking (e.g. asking for participation in an online MM-community). Requests were sent when a participant was considered eligible. If the contacted participant agreed, an appointment was arranged, either face-to-face or online.

3.2.2 Selection of MM comparison

An MM made by a human will be compared with one that is AI-generated. Along with that, the FAMM framework was chosen. Structurally, FAMM's have shown to differ in aspects such as capabilities and size of focus area, as do whether it contains assessment questions or a max stage level descriptor.

In order to properly compare different FAMMs, the FAMMs must follow a similar structure (or set of rules). The purpose is to choose a specific structure and input this structure into instructions for GenAI. Consequently, with these instructions, the AI will be able to understand what structure the generated FAMM is supposed to have. The GenAI instructions are geared towards creating an AI variant of the selected FAMMs. If the AI variant that is generated does not follow the same structure, then there would not have been a proper comparison.

To enable a proper comparison, specific structures within FAMMs need to be first made clear. To do this, all the existing FAMMs (to date) were gathered and analysed. Google scholar was used to find out which FAMMs had been created. A total of 31 FAMMs were found and listed in table 3.2. In the analysis of all these FAMMs, five different structural compositions were found. By structural composition is meant the arrangement and organisation of the elements within a FAMM, including the set of rules it follows. The author named these five compositions and each FAMM can have one or more of these compositions. The compositions are explained in the following paragraphs:

[1] Interviews conducted often lasted less than one hour. There was also one interview that lasted only half an hour. It was decided that the results of the aforementioned interview were still going to be included

Level difference:

This composition is a bit special, because almost each FAMM that was analyzed has this structure. The level difference structure entails that a focus area has maturity levels expressed in numbers. To reach the next level, levels have to be passed progressively (to reach level 2, you have to have reached level 1. To reach level 3, you have to have reached level 2 and 1, etc.). (Steenbergen et al., 2010). Only a few exceptions (for example, with the IT governance FAMM (Smits and Hillegersberg, 2015). In this FAMM no numbers were used and the reader has to count the amount of maturity levels) do it differently. The pool of collected FAMMs showed that the lowest amount of maturity levels found were 4 and the highest 20. An MM can have varying amount of levels depending on its design, as there is no universal standard dictating the exact number of levels required. Figure 3.2 shows that the example has 4 maturity levels and that more could also be added.

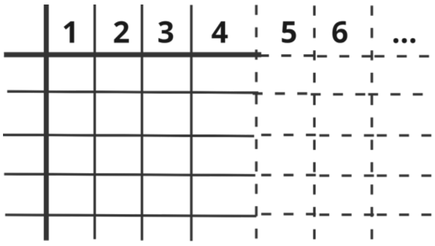


FIGURE 3.2: Focus area architecture: Flexibility in Defining the amount of maturity levels

Capability Maturity Model (CMM) Style:

The CMM style, as the name suggest, is derived from the CMM. The CMM style has a structure of capabilities with five levels (or less) within each row. In general the first level has no capability assigned to it (An example of an exception is in the IT governance MM where the first CMM level starts at the first capability in a row). Each of these levels have a meaning. The meaning of these level are the following (Team et al., 2010):

Level 1: Initial

Level 2: Repeatable

Level 3: Standardized

Level 4: Managed

Level 5: Optimized

Each row starts with the initial as the first level and the highest level achievable is optimized (an example is shown in figure 3.3). One can only achieve the next CMM level, by also having fulfilled the requirements of the CMM level(s) prior (To reach level 5, you need to have reached level 4, 3, 2 and 1). When a FAMM adopts these characteristics (No more than 5 CMM levels, always start at the initial stage & go through the levels in a progressive pattern), the FAMM has a CMM style structural composition.

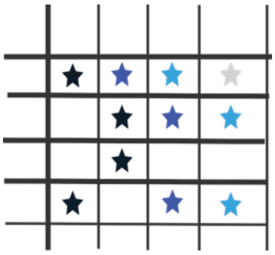


FIGURE 3.3: CMM Style (example: black = initial | Dark blue = Repeatable | light blue = Managed, ...

Focus area segmentation:

This composition involves breaking down a broad focus area into smaller, more detailed, sub topics. Sub topics are in literature also called focus areas. However, in this paper they will be referred to as sub focus areas. Figure 3.4 shows an example of a segmentation structure. In the example, '1.' is a focus area that is broken down into two sub focus areas (1.1 & 1.2). Only the sub focus areas could have capabilities in them.

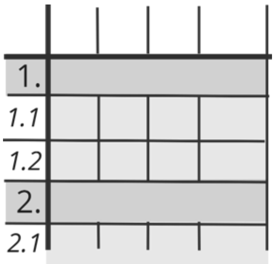


FIGURE 3.4: Focus area segmentation (example: 1. Focus area | 1.1 Sub focus area | 1.2 Sub focus area, ...

Interlinked capabilities:

Capabilities in a focus area are meant to be completed in a sequential manner starting from the lowest to the highest maturity level. In FAMMs with this structure, there are capabilities that are interlinked to each other across focus areas. This means that when reaching a capability, certain other capabilities of another focus area first needs to be completed. This is shown in figure 3.5.

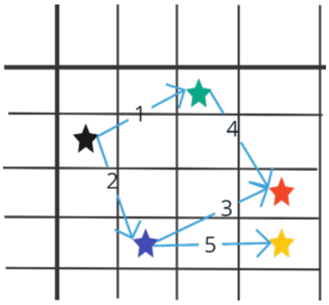


FIGURE 3.5: Interlinked capabilities (example: Red needs (4)green and (3)blue to be priorly completed. Whereas on the same maturity level, Yellow only needs (5)blue to be priorly completed)

Categorized zones:

Within this structure maturity levels get categorized in different zones. These zones have a specific meaning assigned to them. A good example of this is with the ISFAM. The ISFAM assigned maturity level 0-4 with Design, 5-6 with Implementation, 7-9 with Operational effectiveness and 10-12 with Monitoring (Spruit and Röling, 2014). A simplified version of this is visualized in figure 3.6. There are three zones, where maturity level 1 & 2 are assigned to zone "X", 3-5 to zone "Y" and 6 & 7 to zone "Z".

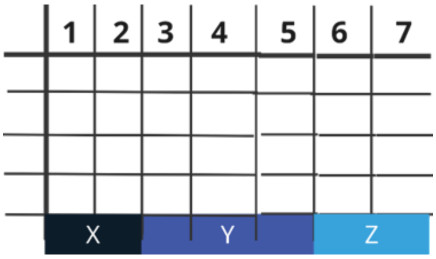


FIGURE 3.6: Example: design of Categorized zones wherein an particular ranges of maturity levels get appointed a meaning

3.2.3 Collected FAMMs

Now all the structural compositions have been explained. A table below is shown, wherein all the (google scholar) accessible FAMMs are organized according to their structural compositions. The information in table 3.2 & table 3.3 made use of an already structural comparative table about FAMMs (Sanchez-Puchol and Pastor-Collado, 2017). Any remaining information was searched and evaluated in a literature review.

The table 3.2 is read as follows. Each row lists a single FAMM along with its name, abbreviation, year of publication, and source. The FAMMs are arranged chronologically by year.

TABLE 3.2: Identified FAMMs in the retrieved literature: FAMMs are linked with abbreviations and ordered by date of publication

#	Name	Abbreviation	Year	Source
1	Test Process Improvement	TPI	1993 & 2004	(Andersin, 2004)
2	Implementation Maturity Matrix	IMM	2003	(Koop, Rooimans, and Theye, 2003)
3	DyA (Enterprise) Architecture	DyAMM	2007	(Steenbergen, Berg, and Brinkkemper, 2008)
4	Software Product Management	SPM-MM	2010	(Weerd, Bekkers, and Brinkkemper, 2010)

#	Name	Abbreviation	Year	Source
5	Software product portfolio	SPPM-MM	2011	(Jagroep et al., 2013)
6	Master Data Management Maturity Assessment	MD3M	2012	(Spruit and Pietzka, 2015)
7	Social Media Maturity Model	SM-MM	2012	(Kerkhof, 2012)
8	Organizational use for information	IUO-MM	2013	(Alves, 2013)
9	Software Product Line Management	SPLM-MM	2013	(Sprockel, 2013)
10	Disaster risk management	DRM-MM	2013	(Waldt, 2013)
11	Statewide Master Person Index	SMP	2013	(Duncan et al., 2013)
12	IT Governance Maturity model for Hard & Software governance	ITGOV-MM	2014	(Smits and Hillegersberg, 2015)
13	General IT practice	H-IT	2014	(Hermanns, 2014)
14	Information Security FAMM	ISFAM	2014	(Spruit and Röling, 2014)
15	IT carve out FAMM	IT-S-FAMM	2015	(Pflügler, Böhm, and Krcmar, 2015)
16	DevOps	DevOps-FAMM	2017	(Feijter et al., 2017)
17	Data as a Platform Ecosystems	DAAP-FAMM	2019	(Mijsters, 2019)
18	Tailoring a Domain-Specific Enterprise Architecture	TDSPEA	2019	(Zwienen et al., 2019)
19	Agile requirements for engineering practices	A-FAMM	2019	(Reeder, 2019)
20	Information Systems Risk Management Maturity Model	ISR3M	2019	(Elmaallam, Bensaid, and Kriouile, 2019)
21	FAMM Software ecosystem governance	FAMM-SEG	2020	(Jansen, 2020)

#	Name	Abbreviation	Year	Source
22	Modelling adaptive information security for SMEs in a cluster	Cluster FAMM	2020	(Yigit Ozkan et al., 2020)
23	API Management	API-m-FAMM	2021	(Overeem, Mathijssen, and Jansen, 2022)
24	Cybersecurity	CYSFAM	2021	(Yigit Ozkan, Lingen, and Spruit, 2021)
25	Gegevensbeschikbaarheid en kwaliteit	D-Qlty-FAMM	2021	(BPMIT, Kistemaker, and Bollen, 2021)
26	Modelbased system engineering	SE-MM	2021	(Amorim, 2021)
27	Assessing Smart Cities	ASC	2023	(Aljowder, Ali, and Kurnia, 2023)
28	Privacy by Design	PbD	2023	(Muszynski, 2023)
29	Prep warehouse for Digital Twin	DT-WA-FAMM	2024	(Quashie, n.d.)
30	Data Driven Decision-making Focus Area Maturity Model	DDDMFAMM	2024	(Raad, 2024)
31	Research Software focus area Maturity Model	RSMM	2024	(Bakhshi et al., 2024)

Table 3.3 categorises FAMM's according to their structural composition. the abbreviations used in table 3.2 are applied instead of the full FAMM names. Check marks in the form of a ('YES') or ('NO') have been used in each column to indicate whether the structural composition is included. Each row is a classification of a FAMM. An example to read the table is given in the bullet point below:

- The fourth row is 'SPM-MM'. which table 3.3 says is the Software Product management maturity model created in 2010. The SPM-MM has segmentation, interlinkage and contains 10 maturity levels. The SPM-MM does not have CMM levels and stages.

TABLE 3.3: FAMM structural composition comparative matrix

Abbreviation	CMM Level	Segmentation	Interlinkage	Stages	Levels
TPI	YES	NO	NO	NO	13
IMM	YES	YES	NO	NO	15
DyAMM	YES	NO	NO	NO	13
SPM-MM	NO	YES	YES	NO	10
SPPM-MM	NO	YES	NO	NO	10
MD3M	YES	YES	YES	NO	5
SM-MM	NO	YES	NO	NO	20
IUO-MM	NO	NO	NO	YES	13
SPLM-MM	NO	YES	YES	NO	10
DRM-MM	NO	NO	NO	NO	5
SMP	NO	NO	NO	NO	7
ITGOV-MM	YES	NO	NO	NO	5
H-IT	NO	YES	NO	NO	11
ISFAM	YES	YES	YES	YES	12
IT-S-FAMM	NO	YES	NO	NO	15
DevOps-FAMM	NO	YES	NO	NO	8
DAAP-FAMM	NO	YES	NO	NO	10
TDSPEA	NO	NO	NO	NO	12
A-FAMM	NO	YES	YES	NO	10
ISR3M	NO	YES	NO	NO	12
FAMM-SEG	NO	YES	NO	NO	7
Cluster FAMM	NO	YES	NO	YES	12
API-m-FAMM	NO	YES	YES	NO	10
CYSFAM	NO	YES	YES	NO	12
D-Qlty-FAMM	NO	YES	NO	YES	5 & 12
SE-MM	NO	YES	NO	NO	11
ASC	NO	YES	NO	NO	4
PbD	NO	NO	NO	NO	10
DT-WA-FAMM	NO	YES	YES	NO	7
DDDMFAMM	NO	YES	YES	NO	9
RSMM	NO	YES	YES	NO	7

Out of these 31 FAMM's, only two have been selected to be used for the study. A three step procedure was done for the selection (the reasons for the importance of the steps are included in the description of the steps).

Step 1: *Structural composition filter:*

The first step is to filter out from all the collected FAMM's, only those FAMM's that follow the structural compositions listed below (the rationale why certain compositions were and were not chosen are added in the third column). Read the second column of table 3.4 for the selection criteria. The filtering was done to avoid too much variation for the customGPT's instructions. If this is not done, then the instructions for generating a FAMM will be too random. This could lead to the AI-generated FAMM to be incomparable to the Human created FAMM.

Structural composition	Selection criteria	Chosen Rationale
Level difference	7 - 12	Frequently used range of maturity levels
CMM style	NO	Brings too much complexity in the interview to be evaluated
Segmentation	YES	Frequently used method
Interlinkage	NO	Brings too much complexity in the interview to be evaluated
Categorized zones	NO	Rarely used structural composition

TABLE 3.4: Selection filters

Filtered: 6 MM's [SPPM-MM, H-IT, DevOps-FAMM, DAAP-FAMM, SE-MM, ISR3M]

Step 2: *Relevance filter*

This study should be able to generalise that the FAMMs selected are representative of a scientifically accepted FAMM. Therefore, it was chosen that the second step filters, from the current pool of filtered FAMMs, published and cited FAMMs. If the FAMM did not have citations or a publication, then the chances of lower quality FAMMs would be higher. This would result in a misguided view on the performance of an accepted FAMM. Besides this, the 'Relevance filter' filters based on documentation. In this study, documentation refers to the paper where the FAMM was published in. In table 3.2 the paper that contains the FAMM can be found in the 'Source' column. To control documentation for the interview, the relevant information should be chosen. For the relevance filter, it was chosen that the paper (of the selected FAMM) needs to have an image of the FAMM and a descriptive documentation of each capability. These were chosen, because a participant would need to see the model and be able to read up on the capabilities within the model. Additional information (e.g. assessment questions) could have benefited this study. However, it was chosen to only include the FAMM and

descriptive documentation on the capabilities because of the time limit of the interview.
Filtered: 3 MM's [SPPM-MM, SE-MM, DevOps-FAMM]

Step 3: *Preference filter*

The third step is to select the final two FAMMs that the author feels are most appropriate. To ensure consistency, each participant should review the same FAMMs. The reason for this is that selecting only one FAMM would leave the uncertainty of what the 'average' accepted FAMM quality is. Choosing two FAMMs removes some of this uncertainty, but is still a small set from which to generalise. Although three FAMMs could have been chosen for this study, evaluating three FAMMs would not leave enough space for the other questions in the interview. From the filtered FAMM pool, the DevOps focus area maturity model (DevOps-FAMM) and the software product portfolio management maturity model (SPPM-MM) were selected. These two FAMMs were selected because they had the highest citations and both published in a Springer journal. Which the author thought was the most representative of the FAMMS within the scientific community. The content of the two FAMM's will be copied into one document. To view the document of the copied FAMM content see appendix B. When creating this document, no changes are made to the original document. For the two selected FAMMs, a document is created containing all the contents of the AI-generated variant. Chapter 4 contains the explanation of how an AI-generated variant was created.

Chosen: DevOps-FAMM, SPPM-MM

3.2.4 Formulating Design Criteria

An MM first needs to be evaluated before an MM is deployed. The evaluation is done to validate the relevance and rigor of an MM. In the FAMM design it is told that the primary type of evaluation is done by expert reviewing the MM and performing a case study (Steenbergen et al., 2010). For this study however, it was chosen to only focus on experts reviewing an MM. This study will also exclude what procedure an expert review should best go through. For example, an often cited paper on performing an expert review in MM, is the delphi method (Becker, Knackstedt, and Pöppelbuß, 2009). The delphi method describes how a procedural expert review would be an appropriate method for conducting an expert review. What this study does use, are the criteria used in an expert review. In a previous paper from Michael Roseman, the evaluation criteria they used were: 'construct validity' (represents both 'face validity', 'content validity') , 'reliability' and 'generalisability' (De Bruin et al., 2005). However, these design criteria were not seen in the evaluation of FAMMs. Reoccurring design criteria that were seen in FAMM evaluations were:

- Completeness
- Ease of use
- Effectiveness
- Operational feasibility
- Usefulness

For this research, these Design criteria were chosen. To determine how well a FAMM performs on a design criteria, each design criteria will get a separate question asking the participant to give it a Likert rating on to it. The questions used will be found in the evaluation in the following paragraph about structure interview.

3.2.5 Structure Interview

Gathering results has been done through a semi-structured interview. A semi structured interview allows asking all the required questions, while still leaving room to dig for additional information. The interview was structured to remain within a time span of 45 to 60 minutes and was conducted in three phases [2]. Before the interview started, the participant was first asked whether they agree on the specified rules from the consent form (found in appendix B). Once the participant agreed, the recording of the interview started. The first phase is the introduction, where the participant was asked questions about their previous experience with MMs. In the second phase, the participant evaluated 4 FAMMs, of which two are AI-generated and the other two Human created. In the last phase, the participant has been asked to envision, what role GenAI could have in MMs. Closure was then done, where the participant is thanked for participating and the recording was stopped. Further details of the interview protocol can be read in appendix B.

3.2.6 Data collection

To answer the research questions, information from the interviews need to be provided. To gather information, participants of the interviews were asked to consent to recording the interview. Commonly, Microsoft Teams was used to record and transcribe the interviews. Once the transcription was made, the author of this study would re-listen to the interview. If there were any mistakes in the transcription, the author would correct it. Every interviewee consented to the recording. A breakdown of qualitative and quantitative information collected is shown in table 3.5.

As can be seen in table 3.5, all the qualitative results fall under the 'transcript'. Firstly, there were the comments where participants gave their reasons for giving certain Likert scores. Direct quotes of their rationale have also been used in chapter 5.3. The second item of the qualitative data, from the transcripts were the comments on why they thought one of the MMs they were given was generated by AI. The third is the information about the participants which can be found in the Appendix B. The information includes: whether they had domain knowledge of the given FAMMs, what particular experience they have with MMs and the order in which they received the FAMMs in the evaluation. The last item in the qualitative data is the role the participant expects GenAI to play in MM. Each of these roles was also further categorised into a zone within the life cycle of an MM. The categorisation was carried out by the author, and the rationale for each categorisation is given in the Appendix B.

For the Quantitative data, Likert ratings and AI guessed FAMMs were collected. The collection of Likert ratings worked as follows. A participant was given a question (on a scale from 1 to 5) related to the design criteria (See Appendix A for the questions). An example of an answer is that a participant gave the lowest rating (e.g. 'Very Difficult' on Ease of Use), a '1', which would be collected. Once all the

[2] See Appendix B for a full overview of the interview protocol

quantitative data had been collected, the averages of each design criteria were calculated. Of the total collected Likert rating scores, box plots were also created. In an effort to understand the general score that were given.

Qualitative Data	**Quantitative Data**
Transcript	Likert ratings (results from ’1-5’)
• Comments: ’why gave this score’	**AI guess** (percentage correct)
- Direct quotes	
• Comments: ’why this MM is AI’	
• Participant	
- Domain knowledge	
- Experience with MM	
- Order of given FAMMs	
•Their expectation of GenAI role	

TABLE 3.5: Overview of Qualitative and Quantitative Data Collection: A breakdown of the information stored within the transcripts and documented evaluation ratings and guesses

Almost all the data collected from the interviews were put into worksheets. These worksheets were used to organize the result, to later be used in an analysis. Part of the results, for example why participants gave a particular Likert scale rating, were copied over from the notes taken in the interviews. Whenever the notes taken during the interview were insufficient or required direct quotes the transcript was used.

3.3 Remaining Literature

Solely the data collected from the interviews, could not answer all the SRQs. For SRQ 1 and SRQ 3 a literature review was used as a method to answer their research question. SRQ 1 required a thorough analysis to find the most appropriate design criteria for the quality assessment of an MM. SRQ 3 required a life cycle into which the participants expectations could be placed. A number of life cycles were reviewed in the literature and one was selected as the most appropriate. If the life cycle was not included in this SRQ, then there would be no basis for including ’life cycle’ in the MRQ. See the next chapter 4.1 for the selected life cycle. For SRQ 2, a literature review was carried out. However, the literature review was not directly used to answer the research question and is therefore not included as a method in Table 3.1. Instead the literature review was indirectly used as a supplement for the research method ’Qualitative surveys’.

Chapter 4

Framework design

In this chapter, it will be explained how GenAI was used to create MMs, that were used in this study. The chapter will start with an explanation of how MMs are developed, followed by how a MM is used for a self assessment. Next, GenAI will describe how, through giving a GPT certain instructions, it can influence what output the GPT generates. More will be explained how a customGPT, with a set of instructions, was utilized to generate FAMM's. The output that the customGPT's generates often contain errors. These errors and how to deal with them, will also be discussed in this chapter. Lastly, an example is shown how the customGPT generated a FAMM.

4.1 Life cycle of Maturity Models

Before diving into what role GenAI can have in MMs, the development and use of MMs need to be made clear. A model created by Tobias Mettler will be used to explain the life cycle of an MM. This model was chosen because it includes the user perspective, known as the application cycle. Other frameworks, such as the FAMM design, only include the developer's perspective on how an MM should be made [1]. The life cycle is shown in figure 4.1.

There are two cycles shown in the figure. On the left is the development cycle, where an MM is created. On the right is the application cycle, where a user selects and applies an MM. As can be seen in the figure, each cycle starts at 1 and ends at 5. Once 5 is reached, the cycle could be repeated. However, the starting point would then be 2. Both cycles meet in the middle. Mettler describes this as follows: "The design of maturity assessment models is intimately connected with the application phase, too. It is therefore our opinion not to analyse both perspectives concurrently" Mettler, 2011. In the literature, the development cycle has been referred to as the developer's perspective. Whereas the application cycle has been referred to as the user perspective Mettler, 2011. Both terms will be used in this paper. It is important to note that both viewpoints excluded the first phase. However, for the purposes of this study, it was decided to include the first phase in both perspectives. Including and recognizing the first phase in each viewpoint was a research design decision made by the author. This was because the life cycle would later be used as a framework for classifying the GenAI role expectations of the participants. Having a framework that includes the first phase, gives more space to make appropriate classifications of the expectations received.

[1] Keep into account, Tobias Mettler's model is a very broad generalisation of the life cycle of an MM. There does not seem to be 'one way' to develop an MM for each action in the literature. For example, when evaluating the design of an MM, different validation methods are used, ranging from subjective to empirical (Lasrado, Vatrapu, and Andersen, 2015)

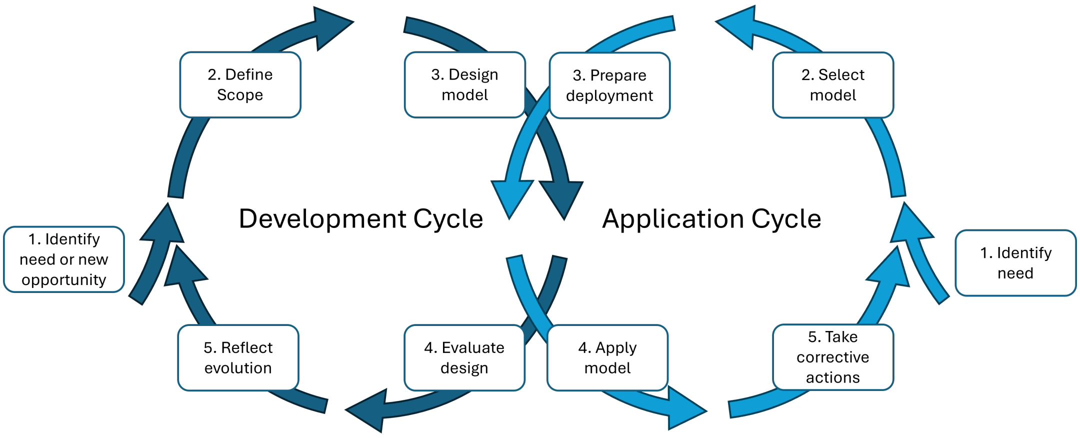


FIGURE 4.1: Mettler's life cycle of MMs: Development and application cycle of maturity assessment models

Below here, there will be a brief explanation about what the viewpoint, and each of the phases within, the viewpoints mean.

4.1.1 Developer's perspective

The main focus of the development cycle is on MMs and their creation. The phases described below cover the steps that developers need to take, to ensure the MMs they create are robust, relevant, and useful. Each phase is important and helps in accomplishing a systematic process of development, throughout the life cycle. The phases include:

1. **Identify need or new opportunity**
   This involves identifying a gap or a new opportunity where a new MM could bring value. It is the first and the most important step of the development process.

2. **Define scope**
   This involves determining the boundaries and the focus areas of the MM. This step makes the model fit the specific needs, so that the model will be neither too general or too particular.

3. **Design model**
   The actual design of the MM happens in this stage. Developers map out the structure, elements, and the working of the MM. This stage is crucial because it creates a baseline for the MM.

4. **Evaluate design**
   In this stage, the designed MM is evaluated in terms of its effectiveness, reliability and applicability. The evaluation takes the form of pilot testing, collection of feedback and consecutive improvements.

5. **Reflect evolution**
   Reflecting evolution means monitoring the performance of the MM and effecting necessary change over time. because of this, the MM becomes relevant and effective, even with the changes in context and requirements.

4.1.2 User's perspective

The application cycle describes the process that users need to go through to appropriately choose and implement MMs. The stages are designed to make sure that the MMs are theoretically correct and that their implementation is of practical use, in service of the business and its goals. These are the following stages:

- **Identify need:**
  Just as a development cycle, this stage includes identifying a need or an opportunity for which an MM can add substantial value. In this phase, users can identify a specific area in their organization that would benefit from using an MM.

- **Select model**
  In this phase, users find the most appropriate MM that fits their needs. Users can use more than one MM to compare different MMs based on their applicability, strengths and weaknesses.

- **Prepare development** MM users can then apply the selected MM in their organization. This process includes assessment, data collection and evaluation of the current maturity level of different processes, or functions, of an organization.

- **Apply model**
  Here users analyze results to know strengths, weaknesses and areas that should be improved. The analysis supports an understanding of the current maturity level. It also gives insight into the steps that must be taken to reach higher maturity.

- **Take corrective actions**
  In the last phase, users put the suggested maturity level improvements into practice. This can include implementing new practices, training and ongoing monitoring to ensure continuous improvement.

4.2 Constructing an AI Generated FAMM

OpenAI's GPT builder was used to create a customGPT. The customGPT was created using text instructions only, without the need for any coding. To let customGPT generate a correct format for a FAMM, iterative testing was done. This testing was done by trial and error, with progressive variations in the instructions. Versions of these variations were tested to evaluate which format gave the best FAMM generation. The evaluation of which format was considered the best, was done by subjective evaluation. How this process went is explained in more detail below.

4.2.1 Maturity Maker

The customGPT created to generate maturity models is called 'Maturity Maker' and will be referred to as MaturityGPT. MaturityGPT uses GPT-4 for its knowledge and understanding when outputting text. In the instructions, the prompt engineering suggested in (White et al., 2023) has been used to get MaturityGPT to output FAMMs.

In the following, each prompt technique will be explained with the corresponding instruction. Only one sentence of the corresponding instruction, relating to the prompt, is used. The full text of the instructions can be found in the Appendix A.

1. Role
Specifying what role MaturityGPT has, is done to provide context. This ensures that the AI's output is relevant to the scenario. In the first instructions of MaturityGPT, the role "MaturityGPT is made to create FAMM's" is given:

> ``This GPT creates a focus area maturity model ... Start with the table in the output``

2. Example
An example is given of what the expected output should be. Variations of instructions were tested which gave the best output to generate a FAMM. An example of the API management MM (Overeem, Mathijssen, and Jansen, 2022) , where only one focus area was filled in with capabilities, showed to worked best based on the subjective evaluation of the author [2] . It has to be noted, that the example, given in the instructions, does not provide information on what level the capabilities, in the filled in focus area, belong to. The understanding of what level a capability should belong to and how maturation of capabilities should flow, within a component, is done entirely through MaturityGPT's model, GPT-4.

> ``This is an example where only focus area 5 is filled in ... 6. Commercial 6.1 Service-Level Agreements 6.2 Monetization Strategy 6.3 Account Management``

3. Mathematical Formulations
Now that MaturityGPT knows its roll and has an example of output, MaturityGPT needs to know what rules the output should adhere to. Luckily, in the "Design of the FAMM" paper (Steenbergen et al., 2010), there is a full mathematical formulation on how a FAMM should be structured. It includes terms as focus area, Levels, capabilities, mapping and gives a breakdown of the mathematical concepts and how these terms should construct a FAMM.

