

**HUMAN AI CO-CREATION IN SOFTWARE DEVELOPMENT:
A CASE STUDY OF SGU MBKM SYSTEM**

By

Dennis Reynaldo
22351039

MASTER'S DEGREE

in

MASTER OF INFORMATION TECHNOLOGY
FACULTY OF ENGINEERING AND INFORMATION TECHNOLOGY



SWISS GERMAN UNIVERSITY
The Prominence Tower
Jalan Jalur Sutera Barat No. 15, Alam Sutera
Tangerang, Banten 15143 - Indonesia

Revision after the Thesis Defense on 13 August 2025

STATEMENT BY THE AUTHOR

I hereby declare that this submission is my own work and to the best of my knowledge, it contains no material previously published or written by another person, nor material which to a substantial extent has been accepted for the award of any other degree or diploma at any educational institution, except where due acknowledgement is made in the thesis.



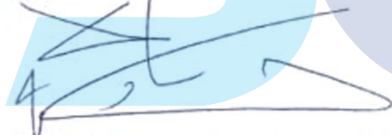
Dennis Reynaldo, S.Ds

Student

27  2025

Date

Approved by:



Dr. Maulahikmah Galinium, S.Kom, M.Sc

Thesis Advisor

27.8.2025

Date



Leonard Rusli, Ph.D

Thesis Co-Advisor

27-8-2025

Date



Dr. Maulahikmah Galinium, S.Kom, M.Sc

Dean

27.8.2025

Date

Dennis Reynaldo

ABSTRACT

HUMAN AI COLABORATION IN SOFTWARE DEVELOPMENT: A CASE STUDY OF SGU MBKM SYSTEM

By

Dennis Reynaldo

Dr. Maulahikmah Galinium, S.Kom, M.Sc, Advisor

Leonard Rusli, Ph.D, Co-Advisor

SWISS GERMAN UNIVERSITY



This study explores the integration of Large Language Models (LLMs), particularly ChatGPT, as co-creators in the software development process of the MBKM system at Swiss German University (SGU). In response to limited IT resources and the need to manage complex student activity data, the development adopted a modified Feature-Driven Development (FDD) approach, with LLMs assisting in key phases such as requirement elicitation, system modelling, UI/UX design, coding, and testing. A structured prompting framework was applied to optimize interactions with the LLM, ensuring more accurate and usable outputs. The final system prototype was evaluated through development artifacts and user feedback, showing that LLM-assisted development improved efficiency by enhancing documentation completeness by 26,7% and reducing development time by 60%. Despite some limitations, such as visual output inconsistencies and contextual limitations, the collaboration between human developers and LLMs proved valuable in accelerating development and enhancing creativity.

Keywords: Software Development, Human-AI collaboration, ChatGPT, Software Development Life Cycle, Prompt Engineering.



DEDICATION

This thesis is dedicated to God, whose grace has sustained me throughout this journey; to my family, for their endless love, patience, and encouragement; and to my advisors, for their invaluable guidance, support, and belief in my work.



ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my advisors for their invaluable guidance, encouragement, and expertise throughout the completion of this thesis. Their insights have been essential in shaping the direction and depth of this research. I am also thankful to the SGU IST team for their collaboration and technical contributions during the development of the MBKM system.

Special thanks to the faculty and staff of Swiss German University for providing a supportive academic environment and access to valuable resources. To my family, thank you for your unwavering love, patience, and encouragement throughout this journey.

Above all, I thank God for granting me the strength, clarity, and perseverance to complete this work.

