

Automated Front-end to Back-end Request Capturing System for Flutter-based Applications

By

Janet Gloria Salim
12002011

BACHELOR'S DEGREE
in

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

Revised after the Thesis Defense on 1 July 2024

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.

Janet Gloria Salim

Student


15 July 2024

Approved by:

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

Thesis Advisor


15.7.2024.

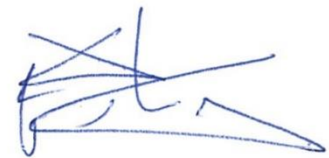
Leonardo Kurnia, S.Kom., B.Eng

Thesis Co-Advisor


15 July 2024

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

Dean


15.7.2024.

Janet Gloria Salim

ABSTRACT

AUTOMATED FRONT-END TO BACK-END REQUEST CAPTURING SYSTEM FOR FLUTTER-BASED APPLICATIONS

By

Janet Gloria Salim

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

Leonardo Kurnia S.Kom., B.Eng, Co-Advisor

SWISS GERMAN UNIVERSITY

Understanding and documenting connections within a software project can be challenging for both developers and non-technical stakeholders. To address this, we developed a Python script for automated request capturing and automated documentation of scripts in a Flutter-based project using its source code. The script pulls code from Bitbucket based on the client's name and captures key details such as class names, methods, HTTP methods, parameters, and request bodies using Regex. It supports various connections like front-end App to Middleware (both client-side and server-side), Middleware to ERP, and back-end App to Middleware services in languages such as Dart and Java. The results are compiled into a YAML file and displayed on Swagger UI for easy access. Initially, the script included AI-generated descriptions using ChatGPT's GPT-3.5 Turbo model, but this feature was removed due to rate limitations. The entire setup is containerized with Dockerization and integrated into Commsult's customer portal, the Customer Portal, where it can be triggered by a button click that activates an Express.js server. This tool has proven effective in helping both developers and non-technical users understand and navigate the project's codebase, despite the removal of the LLM-generated descriptions.

Keywords: Automated Request Capturing, Automated Documentation, Swagger, Front-end, Back-end, Flutter Framework, Dockerization.

© Copyright 2024
By Janet Gloria Salim
All rights reserved

DEDICATION

I dedicate this research for the companies
that has always provided me with
endless learning possibilities and experiences:

Commsult Indonesia and Commsult AG

ACKNOWLEDGEMENTS

My deepest gratitude goes to God, who always provided me with the guidance, strength, and motivation to help me throughout my thesis.

The support of my family – the “Salim Group” as we like to call it – My parents, Mr. Salim Janto Salim and Mrs. Ruth Renti Sisilya, and my younger brother, Mr. Jason Jeremiah Salim – is one of the biggest reasons that I could even make it to a time of my life where I am educated enough to do a research project. My parents have always sacrificed so much of their finances, time, comfort and gave me so much love to make sure I get smarter every day, and my brother somehow tolerated me throughout my life. It is a miracle to be able to do that. Truly the patience of a saint.

I would also love to give my biggest gratitude to Mr. Leonardo Kurnia, the Managing Director for Commsult Indonesia and my Co-Advisor for this thesis, for giving me the opportunity to develop this thesis in Commsult Indonesia and Commsult AG, and for giving me the guidance, and information regarding the technology that is used and developed with this thesis. My previous internship experiences in both the headquarters in Germany and the office in Indonesia has made me learn more about striving for excellence in my programming, and it is thanks to Mr. Leonardo Kurnia. I also want to express my deepest thanks to Dr. Maulahikmah Galinium S.Kom., M.Sc. for always being so supportive and by always giving me the most insightful advice.

I am also very grateful for the support of my Commsult colleagues, such as Mr. Sebastian Minke, Mr. Oliver Henkel, Mr. Tino Hönicke, Mr. Tobias Schenke, Mr. Alexander Redlich, Mrs. Anika Simon, and more, and those in Commsult Indonesia, such as Mr. Brilliant Joan Wongso, Ms. Michaela Hannah, Mr. Dave Andersen, and more as they have always supported me whenever I needed more motivation and helped me whenever I had any questions or issues in the technical side. Special shoutouts to Mr. Thaniel Suteja, Mr. Jesse Young and Mr. Aditya Fidianto from Commsult Indonesia for guiding me throughout this thesis with his great knowledge regarding the API development, Docker usage and general conventions in Commsult Indonesia to

help make my understanding about the topic better, and to make sure that I could create the code in the most optimized and simplest way possible with the habits of the programmers in mind. The interns that were there by my side before, during and after the thesis period in Commsult AG and Commsult Indonesia also made it much more fun – Mr. Hendika Indrawinata Lukman, Mr. Radyan Rahmananta, and Ms. Theodora Andrea Lydia.