> ``In order to provide rigorous fundamentals for focus area maturity models ... doesn't contain all capabilities with scale 1 (in both cases the maturity scale of the organization will be 0)``

4.2.2 Versions

The creation of the MaturityGPT (MaturityMaker) went through a trial and error phase. In order to show the general evolution of the CustomGPT, versioning was used. Figure 4.2 shows each version of the CustomGPT and what the biggest problem within that version was. In the end, version 0.5 was chosen to be used to generate the FAMMs that have been used in this study. Additional text is described below, to go give more context what occurred within a version.

[2] Other variations, as filling the CustomGPT's knowledge with example papers of FAMM's, showed to have less favorable results.

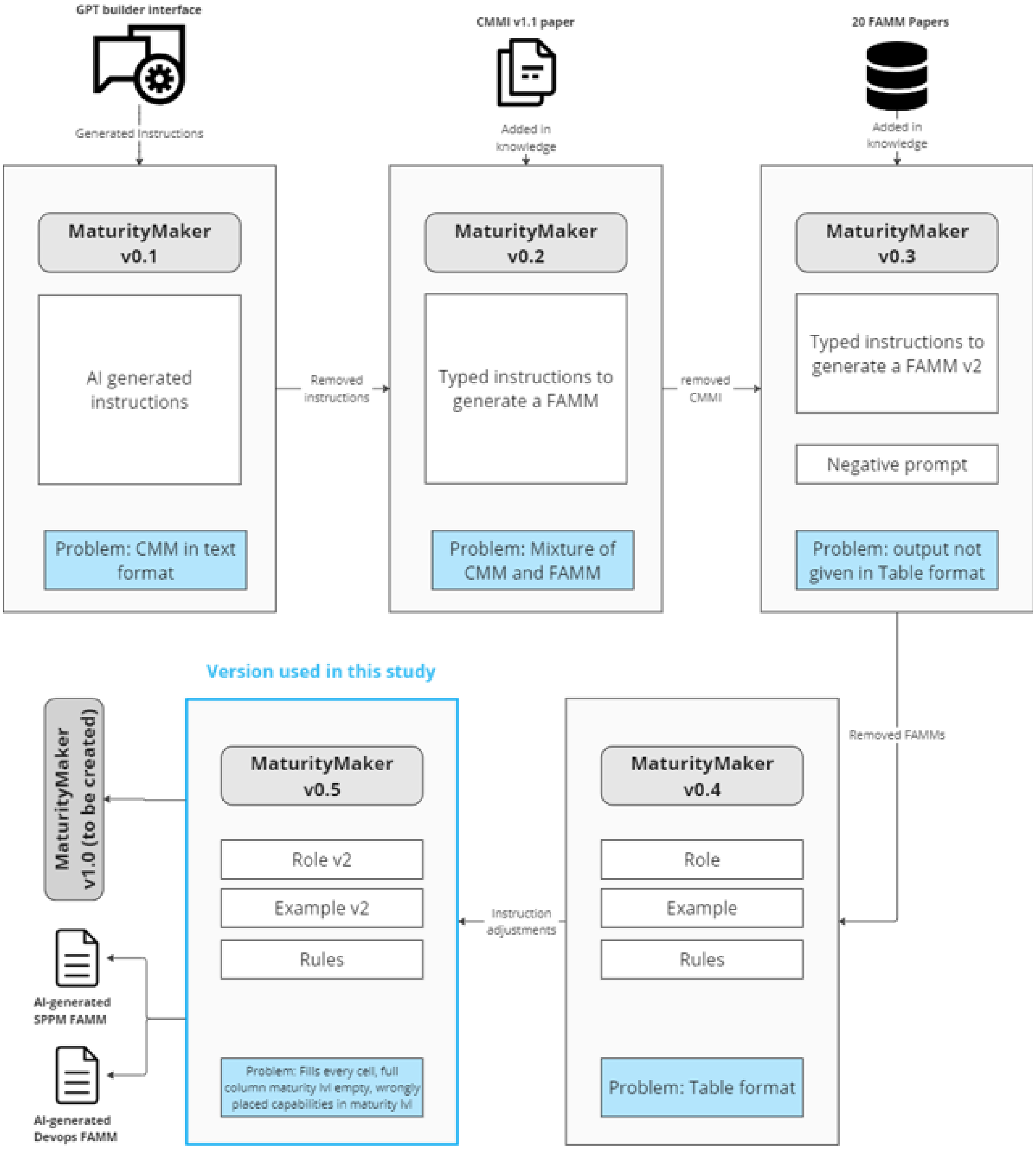


FIGURE 4.2: Evolution of the MaturityMaker: a iterative prompt engineering process to creating an AI tool for generating FAMMs that follows the selected structural composition

Version 0.1 Made use of (Ask in the interface to generate FAMM's that have 10 levels and 2 to 5 capabilities per component, noting that the capabilities should match the maturity level that belongs to it).

- Instruction: tested variations
- Problem: plain text output.

Version 0.2 Used (Relying solely on generating custom instructions proved ineffective. In this version, to test what effect it would have, some knowledge and a role would be enough)

- Instruction: added CMMI, FAMM papers in knowledge. Given the role to generate a FAMM not a CMM.

- Problem: Output often had a mixed version of CMM and FAMM where aswell every cell was filled in.

Version 0.3 Used (CMM was recurring MM output. Prompt was told to exclude CMM in output and 20 papers were fed into MaturityGPT)

- Instruction: Structured FAMM documentation
- Problem: FAMM did not output in tabular format, often showing results in markdown. Do not output tabled format without repair prompt

Version 0.4 (In trial and error, it was found that when given a complete example of what the output should look like, the GPT gave the correct format for a FAMM. Testing variations showed the best output to generate when knowledge was removed.)

- Instruction: Removed all knowledge and gave instructions that included a role, example, rules.
- Problem: Still having problems with the table output.

Version 0.5 (CURRENT) (FAMM's have been created, but often with errors. Minor changes have been made to improve MaturityGPT's output).

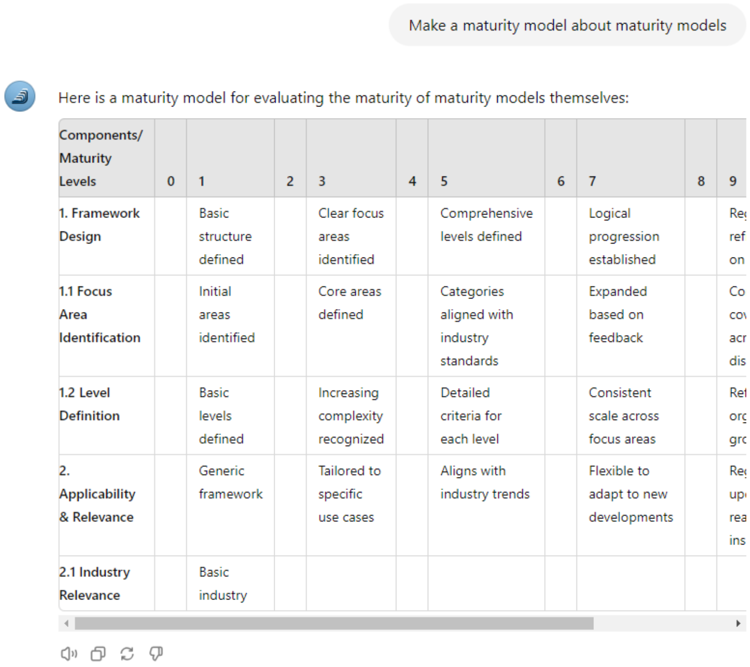
- Instruction: In the role, added a second time that the output should be a table, and in the example changed the text to say that only one focus area was filled.
- Problem: MaturityGPT still rarely generates a FAMM in the correct structure.

Version 1.0 (Future) The goal was to make GenAI generate a FAMM. MaturityGPT 0.5 can do this, but rarely. A more robust version is being worked on, but two generated FAMM outputs from v0.5 were chosen in this study to stay within the time frame.

4.2.3 Generation Errors

MaturityGPT version 0.5 is able to output a FAMM that is in the correct format, as specified in chapter 3.2.2. In the attempt to create a FAMM, it often has one (or more) structural error(s) in the first generation. The most frequent occurring errors are shown in the list below. However, A generated FAMM can still be corrected by giving a prompt, that describes the errors and how the error should be corrected.
MaturityGPT Could have the following errors (only a screenshot is shown of the model. The FAMMs have a total of 10 levels. If scrolled downwards, additional focus areas and capabilities would be seen. Links are provided for each model in order to view the full generated models):

Error #1 Only fills in one or several maturity levels with capabilities:



Make a maturity model about maturity models

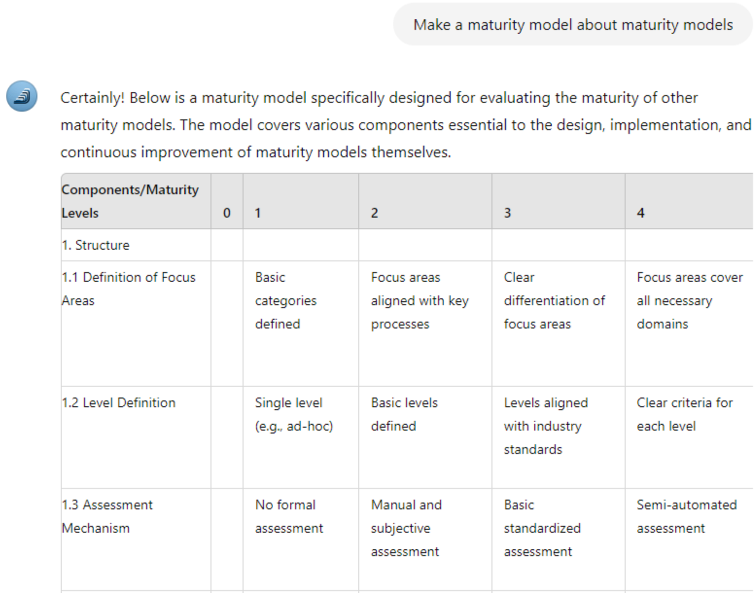
Here is a maturity model for evaluating the maturity of maturity models themselves:

Components/ Maturity Levels	0	1	2	3	4	5	6	7	8	9
1. Framework Design		Basic structure defined		Clear focus areas identified		Comprehensive levels defined		Logical progression established		Re ref on
1.1 Focus Area Identification		Initial areas identified		Core areas defined		Categories aligned with industry standards		Expanded based on feedback		Co co acr dis
1.2 Level Definition		Basic levels defined		Increasing complexity recognized		Detailed criteria for each level		Consistent scale across focus areas		Ret org grc
2. Applicability & Relevance		Generic framework		Tailored to specific use cases		Aligns with industry trends		Flexible to adapt to new developments		Re up rea ins
2.1 Industry Relevance		Basic industry								

FIGURE 4.3: Example error: Each cell is from top to bottom filled with capabilities for specific maturity levels whereas maturity levels are left empty

link: https://chatgpt.com/share/8109e1ff-250c-499b-b272-84fbe8484c0c

Error #2 Fills in every cell of the FAMM with a capabilities (except lvl 0) :



Make a maturity model about maturity models

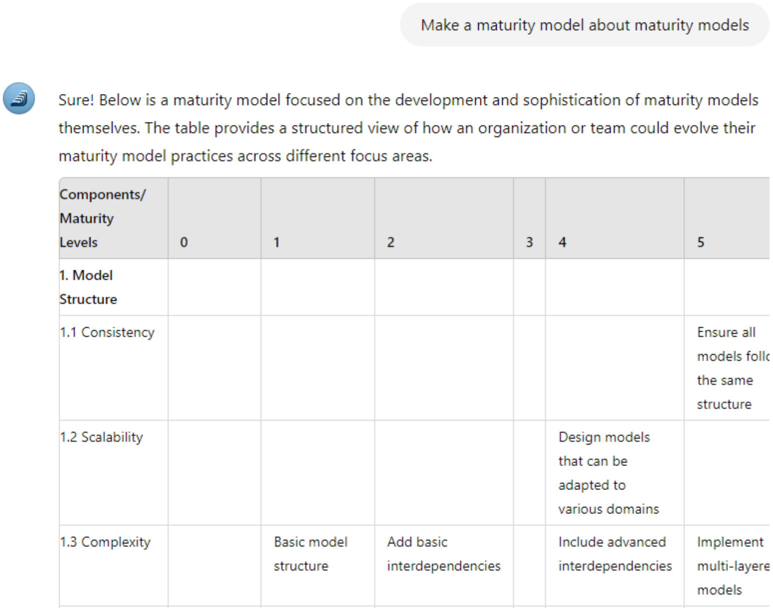
Certainly! Below is a maturity model specifically designed for evaluating the maturity of other maturity models. The model covers various components essential to the design, implementation, and continuous improvement of maturity models themselves.

Components/Maturity Levels	0	1	2	3	4
1. Structure					
1.1 Definition of Focus Areas		Basic categories defined	Focus areas aligned with key processes	Clear differentiation of focus areas	Focus areas cover all necessary domains
1.2 Level Definition		Single level (e.g., ad-hoc)	Basic levels defined	Levels aligned with industry standards	Clear criteria for each level
1.3 Assessment Mechanism		No formal assessment	Manual and subjective assessment	Basic standardized assessment	Semi-automated assessment

FIGURE 4.4: Example error: Every single cell in the sub focus areas are filled with capabilities (excluding the focus areas in the table)

link: https://chatgpt.com/share/2bc58914-9b3a-4d31-8f04-04d5144df300

Error #3 Keeps one or several maturity levels without a capability:



Make a maturity model about maturity models

Sure! Below is a maturity model focused on the development and sophistication of maturity models themselves. The table provides a structured view of how an organization or team could evolve their maturity model practices across different focus areas.

Components/ Maturity Levels	**0**	**1**	**2**	**3**	**4**	**5**
1. Model Structure						
1.1 Consistency						Ensure all models follc the same structure
1.2 Scalability					Design models that can be adapted to various domains	
1.3 Complexity		Basic model structure	Add basic interdependencies		Include advanced interdependencies	Implement multi-layere models

FIGURE 4.5: Example error: As shown in the figure, level 3 is left completely empty

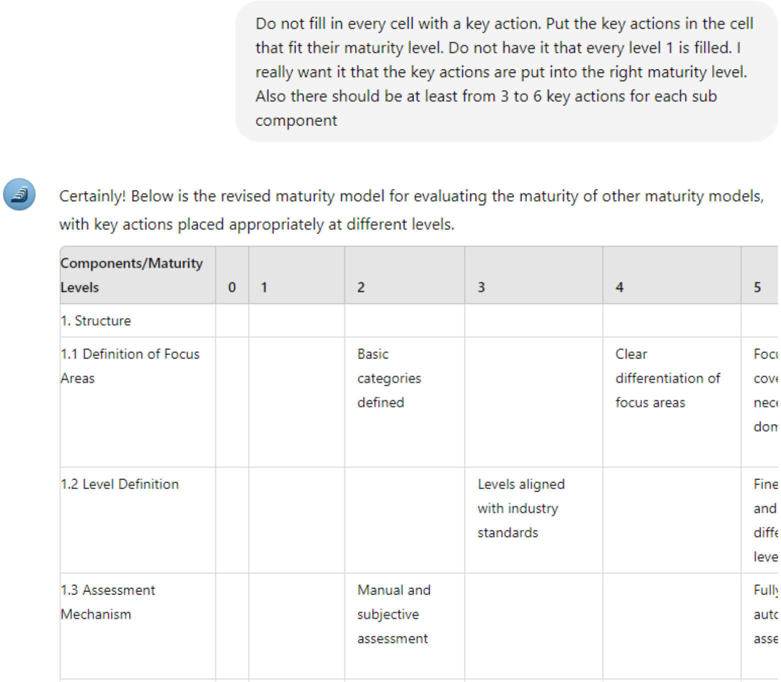
link: https://chatgpt.com/share/39d8587a-7432-4602-bd58-f4654fdd2a3c

How to deal with generated errors?

An effective strategy is with a 'repair prompt'. In this study, only the 'SPPM-FAMM' used a repair prompt. The DevOps-FAMM did not require a repair prompt.

In order to correct a previous generation error, the following prompt can be given: `"Do not fill in every cell with a key action. Put the key actions in the cell that fit their maturity level. Do not have it that every level 1 is filled. I really want it that the key actions are put into the right maturity level. Also there should be at least from 3 to 6 key actions for each sub component"`

Figure 4.6 shows how a previous error was fixed with a repair prompt.



Do not fill in every cell with a key action. Put the key actions in the cell that fit their maturity level. Do not have it that every level 1 is filled. I really want it that the key actions are put into the right maturity level. Also there should be at least from 3 to 6 key actions for each sub component

Certainly! Below is the revised maturity model for evaluating the maturity of other maturity models, with key actions placed appropriately at different levels.

Components/Maturity Levels	0	1	2	3	4	5
1. Structure						
1.1 Definition of Focus Areas			Basic categories defined		Clear differentiation of focus areas	Focu cove nece don
1.2 Level Definition				Levels aligned with industry standards		Fine and diffe leve
1.3 Assessment Mechanism			Manual and subjective assessment			Fully auto asse

FIGURE 4.6: Used repairprompt (Continuation of Error #2)

link: https://chatgpt.com/share/2bc58914-9b3a-4d31-8f04-04d5144df300

4.2.4 Prompt to design

As mentioned before, MaturityGPT does not always produce the correct output. A workflow was created, shown in figure 4.7, that provides procedures on handling incorrect outputs. This workflow was followed to create an AI variant of the two previously selected, human created FAMMs (DevOps and SPPM). It is important to note that *no* content of the AI-generated variant was changed. Which means that all content generation and capability positioning was done solely by the MaturityGPT. Furthermore, the creation of an AI-generated MM was significantly less time intensive, than a human created MM. Since, a human created MM takes commonly the time of a full Master thesis or PhD (as stated in sub chapter 1.2), while the author created the AI-generated MMs in a few minutes (excluding the time of creating the customGPT, which was around two weeks).

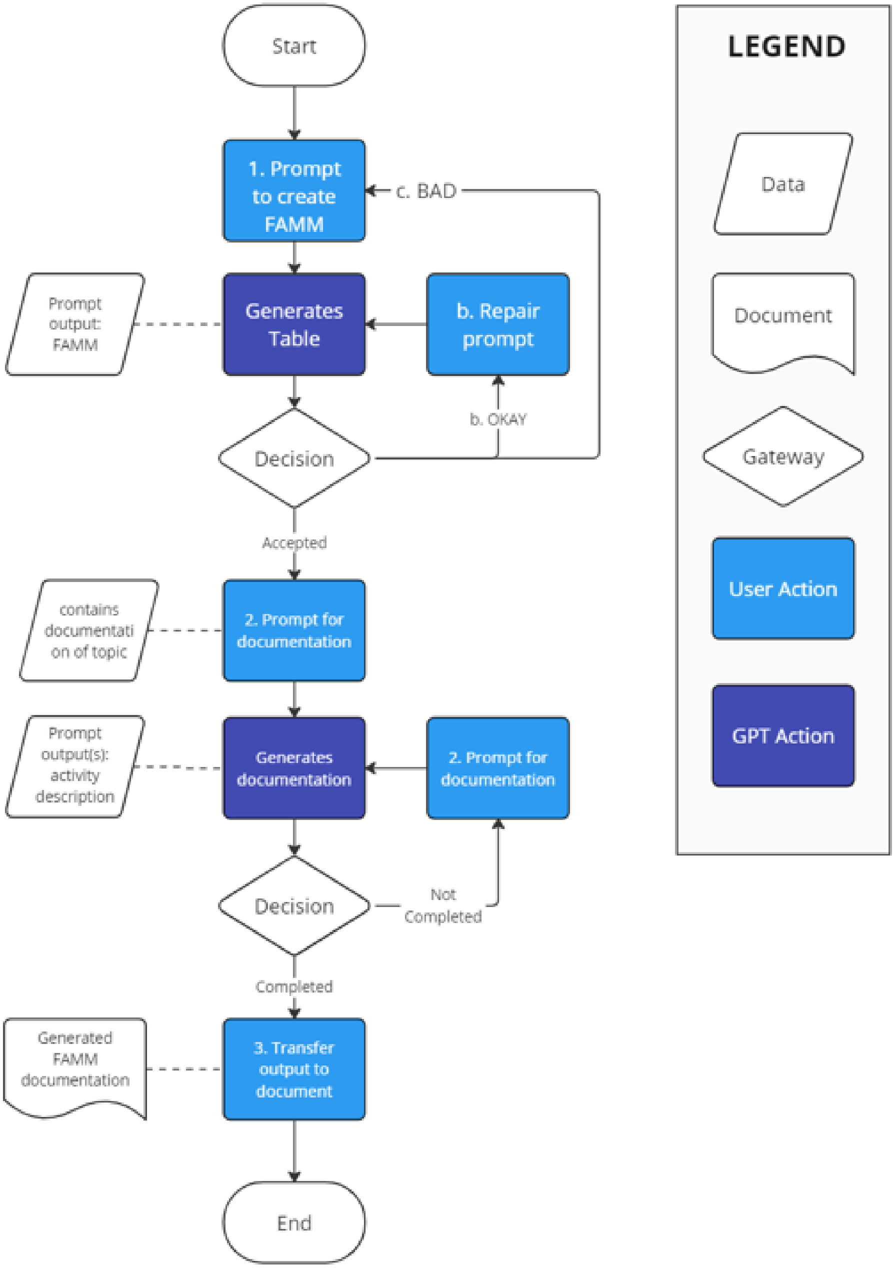


FIGURE 4.7: Prompt to design workflow: A procedure to prompt into MaturityGPT, to get to an AI-generated FAMM that is transferred into a document used for the research

The following procedure was used to get from the MaturityGPT prompts to the two documents containing AI-generated FAMMs used in the interviews.

Process for Generating a Maturity Model using GPT

1. Prompt to create FAMM

Prompt: `"Make a Maturity Model about [X] in [Y]."`

[X]: Maturity model topic

[Y]: Additional context (optional)

Examples:

X1 = DevOps

Y1 = Software organization

X2 = Software Product Portfolio Management

Y2 = n.a.

Evaluation of Initial Prompt Output

a **GOOD:** Output is acceptable; proceed to the next step.

b **OKAY:** Some issues; Use a repair prompt and review if the correction is acceptable.

c **BAD:** Output has too many mistakes; re-prompt.

Common Issues:

See chapter 4.2.3.

2. Prompt for documentation
Prompt: `"Make documentation for the first focus area following this example: [Z]."`

[Z]: Documentation style used from the selected maturity model.

Examples:

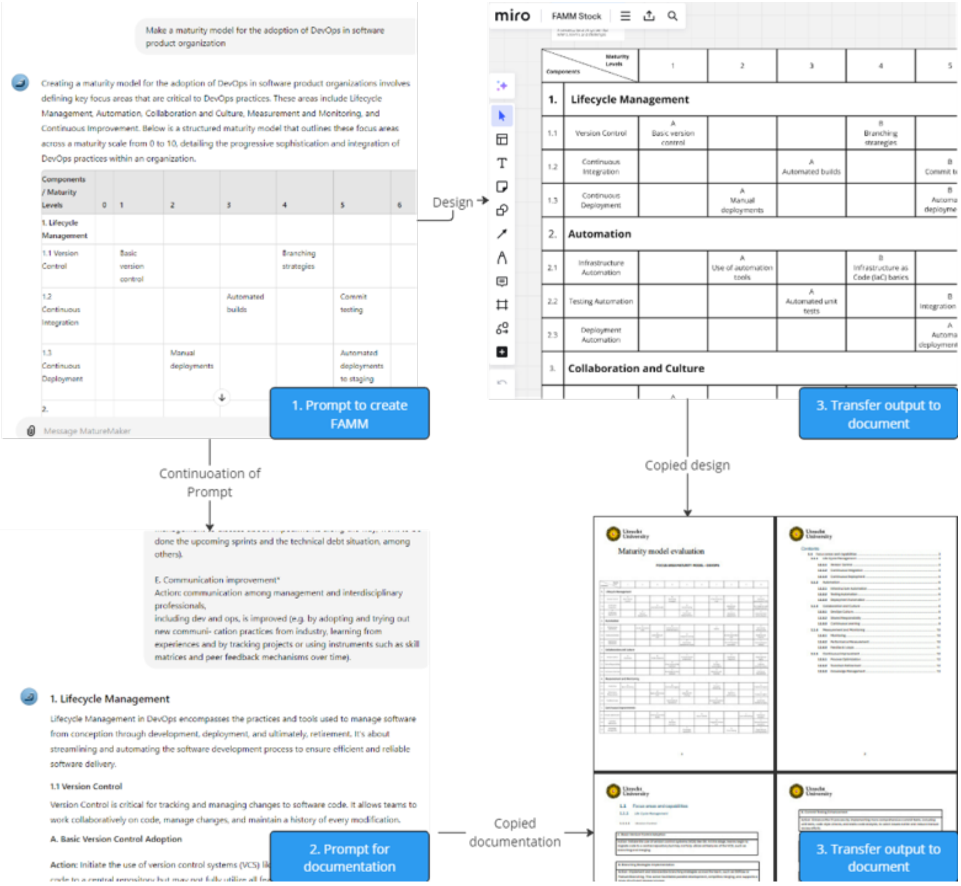
Z1 = Example from a previous DevOps FAMM.

Z2 = Example from a previous SPPM FAMM

3. Transfer Output to Document

The FAMM output will be designed into a MIRO, then both the model and text is copied over into a WORD document.

Example of Transfer Process shown in figure 4.8



Make a maturity model for the adoption of DevOps in software product organization

Creating a maturity model for the adoption of DevOps in software product organizations involves defining key focus areas that are critical to DevOps practices. These areas include Lifecycle Management, Automation, Collaboration and Culture, Measurement and Monitoring, and Continuous Improvement. Below is a structured maturity model that outlines these focus areas across a maturity scale from 0 to 10, detailing the progressive sophistication and integration of DevOps practices within an organization.

Components / Maturity Levels	0	1	2	3	4	5	6
1. Lifecycle Management							
1.1 Version Control		Basic version control			Branching strategies		
1.2 Continuous Integration				Automated builds		Commit testing	
1.3 Continuous Deployment			Manual deployments			Automated deployments to staging	
2.							

done the upcoming sprints and the technical debt situation, among others).

E. Communication improvement"
Action: communication among management and interdisciplinary professionals,
including dev and ops, is improved (e.g. by adopting and trying out new communi- cation practices from industry, learning from experiences and by tracking projects or using instruments such as skill matrices and peer feedback mechanisms over time).

1. Lifecycle Management

Lifecycle Management in DevOps encompasses the practices and tools used to manage software from conception through development, deployment, and ultimately, retirement. It's about streamlining and automating the software development process to ensure efficient and reliable software delivery.

1.1 Version Control

Version Control is critical for tracking and managing changes to software code. It allows teams to work collaboratively on code, manage changes, and maintain a history of every modification.

A. Basic Version Control Adoption

Action: Initiate the use of version control systems (VCS) li

FIGURE 4.8: Overview of practical steps followed in the Prompt to document workflow

4.3 Meta-Model of Interaction

A meta-model was created to aid in understanding GenAI generating MMs. Prior to the creation of the meta-model, a review of the literature was conducted to identify the existing meta-models related to GenAI and FAMMs. In total, three meta-models were identified. Each of the three meta-models contained information that was used for the creation of the meta-model. Figure 4.9 shows a meta-meta-model wherein, of the identified papers, what information was used into the meta-model shown in figure 4.10. Besides that, the meta-meta-model provides an easy to comprehend version of the interactions between the association of an MM and GenAI. In figure 4.9, *'Generative AI'* is informed what structural design a *'Maturity Model'* follows and from that, *'creates'* a *'Maturity Model'*. The meta-model is divided into two sections. One section is the meta-model, where elements of the 'Domain' and 'FAMM' from Steenbergen et al., 2013 were provided. Further, the Model components from Jansen, 2020 were used. It is to be noted that the naming of the model components were slightly altered. The Meta-model of the found paper (Jansen, 2020) uses the naming: 'practices', whereas the meta-model calls [practices -> capabilities] and [capabilities -> sub focus areas].

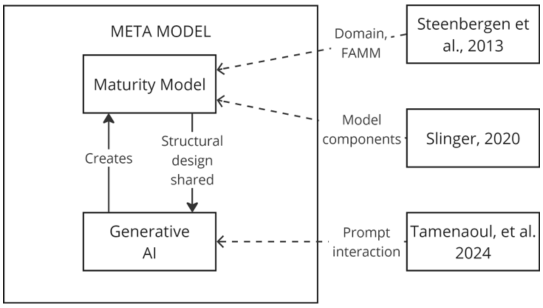


FIGURE 4.9: Meta-meta-model of the associations between GenAI and MM

Go to the next page to view the author's meta-model. An example, whereof a starting point was taken, can be read below to aid in understanding the meta-model:

1. **User Prompt:** Start in the section for *'Generative AI'* with the *'User prompt'*, which is shown in the middle. For the first user prompt, it contains the *'domain topic'* of a FAMM. A *'0...1'* is indicated as to whether the prompt contains the Domain topic. With this being the first prompt, A *'1'* is given, thus claiming that the *'Domain Topic'* is contained in prompt in a *'User Prompt'*.