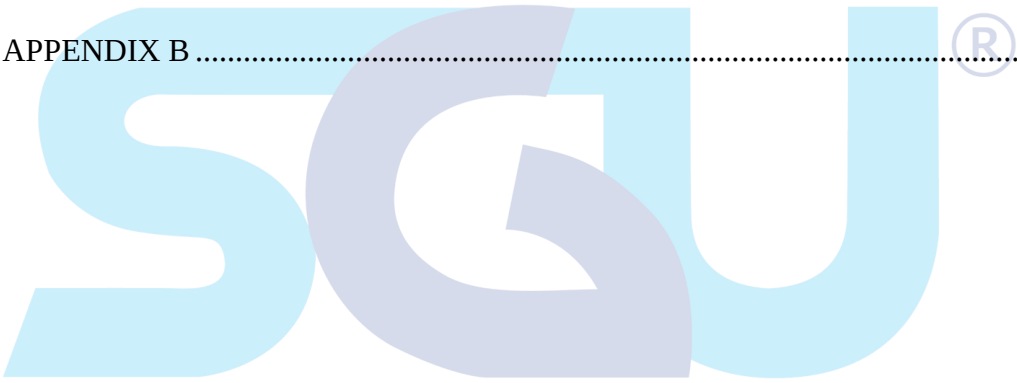
TABLE OF CONTENTS

	Page
STATEMENT BY THE AUTHOR.....	Error! Bookmark not defined.
ABSTRACT.....	3
DEDICATION	5
ACKNOWLEDGEMENTS	6
TABLE OF CONTENTS.....	7
LIST OF FIGURES	11
LIST OF TABLES	13
CHAPTER 1 – INTRODUCTION	14
1.1 Background.....	14
1.2 Research Problem	15
1.3 Research Objective	16
1.4 Hypothesis	17
1.5 Research Scope and Limitation	17
1.6 Significance of Study	18
1.7 Thesis Structure	18
CHAPTER 2 - LITERATURE REVIEW	19
2.1 Swiss German University	19
2.2 Merdeka Belajar-Kampus Merdeka (MBKM)	19
2.3 Software Engineering	20
2.3.1 Software Requirements	20
2.3.2 Software Design	23
2.3.3 Software Construction	27

2.3.4	Software Testing.....	32
2.4	Software Development Life Cycle Agile Methodology.....	37
2.4.1	Feature Driven Development Agile Methodology.....	39
2.5	LLMs.....	40
2.5.1	ChatGPT.....	40
2.6	Prompt Engineering.....	43
2.6.1	Factors Influencing Prompt.....	43
2.6.2	Prompt Principle.....	44
2.6.3	Prompting Technique.....	45
2.7	Previous Studies.....	46
2.7.1	Prompt Pattern for Software Engineering.....	46
2.7.2	LLMs in Requirement Engineering.....	48
2.7.3	LLMs in Software Design.....	50
2.7.4	LLMs in Software Development.....	53
2.7.5	LLMs in Software Testing.....	55
2.8	Formulating Research Methodology.....	57
CHAPTER 3 – RESEARCH METHODS.....		61
3.1	Research Design.....	61
3.2	ChatGPT as the Large Language Model (LLM) Used.....	62
3.3	Instructional Pattern Framework Implementation.....	62
3.4	System Development Methodology.....	64
3.4.1	Allocation of Activities.....	65
3.4.2	Requirement Elicitation.....	66
3.4.3	Develop Overall Model.....	67
3.4.4	Build Feature List.....	68
3.4.5	Plan by Feature.....	69
3.4.6	Design by Feature.....	69
3.4.7	Build by Feature.....	70
3.4.8	Feature Testing.....	71
3.5	Evaluation Framework.....	71

3.6	Data Collection	72
3.6.1	Development Artifacts and Interaction Logs	72
3.6.2	Evaluation Data: Survey and Interview.....	72
3.6.3	Data Gathering Instruments.....	73
3.7	Success Criteria	74
CHAPTER 4 – RESULTS AND DISCUSSIONS.....		76
4.1	Initial Instruction	76
4.2	Requirement Elicitation.....	77
4.2.1	Initial preparation	77
4.2.2	Stakeholder Interviews	78
4.2.3	Input Pre-processing using ChatGPT	78
4.3	Build Overall Model Phase.....	80
4.3.1	Collaborative Modelling Process	80
4.3.2	UML Artifact Generation.....	82
4.4	Build Feature List	84
4.4.1	ChatGPT-Assisted Feature Breakdown.....	84
4.4.2	Prioritization and Refinement	86
4.5	Plan by Feature	86
4.6	Design by Feature.....	87
4.6.1	Wireframe Suggestions via ChatGPT	88
4.6.2	UI Design in Figma	89
4.6.3	Reflection on LLM-Assisted UI Design	91
4.7	Build by Feature	91
4.8	Testing	93
4.8.1	Uploading the SRD to ChatGPT for Test Case Generation	94
4.8.2	Iterative Refinement through Prompt Engineering	94
4.8.3	Test Execution and Validation	95
4.9	Summary of System Development	96
4.10	Comparison Between Traditional and AI-Assisted Development.....	99

