

Pemodelan Pengembangan Aplikasi Manajemen Informasi Fakultas Industri Kreatif (MI-FIK) Universitas Telkom Dengan Unified Model Language

1st Fauzan Reza Arnanda
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
arzafr@students.telkomuniversity.ac.id

2nd Mira Kania Sabariah
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
mirakania@telkomuniversity.ac.id

3rd Monterico Adrian
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
monterico@telkomuniversity.ac.id

Abstrak - Proyek pengembangan Aplikasi MI-FIK merupakan proyek lanjutan dari perancangan prototype Aplikasi Pengelolaan Informasi di Fakultas Industri Kreatif yang sebelumnya telah dikerjakan oleh Ceka Elgy Dwi Putra, mahasiswa Program studi S1 Desain Komunikasi Visual angkatan 2018. Aplikasi ini berfungsi sebagai pengelola informasi untuk civitas akademika Fakultas Industri Kreatif, dan dirancang untuk menyelesaikan permasalahan pengelolaan informasi pada lingkungan Fakultas Industri Kreatif, yang kemudian dialih tugaskan kepada Fakultas Informatika untuk dilanjutkan ke tahap development, dikarenakan keterbatasan pengetahuan dalam ranah pengembangan dan pemrograman aplikasi, serta kemampuan yang baru sebatas desain dan requirement elicitation. Oleh karena itu, mengingat pentingnya pengelolaan informasi untuk memastikan setiap kegiatan yang direncanakan dapat terlaksana dengan baik, maka timbullah adanya urgensi dan kebutuhan untuk membuat aplikasi pengelolaan informasi. Untuk membuat sebuah aplikasi, maka kita harus melalui beberapa tahap pengembangan, atau yang biasa disebut SDLC (Software Development Life Cycle). Pada tahap awal SDLC, tahap planning diperlukan untuk mencari requirement yang dibutuhkan aplikasi nantinya. Setelah tahap planning selesai, maka langkah selanjutnya adalah desain permodelan. Dalam tahap ini, model atau struktur aplikasi akan dibangun. Model atau struktur ini juga menjadi pedoman untuk selanjutnya seperti tahap pemrograman dan testing.

Kata Kunci: perancangan sistem, pemodelan perangkat lunak, uml, aplikasi manajemen informasi

I. PENDAHULUAN

A. Latar Belakang

Aplikasi MI-FIK merupakan proyek lanjutan dari perancangan prototype aplikasi pengelolaan informasi di lingkungan Fakultas Industri Kreatif - Telkom University. Aplikasi ini berfungsi sebagai pengelola dan reminder informasi untuk civitas akademika Fakultas Industri Kreatif, agar dapat

menyelesaikan permasalahan yang terjadi pada lingkungan Fakultas Industri Kreatif. Proyek ini telah dikelola sebelumnya oleh Ceka Elgy Dwi Putra, salah satu mahasiswa program studi S1 Desain Komunikasi Visual angkatan 2018, yang kemudian dialih tugaskan kepada Fakultas Informatika untuk dilanjutkan ke tahap development, dikarenakan keterbatasan pengetahuan dalam ranah pengembangan dan pemrograman aplikasi, serta kemampuan yang baru sebatas desain dan requirement elicitation.

Berdasarkan hasil survei sebelumnya, dikatakan bahwa terdapat beberapa masalah dalam hal penyebaran informasi di Fakultas Industri Kreatif, seperti terjadinya penumpukan informasi, informasi yang tidak terkategori, informasi yang tidak dikelola dengan baik, penggunaan media informasi yang terlalu banyak, hingga informasi yang telah tersampaikan namun sulit untuk dilihat kembali. Informasi tersebut meliputi hal-hal akademik seperti jadwal kuliah, informasi tentang fakultas, informasi kurikulum, jadwal ujian, jadwal kuliah, jadwal mengajar, jadwal pertemuan, event, seminar, dan lainnya. Oleh karena itu, mengingat pentingnya pengelolaan informasi untuk memastikan setiap kegiatan yang direncanakan dapat terlaksana dengan baik, maka timbullah adanya urgensi dan kebutuhan untuk membuat aplikasi pengelolaan informasi.

Agar proyek pengembangan aplikasi dapat berjalan dengan baik, maka diperlukan beberapa tahap pengembangan, atau yang biasa disebut SDLC (Software Development Life Cycle) [1], [2]. Berdasarkan alur SDLC (Software Development Life Cycle), maka tahap awal yang perlu dilakukan sebelum memasuki tahap pemrograman adalah tahap requirement analysis, dimana requirement tersebut menjadi suatu kebutuhan utama dari aplikasi yang akan dibangun dan berfungsi untuk memahami kebutuhan klien serta mendokumentasikannya dengan baik [1], [2]. Namun, requirement saja tidak cukup untuk melanjutkan proyek ke tahap pemrograman, karena berpotensi terjadinya