2. **Behind the system prompt:** As the *'User Prompt' 'prompts'* for an *'System prompt'*, the following occurs: the *'User Prompt'* is *'prompted through'* a *'User log'*. Also the prompt is *'Added to'* the *'Memory of responses'* that is stored in the *'User log'*. Within the *'User log'*, a GPT had to be chosen. For this meta-model, the *'CustomGPT (Maturity Maker)'* was chosen, whereof the *'User log' 'uses'* this *'CustomGPT (Maturity Maker)'*. The *'CustomGPT (Maturity Maker)'* 'contain' *'Instructions'*. The *'Role'*, *'Example'* and *'Rules'* are an aggregation of the *'Instructions'*. Further, the *'CustomGPT (Maturity Maker)'* utilizes a *'GPT model (GPT-4)'* which is the algorithm that does the 'next word production' to give a *'System prompt'*. The *'GPT model (GPT-4)' 'contains'* the *'Memory of responses'* as input for the model to predict the next word on.

3. **System prompt:** As response to the *'prompt'* of the *'User Prompt'*, a *'System prompt'* outputs a *'Textual output prompt'* which is also added to the *'Memory of responses'*. The *'Textual output prompt'* would starts with the initial textual format of a *'Model'*, which represents the FAMM of the prompted Domain topic. The following prompt is expected (Following workflow in chapter 3) to be a prompt for the *'Description'*, that based on the *'Memory of responses'*, would generate a *'Description'*, which 'is based on the *'model'*. Once the *'System prompts'* has done all the *'outputs'* for the 'Textual output prompt', then the *'Textual output prompt'* can be *'Copied to'* a *'Document MM AI Generated'*.

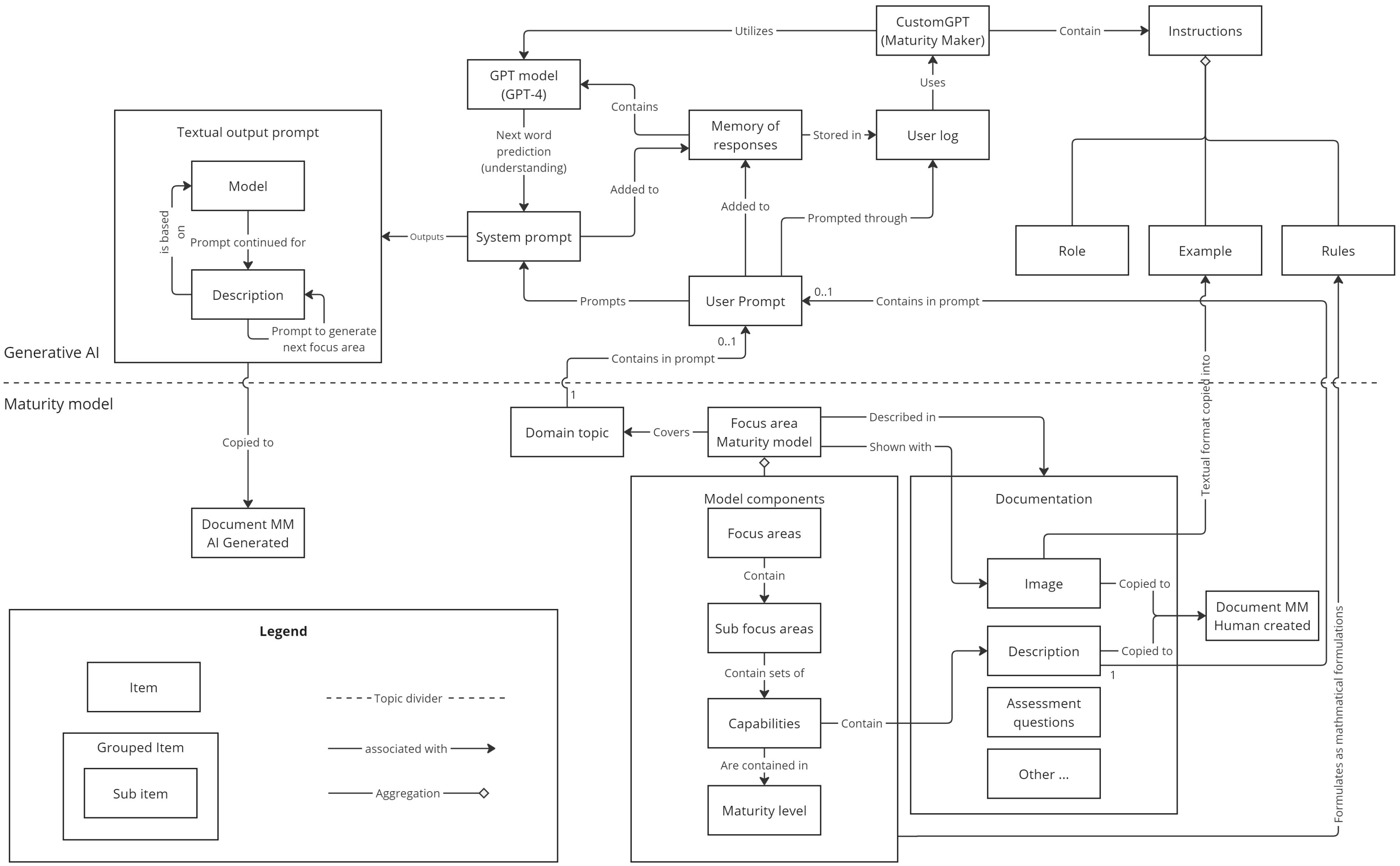


FIGURE 4.10: meta-model of the interactions between the User prompting in MaturityGPT to create a Focus Area Maturity model

Chapter 5

Findings

This chapter shows the results of the interviews. The method used in the interviews, were explained in chapter 3.2. To show the results of the interviews, there will be first a summary of the research method, as described in chapter 3. Additionally, quantitative results will be shown. Second, for each design criteria, the human created MM quantitative results will be compared to the AI-generated MM results. These quantitative results will also use the qualitative results in the comparison. Third, the results will be averaged and any notable highlights will be discussed. Fourth, the participants responses to the question ”What role, if any, do you envision GenAI will have with MM?” will be provided.

5.1 Interviews

For this research, 17 people were interviewed. These 17 people have a wide variety of professions. For example, consultant, professor or partner. The backgrounds of the participants are irrelevant to the research results. To read further upon the professions of the interviewees, check-up on appendix B. Of the 17 interviews, there were 2 interviews that had a time constraint. For those two interviews, the time for the experience and closure phase was shortened. Despite the time constraints, there was enough time to conduct the evaluation phase. All the findings collected in this research is to be found in appendix C.

5.1.1 Overview of the evaluation results

The evaluation was done with 4 minutes for the participant to observe and understand the MM, and 3 minutes of questioning. In total a participant evaluates 4 MMs. Two MMs have as domain topic ”DevOps” and the other two are about ”Software Product Portfolio Management”. Each domain topic has an MM that is AI generated and a published human created one. After all 4 MMs have been evaluated, the participant was then asked: ”Which MM do you think is AI generated and why do you think it is AI generated?”.

Each participant may have prejudice for or against AI-generated content. To prevent any bias, no participant was informed which MM was AI-generated. Which MM they saw first, was also mixed in the evaluation. Half of the participants start off with the MM that was AI generated about DevOps, whereas the other half of the participant start off with the human created MM for SPPM. Mixing these two pools is to ensure that the order, in which the MM’s are shown, will have less influence on the results. The demographics of the participants, and these pools, are not taken into account when answering the research questions. In the evaluation, it was notable that the pool that *ended* with evaluating the human created SPPM scored with a 2. This evaluating is significantly lower ratings than the group that *started* with the human created SPPM, that rated it a 3,11. The other three MMs remained roughly the same point difference (range 0,1-0,3).

The recorded results based, on the likert rating evaluation questions, are given in table 5.1 and 5.2.

How this table should be read, goes as follows. The first column contains the interviewees. [1]. The following two top columns are indicating, if the results recorded were from the AI generated FAMM or the Human created FAMM. Each of these top columns contains the 5 design criteria as their own column. These Design criteria are given in a shortened version, named as D1, D2, D3, D4 and D5. A more detailed explanation that contains the qualitative results for each design criteria will be described in sub chapter 5.3

TABLE 5.1: Results of AI Generated and Human Created MMs for DevOps

	[A] AI Generated					[B] Human created				
Participants	D1	D2	D3	D4	D5	D1	D2	D3	D4	D5
Interviewee A	4	2	5	4	5	3	2	4	3	4
Interviewee B	4	3	2	3	2	4	2	2	3	3
Interviewee C	4	4	2	3	3	2	2	3	2	2
Interviewee D	4	4	4	2	4	3	3	2	2	2
Interviewee E	3	4	2	3	4	3	3	2	2	3
Interviewee F	4	3	4	4	4	3	3	3	3	4
Interviewee G	3	3	3	3	4	4	3	4	4	4
Interviewee H	2	4	3	4	3	2	4	2	4	2
Interviewee I	3	4	2	2	5	3	4	2	2	5
Interviewee J	4	4	4	4	4	2	4	4	4	3
Interviewee K	1	3	2	2	2	2	3	4	4	4
Interviewee L	3	3	4	4	3	3	5	4	5	3
Interviewee M	4	4	5	4	5	4	5	4	5	4
Interviewee N	2	4	3	4	2	4	4	4	4	2
Interviewee O	3	4	5	4	4	4	4	4	4	4
Interviewee P	3	4	3	4	3	4	3	4	4	4

Design Criteria used for DevOps and SPPM:

Design Criteria	**Description**
D1	Completeness
D2	Ease of Use
D3	Effectiveness
D4	Operational Feasibility
D5	Usefulness

[1] A description of each of the interviewees can be found in Appendix B

TABLE 5.2: Results of AI Generated and Human Created MMs for SPPM

	[C] AI Generated					[D] Human created				
Participants	D1	D2	D3	D4	D5	D1	D2	D3	D4	D5
Interviewee A	3	2	4	4	4	2	2	2	1	2
Interviewee B	3	4	2	4	2	2	1	4	1	2
Interviewee C	1	4	2	3	3	1	2	2	2	2
Interviewee D	3	3	2	3	3	2	2	2	1	2
Interviewee E	3	4	3	3	3	1	3	3	2	2
Interviewee F	4	4	4	4	3	2	2	2	3	2
Interviewee G	1	2	3	4	2	4	2	2	2	3
Interviewee H	2	4	1	4	2	4	4	4	5	4
Interviewee I	3	4	4	2	5	3	4	2	2	4
Interviewee J	4	4	4	5	5	3	2	2	2	2
Interviewee K	5	4	4	4	4	4	4	5	5	4
Interviewee L	3	4	4	5	3	3	4	4	5	3
Interviewee M	4	3	4	4	3	2	3	2	3	2
Interviewee N	5	4	5	4	5	2	1	1	4	1
Interviewee O	3	5	4	4	5	3	4	4	3	4
Interviewee P	3	3	3	4	4	2	2	2	2	2

5.2 Comparison

This section explains the evaluation findings from the total of 16 [2] interviews conducted. Each of the following subsections will describe one of the design criteria. In each of the design criteria, both the AI-generated and human created FAMMs (DevOps and SPPM) will be addressed. Both quantitative and qualitative data are used to explain the evaluation results. First, a box plot is shown to help understand the particular spread of ratings given per FAMM. See table 5.1 & 5.2 for the full results. Below the box plot is a table showing the mean score and the standard deviation of each of the FAMMs. Following that, there is a description to explain what notable results can be seen in the collected quantitative data. Qualitative data is also included in the text to explain, for example, why participants gave certain ratings. Specific codes, which refer to a row in appendix C, are added to identify where the qualitative data comes from. At the end of the paragraph a quote from one of the participants is given. The quote is meant to complement the previous description, by directly stating what one of the participants said in relation to the design criteria. Both domain topics are given a separate table, description and quote. This was done to ensure that the domain topic was controlled for, in the results. As a result, a clear distinction in the comparison is maintained. Additionally, at the end of each subsection a short summary is given that combines both MM domain topics. The summary was given in order to generalise how well AI-generated MM compares to human created MM (in terms of the design criteria).

[2] the 17th Interviews did not contain an evaluation and was therefore excluded

5.2.1 Completeness

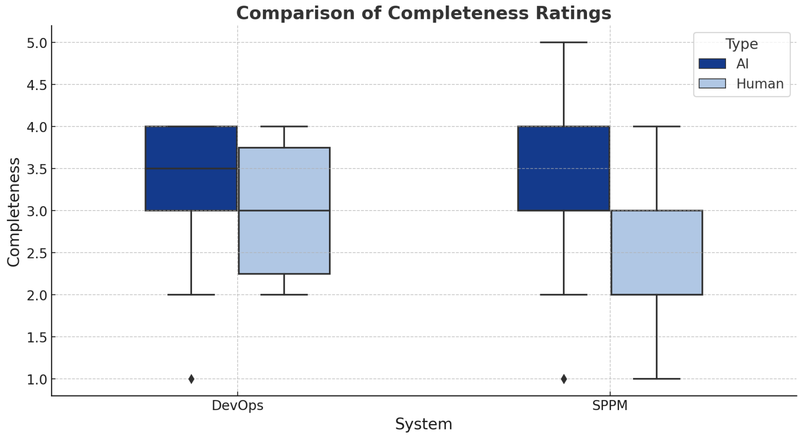


FIGURE 5.1: Boxplot Analysis of (D1) Completeness Scores: Comparative Evaluation of AI-Generated and Human-Created FAMMs Across DevOps and SPPM Domains

DevOps The completeness of the DevOps FAMMs generated by AI and those created by a human were assessed. On average, as can be read in table 5.3, the AI-generated DevOps FAMM scored higher than the human created DevOps FAMM. It is interesting to note that only the AI generated FAMM received a 1 ('strongly disagree') in the ratings, as shown in Figure 5.1. Interviewee K, who is knowledgeable about DevOps, gave the AI-generated FAMM a 1 because the interviewee said the FAMM did not take into account the social context (ev-a-1). Interviewee K made the same comment about the human created FAMM. However, they gave the human created FAMM a 2 ('slightly disagree') as it adapted slightly better based on the social context (ev-b-1). Another comment generally remarked that the AI-generated FAMM proved to be robust (ev-a-2). Despite seeming robust, the AI-generated FAMM was criticized for lacking in areas such as product management, security (ev-a-4) and collaboration (ev-a-3;ev-a-5). Similarly, the human created model was also noted to be deficient in security (ev-b-2), though it did address cross-functional team dynamics better (ev-b-3). The evaluation did not show any significant differences in completeness.

> *"The most important thing in DevOps is cross functional. And I don't think that shared responsibility really addresses that"* (ev-a-3)
> – Interviewee N, (rated a 2) reflecting on document A [3]

Group	Mean	Standard Deviation (std)
DevOps AI	3.21	0.97
DevOps Human	3.00	0.78

TABLE 5.3: Comparison of Mean and Standard Deviation between DevOps AI and Human Evaluations

[3] Responses in the interviews originally in Dutch have been translated into English.

SPPM The average score in table 5.4 for the AI-generated SPPM FAMM showed a more significant difference compared to the previous DevOps average. This was partly due to the fact that the human created FAMM scored lower, while the AI-generated FAMM scored about the same. Figure 5.1 shows that the human created FAMM scored with more variable results, ranging from a 1 to a 5. Interestingly, some participants said that the AI variant was presented in a less specific (ev-c-1;ev-c-2) and more corporate viewpoint (ev-c-3), in terms of the content in the FAMM. For the human created FAMM, participants said it was very thin(ev-d-3) and lacked a lot of detail (ev-d-2). Often these comments were accompanied by suggestions about should have been included in the documentation. They also often claimed that it contained vague (ev-d-1) and unrelated content (ev-d-4) that they felt should not be included in an SPPM FAMM. These findings highlight the importance of having recognizable text in order to achieve a higher score for completeness.

> *"I like this one better, seems to have a better evolution from 1 to 10. . . . I think this has a lot of good ideas"* (ev-c-13)
> – Interviewee O, (rated a 3) reflecting on document C

Group	Mean	Standard Deviation (std)
SPPM AI	3.21	1.25
SPPM Human	2.50	1.02

TABLE 5.4: Comparison of Mean and Standard Deviation between SPPM AI and Human Evaluations

Summary: In summary, for the design criteria 'completeness', AI-generated FAMMs generally outperformed their human created variants. However, when experts began to review the documentation, a common comment was that there was not enough information. Specific feedback pointed to the need for a clearer 'desired state' and the inclusion of 'additional literature'. These were some of the reasons why participants gave lower scores.

5.2.2 Ease of Use

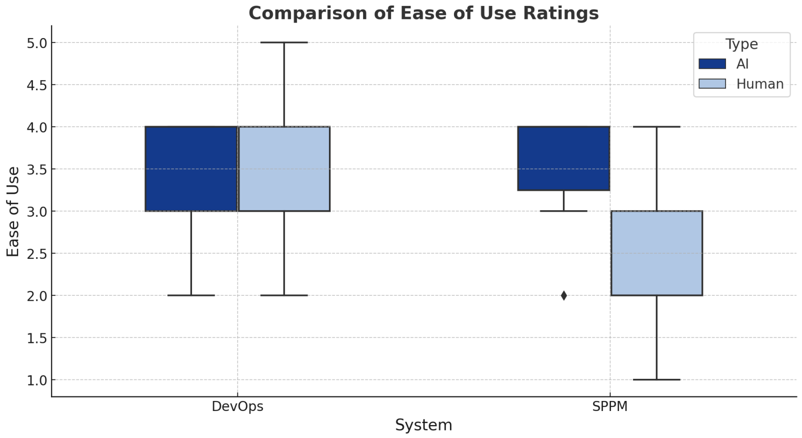


FIGURE 5.2: Boxplot Analysis of (D2) Ease of Use Scores: Comparative Evaluation of AI-Generated and Human-Created FAMMs Across DevOps and SPPM Domains

DevOps: Ease of use was assessed and table 5.5 shows that the AI-generated FAMM scored a mean of 3.56. This is slightly higher than the human created FAMM which scored 3.44. Further analysis of the results shows, as can be seen in figure 5.2, that the AI-generated FAMM received a consistent rating between 2 and 4. Which indicates a modest ease of use. It is important to note that the human created FAMM received two ratings of '5' (very easy). These ratings were given by interviewees M and N, who commented that they liked the additional documentation that the human created FAMM had (ev-b-4;ev-b-5). Although both participants were not fully knowledgeable about the domain, these comments hint to a correlation between the amount of documentation and the interviewees ratings. The most striking comment from the data is the criticism of the AI generated FAMM, specifically for sub-focus area 5.1 (process optimisation). Within this sub-focus area, capabilities (such as, Kaizen and Lean) were considered to be have an insufficient amount of information provided (ev-a-6) or overly complex (ev-a-7). The criticism is well founded, as the Kaizen methodology, for example, is too large to be considered as a single capability to be implemented. Despite the valid criticism of an AI-generated error, the AI-generated FAMM was still rated higher for ease of use. Which could be partly because the human created FAMM was noted for its complexity and vagueness, particularly in some of the measurement criteria (ev-b-6;ev-b-8). This may have led to a lower rating for ease of use. It was further commented that the human created FAMM tended to be created from a lower level viewpoint (ev-b-7). The general response was that such a viewpoint could be beneficial or counter-intuitive depending on the specific person using the FAMM. Overall, the human created FAMM was more often commented as too complex and vague.

> *"Too much in definition form. . . . I'd have to search additional information up. The action points were better, as 'now we have to fix things'."* (ev-a-6)
> – Interviewee K, (rated a 2) reflecting on document A

Group	Mean	Standard Deviation (std)
DevOps AI	3.56	0.63
DevOps Human	3.44	0.96

TABLE 5.5: Comparison of Mean and Standard Deviation between DevOps AI and Human Evaluations

SPPM: When comparing the AI-generated SPPM FAMM to the human variant, the AI-generated SPPM FAMM, again, scores significantly higher than the human variant. Table 5.6 shows that the mean score for the AI-generated FAMM is 3.63, while the human variant scores 2.5. Figure 5.2 also shows that the human SPPM FAMM was the only FAMM to receive a 1 ('not easy'). This 1 was given several times, with Interviewee E being one of those. Interviewee E stated in their reasoning that a 1 was given due to the lack of concrete terms in the capabilities (ev-d-5). Other comments, criticising the human created FAMM, often refer to the lack of documentation (ev-d-6). It was mainly commented that the capability descriptions were very thin. This makes it difficult for respondents to see the FAMM as "easy to use" when applying the model. The most striking results come in contrast to the human created FAMM. The AI-generated FAMM was praised as being easier to follow due to a clearer maturation path(ev-c-4). This is a significant finding. The finding provides compelling evidence that GenAI could play a role in the development of MM through its ability to capture a maturation pathway.

> *"Two sentences, as a description, is fine to start off with. But the moment when you(organization) are on a higher maturity, then this would no longer be usable"* (ev-d-6)
> – Interviewee H, (rated a 2) reflecting on document D

Group	Mean	Standard Deviation (std)
SPPM AI	3.63	0.81
SPPM Human	2.50	1.15

TABLE 5.6: Comparison of Mean and Standard Deviation between SPPM AI and Human Evaluations

Summary: In summary, for the "ease of use" design criterion, the AI-generated FAMMs were rated higher in ease of use than the human created FAMMs. The most notable result that emerged from the qualitative feedback, on this design criterion, was that some participants claimed that the AI-generated FAMM was able to capture maturity correctly. This finding reinforces the belief that GenAI could play a role in MM development. However, the AI generation of an MM was shown to have some notable errors in capability placement, which were highlighted in the 'ease of use' design criteria of the DevOps FAMM.

5.2.3 Effectiveness

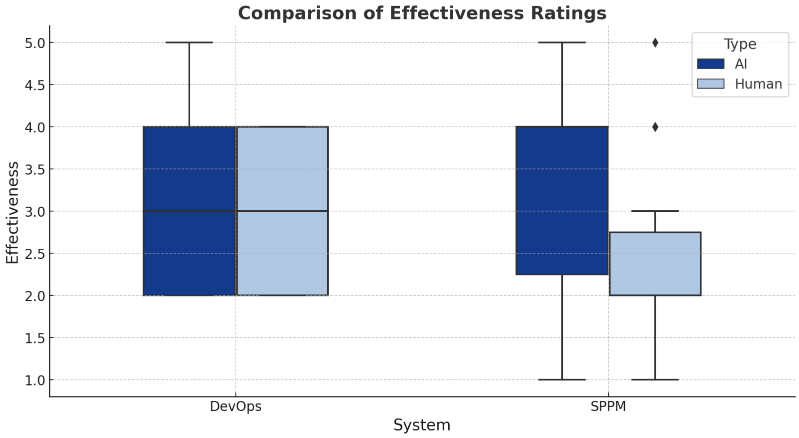


FIGURE 5.3: Boxplot Analysis of (D3) Effectiveness Scores: Comparative Evaluation of AI-Generated and Human-Created FAMMs Across DevOps and SPPM Domains

DevOps The effectiveness of the FAMMs was evaluated. As can be read in table 5.7, the evaluations showed that the AI-generated DevOps received a higher mean of 3.31 compared to the human created DevOps FAMM, which received a 3.19. Figure 5.3 also shows that the AI-generated variant received a 5 ('very effective'). Further analysis showed that the AI-generated variant was rated three times: 'very effective'. One of these was given by Interviewee O, who is very knowledgeable about DevOps. Interviewee O praised the AI-generated variant, as it provided a clear starting point and a solid foundation on which a potential user of the FAMM model could build (ev-a-8). However, the AI-variant was not without its critics. One such criticism was that the AI-generated FAMM was too much from a technical perspective (ev-a-9). This perspective would be appropriate for ICT staff, but would not be useful if a manager were to use this FAMM for an assessment. Further comments on the AI-generated FAMM were generally that the FAMM contained a sufficient amount of skills (ev-a-10). However, it was suggested that a more detailed description would have been beneficial (ev-a-11). For the human created FAMM, respondents were more complimentary of the FAMM's organisational focus (ev-b-9) and emphasis on social skills (ev-b-10). Despite the compliments on including the social context, the AI-generated FAMM still scored higher in terms of effectiveness.

> *"I interpret this question as how can they(organization) use this model to identify the next course of actions. This is done with a FAMM, which handles this well. Expecting that they can fill in the assessment, then that would sort itself out. .. It(documentation) does not state 'how', otherwise it would have been a 'very effective'"* (ev-a-16)
> – Interviewee D, (rated a 4) reflecting on document A

SPPM The effectiveness ratings, as shown in table 5.8, also showed that for SPPM, the AI-generated variant scored higher. Based on the mean, the AI-generated SPPM FAMM scored 3.31, while the human created SPPM FAMM scored 2.5. It is interesting to note that these scores have exactly the same mean as previously reported for the 'completeness' design criterion. As with the distribution of ratings, both FAMMs

Group	Mean	Standard Deviation (std)
DevOps AI	3.31	1.14
DevOps Human	3.19	0.91

TABLE 5.7: Comparison of Mean and Standard Deviation between DevOps AI and Human Evaluations

share a range from 1 to 5. One of the most interesting ratings was given by interviewee N, who gave the human created FAMM a 1 ('not effective') and the AI variant a 5 ('very effective'). Interviewee N's reasoning was that the AI-generated FAMM was more effective in capturing the basic ideas through a technical lens, providing a clear evolutionary step (ev-c-5). This comment reinforces an already commented on potential (see design criteria 2), that GenAI is able to capture the required functionality of understanding maturity. A key functionality that would be required, if GenAI were given the role of developing an MM. For the human created SPPM, participants noted that it lacked clarity. Specifically, that clarification was needed as to what the starting points for the sub-focus areas were (ev-d-7). The AI-generated SPPM, on the other hand, was able to document certain essential aspects. For example, the aspect of facilitating productive discussions about the current state of the company's SPPM (ev-c-6). The fact that a FAMM formed productive discussions, after an assessment, was often an important comment as a reason for the AI-generated FAMM scoring higher on 'effectiveness'.

> *"Feeling a '1', because I think if the company cannot get anything out of it. They can just fill the boxes, but yeah. But, what can you do beyond that, what are we missing? ... What protocols and how do I do this then"* (ev-d-10)
> – Interviewee B, (rated a 1) reflecting on document D

Group	Mean	Standard Deviation (std)
SPPM AI	3.31	1.08
SPPM Human	2.50	1.15

TABLE 5.8: Comparison of Mean and Standard Deviation between SPPM AI and Human Evaluations

Summary: In summary, for the design criterion 'Effectiveness', the AI-generated FAMMs still show a higher mean score than the human variant. The AI-generated FAMM is complimented for capturing the technical aspects of the capabilities, but tends to overlook the 'people' aspect. The 'people' aspect refers to the social dynamics of the stakeholders who will be using the capabilities. Which is critical to consider when progressing through the maturity levels. This oversight is similar to the criticism given to CMMI in the past. When CMM was changed from a less organisational focus to CMMI, a more capability and therefore technical focus.

5.2.4 Operational Feasibility

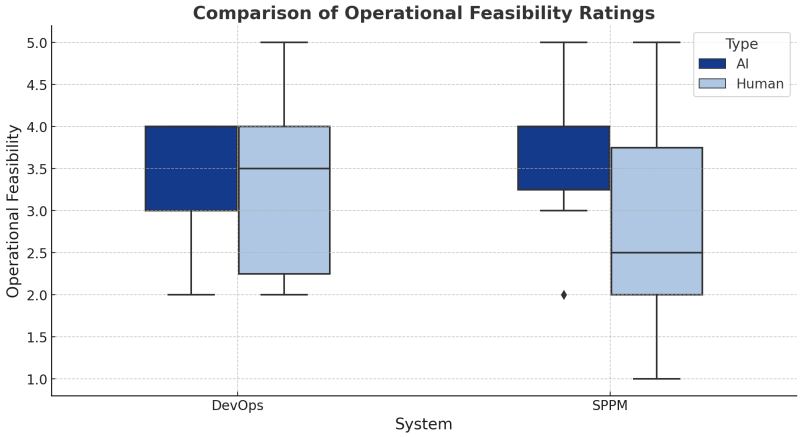


FIGURE 5.4: Boxplot Analysis of (D4) Operational Feasibility: Comparative Evaluation of AI-Generated and Human-Created FAMMs Across DevOps and SPPM Domains

DevOps: Next, ”operational feasibility” was assessed. Based on the mean scores, the human created DevOps FAMMs scored slightly higher than the AI-generated FAMMs. As can be seen from the table 5.9 The AI-generated DevOps FAMM had a mean of 3.38, while the human created variant had a mean of 3.44. Notably, this is the only case where the human created variant scored higher than the AI-generated variant. Figure 5.4 also shows that the human created Devops FAMM was the only variant to receive a 5 (’very feasible’) as a rating. In total, two ’very feasible’ ratings were given to the human-created DevOps FAMM. One was given by interviewee L, who commented that the human variant seemed very straightforward (ev-b-11). The other rating was given by Interviewee M, who praised the FAMM for its incremental approach (ev-b-12). Criticisms, often made about the lower scores of the AI-generated variant, frequently mentioned that: if the AI-generated DevOps FAMM were to be put into practice, then there would be complications due to Focus Area 5.1 (ev-a-12). These comments are related to the same criticisms mentioned in design criterion 2.