4.10.1	Documentation Completeness	100
4.10.2	Development Time	101
4.10.3	User Acceptance	101
4.11	Challenges and Recommendations	103
CHAPTER 5 – CONCLUSIONS AND RECOMENDATIONS.....		105
5.1	Conclusions	105
5.2	Future Works	106
GLOSSARY		108
REFERENCES		110
CURRICULUM VITAE		114
APPENDIX A		115
APPENDIX B		117



LIST OF FIGURES

Figures	Page
Figure 1.1 Fishbone Diagram on potential issue in software development	16
Figure 2.1 FDD (Alsaqqa et al. 2020).....	39
Figure 2.2 Text Evaluation Performance (OpenAI 2024).....	41
Figure 2.3 Feature-Drive Development	58
Figure 2.4 Proposed prompting framework based on previous study	60
Figure 3.1 Research Framework	61
Figure 3.2 Prompting Framework.....	63
Figure 3.3 System Development Framework	65
Figure 3.4 Requirement Elicitation Process Diagram.....	67
Figure 3.5 Develop Overall Model Process Diagram.....	68
Figure 3.6 Build and Plan Feature List Process Diagram	69
Figure 3.7 Design by Feature Process Diagram.....	70
Figure 3.8 Build by Feature Process Diagram	70
Figure 3.9 Feature Testing Process Diagram	71
Figure 4.1 Initial Instructions for ChatGPT	76
Figure 4.2 ChatGPT output based on the prompt	78
Figure 4.3 Prompt for requirement analysis	79
Figure 4.4 SRD generated by ChatGPT.....	79
Figure 4.5 Snippet of modelling process	80
Figure 4.6 ChatGPT generated PlantUML script for architecture	81
Figure 4.7 Initial component diagram generated using PlantUML	81
Figure 4.8 ChatGPT initial output for Use Case diagram.....	83
Figure 4.9 Use Case diagram after several iteration	83
Figure 4.10 Finalized ERD for MBKM registration.....	84
Figure 4.11 ChatGPT out in feature list phase.....	85
Figure 4.12 ChatGPT prioritization recommendation	86

Figure 4.13 ChatGPT output for development task breakdown	87
Figure 4.14 ChatGPT text-based wireframe output	88
Figure 4.15 ChatGPT PlantUML script and the visualization.	89
Figure 4.16 Student Dashboard design	90
Figure 4.17 Student MBKM Registration design	90
Figure 4.18 Table Layout Adjustment by ChatGPT	91
Figure 4.19 ChatGPT Role Routing Logic Validation	92
Figure 4.20 ChatGPT for debugging	93
Figure 4.21 ChatGPT generate Test Case	94
Figure 4.22 Iteration done to enhance test case	95



LIST OF TABLES

Table	Page
Table 2.1 Agile Development vs Traditional Development	38
Table 2.2 Prompt Principles adopted from (Bsharat et al. 2023).....	44
Table 2.3 Previous Study on Prompt Pattern for Software Engineering	47
Table 2.4 Previous Study on LLMs in Requirement Engineering	49
Table 2.5 LLM Capability in Requirement Engineering	50
Table 2.6 Previous Study on LLMs in Software Design	51
Table 2.7 LLM Capability in Software Design.....	52
Table 2.8 Previous Study on LLMs in Software Development	53
Table 2.9 LLM Capability in Software Development	54
Table 2.10 Previous Study on LLMs in Software Testing.....	55
Table 2.11 LLM Capability in Software Testing.....	56
Table 2.12 Software Development Lifecycle Activities.....	57
Table 2.13 Mapping Activities to LLM Capabilities.....	59
Table 3.1 Human & AI Role.....	65
Table 4.1 Strengths and limitations of ChatGPT assistance	98
Table 4.2 Development Approach Comparison.....	99
Table 4.3 Documentation Completeness Scoring.....	100
Table 4.4 Development Time Comparison	101
Table 4.5 User Acceptance Scoring.....	101