I would also like to thank my friends, and the pastors in Germany from Lighthouse Church Berlin – Mrs. Rika Rianti and Mr. Andre Kho, Mr. Ashish Bhonsle, Ms. Pazia Alberta, Mr. James Reynaldo, Mr. James Reynaldi, Ms. Teressia Ciuputri, Ms. Wynnie Valencia, Ms. Nathalia Runtuwene, Ms. Virginia Tjan and Mr. Daniel Overröder for always giving me hope and strength through our times together and through prayer. They have taught me that through enough rest by exploring the city and coming to church, great ideas and motivation can come.

The St. Peter Canisius Parish have always been lovely and reminded me to always love God and see Him in my daily life. It was always refreshing to be part of events such as the Live Stations of the Cross, and meeting other young parishioners such as Ms. Clarissa Felicya, Mr. Winston Tjahaja, Mr. Patrik Elang Prakasa, Mrs. Alfa Winarno and Mr. Donny Winarno, Mrs. Widya Hotate, Mrs. Maria Rosario Tan along with Mrs. Eunice Tirtalukita and more made me feel welcomed and warm.

My friends have always never failed to support me throughout the thesis development series. I would like to thank Mr. Joseph Andreas, Ms. Rehmia Eunike and Mr. Nazree Nadhir for always being part of my main gang “Jeden Tag Trinken!” I am so glad we all got to know each other during our life-changing experience in Germany. I would also like to thank my friends from “GOKUDOLLS,” Mr. James Adipoetra, Mr. Lius Vigo and Mr. Kenny Gani for always being super entertaining and making me laugh when tough times come. I would also like to thank my friend group in high school, Ms. Kelly Wijaya, Ms. Maria Christina, Ms. Cayley Wijaya and Ms. Zelda Arwen for always keeping me company, though only through chat messages due to the time differences. My friends from middle school also cannot be forgotten – Mr. Edbert

Mulyawan, Mr. Inyo Hansen, Mr. Rafael Alessandro, and Ms. Jessica Ng, truly the “OG.”

I would also love to mention my friends who are not in particular groups – Mr. Peter Yudhistira, Mr. Ariel Yuwono, Mr. Ravell Revano, Mr. Pine Hein Swe, Mr. Matthew Bintang, Mr. Seidi Lefrandy, Mr. Nicholas Santoso and Mr Edwin Wibowo. The funny meme pictures and videos, together with the life stories you told me, always made me happy.

Finally, I would like to thank Mr. Christian Joseph De Silva Tan, my thesis and internship partner, for always supporting me from the beginning of my internship journey in the third semester of our studies in Swiss German University up till the end of our thesis journey. He has always been one of the greatest helps I could ever have as he has always been more than supportive in all aspects and supported my growth as both a software developer and I as a person.

Thank you Mr. Lando Norris, McLaren Formula One Racing Driver, for winning Miami Grand Prix. That alone has increased my motivation to complete my thesis and motivate me to rise to glory, along with Mr. Charles Leclerc’s Home Win at Monaco and Mr. Carlos Sainz Jr.’s win at the Australian Grand Prix. Sir Lewis Hamilton’s win at his home race in Silverstone was also a moment to remember.

I would also love to thank the bands Radiohead, My Chemical Romance, Falling In Reverse, BABYMETAL, Electric Callboy, Bring Me The Horizon, The Strokes and ONE OK ROCK for always having the best music to accompany me during my thesis writing process. This gratitude also goes to individual singers such as Joost, Ado, JJ Lin, Eric Chou, Jay Chou and more.

TABLE OF CONTENTS

STATEMENT BY THE AUTHOR.....	2
ABSTRACT.....	3
DEDICATION.....	5
ACKNOWLEDGEMENTS.....	6
TABLE OF CONTENTS.....	9
LIST OF FIGURES	12
LIST OF TABLES.....	15
LIST OF EQUATIONS	16
Chapter 1 - INTRODUCTION	17
1.1 Background.....	17
1.2 Research Problems.....	19
1.2.1 Developer-Facing Problems.....	19
1.2.2 Client-Facing Problems	20
1.3 Research Objectives.....	21
1.4 Significance of Study.....	21
1.4.1 Developer-Facing Significance.....	22
1.4.2 Client-Facing Significance.....	23
1.5 Research Questions	23
1.6 Hypothesis.....	24
1.7 Limitations	24
1.7.1 General Limitations	24
1.7.2 Architecture and System-based Limitations	26
1.8 Scope of Research.....	26
Chapter 2 - LITERATURE REVIEW	28
2.1 Theoretical Perspectives	28
2.1.1 Commsult Products.....	28
2.1.2 Software Engineering.....	29
2.1.3 Regular Expressions.....	29
2.1.4 Python Scripting.....	30
2.1.5 Large Language Models.....	30
2.1.6 Docker.....	30
2.1.7 Software Visualization.....	31