> *”This should be able to be applied in the organization with enough pointers. What should be considered, but depends on the maturity of the company, is having the pointers indicate a ’to’. Okay: ’that are the options’ and ’where can I find the documentation’ ... Blue/green, fun that it’s written that way, but what does it mean and how am I that.”* (ev-a-17)
> – Interviewee B, (rated a 3) reflecting on document A

Group	Mean	Standard Deviation (std)
DevOps AI	3.38	0.81
DevOps Human	3.44	1.03

TABLE 5.9: Comparison of Mean and Standard Deviation between SPPM AI and Human Evaluations

SPPM: For the SPPM FAMMs, table 5.10 shows that the AI generated SPPM FAMM has a score of 3.81 while the human created variant has a score of 2.94. Notable is that the AI-generated SPPM FAMM has the highest mean design criteria score of all the variants evaluated. Comments on the AI-generated variant range from it being concrete (ev-c-7), to it being integratable (ev-c-8), to it having "enough juice" (ev-c-9) to be effective. Interestingly, the human created SPPM FAMM also scored higher on average compared to the previous design criteria. Despite the higher rating of the human created FAMM, there was still criticism, whereof one interviewee referred to the FAMM being too ambiguous for the user to understand (ev-d-8). Interviewees noted that the documentation left many questions unanswered, which affected the perceived feasibility of use.

> *"You need like a database of best practices to more fill in what should be done, otherwise it would stay with high level statements. The challenge is often 'what should I actually do once the assessment is done'"* (ev-d-11)
> – Interviewee P, (rated a 2) reflecting on document D

Group	Mean	Standard Deviation (std)
SPPM AI	3.81	0.75
SPPM Human	2.94	1.34

TABLE 5.10: Comparison of Mean and Standard Deviation between SPPM AI and Human Evaluations

Summary: In summary, the AI-generated FAMM scored very well on the 'operational feasibility' design criteria. Especially, when considering the results from the SPPM evaluation. On average, the human created FAMMs scored lower than the AI-generated variant. However, for the DevOps evaluation, the human created FAMM did score higher than the AI-generated variant and was also praised for its clarity and integratable. For the SPPM, the human created FAMM scored lower due to its ambiguity, highlighting the advantage that the AI-generated FAMM could have in providing a clearer, more understandable FAMM.

5.2.5 Usefulness

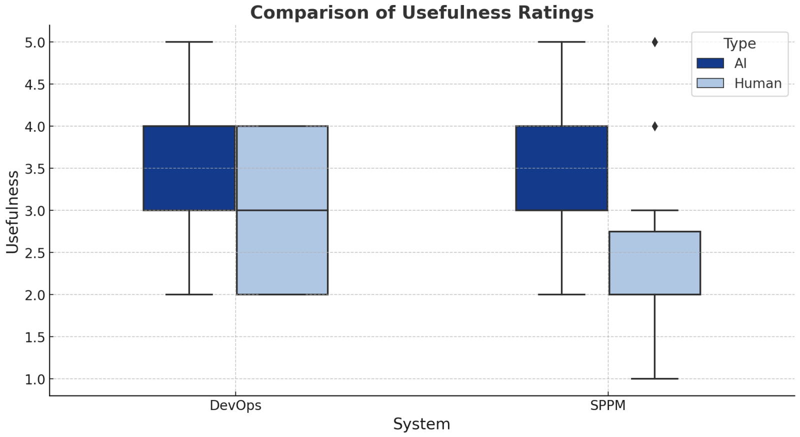


FIGURE 5.5: Boxplot Analysis of (D5) Usefulness Scores: Comparative Evaluation of AI-Generated and Human-Created FAMMs Across DevOps and SPPM Domains

DevOps The last design criteria evaluated was 'usefulness'. Table 5.11 shows that for Usefulness, the AI-generated DevOps FAMM received a mean score of 3.63, which is slightly higher than the human-created FAMM with a score of 3.31. Interestingly, Interviewee N is knowledgeable about DevOps and was one of the participants, shown in Figure 5.5, who gave both FAMMs a score of 2 ('Slightly Useful'). Interviewee N commented that there are better sources of information, such as case studies, than the two FAMMs shown (ev-a-13). This comment emphasises that there could be more useful manners of presenting MMs. Alongside the comment, the AI-generated FAMM was generally praised as being easier to use and therefore more useful (ev-a-14;ev-a-15). Both FAMMs were found to be effective in starting productive discussions (ev-b-13).

> *"I think that there is more that is needed, to be really helpful, than the 13 pages that I see here. If I was looking for something I would find very useful, then I would pick up a book or a case study on devops, and figure out how I would do that, than use this model. ... Still 'what' level material, not 'how' level material"* (ev-a-13)
> – Interviewee N, (rated a 2) reflecting on document A

Group	Mean	Standard Deviation (std)
DevOps AI	3.63	1.02
DevOps Human	3.31	0.95

TABLE 5.11: Comparison of Mean and Standard Deviation between DevOps AI and Human Evaluations

SPPM For the SPPM FAMMs, table 5.12 shows that the AI-generated SPPM scored a full point higher than the human created FAMM. The AI-generated FAMM scored an average of 3.5, while the human created FAMM scored an average of 2.5. Notably, interviewees often commented that the human variant lacked clear instructions. These missing instructions specifically related to guiding (of what next steps needed to be taken to reach a higher level of maturity) a potential user of the model (ev-d-9). Contrary to this, the AI variant was praised for providing a more logical path (ev-c-10;ev-c-11). However, it was also commented that the AI generated pathway was seen as somewhat rigid and did not offer much flexibility in how a higher maturity could be achieved (ev-c-12).

> *"It got a nice progression of activities. For example in compliance, once you are aware of the compliance, it builds a way, anticipating what you have to next comply with. That kind of evolution seems much better"* (ev-c-11)
> – Interviewee O, (rated a 5) reflecting on document C

Group	**Mean**	**Standard Deviation (std)**
SPPM AI	3.5	1.096
SPPM Human	2.5	1.03

TABLE 5.12: Comparison of Mean and Standard Deviation between SPPM AI and Human Evaluations

Summary In summary, both AI-generated FAMMs were rated higher in terms of usefulness than the human created FAMMs for the design criterion 'usefulness'. However, neither FAMM provided sufficient guidance. This criticism could largely be attributed to the lack of additional information in the documents presented in the evaluation. The lack of information could possibly have been provided by the additional information from the original FAMM. However, that information was excluded because only the FAMM and the descriptions of capabilities within the FAMM were controlled for in the comparative analysis. Guidelines, such as the assessment questions, could have been added. Nevertheless, this would not have been possible due to time constraints. The same could be said for the AI-generated variant, where additional information could have been generated.

5.3 Evaluation analysis

This section discusses the overall quantified results gathered from the evaluations. While the previous section went into details for each of the design criteria, here only the highlights will be described. It begins by highlighting the main findings from the comparative analysis. Subsequently a discussion is provided of the accuracy of participants in guessing which FAMM was generated by AI. These results will also be discussed in more detail, in order to explain how accuracy might affect the overall results. Finally, possible implications will be drawn from the results.

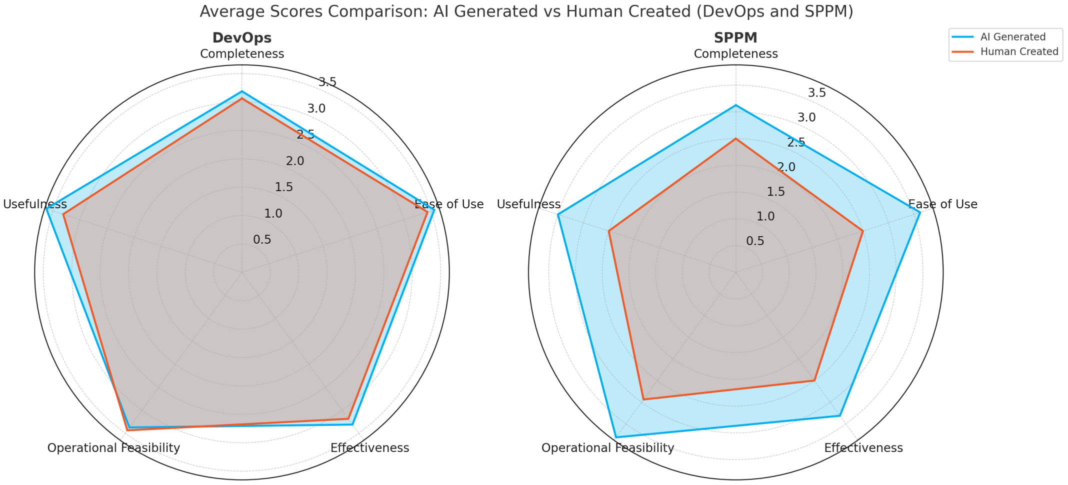


FIGURE 5.6: Radar Chart Comparison of Average Design Criteria Scores Across AI-Generated and Human-Created FAMMs in DevOps and SPPM Domains

Of the 16 conducted evaluations, the results show that the AI-generated FAMMs received higher scores than the human created FAMMs. Figure 5.6 shows a radar diagram with the quantitative results of the evaluations. Intriguingly, only the 'Operational Feasibility' design criterion for the human created DevOp FAMM showed a higher score than the AI-generated variant. When comparing the overall average, the AI-generated variant scored 0.51[4] higher. See table 5.13 for the full breakdown of means and variances. On average, the human created FAMM scored 2.94, while the AI-generated variants scored 3.44. Further analysis of the data showed that the two AI-generated FAMMs were almost identical on average. The small variance of only 0.07 is strong evidence that both AI-generated FAMMs are of approximately the same quality (based on the design criteria). On the contrary, the human created FAMMs show, their variance of 0.7 is much higher.

[4] This and the following results are rounded up to two decimal places. It is important to note that the rounded up variance results in a discrepancy of 0.01 when subtracting the mean of the AI-generated from the human created FAMMs

Domain Topic	AI-Generated	Human-created	Variance
DevOps	3.4125	3.2875	0.125
SPPM	3.475	2.5875	0.8875
Variance	0.0625	0.7	
MEAN	**3.44375**	**2.9375**	**0.50625**

TABLE 5.13: Comparison of Mean Scores and Variance Between AI-Generated and Human-Created FAMMs in DevOps and SPPM Domains

It was also important to check whether participants were able to distinguish the AI generated FAMM from the human created FAMMs. As previously discussed in chapter 3, participants might have a bias when rating AI-generated content. Therefore, it was chosen to check whether this bias could influence the results. From the results collected, strong evidence was found that AI bias did not play a role in influencing the results. Given that, when participants were asked which FAMM was AI-generated, the results show that they could not tell the difference. For the SPPM, participants correctly guessed whether it was AI-generated 44% of the time. Whereas for the DevOps, participants guessed correctly 50% of the time. Participants often commented that they were unsure of their guess. Table 5.14 provides a summarised reason why participants thought the FAMM was AI generated. The reasons, given in the table, were generalised. See Appendix B for the specific reasons [5] each participant gave, for guessing which FAMM was AI-generated.

	Wrong	Correct	Accuracy
DevOps	8	8	50%
Reasons	lvl 1 Stuffed full (ev-aig-9), classical style AI would copy (ev-aig-10), more information (ev-aig-11)	Buzzwords (ev-aig-12), concrete terminology (ev-aig-13), illogical structure (ev-aig-14), unfinished text (ev-aig-15)	
SPPM	7	9	44%
Reasons	More structure (ev-aig-6), illogical (ev-aig-7), terminology (ev-aig-5), misunderstand concept (ev-aig-8)	Generic (ev-aig-1;(ev-aig-2), more robust (ev-aig-3), more spread out (ev-aig-4)	

TABLE 5.14: Accuracy and Reasoning in Participant Identification of AI-Generated Versus Human-Created FAMMs for DevOps and SPPM Domain

The results of AI-generated FAMMs seem to indicate that they are at about the same level (DevOps FAMM), or even better (SPPM FAMM), than human-created and approved FAMMs. In addition, participants have commented that AI generated

[5] Codes, found in C, were added to be able to retrace from where the generalized reasons originate from

FAMMs are able to capture maturity, but may suffer from errors that stem from its AI generation. Correcting an error in the AI generation would still require a human reviewer in the development cycle. Although not analysed in detail, domain experts were more likely to approve the AI-generated FAMMs than the human created FAMMs. The findings that AI-generated FAMMs perform better overall is a significant finding for the question: 'Could AI play a role in the development of MMs?'. This is because the development of MM would take significantly longer, if only a human were to create an MM. Surprisingly, both domain and non-domain experts had a hard time distinguishing which FAMM was generated by AI. The overall results showed that participants were more likely to guess incorrectly, which FAMM was AI-generated. Further analysis showed that the variance between the two FAMMs was low. Given that these findings are based on only two AI-generated FAMMs, the results of such an analysis should therefore be treated with considerable caution. Nevertheless, these results provide compelling evidence that if a new AI-generated FAMM of a different domain topic were to be created, that AI-generated FAMM would be of approximately the same quality as the AI-generated FAMMs presented.

Chapter 6

Analysis

Within this chapter, an analysis of the participants expectations to the question: ”What role do you envision that GenAI could play in MMs?” is shown. Sub-chapter 6.1 explains how the data of the expectations were collected, categorized and combined. Next, sub-chapter 6.2 shows all the envisioned roles given by the participants. These roles are plotted on Mettler's MM life cycle. For extra clarity, a description of each phase, of Mettler's MM life cycle, is also provided.

6.1 Through the Three C's

The data for the participants expectations went through the three C's.

1. The first C is 'Collect'. Every expectation was collected and can be found in appendix C. It is worth mentioning that, within appendix C, each expectation has been summarized to a single word or sentence, by the author. This was with the aim of getting to the essence of the elaborate answers, that the participants gave.

2. The second C is 'Categorize'. To categorize the collected information, the author divided the expectations to 'user viewpoint' expectations and 'developer viewpoint' expectations. Furthermore, every expectation was categorized in line with one of the phases from Mettler's MM Life Cycle. Each rationalisation for each categorization can be found in appendix C. Categorising was needed to identify which specific phase shows the highest expected role for GenAI.

3. The last C is 'Combine'. To combine the expectations, the author identified which expectations had overlap. If two (or more) expectations, in essence, expressed the same expectation, then those were combined.

6.2 Role expectations of genAI in the life cycle

This subchapter gives an analysis of all the participants expectations given. We will starts with showing figure 6.1 wherein Mettler's life cycle is displayed. In the figure, for each phase the amount of expectations are indicated. If, within a phase, a specific role was envisioned two (or more) times, then this role would be noted. Following that, each viewpoint is separately discussed. Both viewpoints start wherein all the expectations, which have already been categorized and combined, are stated for each phase. As can be seen in table 6.1 (developer viewpoint [1]) and 6.2 (user viewpoint [2]), 'Categorize' and 'Combine' together show which specific roles yielded

[1] Developer cycle of Mettler's MM life cycle

[2] Application cycle of Mettler's MM life cycle

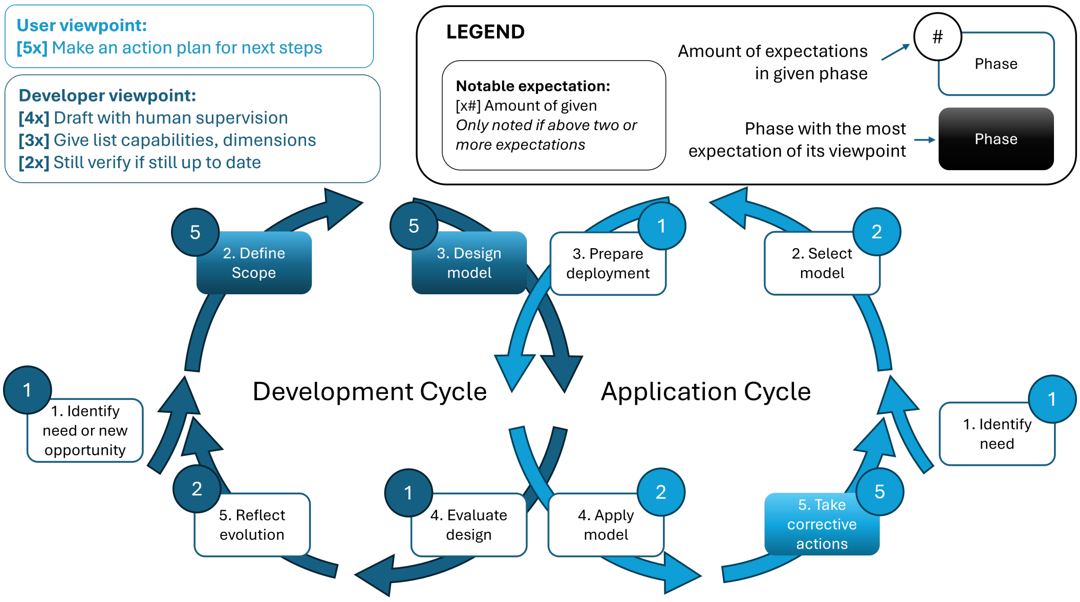

User viewpoint:
[5x] Make an action plan for next steps
Developer viewpoint:
[4x] Draft with human supervision
[3x] Give list capabilities, dimensions
[2x] Still verify if still up to date
LEGEND
Notable expectation:
[x#] Amount of given
Only noted if above two or more expectations
Amount of expectations in given phase
#
Phase
Phase with the most expectation of its viewpoint
Phase
Development Cycle
Application Cycle
1
1. Identify need or new opportunity
5
2. Define Scope
5
3. Design model
1
4. Evaluate design
2
5. Reflect evolution
1
1. Identify need
2
2. Select model
1
3. Prepare deployment
2
4. Apply model
5
5. Take corrective actions

FIGURE 6.1: Mettler's Life Cycle with Participant Expectations: The figure highlights the phases with the most received expecations and names the ones that were given multiple times.

the highest expectation. The top of the tables are the phases of each viewpoint. Under every phase the different expectations were given in a summarized fashion. The number in front of the expectations is the amount of participants that gave the expectation. Additionally, in the parentheses, at the end of each expectation, a code can be found which links to C.

6.2.1 Expectations for the Developer viewpoint

Identify opportunity	Scope	Design	Evaluation	Reflect
(1) Inspiration (c-d-9)	[3] Give list capabilities, dimensions (c-d-5;c-d-8;c-d-12)	[4] Draft with human supervision (c-d-1;c-d-6;c-d-10;c-d-14)	[1] Use case generation tests (c-d-5)	[2] Verify if still up to date (c-d-7;c-d-11)
	[1] Condense gathered info (c-d-2)	[1] Draft creation as it can be personalized (c-d-1)		
	[1] Separate AI/creativity into dimension (c-d-13)			

TABLE 6.1: Envisioned Roles for GenAI in the Phases of Maturity Model Development (Developer Viewpoint)

identify opportunity: An expectation was given for the first phase, where GenAI could serve as inspiration for the creation of a FAMM.

(Most suggested) Scope: This and the next phase (design), received the most expectations. In total, 10 out of the 14 expectations were made in the Scope and Design phases. Both phases are closely related to the information development in an MM. It is important to note that almost all participants clarified, that the information generated should first be validated by a domain expert or MM-developer. With regard to the expectations given, 3 were related to giving a list of capabilities that could serve as a basis for the model. Interviewee I, who made one of these three expectations, referred to the expectation as: "checking that nothing is missing in a lecture". Interviewee I explained that as a professor, before a lecture would start, he would prompt to GenAI how the lecture should be structured. Following that, GenAI's response would then be used to compare how lecture was initially setup. If GenAI's Lecture indicated gaps (that are too high importance) from the initial lecture structure, adjustments would be made. It is advised to have a goal when generating a list of capabilities. As stated in the paper 'A critical Review of Maturity Model Development ...', the following was said 'To successfully define the scope, the reason and relevance of the model must be defined clearly' (Adekunle et al., 2022). In addition to a list of capabilities that could be generated, there was also an expectation to condense all the information gathered, for a potential MM, (e.g. the interviews conducted for an MM) into a clear format. This would make it easier to see, which priority issues should be included in an MM. The final expectation was to include AI as a separate dimension of maturity.

(Most suggested) Design: As mentioned in the previous phase, participants stressed the importance of human supervision in the use AI-generated content. 4 of the 5 expectations indicated that AI-generating a draft could play a role, but only if it is done under human supervision. While only one of these five expectations mentioned that the reason why genAI could play a role, in the generation of an MM, is that it can be completely personalised for a specific topic. When looking into validations methods for AI systems, it does come up that expert opinion are rarely reported method Myllyaho et al., 2021. Further work can be researched whether Simulation, trial or model-centered could be a more appropriate validation method for AI-generated MMs.

Evaluation: Once an MM has been created, it needs to be tested. Only one expectation was made in the evaluation phase for an MM. The expectation is to use AI-generated use-cases to evaluate an MM.

Reflect: The last stage is to reflect if the MM is still up to date with changes in the market. Two similar expectations were made. The expectation goes as follows: An MM would be given to a GenAI, which would then assess, if it is still up to date or suggest if the MM needs to be changed. This phase is particularly relevant for the participants in this study, who created their own MM. Often, in the semi-structured interview, the participant was asked if their MM was still up to date. The most common answer being no. Previous papers have evaluated whether certain maturity models revision (Muller and Hart, 2016). However, these revisions take a significant effort. Added to that, these revisions would only stay temporarily applicable.

6.2.2 Expectations on User viewpoint:

Identify need	Select model	Prepare deployment	Apply model	Take corrective action
(1) Identify starting point related to problem (c-u-9)	[1] Find MM (c-u-10)	[1] Bind assessment tool by using AI (c-u-4)	[1] Let genAI assist as copilot in creating MM (c-u-11)	[5] Make an action plan for next steps (c-u-1;c-u-2;c-u-5;c-u-7;c-u-8)
	[1] Create MM to skip search process (c-u-3)		[1] Generate extra documentation to understand MM (c-u-6)	

TABLE 6.2: Envisioned Roles for GenAI in the Phases of Maturity Model Application (User Viewpoint)

Identify need: The application cycle starts with the phase of identifying if there is a need. An expectation was made, where GenAI can play a role in identifying a starting point. This would be done by first describing the context and problem of the business to GenAI. Following this, GenAI can describe how the user should deal with the given information by provide advice.

Select model: Once the need has been identified, an MM could be selected. Two expectations were made at this stage. The first expectation was to let GenAI do the search process of finding an MM. This could be done by prompting, in GenAI, the business context that the user is currently in. Following this, GenAI can search in the existing pools of created MMs and suggest the most appropriate MM to apply. The second expectation is to skip the search process of finding an existing MM entirely. Instead the user can let GenAI create an MM based on the context that it was prompted for. It is imported to note that generating an MM from the user's point of view, differs from the developer's point of view. This is because the user would use the generated MM as an already definitive MM. The MM that would be generated, would *not* also go through the process of human evaluation.

Prepare deployment: Before a model is applied, the necessary preparation for an assessment needs to be done. One expectation for preparing for use, was to use GenAI as a tool to generate assessment questions. These assessment questions can help the user with the process of evaluating what capabilities have been implemented.

Apply model: Two expectations were made for applying the model. The first is to use a GenAI as a co-pilot to guide a user through applying an MM. The second is that, whenever a user does not find sufficient information in the descriptions of an MM, the AI could generate additional documentation.

(Most suggested) Take corrective action: Most of the participants' expectations were made in the last phase. Of the 11 expectations made in the user perspective, 5 suggested that GenAI should be used to create an action plan on what next steps a user should take. An action plan should help the user make sense of what next capabilities would be most beneficial, to reach a higher level of maturity. Design principles for prescriptive purpose of use can be followed Pöppelbuß and Röglinger, 2011. The design principles "decision calculus for selecting improvement measures", which helps the user in the appropriate corrective actions. There are papers that report on different approaches for deciding the next corrective action. An example was in a paper by Burnstein et al., 1998, wherein stated a separate assessment and action planning team was created. For the action planning, a group of experts were gathered to decide in a workshop what the next corrective actions should be. The literature found no approach which would be most suitable in taking corrective actions.

Chapter 7

Discussion

The following chapter is aimed at reflecting on the method and findings of the research. Each sub chapter includes a quote from a Bill Curtis (lead creator of the CMM). These quotes were gathered from an interview the author had with professor Curtis, whereof each quote is related to a sub chapter. For the first sub chapter discusses the possible implications of the findings for the industry. Second, the scientific contribution this paper has, are stated. Third, the validities of the research are called into question. Fourth, the limitations of the research are considered. The last sub chapter proposes what future work could be conducted.

7.1 Industry implications

> "... eventually DoD dropped it (CMMI), DoD said. You know, we're not getting the benefit from this. We've got these guys that claim they're level 5, but frankly their work isn't that much better. Because it became a compliance model."
>
> *Bill Curtis*

As stated in the quote above, Bill Curtis referred to CMMI becoming a compliance model. Professor Curtis goes on to mention that the CMMI became like an ISO standard, by following the purpose of a compliance model. Certain standards were being pursued (or maintained) to get a contract. Whereas, the essence of an MM is, meeting certain standards *and* improving a certain category. This paper wants to evade the continuation of creating MMs that are used as compliance models. Instead, the research is meant to give the industry a tool, where personalized MMs can be created, in order to fuel continuous improvement. Continuous improvements, whereby next quality improvement steps become clear by using an AI-generated MM. With the tool, Quality management no longer has to worry about staying still by solely upholding standards. Quality management can now realise what is in their name, Managing and *improving* quality. At a baseline, the AI-generated MMs provide organizations with a brainstorm tool. Providing them with a quickly made framework to reflect on where potential gaps might lie and could be filled.

7.2 Scientific Contributions

> "quality became a dirty word ... We talk about risks, today when I give talks, I start with 'We're in the era of nine digit defects' and they say 'Is that bits or bytes?'. I said 'No, that's dollars and euro'."
>
> *Bill Curtis*

With this paper, the first steps in AI-generated MMs are taken. A breakdown is given of how GenAI takes these steps in chapter 4.3. Despite this being done from a top down view, the steps described can be passed over as a torch to guide other researchers. As of now, the first data, showing the potential of AI-generated MMs, has been collected. The data, and the methods described on how to collect the data, serve as a snapshot into the prospect of GenAI in MMs. Not only does the data serve MM research, it also serves to show potential for GenAI in models from other information science domains. Similar methods, as described in chapter 3.1, can be followed to compare human created models with AI variants. To bring us back to MMs, the methodology of the framework design (given in chapter 4.1) can also be tested in different arches of science. For example, within organizational psychology, an MM can now easily be instantaneously created fora topic like preventing burnouts. An GenAI MM variant of a model, such as JD-R model of Arnold Bakker (Demerouti et al., 2001), could be generated. Subsequently, the AI-variant could be empirically assessed. With the empirical data it could be uncovered whether the created variant shows a lower percentage of burn-outs for each higher achieved level of maturity. This is just one of the many opportunities within scientific understanding ahead of us.

7.3 Validity

In this subchapter, I am going to discuss the validity of this research. The reason for this, is that GenAI is rapidly growing and the concept of using GenAI with MM's is relatively new. The validities I will discuss are: Construct validity, which describes whether the measurements accurately reflect the theoretical constructs. Internal validity, which examines whether any preconceived bias might affect the results. External validity, which is whether the results of this study can be generalised to other MMs. Whilst other validity's are important considerations in the research design, they are not relevant to the purpose of the research.

7.3.1 Construct validity

> "to really build a good maturity model, you've got to know the precedence order of practices ... and that's a problem with most maturity models is they just start putting practices in at levels and some cases they're right and in some cases they're not right."
>
> *Bill Curtis*

A number of questions could be asked about the validity of the findings of this study. The most important ones are listed below, with a rationale for each.

I. **The MM evaluation in the interview was too short. How can the results be trusted if some participants did not read the documentation completely?**

   It is true that the evaluation was very short. One participant had a total of 7 minutes per FAMM. It is more common in expert reviews for a participant to spend more than an hour reviewing the model. For example, in the API-m FAMM, experts spent more than an hour reviewing each capability (the API-m FAMM calls capabilities 'practices' and calls sub focus area 'capabilities'). This check will go through each practices within a capability, to assess if it contains the right information and has the appropriate maturity level. However, in the interest of time, it was decided to significantly reduce the time taken to assess an MM in this study. This was done in order to have enough time in the interview. This extra time was used to ask the participants exploratory questions, that were not just about the assessment. Additionally, the extra time was used to assess two FAMMs in one interview. If only one FAMM was assessed in this study, but it took twice as long to assess, the results would be less generalisable. Alternatively, two FAMMs could have been split between each half of the participant pool for scoring. However, certain participants might have tended to score higher overall, which could affect the validity of the results. For these reasons, a much shorter evaluation form was chosen. This allowed two FAMMs to be evaluated and allowed additional questions to be asked to the participants.

II. **Not all participants had domain knowledge of the topic. The requirement for selecting a participant was that they had knowledge of MMs. However, this raises the question: if a participant had domain knowledge, could they really give an evaluation that would be useful for this study?**

   It would be preferable to have a participant with knowledge of both MMs and the domain topic. However, then this pool of participants would be too small. It was therefore decided to select either a participant with knowledge of an MM or a participant with knowledge of a domain. In the case of how expert reviews are conducted for a maturity assessment, the requirement is always that the participant should be a domain expert. However, it was decided that the participant should be an expert in the MM. This is because, when observing the MM, it is important for the participant to be able to understand the underlying meaning of what an MM is supposed to be used for. This could have been explained to a domain expert, but it was uncertain whether the domain expert would be able to quickly grasp how MMs work. However, it was still recorded if the participant had domain knowledge. There was no full analysis of whether domain knowledge affected the results. In general, the results showed that those who did not have domain knowledge, often gave a lower rating. The reason they gave a lower rating was that they said they did not know enough about the subject.

III. **The 'right maturity' was only captured by a small set of observations. There was no real indepth assessment of whether the MM captured a maturity path as MMs are supposed to work.**

True, however, it could also be argued that this statement applies to many MMs published to date. The original CMM was measured through each maturity level if it gave a better result based on (cost, time, quality). Besides that, such a quantitative, rigorous evaluation is not often/no longer seen. MMs are more likely to be evaluated through a subjective lens. This subjective lens works as follows. If there is a consensus among domain experts, who generally agree that the MM is well structured, and if a case study appears to be generally representative of the created MM, then it is up to the MM developer to deem it accurate. It should be noted that experts may disagree on how maturity should be captured. For example, in my discussion with Bill Curtis, he argues that the validity of MMs has gone astray. This is because the MM started focusing too much on becoming a capability audit tool without taking the organisational context into account.

IV. **Is it possible that GenAI scores higher because of its writing style?**

Based on the quantitative results, AI-generated FAMMs performed better than a human created FAMM. Questions about the validity of this data have already been raised in this discussion. In this subsection, we'll discuss a hypothesis that AI-generated content may actually receive higher scores simply because of the way it's written. This might mislead people about the actual usefulness of the model. The hypothesis is as follows: *"AI-generated text is much easier to read than human created text. Because AI-generated text is easier to read, for example by using more general terms, participants rated the AI-generated MM higher than the human created MM. However, when an AI-generated MM is actually used in practice, hidden errors in the MM could occur due to the limitations of GenAI. These possible errors may not have occurred in a human created MM because the developer behind the MM may have a better understanding of how maturity should work within the domain. Although the human created MM would have scored lower in the same evaluation, using the research methods of this study, it would actually have been the better MM. This would imply that the results of the AI-generated MM assessment are "untrustworthy*". So far, this hypothesis cannot be proven wrong or right, unless an actual case study is conducted using an AI-generated MM. Despite this hypothesis, the results presented in the previous chapter 5 showed that even participants who had domain knowledge claimed that the AI-generated MM was able to model maturation correctly. The remarkable results of this study should not be simply discredited on the basis of the above stated hypothesis.

7.3.2 Internal validity

> "I've seen too much the difference between reality and what people say in questionnaires. ... I wanna see hard data"
>
> *Bill Curtis*

I. **A comparison of MMs, based on Likert scale ratings, has never been done before. How can such a ranking of the quality of an MM be made, when the AI-generated MM has not even been subjected to a Delphi process or a case study?**

It is true that a comparative analysis has never been done before. In addition, GenAI is also a new emerging topic that is being researched. So far, no work has explored the potential of GenAI in MMs. This research could have alternatively been done by only reporting the use of GenAI, in an full MM development process. However, it would have been impossible to know whether GenAI would have been a benefactor in the development of MMs. This is because such a research project would only rely on the author's subjective opinion. Although not much is known yet, the results of the study serve as a foundation for future work. With the expectations given and the data collected, this comparative analysis provides some basis for what role GenAI could play in MMs.

7.3.3 External validity

> "We have a library of like 4000 different articles on maturity models and just hundreds and hundreds of different maturity models, most of which don't work, most of which haven't been used."
>
> *Bill Curtis*

I. **Can two FAMM (sub)types be generalized to MMs?**

This study seeks to answer the question of what role GenAI could play in MMs. Results were collected by comparing a selected type of human created FAMM with an AI-generated FAMM. The results showed that the AI-generated FAMM scored higher and was presumably better, based on the design criteria, than the human created FAMM. However, the question remains: 'Can these results be generalised to say that GenAI can create an MM that is as good or even better than a human-created one?'

Yes

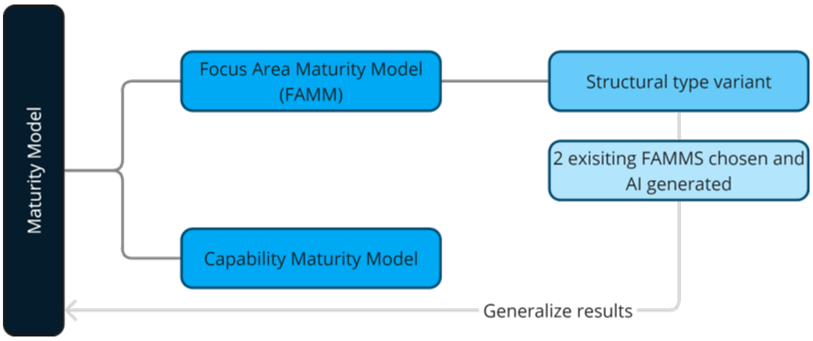

Maturity Model
Focus Area Maturity Model (FAMM)
Structural type variant
2 exisiting FAMMS chosen and AI generated
Capability Maturity Model
Generalize results

FIGURE 7.1: Overview of Maturity Model Generalization: An illustration of the selected sub-branch of MM whereof the findings were gathered

In order to answer the question, it is important to realise that only one sub-branch of MMs were used for the comparative analysis. Consequently, to get an overview of which sub-topic was used, figure 7.1 was created. In figure 7.1 it is shown that MMs start from the left and is then split into two branches. This first split is meant to represent a different method of how an MM could be structured. There were more than two different approaches on how to structure an MM(e.g. maturity grid). The CMM has been included because it is a popular approach and is only shown in, the figure, as an example of a branch. Within the FAMM branch there is a subbranch where a specific set of structural compositions were selected. In this sub-branch, 2 existing FAMMs were selected and AI was generated. In summary, the results of 2 existing structural variant types of FAMMS are to be generalised to MMs.

GenAI was able to work out the right maturity path based on the participants' comments. No direct instructions were given to explain which capabilities should be at which maturity level. Based on the research, GenAI did the same for both topics. So if this research were to be repeated, but for a different branch of MMs, it would probably perform at the same level based on the design criteria measured as in this study. However, how well it compares to human created MMs will largely depend on the quality of the human created MMs it is compared to. This is based on the fact that the human created FAMMs, in this study, had a high variance rating. In addition, the FAMMs did not appear to be publicly known, as each participant stated that they were seeing the FAMM for the first time. It is possible that an MM that is more widely known, and therefore of higher quality, because it has undergone more scrutiny, could score higher than an AI-generated MM on the same topic.

7.4 Limitations

> "if I've got a process, but there's a weak practice in it, that's gonna undermine the process."
>
> *Bill Curtis*

A number of potential limitations need to be considered. Three of the main limitations of have been identified and will be explained in this section. It is important to note that the author was aware of these limitations before starting the research.

The limitations of this study reflect the choices made by the author in the research design.

The first limitation of this study is that there was only one coder (the author), who reviewed and coded the transcripts of the interviews. The author chose this because he thought it would be better to remain the independent researcher of this study. Added to this, it would take too much time to set up such a process. However, the author later regretted the decision not to include a second coder. A disadvantage of not using a second coder is that it reduces reliability. If only the author codes the transcripts, their bias or subjective opinion could affect the consistency and replicability of the findings.

The second limitation is that this study only evaluates a created MM. This study does not evaluate the MM development process. To evaluate the MM development process, a case study should have been conducted. For example, a participant could have developed their own MM using GenAI. Subsequently, within this example, the participant could have reported (or a researcher could have observed) how the MM development process went. It is important to note that the research objective, of this study, assumes that the evaluation of the MMs also provides insight into the MM development process. The insight is limited however. To go into depth what specific roles GenAI could fulfill, additional research should be done.

The third limitation was outlined in the external validity. In short, the findings from the selection of only two FAMMs in this study may not be representative of all MMs. To expand on this limitation, an alternative research method is described below that could have addressed this limitation.

Alternative research method (tackling the third limitation) During and at the end of the research, questions arose how the research could have been done differently. One of these questions was: 'What if there was a method to have participants who were both knowledgeable in MMs and the domain topic?'. A possible solution for this was later thought of and will now be discussed. To aid in the understanding of the alternative research method, a visualization was made which is shown in figure 7.2 . The possible solution is first finding an MM that was already created. Second, another later published MM, with the same domain topic, could be searched. From this second MM found, an AI variant could be created with a CustomGPT. Following that, both MMs could be put into their own separate document for an evaluation. The participant(s) who would be asked to do this evaluation, are the MM developer(s) who created the first found MM. This research method would lead to having a participant (or participants) who is both knowledgeable about the MM and it's domain. A larger variety of MM could also be analyzed through this method. This makes it better to generalize the sample results to all MMs. However, it is to be noted that this possible solution does have downsides. One of the downside is the possibility that each AI generated MM would need it's own custom instructions to create the AI-generated MM. This is because a customGPT creating these MMs needs to know the structural composition of the human created MM. If the format is not specified, the AI-generated MM might not be representative of the human created MM, it could lead to an unequal comparison.

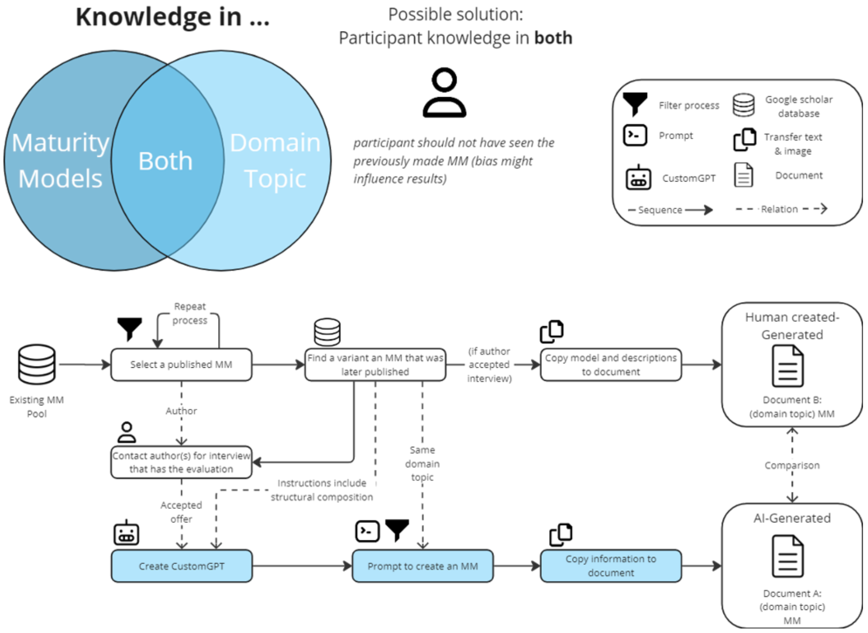

Knowledge in ...
Maturity Models
Both
Domain Topic
Possible solution:
Participant knowledge in **both**
participant should not have seen the previously made MM (bias might influence results)
Filter process
Google scholar database
Prompt
Transfer text & image
CustomGPT
Document
Sequence
Relation
Existing MM Pool
Repeat process
Select a published MM
Find a variant an MM that was later published
(if author accepted interview)
Copy model and descriptions to document
Human created-Generated
Document B: (domain topic) MM
Author
Contact author(s) for interview that has the evaluation
Same domain topic
Instructions include structural composition
Accepted offer
Comparison
AI-Generated
Create CustomGPT
Prompt to create an MM
Copy information to document
Document A: (domain topic) MM

FIGURE 7.2: Alternative Research Method: Evaluating MM with a participant who would have both MM and domain expertise (not included in future work as it is not a continuation but rather an alternative method of the conducted work)

7.5 Future work

> “my opinion is that they (AI-generated MM) are probably going to be as good as 90% or 99% of maturity models that are already out there that are getting people tenure. However, I’d question if they’re as good as the really well-developed maturity models. That would be interesting.’
>
> *Bill Curtis*

At the time of writing this paper, a better version for MaturityGPT is being created by the author. Version 1.0, which more reliably creates, on the prompt of a user, a desired structure for a FAMM. Besides this development, further research should be conducted. The first research possibility is to look into how prompt engineering, on its own, affects the desired output. Research can be conducted by analysing different prompt engineering strategies. Additionally, the success rate of a desired output for each strategy can be measured. By doing such research a more thorough analysis is done on prompt engineering for MM, which would expand on the research done in this paper. Second, could be the replication of the results. Replication is done by comparing more or other types (e.g. CMM style) of human created MMs with AI-generated MMs. As of now, only two FAMMs were compared. There is no strong external validity that the findings of this study can be generalized. To support the generalization, that AI-generated MMs are better than human created MMs, more results need to be gathered. Third, the author proposes to perform case studies. Performing a case study is vital for supporting the findings. If no case study would be conducted, then the findings, of the present study, would only be based on the opinions of participants. The importance of this work, aligns with the continuation of the quote given above: *"And if you can get some sustainable long-term results that are dramatic, especially at the higher levels then then you got something." - Bill Curtis*. Fourth, the focus could be what the best next course of actions, on the existing MMs, is. Questions like 'Is it better for GenAI to replicate or to refine existing created MMs?' come into view. It has to be noted that future work could show that AI-generated MMs are not as good as human created MMs. Nevertheless, that would only be temporary. As stated in chapter 2 of the literature review, GenAI will only get better throughout the years. With regards to AI-replication of existing created MMS, a comparison wherein a popular MM is chosen. During the replication, the time, effort and quality required for developing and using an AI-variant would be observed. Following this, the same popular MM would be used for researching the potential of refining an MM. For the refinement, the development and usage will also be observed. Both these methods can then be compared. Based on these results, that research can establish the superior method. Last, the expected roles for GenAI in the MM life cycle, could be further looked into. An example would be the action plan suggested for the user viewpoint (of Mettlers MM life cycle Mettler, 2011). This research could establish a way of connecting an existing MM with GenAI. Further, the research could investigate which standard practice is most appropriate to do for the AI corrective actions of MMs. Combining corrective actions and testing whether GenAI can generate these corrective actions, while being connected with an MM, would provide valuable information. Information such as, the ability of GenAI to generate, context specific, corrective actions. Having the ability to generate valuable context specific actions, would improve the usability of MMs.

Chapter 8

Conclusion

Throughout this work, a method was outlined on how a variant of a human created FAMM could be generated with the help of AI. Of both these FAMMs, an evaluation was done to compare them in terms of quality. Additionally, the expectations, for the possible role's of GenAI in MMs, were gathered. The goal was to understand whether GenAI has the potential to alleviate the current challenges that MMs face. Challenges such as choosing, maintaining and developing an MM. The study was guided by the MRQ, which is answered at the end of the conclusion.

In order to answer the MRQ, several SRQs were formulated. Each of the answers to the SRQs provides a basis for answering the MRQ. Once each SRQ has been answered, the MRQ can be answered, which is presented in the last section of this chapter.

SRQ1: What are the design criteria for evaluating a maturity model?

In order to determine the quality of an MM for an evaluation, a literature review was conducted (Chapter 3). The literature review was carried out to find the design criteria which would be most appropriate in evaluating the quality of an MM. The findings of the literature review suggest that there is no unified approach for assessing the quality of MMs. There has been previous research that did identify 'what makes an useful maturity model' (Pöppelbuß and Röglinger, 2011). This was done by identifying the design principles (Pöppelbuß and Röglinger, 2011). Nevertheless, that approach held little relevancy, due to the various ways an MM have been (and are being) validated. A common way of validating an MM was 'the delphi Method' (Linstone, Turoff, et al., 1975). However, the method was not used to evaluate an MM. The reason is that, the delphi method required multiple iterative feedback evaluation sessions. While, this study only has one evaluation session, wherein the design criteria are validated (and measured). After careful analysis, a suitable approach to measuring quality was found in the FAMM literature. The evaluation included 5 design criteria[1] that, in a quantified manner, asked participant(s) to give a rating. The design criteria found, were: Completeness, Ease of Use, Effectiveness, Operational Feasibility and Usefulness. Each of these design criteria originated from the paper by Prat et al. 'A Taxonomy of Evaluations Methods in Information Systems' (Prat, Comyn-Wattiau, and Akoka, 2015).

SRQ2: How do AI-generated MMs compare to traditional human-created models in terms of the design criteria?

To answer SRQ2, a comparative analysis was done (See chapter 5). In this analysis participants were given an AI-generated FAMM and a human created one. Each of

[1] Some FAMMs only had 4 design criteria

the participants received five evaluation questions, where each question was related to one of the design criteria of SQ1. The participants were asked to give a Likert rating, that range from 1-5, to score each FAMM given in terms of quality. These ratings were collected to do the comparative analysis. The results are detailed in sub chapter 5.3. These results show that the AI-generated MMs perform, in most design criteria, better (and in some equal) than the human created MMs. Furthermore, of the given FAMMs, participants were (at best) only half the time able to guess correctly which MM was AI-generated. It is to be noted, that in the comparison, the human created MMs showed a high variance, while the AI-generated MMs showed a low variance. These results seem to indicate that new AI-generated MMs, would be roughly of the same quality Despite this, it still begs the question how an AI-generated MM would compare, in terms of applicability, to a human created one. Since, the results gathered from the study are only based on subjective quality evaluations. To compare in terms of applicability, additional research needs to be done, where the application of AI-generated MMs are tested. In conclusion, the results of this research suggest that AI-generated MMs perform equal (or even better) than human created MMs, in terms of quality.

SRQ3: What expectations does the maturity model community envision for GenAI in the usage and development of maturity models?

In order to answer this question, the participants were asked: ’What role do you envision GenAI would have in MMs?’. All of the answers were collected and categorized in the life cycle of an MM Mettler, 2011. With the life cycle, categorization of the participants expectations could be done in specific areas. Within the cycle there are two viewpoints (with each their own areas), the user viewpoint and the developer viewpoint. In both these viewpoint there were around an equal amount of expectations. Interestingly, certain areas showed reoccurring expectations. One of areas with the most expectations was the ’Take corrective action’ area of the user viewpoint. The expectations were all about the creation of an action plan. Here GenAI could help with reaching a higher maturity level, by showing which capabilities are the most beneficial to be implemented next. On the developer viewpoint there were two areas with the most expectations. The first was ’Design’. Here GenAI could assist in making a draft for an MM. The second was ’Scope’. In this Area, GenAI could provide a general outline, in the form of a list, of the information which should be put in an MM. It is worth mentioning that, for both the scope and design of the developer viewpoint, the participants emphasized the need for human supervision while using GenAI in these areas. In short, based on this research, GenAI has possible novel approaches in the usage and development of an MM. For the user viewpoint, it is mainly taking corrective action by helping in the Creation of an action plan. For the development viewpoint, GenAI could help in the design of an MM draft and provide a better scope, by giving a list of information.

MRQ: What role can Generative AI play in the life cycle of maturity models?

In conclusion, GenAI shows potential for roles in the MM life cycle. The results show that AI is able to create MMs. Those MMs seem to be of equal or better value than human created ones. While the process of creating an MM through AI is significantly less time intensive and able to provide personalized up to date content. The community does note that, if an MM would be AI-generated, human supervision needs to be applied. Other potentials can be seen in the assistance of corrective actions. Common expectations given, were that GenAI could formulate an action plan. An action plan could aid the MM user in selecting the capabilities needed to reach a higher maturity level. Taken together, GenAI shows potential in alleviating the current challenges of MMs (see sub chapter 1.2 for the specific problems). This study serves as a springboard for future work, wherein possible specific roles for GenAI would be researched. With this the author concludes, as referred to in the paper's title, that GenAI is mature enough for MMs.

Appendix A

AI generated artifacts

A.1 Research design

A visualization was made to get a topdown view of how the research was down. In the visualization, a legend was also included to refer what each symbol as meaning has. There also includes a ’chptr’ label, what stands for chapter, to understand where additional information of the labelled item could be found.

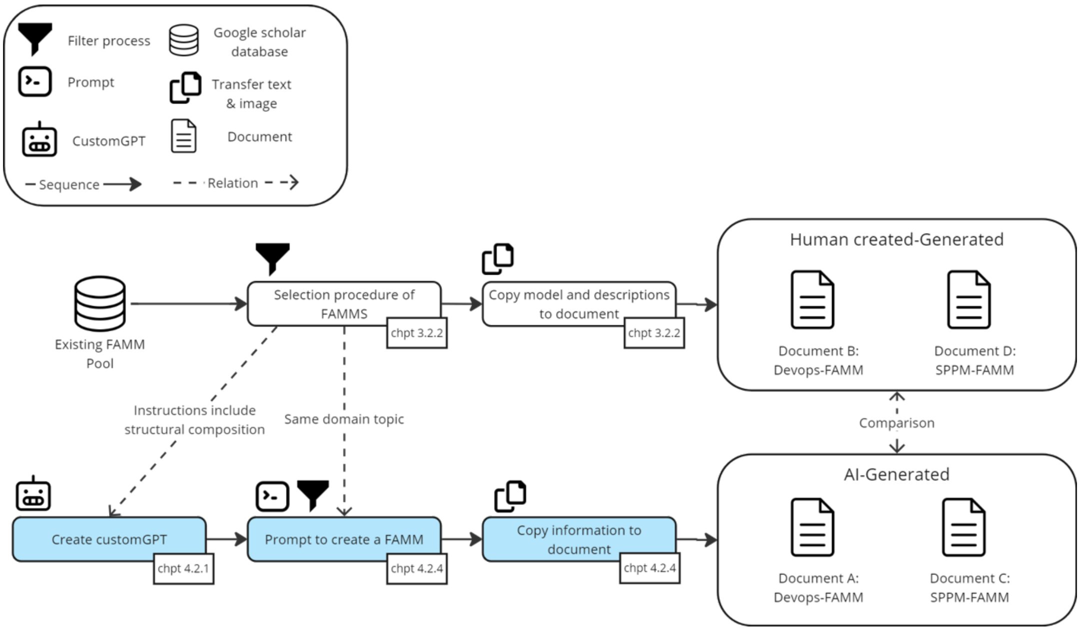


FIGURE A.1: Overview of Research Design and Process Flow: A Visual Representation with Chapter References

A.2 MaturityGPT instructions

This GPT creates a focus area maturity model. The output is given in a table and follows this a structure of level 1 to 10 that is on the head the . I do not want every cell to be filled with a key actions. Start with the table in the output.

This is an example where only focus area 5 is filled in. this is how I want the general structure to be:

Components/ maturity levels 0 1 2 3 4 5 6 7 8 9 10

1. Lifecycle Management 1.1 Version Management 1.2 Decoupling API & Application 1.3 Update Notification 2. Security 2.1 Authentication 2.2 Authorization

2.3 Threat Detection & Protection 2.4 Encryption 3. Performance 3.1 Resource management 3.2 Traffic management 4. Observability 4.1 Monitoring 4.2 Logging 4.3 Analytics 5. Community 5.1 Developer onboarding Facilitate Developer Registration Provide SDK Support Implement Interactive API Console Provide Sandbox Environment 5.2 Support Establish Communication Channel Manage Support Issues Dedicate Developer Support Team 5.3 Documentation Use Standard for Reference Documentation Provide Start-up Documentation Including Samples 5.4 Community Engagement Maintain Social Media Presence Provide Community Forum Provide Developer Portal Organize Events Dedicate API Evangelist 5.5 Portfolio Management Enable API Discovery Provide API Catalog Bundle APIs 6. Commercial 6.1 Service-Level Agreements 6.2 Monetization Strategy 6.3 Account Management

Here is the mathematical formalization you should use: In order to provide rigorous fundamentals for focus area maturity models, we need to abstract the commonalities from the cases into a mathematical model. To introduce this model, we first have to define the fundamental concepts defining the maturity matrices. For convenience we will refer in the following way to the different types of matrices: the EA-matrix will refer to the DyA architecture maturity matrix, and the SPM-matrix will refer to the software product management maturity matrix.

Both types of matrices use the concept of focus area (the rows of the matrices) for which we introduce the set F of focus areas. The number of focus areas within each matrix differs slightly: 18 for the EA-matrix and 16 for the SPM-matrix.

Another fundamental concept comes from the assessments organizations have to pass in order to reach a certain level for a specific focus area. We therefore introduce a totally ordered set $(L, \leq_L)$ of levels and since an assessment is specific for a pair consisting of a focus area and a level, we are interested in the Cartesian product $F \times L$. We abstract away from the 'assessment' and concentrate on the set $F \times L$. Since not every element of F needs to have the same number of levels, this Cartesian product is in general a little bit too large. For the general definition of maturity matrix we allow subsets C of $F \times L$. In the two example matrices, C denotes the set of capabilities and the pairs $(f, l) \in C$ correspond to the cells in the matrix that are filled with a capital letter. The columns in the example matrices are the final concept we need and are formally described by a specific mapping S from C to the natural numbers. This puts us now in a position to give the following:

Definition: A maturity matrix consists of 1. A triple $(F, (L, \leq_L), (C, \leq_C))$ where F is a set, $(L, \leq_L)$ is a completely ordered set and $(C, \leq_C)$ is a partially ordered set with $C \subseteq F \times L$. Moreover, the ordering on C respects the ordering on L in the sense that if $c_1 = (f, l_1), c_2 = (f, l_2) \in C$ and $l_1 \leq_L l_2$ then $c_1 \leq_C c_2$. 2. An order preserving mapping $S : C \to \mathbb{N}$ with $\text{Im}(S) = \{1, \ldots, m\}$ for some $m \in \mathbb{N}$.

As an example take the SPM-matrix where F is the set of 16 focus areas, L = {A, B, ...} is the set of 6 levels (so $F \times L$ consists of 96 elements), and L is totally ordered in the obvious way (A < B < ...). Furthermore, C is the set of 63 capabilities, consisting of specific pairs (f, l) where $f \in F$ and $l \in L$ and C is partially ordered by the intra- and inter-process capability dependencies, e.g. relations of the form $(f, A) < (f, B)$ (intra-process) and relations of the form $(f_1, l_1) < (f_2, l_2)$ where $f_1 \neq f_2$ (inter-process).

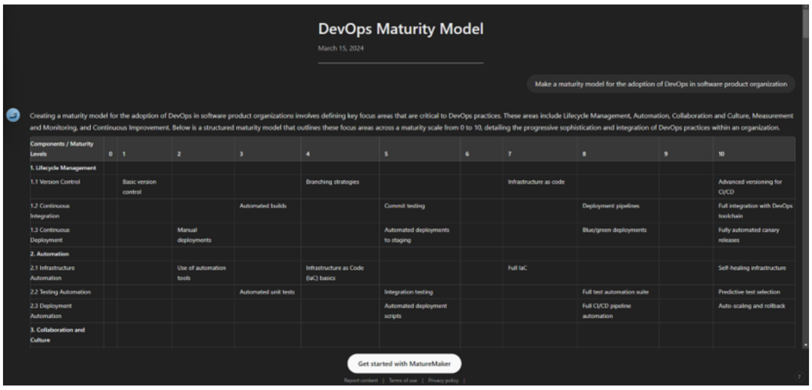
Finally, the mapping S assigns every capability to one of the numbers 1 through 12 while preserving the order (so if $c_1 \leq_C c_2$, then $S(c_1) \leq S(c_2)$).

The maturity scale of an organization can now be defined. Since an organization that just started the development of a functional domain could very well have none of the capabilities defined for this domain, it makes sense to allow a zero scale. Even if they have acquired some capabilities of scale 1, but not all of them, we still define their scale as zero. Only if they have acquired all capabilities of scale 1 (i.e. all capabilities of the set $S^{-1}(1)$), then their scale will be 1 or higher.

In general, if the set of capabilities acquired by the organization is denoted by CA (a subset of C), then the scale of that organization is the maximum value s for which $S^{-1}(\{1,\ldots,s\}) \subseteq CA$. Note that if we substitute $s = 0$, the set $S^{-1}(\{1,\ldots,0\}) = \emptyset$ is a subset of CA, so this definition also holds if CA is empty or if CA doesn't contain all capabilities with scale 1 (in both cases the maturity scale of the organization will be 0).