2.1.8 Human-Computer Interaction (HCI).....	31
2.2 Previous Works	32
Chapter 3 – RESEARCH METHODS.....	42
3.1 Theoretical Perspectives	42
3.1.1 Gathering and Analysis.....	43
3.1.2 Design	43
3.1.3 Implementation	44
3.1.4 Testing.....	45
3.1.5 Additional Experiments	45
3.3 Architecture Diagram.....	45
3.3.1 General Architecture Diagram for Two Capturing Tools.....	46
3.3.2 Focused Architecture Diagram for Request Capturing Tool	47
3.3.2.1 “re” Library	48
3.3.2.2 “os” Library	48
3.3.2.3 “glob” Library	49
3.3.2.4 “argparse” Library.....	49
3.4 Use-Case Diagram	49
3.5 Activity Diagram	50
Chapter 4 – RESULTS AND DISCUSSIONS	52
4.1 Initial Evaluation.....	52
4.2 Preliminary Results	55
4.2.1 Source Code Detection	55
4.2.2 Accessibility to Non-Technical Users.....	56
4.2.3 Time Comparison.....	56
4.2.4 Requirements Met for Developers	57
4.3 Early Script Prototypes and Development Process.....	59
4.3.1 Prototype One (script-learning.py)	59
4.3.2 Prototype Two (scriptv2.py)	60
4.3.3 Prototype Three (scriptandswagger.py)	60
4.3.4 Prototype Four (scriptswaggerbitbucket.py).....	61
4.3.5 Prototype Five (scriptswaggerbitbucket.py – SOAP File and Client-Side).....	62
4.3.6 Prototype Six (AIpowereddescriptions.py).....	62
4.4 Final Prototype of Local Docker and Script	63
4.4.1 General Overview Flowchart	64
4.4.2 Technical Flowchart.....	65

4.4.2.1 Process of Bitbucket Source Code Cloning and Saving	69
4.4.2.2 Process of Capturing App to Middleware (Server-Side) Services.....	69
4.4.2.3 Process of Capturing App to Middleware (Client-Side) Services	70
4.4.2.4 Process of Capturing Middleware to ERP Services.....	74
4.4.2.5 Process of Capturing App to ERP Services	77
4.4.2.6 Process of YAML Creation with All Results.....	81
4.4.2.7 Process of Automated YAML Input and Swagger UI Launching	81
4.4.2.8 Express Service and Docker	85
4.5 Integration of Script, Docker and the Customer Portal.....	86
4.5.1 Integration Methodology	87
4.5.2 Flow of Integrated Request Capturing System	89
4.5.2.1 Swagger UI Design and Details	91
4.5.2.2 Customer Portal's Functionality's Further Implementation	95
4.6 Discussion of Findings.....	97
4.6.1 Usability Testing	99
4.6.2 Cost-Savings Additional Experiment Result	101
4.6.3 Tool Performance Evaluation	103
4.6.3.1 Accuracy of Tool	103
4.6.3.2 Quality of Tool.....	104
4.6.4 Comparison of Previous Technical Documentation and Swagger UI	104
4.6.4.1 ChatGPT 3.5 LLM Descriptions.....	108
4.6.5 Final Python Script Flowchart	110
Chapter 5 – CONCLUSIONS AND RECOMMENDATIONS.....	115
5.1 Conclusions.....	115
5.2 Future Works	116
GLOSSARY	118
APPENDICES	120
REFERENCES	131
CURRICULUM VITAE	137

LIST OF FIGURES

Figures	Page
Figure 1.1 - Flutter Development Process without Tool.....	22
Figure 1.2 - Flutter Development Process with Tool.....	22
Figure 3.1 - Research Framework based on the Software Development Life Cycle...	42
Figure 3.2 - General Architecture Diagram of Two Capturing Tools	46
Figure 3.3 - Architecture Diagram of Request Capturing Tool	47
Figure 3.4 - Use Case Diagram of Request Capturing Tool	49
Figure 3.5 - Activity Diagram of Request Capturing Tool.....	50
Figure 4.1 - Prototype YAML Naming.....	61
Figure 4.2 - "client_name" variable usage for path setting in Prototype Four.....	61
Figure 4.3 - First AI Prototype Prompt for Description Generation.....	62
Figure 4.4 - "gendesc" variable usage to store generated descriptions.....	63
Figure 4.5 - Implementation of "gendesc" variable inside of generated YAML file...	63
Figure 4.6 - General Overview of the Request Capturing Process	64
Figure 4.7 - Flowchart of Service Capturing Tool (Part 1).....	65
Figure 4.8 - Flowchart of Request Capturing Tool (Part 2).....	66
Figure 4.9 - Flowchart of Request Capturing Tool (Part 3).....	66
Figure 4.10 - Flowchart of Request Capturing Tool (Part 4).....	67
Figure 4.11 - Flowchart of Request Capturing Tool (Part 5).....	67
Figure 4.12 - Flowchart of Request Capturing Tool (Part 6).....	68
Figure 4.13 - Flowchart of Request Capturing Tool (Part 7).....	68
Figure 4.14 - App to Middleware Path Selection 1.....	69
Figure 4.15 - App to Middleware Path Selection 2.....	69
Figure 4.16 - Client-Side App to Middleware Path Selection 1	70
Figure 4.17 - Client-Side App to Middleware Path Selection 2	70
Figure 4.18 - "dir_name" variable's methodology	70
Figure 4.19 - Client-Side App to Middleware Class Name & Method Info Regex.....	71