A.3 Full generated AI FAMM

A.3.1 Example of prompt in MaturityGPT



DevOps Maturity Model

March 15, 2024

Make a maturity model for the adoption of DevOps in software product organization

Creating a maturity model for the adoption of DevOps in software product organizations involves defining key focus areas that are critical to DevOps practices. These areas include Lifecycle Management, Automation, Collaboration and Culture, Measurement and Monitoring, and Continuous Improvement. Below is a structured maturity model that outlines these focus areas across a maturity scale from 0 to 10, detailing the progressive sophistication and integration of DevOps practices within an organization.

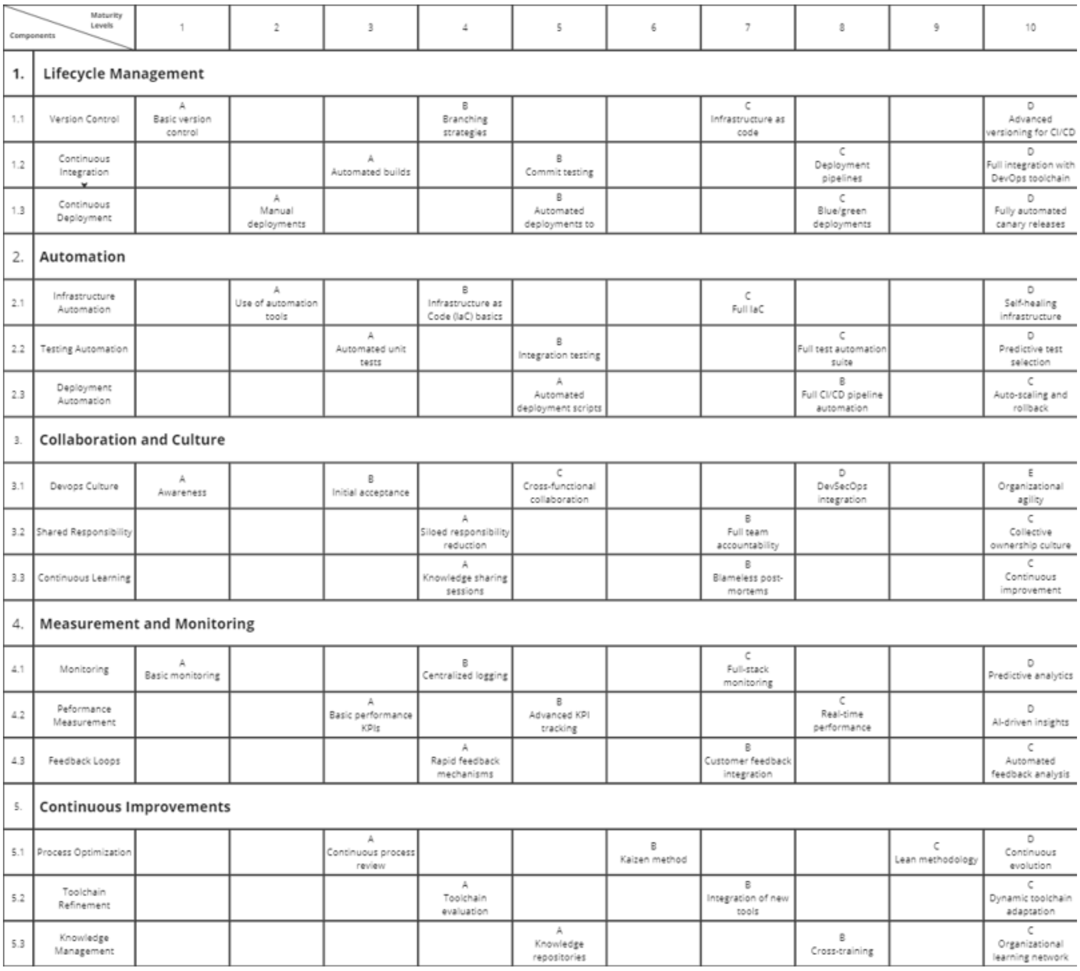
Components / Maturity Levels	0	1	2	3	4	5	6	7	8	9	10
1. Lifecycle Management											
1.1 Version Control		Basic version control			Branching strategies			Infrastructure as code			Advanced versioning for CI/CD
1.2 Continuous Integration				Automated builds		Commit testing			Deployment pipelines		Full integration with DevOps toolchain
1.3 Continuous Deployment			Manual deployments			Automated deployments to staging			Blue/green deployments		Fully automated canary releases
2. Automation											
2.1 Infrastructure Automation			Use of automation tools		Infrastructure as Code (IaC) basics			Full IaC			Self-healing infrastructure
2.2 Testing Automation				Automated unit tests		Integration testing			Full test automation suite		Predictive test selection
2.3 Deployment Automation						Automated deployment scripts			Full CI/CD pipeline automation		Auto-scaling and rollback
3. Collaboration and Culture											

Get started with MatureMaker
Report content | Terms of use | Privacy policy |

FIGURE A.2: Output from DevOps FAMM MaturityGPT

A.3.2 DevOps FAMM document

link: https://chatgpt.com/share/64a30ea6-57e1-432c-83a4-8073e45000ff



	Maturity Levels Components	1	2	3	4	5	6	7	8	9	10
1.	Lifecycle Management										
1.1	Version Control	A Basic version control			B Branching strategies			C Infrastructure as code			D Advanced versioning for CI/CD
1.2	Continuous Integration			A Automated builds		B Commit testing			C Deployment pipelines		D Full integration with DevOps toolchain
1.3	Continuous Deployment		A Manual deployments			B Automated deployments to			C Blue/green deployments		D Fully automated canary releases
2.	Automation										
2.1	Infrastructure Automation		A Use of automation tools		B Infrastructure as Code (IaC) basics			C Full IaC			D Self-healing infrastructure
2.2	Testing Automation			A Automated unit tests		B Integration testing			C Full test automation suite		D Predictive test selection
2.3	Deployment Automation					A Automated deployment scripts			B Full CI/CD pipeline automation		C Auto-scaling and rollback
3.	Collaboration and Culture										
3.1	Devops Culture	A Awareness		B Initial acceptance		C Cross-functional collaboration			D DevSecOps integration		E Organizational agility
3.2	Shared Responsibility				A Siloed responsibility reduction			B Full team accountability			C Collective ownership culture
3.3	Continuous Learning				A Knowledge sharing sessions			B Blameless post-mortems			C Continuous improvement
4.	Measurement and Monitoring										
4.1	Monitoring	A Basic monitoring			B Centralized logging			C Full-stack monitoring			D Predictive analytics
4.2	Peformance Measurement			A Basic performance KPIs		B Advanced KPI tracking			C Real-time performance		D AI-driven insights
4.3	Feedback Loops				A Rapid feedback mechanisms			B Customer feedback integration			C Automated feedback analysis
5.	Continuous Improvements										
5.1	Process Optimization			A Continuous process review			B Kaizen method			C Lean methodology	D Continuous evolution
5.2	Toolchain Refinement				A Toolchain evaluation			B Integration of new tools			C Dynamic toolchain adaptation
5.3	Knowledge Management					A Knowledge repositories			B Cross-training		C Organizational learning network

FIGURE A.3: AI-generated DevOps FAMM

Documentation

1.1 Focus areas and capabilities
1.1.1 Culture and collaboration
1.1.1.1 Communication

A. Indirect communication

Action: communication between interdisciplinary professionals, among which are dev and ops professionals, is indirectly established (e.g. through procedures,managers, software architects).

B. Facilitated communication

Action: direct communication between interdisciplinary professionals, among which are dev and ops professionals, is facilitated by management by stimulat- ing professionals to communicate directly (e.g. by giving professionals time to prepare sessions).

C. Direct communication
Action: direct interdisciplinary communication between professionals, among which are dev and ops professionals while working towards a release is present. This direct communication could occur through mailing lists, personal contact etc.

D. Structured communication
Action: a structure for interdisciplinary communication is in place (e.g. by holding daily standups and retrospectives with interdisciplinary professionals including dev and ops, and by maintaining contact with (product) management to discuss about impediments along the way, work to be done the upcoming sprints and the technical debt situation, among others).

E. Communication improvement
Action: communication among management and interdisciplinary professionals, including dev and ops, is improved (e.g. by adopting and trying out new communi- cation practices from industry, learning from experiences and by tracking projects or using instruments such as skill matrices and peer feedback mechanisms over time).

1.1.1.1 Knowledge sharing

A. Decentralized knowledge sharing
Action: knowledge is shared between interdisciplinary professionals, among which are dev and ops professionals in a decentralized way (i.e. through notes or documents).

...

Skip to last capability

...

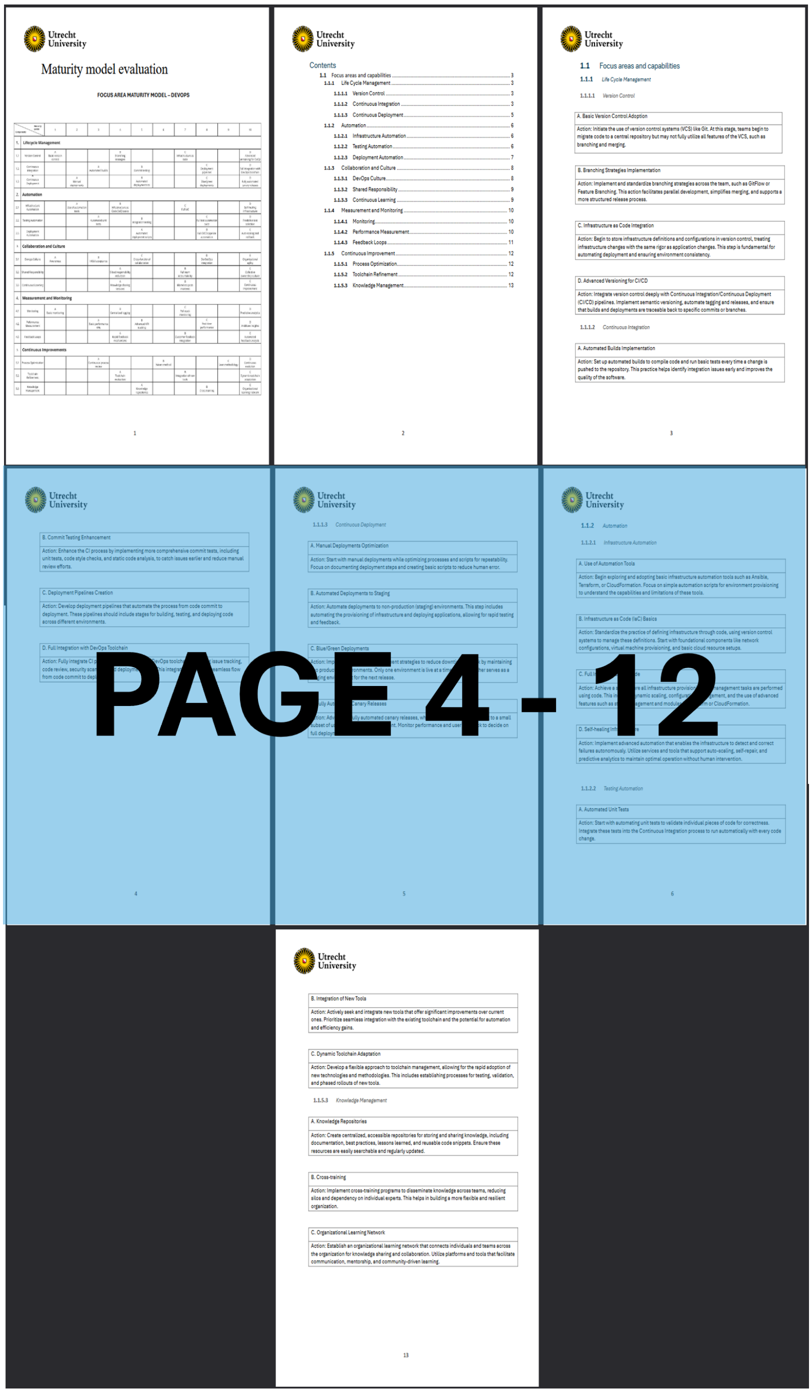
1.1.1.1 Infrastructure

A. Manually provisioned infrastructure
Action: infrastructure such as development, test, acceptance and production in- frastructure is available and provisioned manually (e.g. by walking through a manual and provision an environment by hand).

B. Automatically provisioned infrastructure
Action: infrastructure between development and production is equivalent in terms of configuration and hardware and provisioned automatically (e.g. by pushing a declarative configuration (i.e. a reproducible configuration in code to a virtual machine that represents an environment)).

C. Managed platform services
Action: platform services (such as a web server and a database server) are precon- figured in the platform and allow for applications being directly de- ployed, among others, while rights and rolls are managed per environment. This is also known as platform as a service.

Overview



Utrecht University

Maturity model evaluation

FOCUS AREA MATURITY MODEL – DEVOPS

Component	Maturity level	1	2	3	4	5	6	7	8	9	10
1.	Lifecycle Management										
1.1	Version Control	A Basic version control			B Branching strategies			C Infrastructure as code			D Advanced versioning for CI/CD
1.2	Continuous Integration			A Automated builds		B Commit testing			C Deployment pipelines		D Full integration with DevOps toolchain
1.3	Continuous Deployment		A Manual deployments			B Automated deployments to staging			C Blue/green deployments		D Fully automated canary releases
2.	Automation										
2.1	Infrastructure Automation		A Use of automation tools		B Infrastructure as Code (IaC) basics			C Full IaC			D Self-healing infrastructure
2.2	Testing Automation			A Automated unit tests		B Integration testing			C Full test automation suite		D Predictive test selection
2.3	Deployment Automation					A Automated deployment scripts			B Full CI/CD pipeline automation		C Auto-scaling and rollback
3.	Collaboration and Culture										
3.1	DevOps Culture	A Awareness		B Initial acceptance		C Cross-functional collaboration			D DevSecOps integration		E Organizational agility
3.2	Shared Responsibility				A Siloed responsibility reduction			B Full team accountability			C Collective ownership culture
3.3	Continuous Learning				A Knowledge sharing sessions			B Blameless post-mortems			C Continuous improvement
4.	Measurement and Monitoring										
4.1	Monitoring	A Basic monitoring			B Centralized logging			C Full-stack monitoring			D Predictive analytics
4.2	Performance Measurement			A Basic performance KPIs		B Advanced KPI tracking			C Real-time performance		D AI-driven insights
4.3	Feedback Loops				A Rapid feedback mechanisms			B Customer feedback integration			C Automated feedback analysis
5.	Continuous Improvements										
5.1	Process Optimization			A Continuous process review			B Kaizen method			C Lean methodology	D Continuous evolution
5.2	Toolchain Refinement				A Toolchain evaluation			B Integration of new tools			C Dynamic toolchain adaptation
5.3	Knowledge Management					A Knowledge repositories			B Cross-training		C Organizational learning network

1

Utrecht University

Contents

2

Utrecht University

1.1 Focus areas and capabilities

1.1.1 *Life Cycle Management*

1.1.1.1 *Version Control*

A. Basic Version Control Adoption
Action: Initiate the use of version control systems (VCS) like Git. At this stage, teams begin to migrate code to a central repository but may not fully utilize all features of the VCS, such as branching and merging.

B. Branching Strategies Implementation
Action: Implement and standardize branching strategies across the team, such as GitFlow or Feature Branching. This action facilitates parallel development, simplifies merging, and supports a more structured release process.

C. Infrastructure as Code Integration
Action: Begin to store infrastructure definitions and configurations in version control, treating infrastructure changes with the same rigor as application changes. This step is fundamental for automating deployment and ensuring environment consistency.

D. Advanced Versioning for CI/CD
Action: Integrate version control deeply with Continuous Integration/Continuous Deployment (CI/CD) pipelines. Implement semantic versioning, automate tagging and releases, and ensure that builds and deployments are traceable back to specific commits or branches.

1.1.1.2 *Continuous Integration*

A. Automated Builds Implementation
Action: Set up automated builds to compile code and run basic tests every time a change is pushed to the repository. This practice helps identify integration issues early and improves the quality of the software.

3

Utrecht University

B. Commit Testing Enhancement
Action: Enhance the CI process by implementing more comprehensive commit tests, including unit tests, code style checks, and static code analysis, to catch issues earlier and reduce manual review efforts.

C. Deployment Pipelines Creation
Action: Develop deployment pipelines that automate the process from code commit to deployment. These pipelines should include stages for building, testing, and deploying code across different environments.

D. Full Integration with DevOps Toolchain
Action: Fully integrate CI p[illegible] DevOps toolch[illegible] issue tracking, code review, security scan[illegible] deployme[illegible] This integrat[illegible] seamless flow from code commit to depl[illegible]

4

Utrecht University

1.1.1.3 *Continuous Deployment*

A. Manual Deployments Optimization
Action: Start with manual deployments while optimizing processes and scripts for repeatability. Focus on documenting deployment steps and creating basic scripts to reduce human error.

B. Automated Deployments to Staging
Action: Automate deployments to non-production (staging) environments. This step includes automating the provisioning of infrastructure and deploying applications, allowing for rapid testing and feedback.

C. Blue/Green Deployments
Action: Imp[illegible] ent strategies to reduce downti[illegible] k by maintaining [illegible] o produc[illegible] ronments. Only one environment is live at a tim[illegible] her serves as a [illegible] ging env[illegible] t for the next release.

[illegible] ully Aut[illegible] Canary Releases
[illegible] ction: Adv[illegible] ully automated canary releases, wh[illegible] to a small subset of us[illegible] nt. Monitor performance and user[illegible] ck to decide on full deploym[illegible]

5

Utrecht University

1.1.2 *Automation*

1.1.2.1 *Infrastructure Automation*

A. Use of Automation Tools
Action: Begin exploring and adopting basic infrastructure automation tools such as Ansible, Terraform, or CloudFormation. Focus on simple automation scripts for environment provisioning to understand the capabilities and limitations of these tools.

B. Infrastructure as Code (IaC) Basics
Action: Standardize the practice of defining infrastructure through code, using version control systems to manage these definitions. Start with foundational components like network configurations, virtual machine provisioning, and basic cloud resource setups.

C. Full I[illegible] de
Action: Achieve a s[illegible] re all infrastructure provision[illegible] management tasks are performed using code. This in[illegible] ynamic scaling, configurati[illegible] gement, and the use of advanced features such as st[illegible] agement and module[illegible] rm or CloudFormation.

D. Self-healing Infr[illegible] re
Action: Implement advanced automation that enables the infrastructure to detect and correct failures autonomously. Utilize services and tools that support auto-scaling, self-repair, and predictive analytics to maintain optimal operation without human intervention.

1.1.2.2 *Testing Automation*

A. Automated Unit Tests
Action: Start with automating unit tests to validate individual pieces of code for correctness. Integrate these tests into the Continuous Integration process to run automatically with every code change.

6

Utrecht University

B. Integration of New Tools
Action: Actively seek and integrate new tools that offer significant improvements over current ones. Prioritize seamless integration with the existing toolchain and the potential for automation and efficiency gains.

C. Dynamic Toolchain Adaptation
Action: Develop a flexible approach to toolchain management, allowing for the rapid adoption of new technologies and methodologies. This includes establishing processes for testing, validation, and phased rollouts of new tools.

1.1.5.3 *Knowledge Management*

A. Knowledge Repositories
Action: Create centralized, accessible repositories for storing and sharing knowledge, including documentation, best practices, lessons learned, and reusable code snippets. Ensure these resources are easily searchable and regularly updated.

B. Cross-training
Action: Implement cross-training programs to disseminate knowledge across teams, reducing silos and dependency on individual experts. This helps in building a more flexible and resilient organization.

C. Organizational Learning Network
Action: Establish an organizational learning network that connects individuals and teams across the organization for knowledge sharing and collaboration. Utilize platforms and tools that facilitate communication, mentorship, and community-driven learning.

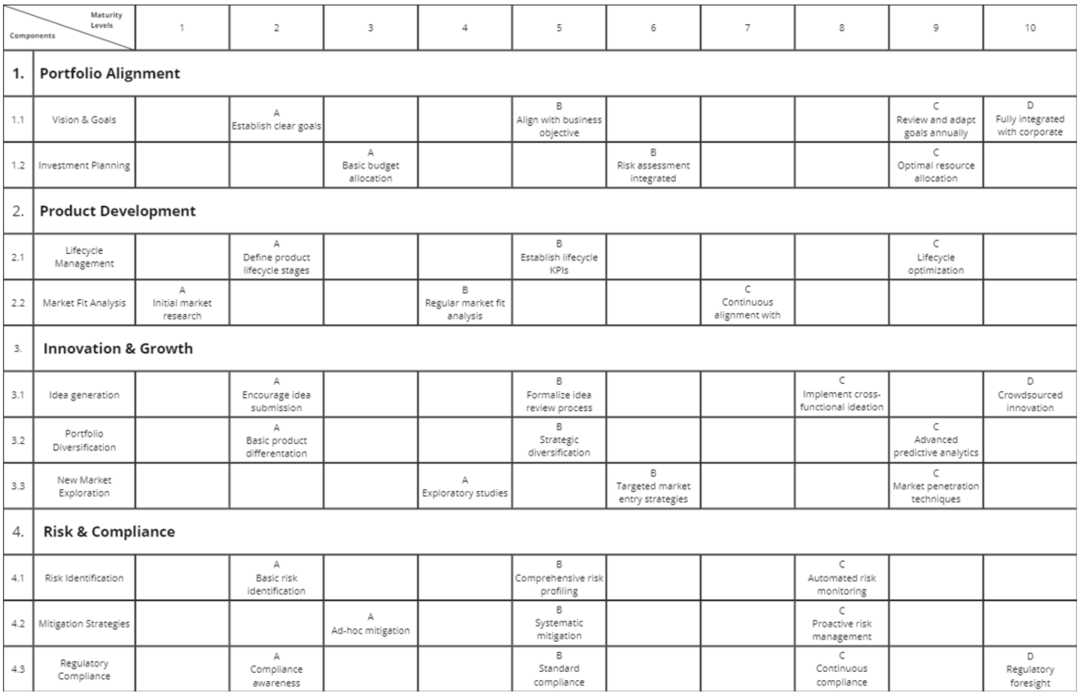
13

FIGURE A.4: Document A: DevOps

A.3.3 SPPM FAMM document

link: https://chatgpt.com/share/2b3e5afe-c03e-492b-8634-f11d63029da3

FAMM



Maturity Levels Components		1	2	3	4	5	6	7	8	9	10
1.	**Portfolio Alignment**										
1.1	Vision & Goals		A Establish clear goals			B Align with business objective				C Review and adapt goals annually	D Fully integrated with corporate
1.2	Investment Planning			A Basic budget allocation			B Risk assessment integrated			C Optimal resource allocation	
2.	**Product Development**										
2.1	Lifecycle Management		A Define product lifecycle stages			B Establish lifecycle KPIs				C Lifecycle optimization	
2.2	Market Fit Analysis	A Initial market research			B Regular market fit analysis			C Continuous alignment with			
3.	**Innovation & Growth**										
3.1	Idea generation		A Encourage idea submission			B Formalize idea review process			C Implement cross-functional ideation		D Crowdsourced innovation
3.2	Portfolio Diversification		A Basic product differentation			B Strategic diversification				C Advanced predictive analytics	
3.3	New Market Exploration				A Exploratory studies		B Targeted market entry strategies			C Market penetration techniques	
4.	**Risk & Compliance**										
4.1	Risk Identification		A Basic risk identification			B Comprehensive risk profiling			C Automated risk monitoring		
4.2	Mitigation Strategies			A Ad-hoc mitigation		B Systematic mitigation			C Proactive risk management		
4.3	Regulatory Compliance		A Compliance awareness			B Standard compliance			C Continuous compliance		D Regulatory foresight

FIGURE A.5: AI-generated SPPM FAMM

Documentation

1.1 Focus areas and capabilities
1.1.1 Culture and collaboration
1.1.1.1 Communication

A. Established Clear Goals
An initial evaluation that includes setting precise, measurable objectives that align with the overarching business mission. This assessment involves understanding current business conditions, market opportunities, and strategic imperatives.

B. Aligned Goals with Business Objectives
he insights from the initial goal-setting phase are utilized to align these goals directly with broader business objectives. This ensures that every aspect of the strategic alignment supports the primary aims of the organization, facilitating unified progress.

C. Reviewed and Adapted Goals Annually
This involves a structured review process where existing goals are evaluated against the current business and market context. Adjustments are made to ensure that the strategic direction remains relevant and effective in driving business success.

D. Integrated Goals with Corporate Strategy
The culmination of this process is the full integration of these specific goals with the wider corporate strategy. This step ensures that the portfolio strategy is not only aligned but also embedded within the company's strategic initiatives, leading to coherent and unified organizational progress.

1.1.1.2 Investment Planning

A. Basic Budget Allocation
Initiate the investment planning process with the establishment of a basic budget framework. This stage involves setting preliminary financial allocations based on initial assessments of needs and strategic objectives, providing a foundational structure for further refinement.

...

Skip to last capability

...

1.1.1.1 Regulatory Compliance

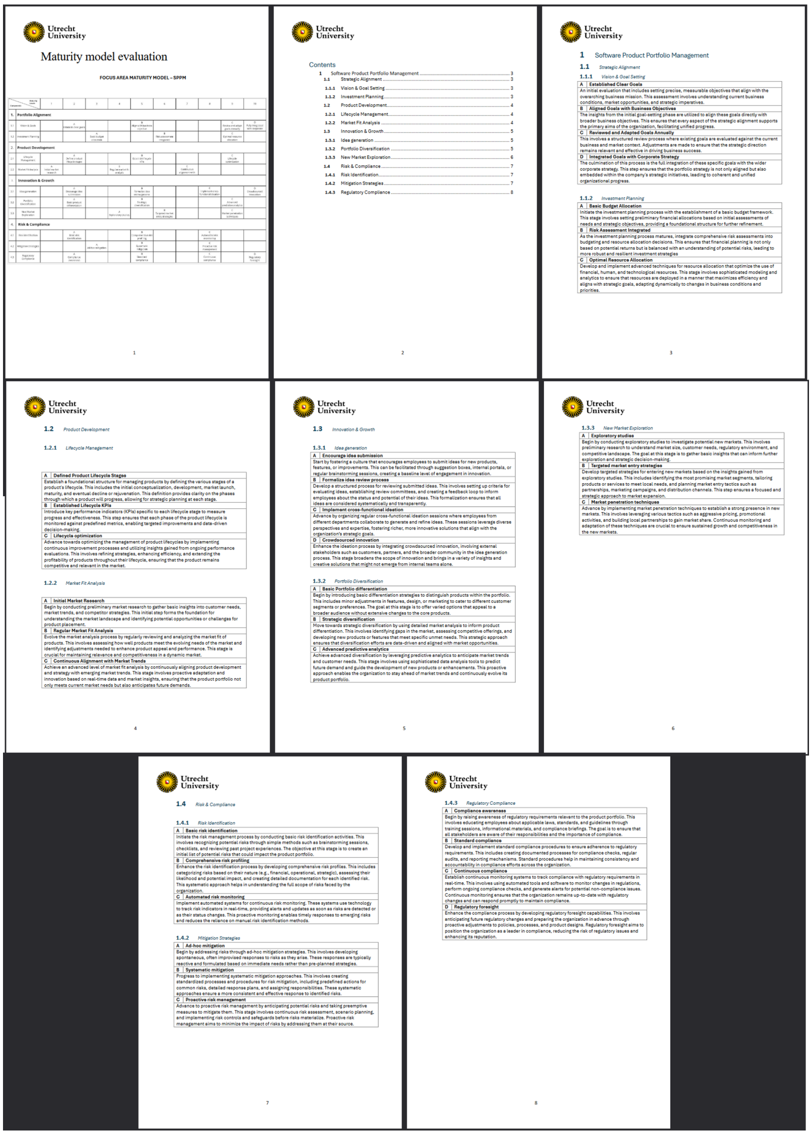
A. Compliance awareness
Begin by raising awareness of regulatory requirements relevant to the product portfolio. This involves educating employees about applicable laws, standards, and guidelines through training sessions, informational materials, and compliance briefings. The goal is to ensure that all stakeholders are aware of their responsibilities and the importance of compliance.

B. Standard compliance
Develop and implement standard compliance procedures to ensure adherence to regulatory requirements. This includes creating documented processes for compliance checks, regular audits, and reporting mechanisms. Standard procedures help in maintaining consistency and accountability in compliance efforts across the organization.

C. Continuous compliance
Establish continuous monitoring systems to track compliance with regulatory requirements in real-time. This involves using automated tools and software to monitor changes in regulations, perform ongoing compliance checks, and generate alerts for potential non-compliance issues. Continuous monitoring ensures that the organization remains up-to-date with regulatory changes and can respond promptly to maintain compliance.