Figure 4.20 - Method Info and Ignore List (Part 1)	71
Figure 4.21 - Method Info and Ignore List (Part 2)	71
Figure 4.22 - HTTP Method Setting Methodology	73
Figure 4.23 - Middleware to ERP Path	74
Figure 4.24 - Middleware to ERP Class Name & Method Info Regex	75
Figure 4.25 - Middleware to ERP Parameters Regex	76
Figure 4.26 - SOAP File Path Selection 1	77
Figure 4.27 - SOAP File Path Selection 2	77
Figure 4.28 - SOAP File Class Name & Method Info Regex.....	78
Figure 4.29 - SOAP File Parameters Regex	79
Figure 4.30 - SOAP Directory Path Selection 1	79
Figure 4.31 - SOAP Directory Path Selection 2	79
Figure 4.32 - SOAP Directory Class Name & Method Info Regex.....	80
Figure 4.33 - SOAP Directory Parameters Regex	81
Figure 4.34 - YAML Creation, Saving, and Swagger UI Launching Process	84
Figure 4.35 - Express Service "/build" Route	85
Figure 4.36 - Express Service "/run" Route	86
Figure 4.37 - Docker and Customer Portal Integration Flow	87
Figure 4.38 - Shared Folder Items	88
Figure 4.39 - "latest.yaml" renaming process on Python.....	88
Figure 4.40 - Research API in Button on Customer Portal.....	89
Figure 4.41 - Integrated System's API Endpoint	89
Figure 4.42 - Network Response after Button Click.....	90
Figure 4.43 - Integrated System's Swagger UI Staging Server URL.....	90
Figure 4.44 - Swagger UI Design (Part 1)	91
Figure 4.45 - Swagger UI Design (Part 2)	92
Figure 4.46 - Captured Request Details (Part 1).....	92
Figure 4.47 - Captured Request Details (Part 2).....	94
Figure 4.48 - Captured Request Details (Part 3).....	94
Figure 4.49 - Captured Request Details (Part 4).....	95
Figure 4.50 - Button Implementation on the Customer Portal's Management Page ...	95
Figure 4.51 - JanetThesisFunc.js Code	96

Figure 4.52 - Gojek API Documentation (Part 1).....	106
Figure 4.53- Gojek API Documentation (Part 2).....	106
Figure 4.54 - Swagger UI Navigation and Categorization	107
Figure 4.55 - Swagger UI Color Coding for Requests.....	108
Figure 4.56 – Final Python Script Flowchart (Part 1).....	110
Figure 4.57 - Final Python Script Flowchart (Part 2)	111
Figure 4.58 - Final Python Script Flowchart (Part 3)	111
Figure 4.59 - Final Python Script Flowchart (Part 4)	112
Figure 4.60 - Final Python Script Flowchart (Part 5)	112
Figure 4.61 - Final Python Script Flowchart (Part 6)	113
Figure 4.62 - Final Python Script Flowchart (Part 7)	113

LIST OF TABLES

Table	Page
Table 1 - Comparison of Papers and the Thesis Topic Research.....	38
Table 2 - Commsult Indonesia Developers' Interview Result	52
Table 3 - Time Comparison of Manual Documentation and Tool Usage.....	56
Table 4 - Comparison of Manual Documentation, Local Script, and Customer Portal Button.....	98
Table 5 - Developers' Usability Testing Results.....	99
Table 6 - Non-Technical Users' Usability Testing Results	100
Table 7 - Accuracy Calculation of Manual and Automatic Captured Requests (Client A)	103
Table 8 - Accuracy Calculation of Manual and Automatic Captured Requests (Client B).....	104
Table 9 - AI-Generated Descriptions Examples	109

LIST OF EQUATIONS

Equation	Page
Equation 1 - Weekly Salary Formula.....	101
Equation 2 - Daily Salary Formula	102
Equation 3 - Hourly Rate Formula.....	102
Equation 4 - Cost Savings (Hourly) for Company.....	102
Equation 5 - Company Savings for 40 Developers.....	102