D. Regulatory foresight
Enhance the compliance process by developing regulatory foresight capabilities. This involves anticipating future regulatory changes and preparing the organization in advance through proactive adjustments to policies, processes, and product designs. Regulatory foresight aims to position the organization as a leader in compliance, reducing the risk of regulatory issues and enhancing its reputation

Overview



Utrecht University

Maturity model evaluation

FOCUS AREA MATURITY MODEL – SPPM

	Components \ Maturity Levels	1	2	3	4	5	6	7	8	9	10
1.	**Portfolio Alignment**										
1.1	Vision & Goals		A Establish clear goals			B Align with business objective				C Review and adopt goals annually	D Fully integrated with corporate
1.2	Investment Planning			A Basic budget allocation			B Risk assessment integrated			C Optimal resource allocation	
2.	**Product Development**										
2.1	Lifecycle Management		A Define product lifecycle stages			B Establish lifecycle KPIs				C Lifecycle optimization	
2.2	Market Fit Analysis	A Initial market research			B Regular market fit analysis			C Continuous alignment with			
3.	**Innovation & Growth**										
3.1	Idea generation		A Encourage idea submission			B Formalize idea review process			C Implement cross-functional ideation		D Crowdsourced innovation
3.2	Portfolio Diversification		A Basic product differentiation			B Strategic diversification				C Advanced predictive analytics	
3.3	New Market Exploration				A Exploratory studies		B Targeted market entry strategies			C Market penetration techniques	
4.	**Risk & Compliance**										
4.1	Risk Identification		A Basic risk identification			B Comprehensive risk profiling			C Automated risk monitoring		
4.2	Mitigation Strategies			A Ad-hoc mitigation		B Systematic mitigation			C Proactive risk management		
4.3	Regulatory Compliance		A Compliance awareness			B Standard compliance			C Continuous compliance		D Regulatory foresight

1

Contents

2

1 Software Product Portfolio Management

1.1 *Strategic Alignment*

1.1.1 *Vision & Goal Setting*

A Established Clear Goals

An initial evaluation that includes setting precise, measurable objectives that align with the overarching business mission. This assessment involves understanding current business conditions, market opportunities, and strategic imperatives.

B Aligned Goals with Business Objectives

The insights from the initial goal-setting phase are utilized to align these goals directly with broader business objectives. This ensures that every aspect of the strategic alignment supports the primary aims of the organization, facilitating unified progress.

C Reviewed and Adapted Goals Annually

This involves a structured review process where existing goals are evaluated against the current business and market context. Adjustments are made to ensure that the strategic direction remains relevant and effective in driving business success.

D Integrated Goals with Corporate Strategy

The culmination of this process is the full integration of these specific goals with the wider corporate strategy. This step ensures that the portfolio strategy is not only aligned but also embedded within the company's strategic initiatives, leading to coherent and unified organizational progress.

1.1.2 *Investment Planning*

A Basic Budget Allocation

Initiate the investment planning process with the establishment of a basic budget framework. This stage involves setting preliminary financial allocations based on initial assessments of needs and strategic objectives, providing a foundational structure for further refinement.

B Risk Assessment Integrated

As the investment planning process matures, integrate comprehensive risk assessments into budgeting and resource allocation decisions. This ensures that financial planning is not only based on potential returns but is balanced with an understanding of potential risks, leading to more robust and resilient investment strategies

C Optimal Resource Allocation

Develop and implement advanced techniques for resource allocation that optimize the use of financial, human, and technological resources. This stage involves sophisticated modeling and analytics to ensure that resources are deployed in a manner that maximizes efficiency and aligns with strategic goals, adapting dynamically to changes in business conditions and priorities.

3

1.2 *Product Development*

1.2.1 *Lifecycle Management*

A Defined Product Lifecycle Stages

Establish a foundational structure for managing products by defining the various stages of a product's lifecycle. This includes the initial conceptualization, development, market launch, maturity, and eventual decline or rejuvenation. This definition provides clarity on the phases through which a product will progress, allowing for strategic planning at each stage.

B Established Lifecycle KPIs

Introduce key performance indicators (KPIs) specific to each lifecycle stage to measure progress and effectiveness. This step ensures that each phase of the product lifecycle is monitored against predefined metrics, enabling targeted improvements and data-driven decision-making.

C Lifecycle optimization

Advance towards optimizing the management of product lifecycles by implementing continuous improvement processes and utilizing insights gained from ongoing performance evaluations. This involves refining strategies, enhancing efficiency, and extending the profitability of products throughout their lifecycle, ensuring that the product remains competitive and relevant in the market.

1.2.2 *Market Fit Analysis*

A Initial Market Research

Begin by conducting preliminary market research to gather basic insights into customer needs, market trends, and competitor strategies. This initial step forms the foundation for understanding the market landscape and identifying potential opportunities or challenges for product placement.

B Regular Market Fit Analysis

Evolve the market analysis process by regularly reviewing and analyzing the market fit of products. This involves assessing how well products meet the evolving needs of the market and identifying adjustments needed to enhance product appeal and performance. This stage is crucial for maintaining relevance and competitiveness in a dynamic market.

C Continuous Alignment with Market Trends

Achieve an advanced level of market fit analysis by continuously aligning product development and strategy with emerging market trends. This stage involves proactive adaptation and innovation based on real-time data and market insights, ensuring that the product portfolio not only meets current market needs but also anticipates future demands.

4

1.3 *Innovation & Growth*

1.3.1 *Idea generation*

A Encourage idea submission

Start by fostering a culture that encourages employees to submit ideas for new products, features, or improvements. This can be facilitated through suggestion boxes, internal portals, or regular brainstorming sessions, creating a baseline level of engagement in innovation.

B Formalize idea review process

Develop a structured process for reviewing submitted ideas. This involves setting up criteria for evaluating ideas, establishing review committees, and creating a feedback loop to inform employees about the status and potential of their ideas. This formalization ensures that all ideas are considered systematically and transparently.

C Implement cross-functional ideation

Advance by organizing regular cross-functional ideation sessions where employees from different departments collaborate to generate and refine ideas. These sessions leverage diverse perspectives and expertise, fostering richer, more innovative solutions that align with the organization's strategic goals.

D Crowdsourced innovation

Enhance the ideation process by integrating crowdsourced innovation, involving external stakeholders such as customers, partners, and the broader community in the idea generation process. This stage broadens the scope of innovation and brings in a variety of insights and creative solutions that might not emerge from internal teams alone.

1.3.2 *Portfolio Diversification*

A Basic Portfolio differentiation

Begin by introducing basic differentiation strategies to distinguish products within the portfolio. This includes minor adjustments in features, design, or marketing to cater to different customer segments or preferences. The goal at this stage is to offer varied options that appeal to a broader audience without extensive changes to the core products.

B Strategic diversification

Move towards strategic diversification by using detailed market analysis to inform product differentiation. This involves identifying gaps in the market, assessing competitive offerings, and developing new products or features that meet specific unmet needs. This strategic approach ensures that diversification efforts are data-driven and aligned with market opportunities.

C Advanced predictive analytics

Achieve advanced diversification by leveraging predictive analytics to anticipate market trends and customer needs. This stage involves using sophisticated data analysis tools to predict future demand and guide the development of new products or enhancements. This proactive approach enables the organization to stay ahead of market trends and continuously evolve its product portfolio.

5

1.3.3 *New Market Exploration*

A Exploratory studies

Begin by conducting exploratory studies to investigate potential new markets. This involves preliminary research to understand market size, customer needs, regulatory environment, and competitive landscape. The goal at this stage is to gather basic insights that can inform further exploration and strategic decision-making.

B Targeted market entry strategies

Develop targeted strategies for entering new markets based on the insights gained from exploratory studies. This includes identifying the most promising market segments, tailoring products or services to meet local needs, and planning market entry tactics such as partnerships, marketing campaigns, and distribution channels. This step ensures a focused and strategic approach to market expansion.

C Market penetration techniques

Advance by implementing market penetration techniques to establish a strong presence in new markets. This involves leveraging various tactics such as aggressive pricing, promotional activities, and building local partnerships to gain market share. Continuous monitoring and adaptation of these techniques are crucial to ensure sustained growth and competitiveness in the new markets.

6

1.4 *Risk & Compliance*

1.4.1 *Risk Identification*

A Basic risk identification

Initiate the risk management process by conducting basic risk identification activities. This involves recognizing potential risks through simple methods such as brainstorming sessions, checklists, and reviewing past project experiences. The objective at this stage is to create an initial list of potential risks that could impact the product portfolio.

B Comprehensive risk profiling

Enhance the risk identification process by developing comprehensive risk profiles. This includes categorizing risks based on their nature (e.g., financial, operational, strategic), assessing their likelihood and potential impact, and creating detailed documentation for each identified risk. This systematic approach helps in understanding the full scope of risks faced by the organization.

C Automated risk monitoring

Implement automated systems for continuous risk monitoring. These systems use technology to track risk indicators in real-time, providing alerts and updates as soon as risks are detected or as their status changes. This proactive monitoring enables timely responses to emerging risks and reduces the reliance on manual risk identification methods.

1.4.2 *Mitigation Strategies*

A Ad-hoc mitigation

Begin by addressing risks through ad-hoc mitigation strategies. This involves developing spontaneous, often improvised responses to risks as they arise. These responses are typically reactive and formulated based on immediate needs rather than pre-planned strategies.

B Systematic mitigation

Progress to implementing systematic mitigation approaches. This involves creating standardized processes and procedures for risk mitigation, including predefined actions for common risks, detailed response plans, and assigning responsibilities. These systematic approaches ensure a more consistent and effective response to identified risks.

C Proactive risk management

Advance to proactive risk management by anticipating potential risks and taking preemptive measures to mitigate them. This stage involves continuous risk assessment, scenario planning, and implementing risk controls and safeguards before risks materialize. Proactive risk management aims to minimize the impact of risks by addressing them at their source.

7

1.4.3 *Regulatory Compliance*

A Compliance awareness

Begin by raising awareness of regulatory requirements relevant to the product portfolio. This involves educating employees about applicable laws, standards, and guidelines through training sessions, informational materials, and compliance briefings. The goal is to ensure that all stakeholders are aware of their responsibilities and the importance of compliance.

B Standard compliance

Develop and implement standard compliance procedures to ensure adherence to regulatory requirements. This includes creating documented processes for compliance checks, regular audits, and reporting mechanisms. Standard procedures help in maintaining consistency and accountability in compliance efforts across the organization.

C Continuous compliance

Establish continuous monitoring systems to track compliance with regulatory requirements in real-time. This involves using automated tools and software to monitor changes in regulations, perform ongoing compliance checks, and generate alerts for potential non-compliance issues. Continuous monitoring ensures that the organization remains up-to-date with regulatory changes and can respond promptly to maintain compliance.

D Regulatory foresight

Enhance the compliance process by developing regulatory foresight capabilities. This involves anticipating future regulatory changes and preparing the organization in advance through proactive adjustments to policies, processes, and product designs. Regulatory foresight aims to position the organization as a leader in compliance, reducing the risk of regulatory issues and enhancing its reputation.

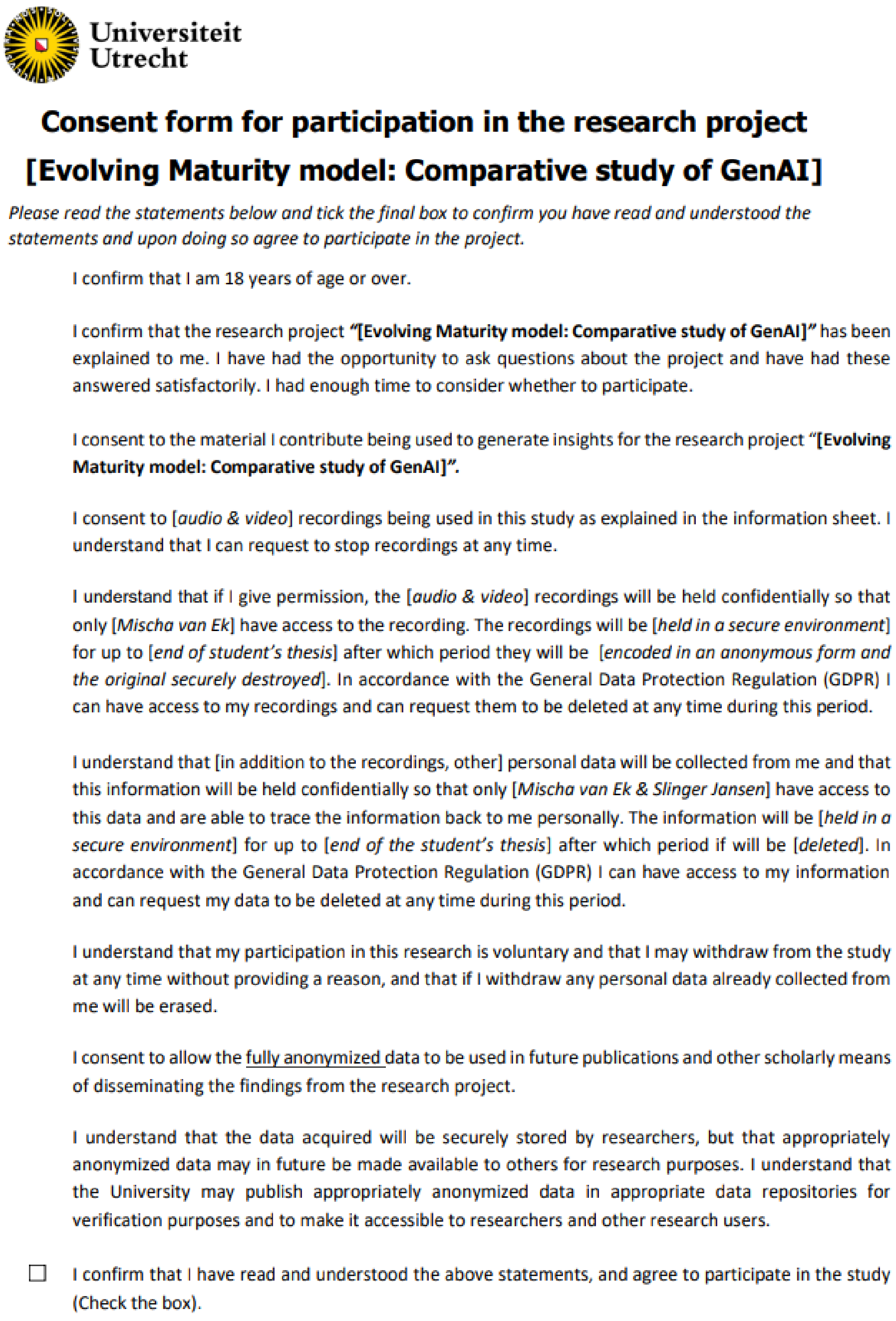
8

FIGURE A.6: Document C: SPPM

Appendix B

Interview

B.1 Informed consent



Universiteit Utrecht

Consent form for participation in the research project

[Evolving Maturity model: Comparative study of GenAI]

Please read the statements below and tick the final box to confirm you have read and understood the statements and upon doing so agree to participate in the project.

I confirm that I am 18 years of age or over.

I confirm that the research project **"[Evolving Maturity model: Comparative study of GenAI]"** has been explained to me. I have had the opportunity to ask questions about the project and have had these answered satisfactorily. I had enough time to consider whether to participate.

I consent to the material I contribute being used to generate insights for the research project "**[Evolving Maturity model: Comparative study of GenAI]".**

I consent to [*audio & video*] recordings being used in this study as explained in the information sheet. I understand that I can request to stop recordings at any time.

I understand that if I give permission, the [*audio & video*] recordings will be held confidentially so that only [*Mischa van Ek*] have access to the recording. The recordings will be [*held in a secure environment*] for up to [*end of student's thesis*] after which period they will be [*encoded in an anonymous form and the original securely destroyed*]. In accordance with the General Data Protection Regulation (GDPR) I can have access to my recordings and can request them to be deleted at any time during this period.

I understand that [in addition to the recordings, other] personal data will be collected from me and that this information will be held confidentially so that only [*Mischa van Ek & Slinger Jansen*] have access to this data and are able to trace the information back to me personally. The information will be [*held in a secure environment*] for up to [*end of the student's thesis*] after which period if will be [*deleted*]. In accordance with the General Data Protection Regulation (GDPR) I can have access to my information and can request my data to be deleted at any time during this period.

I understand that my participation in this research is voluntary and that I may withdraw from the study at any time without providing a reason, and that if I withdraw any personal data already collected from me will be erased.

I consent to allow the fully anonymized data to be used in future publications and other scholarly means of disseminating the findings from the research project.

I understand that the data acquired will be securely stored by researchers, but that appropriately anonymized data may in future be made available to others for research purposes. I understand that the University may publish appropriately anonymized data in appropriate data repositories for verification purposes and to make it accessible to researchers and other research users.

□ I confirm that I have read and understood the above statements, and agree to participate in the study (Check the box).

B.2 Interview protocol

Introduction (5 mins)
The participant has been thanked for participating and asked to confirm the consent form agreement. The consent form is to confirm whether the participant agrees that the interview may be recorded for later use. Once confirmed, the participant was informed of the structure of the interview. The participant was asked about their experience with maturity models, given a FAMM to rate and asked if they noticed which maturity model was AI-generated.

Experience (10-15 mins)
The participant was asked questions related to their experience with MMs. The answers to these questions were used for SRQ 4 by understanding if the participant has any preconceived bias on previous positive or negative experiences with MMs. The questions are:

1. What experience do you have with MMs? Experience could be in developing a maturity model or using a maturity model for an assessment.
    - (if not answered in the previous question) For what purpose did you create/use these maturity model(s)?
    - (if not answered in the previous question) In your experience, was the time spent on creating or utilizing the MM worth it in terms of outcomes?
2. Would you use or develop a maturity model again?

Evaluation (30 mins)
After these questions, the participant have been given four maturity models. Two of these models were related to DevOps and the other two to SPPM. For each topic, one model was created by a human and the other generated by AI. The participant was not informed which model was made by a human or AI. Each MM came with documentation for, if needed, additional understanding of the capabilities within the maturity model.

The participant was tasked with evaluating one maturity model at a time. For each MM evaluation, 4 minutes were given to read the document containing the MM and 3 minutes for the evaluation questions. The order of the documents was mixed within two pools to prevent potential bias. One half of the participant pool started with document A, then B, then C and last D. The other half starts from D, then C, then B and lastly A. A participant could answer each question based on a Likert scale. If time allowed it, the participant was also asked why they gave that Likert rating. The total evaluation should have taken 28 minutes, with the last 2 minutes used to ask which model the participant thought was AI-generated. The evaluation questions that were asked for each model are below.

Completeness (Design criteria 1)

1. Do you think this [subject] maturity model includes all the necessary elements and aspects related to [subject] practices?
    (a) Strongly Disagree
    (b) Disagree
    (c) Neutral

(d) Agree

(e) Strongly Agree

Ease of Use (Design criteria 2)

2. How easy do you think it would be for an organization to use this model to self-assess their [subject]-related business processes?

    (a) Very Difficult

    (b) Difficult

    (c) Neutral

    (d) Easy

    (e) Very Easy

Effectiveness (Design criteria 2)

3. How effective do you think this [subject] maturity model would be in helping organizations improve their [subject]-related business processes?

    (a) Not Effective

    (b) Slightly Effective

    (c) Neutral

    (d) Effective

    (e) Very Effective

Feasibility (Design criteria 2)

4. How feasible do you think it is for organizations to adopt and integrate this [subject] maturity model into their practices?

    (a) Not Feasible

    (b) Slightly Feasible

    (c) Neutral

    (d) Feasible

    (e) Very Feasible

Usefulness (Design criteria 2)

5. How useful do you think this [subject] maturity model is for providing valuable insights regarding [subject] practices?

    (a) Not Useful

    (b) Slightly Useful

    (c) Neutral

    (d) Useful

    (e) Very Useful

Closure (15 mins)
As the interview came to an end, the last points were discussed.

- What role, if any, do you envision GenAI will have with MM?
- Ask if the participant has any remaining questions/topics/comments that have not yet been discussed in the interview.
- Thank the participant.
- Inform the participant when there will be a follow-up on the research.

B.3 Participants

For the domain knowledge,

B.3.1 Domain Knowledge Scale

[1] No knowledge or only a basic understanding of the domain.

[2] Limited knowledge, aware of the basics but not detailed.

[3] Good understanding, competent in the domain but not an expert.

[4] Expert knowledge, fully proficient in the domain.

B.4 Interview Participant Summaries

Author note: ChatGPT was used in the assistance of setting up participant profiles. Further, specific details of participant's experience was left out of their profile to remain compliant to the privacy rules that were stated in the consent form.

Bill Curtis

Role: Initiator of the CMM (1991) and leader of the People CMM development.
Background: Successor of Watts Humphrey as the SEI's Software Process Program Director. Holds a Ph.D. in Organizational Science and Statistics.
Experience: Over 12 years of appraisals, training, and consulting for process maturity improvement. Collaborated with SEI and OMG on 15 standards, including ISO standards. Currently, Vice President of CAST software.

Interviewee A

Role: Professor (University in the Netherlands)
MM Experience: Built and validated a maturity model through assessments and questionnaires. Created a new version to provide a scientific basis, maintaining the core structure. Believes the MM is still up-to-date.

Domain Knowledge:

- **DevOps:** [1] No knowledge, but I know what the term means.
- **SPPM:** [2] Limited knowledge, understands the basics.

AI-Generated Guess:

- **A-B:** B, questioned the model's quick assembly.
- **C-D:** C, suggested spreading the capabilities more.

Evaluation Order: A->D
Expectation: Use GenAI for initial versions and action plans, but an expert should validate the final output.

Interviewee B

Role: Engineering Manager (software company specializing for HR, finance and ERP solutions)
MM Experience: Created an MM used within the organization, but noted its quick obsolescence due to market changes. Concerns about expert bias during refinement.

Domain Knowledge:

- **DevOps:** [3] Competent in the domain, but not an expert.
- **SPPM:** [2] Basic understanding, not detailed.

AI-Generated Guess:

- **A-B:** A, noted the concreteness of the terms.
- **C-D:** D, noted more structure and less content.

Evaluation Order: A->D
Expectation: GenAI useful for condensing information and creating a step plan.

Interviewee C

Role: Product Manager (software company specializing for real-estate solutions)
MM Experience: Uses maturity models occasionally to determine current state and next steps for improvement.

Domain Knowledge:

- **DevOps:** [2] Limited knowledge, not a specialty.
- **SPPM:** [4] Fully knowledgeable, actively works with it.

AI-Generated Guess:

- **A-B:** B, due to the presence of more steps (capabilities), though it was difficult to choose.
- **C-D:** D, thought it was too focused on processes and missed a lot.

Evaluation Order: A->D
Expectation: Specific action plan and creation of a personalized draft.

Interviewee D

Role: Professor (University in the Netherlands)
MM Experience: Created several maturity models and supervised many students in their development. Runs a spinoff company offering paid consultancy on MMs.

Domain Knowledge:

- **DevOps:** [3] Moderate knowledge, akin to a 3 out of 6 on an informal scale.
- **SPPM:** [4] Extensive knowledge, assisted a graduating student with the topic.

AI-Generated Guess:

- **A-B:** B, because level 1 was completely filled up, which is considered a "dead practice."
- **C-D:** D, due to excessive generic jargon and misalignment of labels with capabilities.

Evaluation Order: A->D
Expectation: Skeptical of LLMs due to unreliability. Suggested GenAI could generate a long list of capabilities curated by a researcher and could help in use case generation and self-learning on its model.

Interviewee E

Role: Managing Partner (Company for Business Intelligence platforms)
MM Experience: General knowledge of maturity models, assesses customer maturity using models like Gartner's.

Domain Knowledge:

- **DevOps:** [4] Yes, fully knowledgeable.
- **SPPM:** [1] Limited or no specific knowledge.

AI-Generated Guess:

- **A-B:** A, based on unfinished text, though it was just a guess.
- **C-D:** D, seemed a bit vaguer.

Evaluation Order: A->D
Expectation: Suggested quickly creating an MM with possible mistakes rather than doing thorough research and selection.

Interviewee F

Role: Business Engineer (IT-services and consultancy)
MM Experience: Created an MM and tested it with use cases. MM was sent out with a DIY kit to companies and underwent several rounds of expert reviews.

Domain Knowledge:

- **DevOps:** [3] Decent knowledge, collaborates with DevOps engineers.
- **SPPM:** [2] Basic understanding, defined it as managing the company's software stack.

AI-Generated Guess:

- **A-B:** B, no additional information provided.
- **C-D:** C, because the AI would likely not make such brief descriptions.

Evaluation Order: A->D
Expectation: Suggested using GenAI for first drafts with expert reviews and possibly checking if an existing model is still up-to-date.

Interviewee G

Role: Process Mining Expert (Dutch Bank)
MM Experience: Researched maturity models, particularly in healthcare.

Domain Knowledge:

- **DevOps:** [4] Experienced with DevOps in a company setting.
- **SPPM:** [2] Not a specialty, limited knowledge.

AI-Generated Guess:

- **A-B:** A, due to odd ordering of capabilities.
- **C-D:** C, because the model seemed unrelated to the SPPM topic.

Evaluation Order: D->A
Expectation: Suggested GenAI could generate a measuring instrument, personalized assessments, and provide advice on creating an action plan.

Interviewee H

Role: Information Security Officer (Legal services and consultancy)
MM Experience: Worked with and created several maturity models, particularly in security, focusing on strategy for higher maturity.

Domain Knowledge:

- **DevOps:** [2] Read about it, but has no explicit knowledge.
- **SPPM:** [2] Basic knowledge, had courses on the topic but needs to revisit the knowledge.

AI-Generated Guess:

- **A-B:** A, due to having only three topics and potential for hallucination if it doesn't know what to do.
- **C-D:** C, based on the use of generic knowledge typical of AI.

Evaluation Order: D->A
Expectation: Suggested GenAI might generate additional documentation to clarify specific topics or capabilities.

Interviewee I

Role: Professor (University in the Netherlands)
MM Experience: Supervised maturity model development projects and contributed to academic papers on the subject. Conducted "snapshots" of company maturity levels but lacked long-term hard data due to resource constraints.

Domain Knowledge:

- **DevOps:** [2] Limited knowledge.
- **SPPM:** [2] Not a domain expert.

AI-Generated Guess:

- **A-B:** A, based on the progression stage and inclusion of Kaizen and Lean.
- **C-D:** D, due to how the dimensions were structured.

Evaluation Order: D->A
Expectation: Suggested GenAI could help structure a lecture or MM, provided the user has domain knowledge and trusts the model.

Interviewee J

Role: Consultant (Business consultancy and services)
MM Experience: Created several maturity models and published a book on the subject. Developed MMs primarily through observation and iterative feedback from domain experts.

Domain Knowledge:

- **DevOps:** [3] Aware of microservices, co-authored a book on Agile (rated 3 due to no direct "yes" response).
- **SPPM:** [3] General knowledge, experience from study, and customer information roadmaps.

AI-Generated Guess:

- **A-B:** B, no additional information provided.
- **C-D:** D, misunderstood a comment about one prompt, possibly influencing the guess.

Evaluation Order: D->A
Expectation: Suggested GenAI could provide a clear view, create MMs, and search for relevant information.

Interviewee K

Role: Consultant (Business consultancy and services)
MM Experience: Works with maturity models for clients, focusing on assessment questionnaires and roadmap status.

Domain Knowledge:

- **DevOps:** [4] Fully knowledgeable.
- **SPPM:** [4] Fully knowledgeable, with experience in product portfolio management.

AI-Generated Guess:

- **A-B:** B, due to a weird difference from now going to level 8.
- **C-D:** C, because it aligned better with reality.

Evaluation Order: D->A
Expectation: Suggested using GenAI for updating assessment questions, specific frameworks, and creating action plans for the next steps.

Interviewee L

Role: Consultant (Business consultancy and services)
MM Experience: Created a maturity model related to privacy and uses maturity models for work. Believes a maturity model is never truly "done" and sees information needs as a trigger to start new MM creation.

Domain Knowledge:

- **DevOps:** [2] Slightly more knowledgeable than SPPM, but not much.
- **SPPM:** [1] Basic awareness, no specific knowledge.

AI-Generated Guess:

- **A-B:** A, possibly due to the level 10 distribution.
- **C-D:** C, based on gut feeling, maybe because it had more text, as GenAI would produce.

Evaluation Order: D->A
Expectation: Suggested GenAI could do literature reviews, create roadmaps, business cases, and action plans, but stressed that information needs to be grounded.

Interviewee M

Role: Principal Consultant and owner (International trade and development)
MM Experience: Started with CMM 30 years ago, certified lead appraiser, ambassador, and critic. Hosts appraisal sessions, mostly in China, as it's the largest market for CMMI now.

Domain Knowledge:

- **DevOps:** [1] No knowledge.
- **SPPM:** [1] Not directly knowledgeable.

AI-Generated Guess:

- **A-B:** B, felt it had more insightful information and depth.
- **C-D:** C, similar reason, seemed more robust, while D lacked examples.

Evaluation Order: A->D
Expectation: Suggested CMMI is becoming too broad. GenAI could help pilot solid foundations and refine them over time.

Interviewee N

Role: Professor (University in North America)
MM Experience: Worked at a military company and software engineering institute, wherein (after) became involved with a Software MM. Did later distance themselves of a popular MM that was adopting a different approach, which caused some differences.

Domain Knowledge:

- **DevOps:** [4] Yes, teaches it, though questioned the need for a DevOps MM.
- **SPPM:** [4] Teaches product management and portfolio lines.

AI-Generated Guess:

- **A-B:** B, consistent and less ad-hoc, believed to be AI-generated.
- **C-D:** D, thought AI-generated due to its blind adherence to creating a maturity model.

Evaluation Order: A->D
Expectation: Suggested AI might create poor MMs without expert input. Advocated for combining AI with expert teams to improve MM development.

Interviewee O

Role: Owner (Consultancy business)
MM Experience: Contributed to the CMM, served as a lead appraiser, and focused on keeping business processes up to date.

Domain Knowledge:

- **DevOps:** [4] Yes, manages IT environments for the U.S. government.
- **SPPM:** [4] Yes, developed portfolio management for a military client.

AI-Generated Guess:

- **A-B:** A, included a lot of buzzwords but was just a guess.
- **C-D:** D, thought AI would generate these topics.

Evaluation Order: D->A
Expectation: Suggested GenAI could help identify suitable maturity models for solving business problems, though unsure about its generation capabilities.

Interviewee P

Role: Product Manager (IT consultancy and services)
MM Experience: Authored papers on maturity models but questioned their relevance over time due to shifts from academia to business. Believes that MMs are not one-size-fits-all and must be adapted to organizational context.

Domain Knowledge:

- **DevOps:** [2] Baseline understanding, has not managed or actively worked in it.
- **SPPM:** [3] General knowledge and experience managing products.

AI-Generated Guess:

- **A-B:** A, odd order of Kaizen and Lean in continuous evolution.
- **C-D:** D, based on terminology like "constructed environmental."

Evaluation Order: D->A
Expectation: Suggested GenAI is suitable for the beginning phase of MM development but could struggle with detailed work. Emphasized the influence of personal knowledge on GenAI's effectiveness.

B.5 Overview of comments

TABLE B.1: Classification of Suggestions with Rationale

Interviewee	Suggestion	Phase	Rationale	Viewpoint
a	Generate initial version	Design	This should serve as the basis for MM development. Not in scoping as that would not do maturity leveling.	Developer
a	Make an improvement report	Take corrective action	Need to know what to do next after applying the model is with corrective actions	User
b	Condense gathered info	Scope	chosen in scope as the developer would still be gathering information. No maturity leveling going on yet but a topic is already chosen.	Developer
c	Generate personalized model	Design	This should serve as the basis for MM development. Not in scoping as that would not do maturity leveling.	Developer
c	Create a focused action plan	Take corrective action	Same as previously mentioned. Actions after the MM is applied would be taken	User
d	Generate long list of capabilities	Scope	This is very much what happens in scope phase. Topic is understood but need to get a broad overview what MM should include	Developer
d	Use case generation for evaluation	Evaluation	When evaluating it can also be done through a use case. So therefore in evaluation phase chosen	Developer
e	Skip research for the right MM	Select model	Instead of generating, could just be created. This is bypassing the selection process. But Still would classify under select model	User
f	Create the first draft	Design	This is the basis for creating an MM, with applying maturity levels. So falls under design.	Developer

Continued on next page

Interviewee	Suggestion	Phase	Rationale	Viewpoint
f	Update existing model	Reflect	Checking once a MM is done and evaluated, goes into reflection	Developer
g	Link assessment tool with AI	Prepare deployment	This could also fall under applying the model, however chosen to put in deployment as it is more so a byproduct without actually integrally applying the model.	User
g	Create an action plan	Take corrective action	Once MM is used, time for an action plan. So it is taking the corrective action	User
h	Generate extra documentation	Apply model	This might also fall under prepare deployment. However this was told in the context of when the MM was in use, more documentation might be of use. Therefore applying the model was chosen	User
i	Help choose dimensions	Scope	Participant gave the context of that before they would start their lecture, GenAI would be used to see what should be in the lecture. Based on what was already on the schedule, gaps could be identified. Now with MM context, it is help identify what dimension could also be chosen. as no maturity leveling took place, this falls under scoping.	Developer
j	Provide inspiration	Identify opportunity	Providing inspiration is very broad, but was referred to the beginning phase so it is identifying an opportunity	Developer
j	Create a draft MM	Design	Creating a draft comes back with the same argument as it designing the MM with maturity leveling.	Developer

Continued on next page

Interviewee	Suggestion	Phase	Rationale	Viewpoint
k	Verify all questions are up to date	Reflect	This was reflecting if the assessment questions are still up to date. In some way, It could also fall under user prepare deployment, but was chosen in the developer context.	Developer
k	Create a specific action plan	Take corrective action	Once the MM is applied, time for some action.	User
l	Provide a roadmap	Take corrective action	Once the MM is used, the actions could be put on a roadmap, so such a roadmap would fall under the corrective actions. This is not a roadmap as to MM should later be applied or where we stand. Moreso where should the organization go forward to.	User
m	Use CMMI as a foundation	Scope	Suggests using existing models as the baseline.	Developer
n	Separate AI/creativity into a dimension	Scope	The consideration was within the CMMI that it would get its own separate AI domain dimension. This suggestion was not dug into much detail during the interview. In general thought to be in the scope as it is considering what topics to include in a MM. The suggestion how it was understood is always including it in a MM.	Developer
n	Build bad models but improve with experts	Design	This is making the draft but with human supervision. The suggestion gave strong emphasis on the evaluation, however suggested about partly automating the design with GenAI.	Developer

Continued on next page

Interviewee	Suggestion	Phase	Rationale	Viewpoint
o	Identify starting point related to problem	Identify need	Context was given in that the user would just start out, so it was placed in identify need.	User
o	Find appropriate MM	Select model	self described	User
p	Provide inspiration for the initial phase	Identify opportunity	This could have also been classified under scoping. However it was emphasized to get inspired to help with the first phase. Therfore it is identify opportunity	Developer
p	Ask AI for user actions	Apply model	Asking an AI to help the user with actions was given in the context of when the user would use the MM in an assessment. Therefore applying the model was chosen as a classification	User

Appendix C

Coding Scheme

Interview codes from the Experience phase

Code	APA reference
ex-1	(Interviewee A, person-to-person, 15 May, 2024)
ev-2	(Interviewee B, Online, 16 May, 2024)
ev-3	(Interviewee C, Online, 16 May, 2024)
ev-4	(Interviewee D, Online, 17 May, 2024)
ev-5	(Interviewee E, person-to-person, 27 May, 2024)
ev-6	(Interviewee F, Online, 28 May, 2024)
ev-7	(Interviewee G, person-to-person, 30 Aug, 2024)
ev-8	(Interviewee H, person-to-person, 3 June, 2024)
ex-9	(Interviewee I, person to person, 6 June, 2024)
ex-10	(Interviewee J, person-to-person, 7 June, 2024)
ex-11	(Interviewee K, person-to-person, 7 June, 2024)
ex-12	(Interviewee L, person-to-person, 7 June, 2024)
ex-13	(Interviewee M, Online, 10 June, 2024)
ex-14	(Interviewee N, Online, 17 June, 2024)
ex-15	(Interviewee O, Online, 25 June, 2024)
ex-16	(Interviewee P, Online, 16 July, 2024)

TABLE C.1: Coding scheme: experience

Interviewee codes from the evaluation phase for document A (Ai-Generated DevOps FAMM)

Code	APA Reference
ev-a-1	(Interviewee K, person-to-person, 7 June, 2024)
ev-a-2	(Interviewee M, Online, 10 June, 2024)
ev-a-3	(Interviewee N, Online, 17 June, 2024)
ev-a-4	(Interviewee H, person-to-person, 3 June, 2024)
ev-a-5	(Interviewee K, person-to-person, 7 June, 2024)
ev-a-6	(Interviewee K, person-to-person, 7 June, 2024)
ev-a-7	(Interviewee L, person-to-person, 7 June, 2024)
ev-a-8	(Interviewee O, Online, 25 June, 2024)
ev-a-9	(Interviewee K, person-to-person, 7 June, 2024)
ev-a-10	(Interviewee L, person-to-person, 7 June, 2024)
ev-a-11	(Interviewee G, person-to-person, 30 Aug, 2024)
ev-a-12	(Interviewee L, person-to-person, 7 June, 2024)
ev-a-13	(Interviewee N, Online, 17 June, 2024)
ev-a-14	(Interviewee M, Online, 10 June, 2024)
ev-a-15	(Interviewee E, person-to-person, 27 May, 2024)
ev-a-16	(Interviewee D, Online, 17 May, 2024)
ev-a-17	(Interviewee B, Online, 16 May, 2024)

TABLE C.2: Coding scheme: Evaluation [document A]

Interviewee codes from the evaluation phase for document B (human created DevOps FAMM)

Code	APA Reference
ev-b-1	(Interviewee K, person-to-person, 7 June, 2024)
ev-b-2	(Interviewee H, person-to-person, 3 June, 2024)
ev-b-3	(Interviewee N, Online, 17 June, 2024)
ev-b-4	(Interviewee M, Online, 10 June, 2024)
ev-b-5	(Interviewee N, Online, 17 June, 2024)
ev-b-6	(Interviewee C, Online, 16 May, 2024)
ev-b-7	(Interviewee E, person-to-person, 27 May, 2024)
ev-b-8	(Interviewee G, person-to-person, 30 Aug, 2024)
ev-b-9	(Interviewee L, person-to-person, 7 June, 2024)
ev-b-10	(Interviewee N, Online, 17 June, 2024)
ev-b-11	(Interviewee L, person-to-person, 7 June, 2024)
ev-b-12	(Interviewee M, Online, 10 June, 2024)
ev-b-13	(Interviewee B, Online, 16 May, 2024)

TABLE C.3: Coding scheme: Evaluation [document B]

Interviewee codes from the evaluation phase for document C (Ai-Generated SPPM FAMM)

Code	APA Reference
ev-c-1	(Interviewee D, Online, 17 May, 2024)
ev-c-2	(Interviewee H, person-to-person, 3 June, 2024)
ev-c-3	(Interviewee M, Online, 10 June, 2024)
ev-c-4	(Interviewee D, Online, 17 May, 2024)
ev-c-5	(Interviewee N, Online, 17 June, 2024)
ev-c-6	(Interviewee K, person-to-person, 7 June, 2024)
ev-c-7	(Interviewee F, Online, 28 May, 2024)
ev-c-8	(Interviewee J, person-to-person, 7 June, 2024)
ev-c-9	(Interviewee M, Online, 10 June, 2024)
ev-c-10	(Interviewee N, Online, 17 June, 2024)
ev-c-11	(Interviewee O, Online, 25 June, 2024)
ev-c-12	(Interviewee I, person-to-person, 6 June, 2024)
ev-c-13	(Interviewee O, Online, 25 June, 2024)

TABLE C.4: Coding scheme: Evaluation [document C]

Interviewee codes from the evaluation phase for document D (human created SPPM FAMM)

Code	**APA Reference**
ev-d-1	(Interviewee D, Online, 17 May, 2024)
ev-d-2	(Interviewee B, Online, 16 May, 2024)
ev-d-3	(Interviewee M, Online, 10 June, 2024)
ev-d-4	(Interviewee N, Online, 17 June, 2024)
ev-d-5	(Interviewee E, person-to-person, 27 May, 2024)
ev-d-6	(Interviewee H, person-to-person, 3 June, 2024)
ev-d-7	(Interviewee K, person-to-person, 7 June, 2024)
ev-d-8	(Interviewee I, person-to-person, 6 June, 2024)
ev-d-9	(Interviewee P, Online, 16 July, 2024)
ev-d-10	(Interviewee B, Online, 16 May, 2024)
ev-d-11	(Interviewee P, Online, 16 July, 2024)

TABLE C.5: Coding scheme: Evaluation [document D]

What guess the participant had which document was AI generated

Code	**APA Reference**
ev-aig-1	(Interviewee H, person-to-person, 3 June, 2024)
ev-aig-2	(Interviewee G, person-to-person, 30 Aug, 2024)
ev-aig-3	(Interviewee M, Online, 10 June, 2024)
ev-aig-4	(Interviewee A, person-to-person, 15 May, 2024)
ev-aig-5	(Interviewee P, Online, 16 July, 2024)
ev-aig-6	(Interviewee B, Online, 16 May, 2024)
ev-aig-7	(Interviewee K, person-to-person, 7 June, 2024)
ev-aig-8	(Interviewee N, Online, 17 June, 2024)
ev-aig-9	(Interviewee D, Online, 17 May, 2024)
ev-aig-10	(Interviewee N, Online, 17 June, 2024)
ev-aig-11	(Interviewee M, Online, 10 June, 2024)
ev-aig-12	(Interviewee O, Online, 25 June, 2024)
ev-aig-13	(Interviewee B, Online, 16 May, 2024)
ev-aig-14	(Interviewee P, Online, 16 July, 2024)
ev-aig-15	(Interviewee E, person-to-person, 27 May, 2024)

TABLE C.6: Coding scheme: Evaluation [AI-Guess]

Envisions of Genai from the closure (with regards to Developer viewpoint)

Code	**APA Reference**
c-d-1	(Interviewee A, person-to-person, 15 May, 2024)
c-d-2	(Interviewee B, Online, 16 May, 2024)
c-d-3	(Interviewee C, Online, 16 May, 2024)
c-d-4	(Interviewee D, Online, 17 May, 2024)
c-d-5	(Interviewee D, Online, 17 May, 2024)
c-d-6	(Interviewee F, Online, 28 May, 2024)
c-d-7	(Interviewee F, Online, 28 May, 2024)
c-d-8	(Interviewee I, person-to-person, 6 June, 2024)
c-d-9	(Interviewee J, person-to-person, 7 June, 2024)
c-d-10	(Interviewee J, person-to-person, 7 June, 2024)
c-d-11	(Interviewee K, person-to-person, 7 June, 2024)
c-d-12	(Interviewee M, Online, 10 June, 2024)
c-d-13	(Interviewee N, Online, 17 June, 2024)
c-d-14	(Interviewee N, Online, 17 June, 2024)

TABLE C.7: Coding scheme: Closure [Developer viewpoint]

Envisions of Genai from the closure (with regards to user viewpoint)

Code	**APA Reference**
c-u-1	(Interviewee A, person-to-person, 15 May, 2024)
c-u-2	(Interviewee C, Online, 16 May, 2024)
c-u-3	(Interviewee E, person-to-person, 27 May, 2024)
c-u-4	(Interviewee G, person-to-person, 30 Aug, 2024)
c-u-5	(Interviewee G, person-to-person, 30 Aug, 2024)
c-u-6	(Interviewee H, person-to-person, 3 June, 2024)
c-u-7	(Interviewee K, person-to-person, 7 June, 2024)
c-u-8	(Interviewee L, person-to-person, 7 June, 2024)
c-u-9	(Interviewee O, Online, 25 June, 2024)
c-u-10	(Interviewee O, Online, 25 June, 2024)
c-u-11	(Interviewee P, Online, 16 July, 2024)

TABLE C.8: Coding scheme: Closure [User viewpoint]

Bibliography

Achiam, Josh et al. (2023). “Gpt-4 technical report”. In: *arXiv preprint arXiv:2303.08774*.

(ACQUISITION, OFFICE OF THE UNDER SECRETARY OF DEFENSE and TECHNOLOGY) WASHINGT ON DC (1996). *DoD Guide to Integrated Product and Process Development (Version 1.0).*

Adekunle, Samuel Adeniyi et al. (2022). “A critical review of maturity model development in the digitisation era”. In: *Buildings* 12.6, p. 858.

Aljowder, Thajba, Mazen Ali, and Sherah Kurnia (2023). “Development of a Maturity Model for Assessing Smart Cities: A Focus Area Maturity Model”. In: *Smart Cities* 6.4, pp. 2150–2175.

Alves, Joana Filipa Alemão (2013). “Finding maturity evolution paths for organisational use of information”. PhD thesis. master thesis, Instituto Superior Técnico, June, available at: https://fenix . . .

Amorim, Tiago Luiz Buarque de (2021). “Model-based systems engineering maturity improvement in industry”. PhD thesis. Universität zu Köln.

Andersin, Jari (2004). “TPI–a model for Test Process Improvement”. In: *Seminar. University of Helsinki, Helsinki-Finland*.

Antony, Victor Nikhil and Chien-Ming Huang (2023). “ID. 8: Co-Creating Visual Stories with Generative AI”. In: *arXiv preprint arXiv:2309.14228*.

Bach, James (1994). “The Immaturity of the CMM”. In: *American Programmer* 7, pp. 13–13.

Bakhshi, Rena et al. (2024). “RSMM: A Framework to Assess Maturity of Research Software Project”. In: *arXiv preprint arXiv:2406.01788*.

Bate, Roger et al. (1995). “A systems engineering capability maturity model, version 1.1”. In: *Software Engineering Institute*.

Becker, Joerg et al. (2010). “Maturity models in IS research”. In.

Becker, Jörg, Ralf Knackstedt, and Jens Pöppelbuß (2009). “Developing maturity models for IT management: A procedure model and its application”. In: *Business & information systems engineering* 1, pp. 213–222.

BPMIT, IM0602 Voorbereiden Afstuderen, Rian Kistemaker, and Meelezer Dr Laury Bollen (2021). “Gegevensbeschikbaarheid en-kwaliteit voor Data Analytics. Data availability and quality for Data Analytics.” In.

Brynjolfsson, Erik, Danielle Li, and Lindsey R Raymond (2023). *Generative AI at work*. Tech. rep. National Bureau of Economic Research.

Burnstein, Ilene et al. (1998). “A model to assess testing process maturity”. In: *Crosstalk* 11.11, pp. 26–30.

Chrissis, Mary Beth, Mike Konrad, and Sandra Shrum (2011). *CMMI for development: guidelines for process integration and product improvement*. Pearson Education.

CMMI Institute (2023). *CMMI Version 3.0 Updates*. https://cmmiinstitute.com/getattachment/47a7c84e-472c-4f7f-a473-ddc21c6ae045/attachment.aspx. Accessed: 2023-02-21.

De Bruin, Tonia et al. (2005). “Understanding the main phases of developing a maturity assessment model”. In: *Australasian conference on information systems (ACIS)*. Australasian Chapter of the Association for Information Systems, pp. 8–19.

Defense, United States. Department of and Larry Druffel (1982). *Strategy for a DoD Software Initiative*. Office of the Under Secretary of Defense.

Demerouti, Evangelia et al. (2001). "The job demands-resources model of burnout." In: *Journal of Applied psychology* 86.3, p. 499.

Dhariwal, Prafulla et al. (2020). "Jukebox: A generative model for music". In: *arXiv preprint arXiv:2005.00341*.

Duncan, Jeffrey et al. (2013). "A focus area maturity model for a statewide master person index". In: *Online Journal of Public Health Informatics* 5.2.

Elmaallam, Mina, Hicham Bensaid, and Abdelaziz Kriouile (2019). "A Maturity Model for Assessing IS Risk Management Activity Considering the Dependencies Between Its Elements." In: *Comput. Inf. Sci.* 12.1, pp. 98–111.

Feijter, Rico de et al. (2017). "Towards the adoption of DevOps in software product organizations: A maturity model approach". In: URL: https://api.semanticscholar.org/CorpusID:115646011.

Fraser, Peter, James Moultrie, and Mike Gregory (2002). "The use of maturity models/grids as a tool in assessing product development capability". In: *IEEE international engineering management conference*. Vol. 1. IEEE, pp. 244–249.

Gemini Team, Google (2023). *Gemini: A Family of Highly Capable Multimodal Models*. Tech. rep. Google. URL: https://storage.googleapis.com/deepmind-media/gemini/gemini_1_report.pdf.

Goldenson, Dennis and Diane L Gibson (2003). "Demonstrating the impact and benefits of CMMI: an update and preliminary results". In.

Hermanns, TI (2014). "Towards an IT maturity model for general practice information systems". MA thesis.

Hevner, Alan, Samir Chatterjee, and Juhani Iivari (2010). "Twelve Theses on Design Science Research in Information Systems". In: *Design research in information systems: Theory and practice*, pp. 43–62.

Hevner, Alan R (2007). "A three cycle view of design science research". In: *Scandinavian journal of information systems* 19.2, p. 4.

Hillegersberg, Jos van (2019). "The need for a maturity model for maturity modeling". In: *The Art of Structuring: Bridging the Gap Between Information Systems Research and Practice*, pp. 145–151.

Humphrey, Watts S (1988). "Characterizing the software process: a maturity framework". In: *IEEE software* 5.2, pp. 73–79.

— (1989). *Managing the software process*. Addison-Wesley Longman Publishing Co., Inc.

Humphrey, Watts S and David H Kitson (1987). *Preliminary report on conducting SEI-assisted assessments of software engineering capability*. Carnegie Mellon University, Software Engineering Institute.

Jagroep, Erik et al. (May 2013). "Software Product Portfolio Management: Towards improvement of current practice". In.

Jansen, Slinger (2020). "A focus area maturity model for software ecosystem governance". In: *Information and Software Technology* 118, p. 106219.

Kerkhof, J (2012). "Social media: towards a social media maturity model". MA thesis.

Koop, Reinder, Ruud Rooimans, and Martijn de Theye (2003). *Regatta: ICT-implementaties als uitdaging voor een vier-met-stuurman*. TenHagenStam.

Lasrado, Lester Allan, Ravi Vatrapu, and Kim Normann Andersen (2015). "Maturity models development in is research: a literature review". In.

Linstone, Harold A, Murray Turoff, et al. (1975). *The delphi method*. Addison-Wesley Reading, MA.

Maier, Anja, James Moultrie, and P John Clarkson (2009). "Developing maturity grids for assessing organisational capabilities: Practitioner guidance". In: *4th International Conference on Management Consulting: Academy of Management*.

Mettler, Tobias (2011). "Maturity assessment models: a design science research approach". In: *International Journal of Society Systems Science* 3.1-2, pp. 81–98.

Mettler, Tobias and Peter Rohner (2009). "Situational maturity models as instrumental artifacts for organizational design". In: *Proceedings of the 4th international conference on design science research in information systems and technology*, pp. 1–9.

Mijsters, GY (2019). "A Focus Area Maturity Model for Data as a Platform". MA thesis.

Muller, Louis and Mike Hart (2016). "Updating business intelligence and analytics maturity models for new developments". In: *Decision Support Systems VI-Addressing Sustainability and Societal Challenges: 2nd International Conference, ICDSST 2016, Plymouth, UK, May 23–25, 2016, Proceedings 2*. Springer, pp. 137–151.

Muszynski, Michel (2023). "A Focus Area Maturity Model for Privacy-by-Design". MA thesis.

Myllyaho, Lalli et al. (2021). "Systematic literature review of validation methods for AI systems". In: *Journal of Systems and Software* 181, p. 111050. ISSN: 0164-1212. DOI: https://doi.org/10.1016/j.jss.2021.111050. URL: https://www.sciencedirect.com/science/article/pii/S0164121221001473.

Normann Andersen, Kim et al. (2020). "Ten misunderstandings about maturity models". In: *The 21st Annual International Conference on Digital Government Research*, pp. 261–266.

OpenAI (2024). *Introducing the GPT Store*. https://openai.com/blog/introducing-the-gpt-store.

Overeem, Michiel, Max Mathijssen, and Slinger Jansen (2022). "API-m-FAMM: A focus area maturity model for API Management". In: *Information and Software Technology* 147, p. 106890.

Paulk, Mark C et al. (1991). *Capability maturity model for software*. Citeseer.

Peffers, Ken et al. (2007). "A design science research methodology for information systems research". In: *Journal of management information systems* 24.3, pp. 45–77.

Pflügler, Christoph, Markus Böhm, and Helmut Krcmar (2015). "Coping with IT carve-out projects-towards a maturity model". In.

Pöppelbuß, Jens and Maximilian Röglinger (2011). "What makes a useful maturity model? A framework of general design principles for maturity models and its demonstration in business process management". In.

Prat, Nicolas, Isabelle Comyn-Wattiau, and Jacky Akoka (2015). "A taxonomy of evaluation methods for information systems artifacts". In: *Journal of Management Information Systems* 32.3, pp. 229–267.

Quashie, Stephan (n.d.). "Preparing for Digital Twin Implementation in Warehouses: Assessing and Improving Maturity". In: ().

Raad, Benjamin Carlos (2024). "Developing a Focus Area Maturity Model for Data-driven Decision-making". MA thesis. University of Twente.

Ramesh, Aditya et al. (2021). "Zero-shot text-to-image generation". In: *International conference on machine learning*. Pmlr, pp. 8821–8831.

Reeder, L de (2019). "Defining maturity of agile requirements engineering practices". MA thesis.

Sanchez-Puchol, Felix and Joan A Pastor-Collado (2017). "Focus area maturity models: A comparative review". In: *Information Systems: 14th European, Mediterranean, and Middle Eastern Conference, EMCIS 2017, Coimbra, Portugal, September 7-8, 2017, Proceedings 14*. Springer, pp. 531–544.

Santos, Luis Fernando Lins dos (2023). "Evaluating and Comparing Generative-based Chatbots Based on Process Requirements". MA thesis. University of Waterloo.

Smits, Daniel and Jos van Hillegersberg (2015). "IT Governance Maturity: Developing a Maturity Model Using the Delphi Method". In: *2015 48th Hawaii International Conference on System Sciences*, pp. 4534–4543. DOI: 10.1109/HICSS.2015.541.

Sprockel, YHB (2013). "The impact of Software Product Lines from a Product Management Perspective". MA thesis.

Spruit, Marco and Katharina Pietzka (2015). "MD3M: The master data management maturity model". In: *Computers in Human Behavior* 51, pp. 1068–1076.

Spruit, Marco and Martijn Röling (2014). "ISFAM: the information security focus area maturity model". In.

Staples, Mark et al. (2007). "An exploratory study of why organizations do not adopt CMMI". In: *Journal of systems and software* 80.6, pp. 883–895.

Steenbergen, Marlies van, Martin van den Berg, and Sjaak Brinkkemper (2008). "A balanced approach to developing the enterprise architecture practice". In: *Enterprise Information Systems: 9th International Conference, ICEIS 2007, Funchal, Madeira, June 12-16, 2007, Revised Selected Papers 9*. Springer, pp. 240–253.

Steenbergen, Marlies van et al. (2010). "The design of focus area maturity models". In: *Global Perspectives on Design Science Research: 5th International Conference, DESRIST 2010, St. Gallen, Switzerland, June 4-5, 2010. Proceedings. 5*. Springer, pp. 317–332.

— (2013). "Improving IS functions step by step: the use of focus area maturity models". In: *Scandinavian Journal of Information Systems* 25.2, p. 2.

Team, CMMI Product (2002). "Capability maturity model® integration (CMMI SM), version 1.1". In: *CMMI for systems engineering, software engineering, integrated product and process development, and supplier sourcing (CMMI-SE/SW/IPPD/SS, V1. 1)* 2.

Team, CMMI Product et al. (2010). "CMMI for development, version 1.3. Software Engineering Institute". In: *Carnegie Mellon University, Pittsburgh, PA*.

Van Looy, Amy et al. (2013). "Choosing the right business process maturity model". In: *Information & Management* 50.7, pp. 466–488.

Vaswani, Ashish et al. (2017). "Attention is all you need". In: *Advances in neural information processing systems* 30.

Waldt, Gerrit Van der (2013). "Disaster risk management: Disciplinary status and prospects for a unifying theory". In: *Jamba: journal of disaster risk studies* 5.2, pp. 1–11.

Weerd, Inge van de, Willem Bekkers, and Sjaak Brinkkemper (2010). "Developing a maturity matrix for software product management". In: *Software Business: First International Conference, ICSOB 2010, Jyväskylä, Finland, June 21-23, 2010. Proceedings 1*. Springer, pp. 76–89.

Wendler, Roy (2012). "The maturity of maturity model research: A systematic mapping study". In: *Information and software technology* 54.12, pp. 1317–1339.

White, Jules et al. (2023). "A prompt pattern catalog to enhance prompt engineering with chatgpt". In: *arXiv preprint arXiv:2302.11382*.

Yigit Ozkan, Bilge, Sonny van Lingen, and Marco Spruit (2021). "The cybersecurity focus area maturity (CYSFAM) model". In: *Journal of Cybersecurity and Privacy* 1.1, pp. 119–139.

Yigit Ozkan, Bilge et al. (2020). "Modelling adaptive information security for SMEs in a cluster". In: *Journal of Intellectual Capital* 21.2, pp. 235–256.

Zhang, Peng and Maged N Kamel Boulos (2023). "Generative AI in medicine and healthcare: Promises, opportunities and challenges". In: *Future Internet* 15.9, p. 286.

Zhao, Wayne Xin et al. (2023). "A survey of large language models". In: *arXiv preprint arXiv:2303.18223*.

Zwienen, Mart van et al. (2019). "A Process for Tailoring Domain-Specific Enterprise Architecture Maturity Models". In: *Enterprise, Business-Process and Information Systems Modeling: 20th International Conference, BPMDS 2019, 24th International Conference, EMMSAD 2019, Held at CAiSE 2019, Rome, Italy, June 3–4, 2019, Proceedings 20*. Springer, pp. 196–211.