kekeliruan dan ketidaksepahaman diantara developer. Maka dari itu, kita juga akan melalui proses desain perangkat lunak atau tahap pemodelan. Tahap pemodelan suatu perangkat lunak menjadi aktivitas yang penting dilakukan, karena jika tidak, maka akan berdampak pada terjadinya kesalahpahaman informasi diantara stakeholder karena tidak adanya bahasa standar, potensi masalah akan lebih sulit diidentifikasi sejak awal proses pengembangan, dapat terjadinya ambiguitas atau perbedaan pendapat di antara tim pengembangan, dan besar kemungkinan terjadinya kesalahan pemrograman oleh developer karena melenceng dari requirement sebab tidak adanya dokumen yang menjadi pedoman. Kemudian output dari dibentuknya suatu dokumen kebutuhan perangkat lunak dan desain perangkat lunak juga menjadi hal yang penting, karena dari hal tersebut akan terlihat behavior aplikasi, interaksi yang terjadi di dalam aplikasi, struktur dari sistem aplikasi, dan kebutuhan pengguna akan terdefiniskan dengan jelas.

Kemudian, untuk melakukan pemodelan sistem aplikasi, maka kita perlu membuat use case diagram untuk mendefinisikan peran dan berbagai fungsi dalam sistem aplikasi, activity diagram untuk menggambarkan aliran kerja secara terurut pada aplikasi, sequence diagram untuk menggambarkan interaksi antar objek-objek dalam sistem, dan class diagram untuk menggambarkan struktur dari suatu sistem dengan jelas [3]. Sistem aplikasi juga tidak bisa lepas dengan yang namanya database, maka dari itu kita juga perlu merancang database dengan ERD Diagram yang berfungsi untuk menggambarkan kebutuhan data dari sistem dalam suatu database[4]. Apabila proses development sudah selesai, selanjutnya kita akan menuju tahap pengujian akhir dari pengembangan sebuah produk yaitu UAT (User Acceptance Testing) yang berperan untuk memastikan bahwa sistem yang dibangun telah sesuai dengan kebutuhan pengguna dan sebagai simbol serah terima aplikasi yang sudah dibangun kepada user[5]. Diharapkan aplikasi MI-FIK ini akan menjadi suatu aplikasi pusat informasi berbasis mobile yang dapat diakses oleh seluruh civitas Fakultas Industri Kreatif melalui mobile phone.

B. Topik dan Batasannya

Batasan dalam proyek ini yaitu pemodelan structure diagram, behavior diagram dan interaction diagram menggunakan UML, serta pemodelan system database menggunakan ERD

C. Tujuan

Tujuan yang hendak dicapai dalam TA Capstone ini adalah sebagai berikut:

1. Mengetahui model dari pengembangan aplikasi MI-FIK yang memenuhi requirement pengguna
2. Memberikan gambaran mengenai alur kerja aplikasi manajemen informasi Fakultas Industri Kreatif

II. STUDI TERKAIT

A. UML

Unified Modeling Language (UML) adalah bahasa pemodelan visual standar yang ditujukan untuk melakukan analisis, desain, dan implementasi sistem perangkat lunak. Tidak hanya itu, UML juga dapat digunakan untuk permodelan bisnis. UML merupakan bahasa umum untuk business analysts, software architects dan developer untuk menjelaskan, menentukan, merancang, serta mendokumentasikan proses bisnis yang ada, termasuk structure dan behavior dari artefak sistem perangkat lunak[3], [10].

B. Basis data

Database adalah kumpulan data atau informasi yang terstruktur dan terorganisir, biasanya disimpan secara elektronik dalam sistem komputer. Database pada umumnya dikontrol oleh Database Management System (DBMS). Bersama-sama, mulai dari data dan DBMS, beserta aplikasi yang terkait dengannya, disebut sebagai sistem basis data [15].

Data dengan jenis database paling umum yang beroperasi saat ini biasanya dimodelkan dalam baris dan kolom dalam serangkaian tabel untuk membuat pemrosesan dan kueri data yang efisien. Data kemudian dapat dengan mudah diakses, dikelola, dimodifikasi, diperbarui, dikontrol, dan diatur. Sebagian besar database menggunakan Structured Query Language (SQL) untuk menulis dan membuat kueri data.

1. ERD

Entity Relationship Diagram (ERD) adalah sebuah diagram yang digunakan untuk melakukan perancangan database. Pada umumnya, ERD menggambarkan entitas di dunia nyata, atribut dan relasi diantaranya. Entity Relationship Diagram (ERD) membantu menjelaskan logical structure dari sebuah basis data[16]. Komponen yang terdapat pada ERD terdiri dari

C. SRS

Software requirements specification (SRS) adalah dokumen spesifikasi lengkap dan deskripsi requirement perangkat lunak yang harus dipenuhi untuk keberhasilan pengembangan sistem perangkat lunak. Requirement ini biasanya menggambarkan fitur-fitur dari sistem yang sedang dikembangkan. Fitur-fitur ini tidak hanya menggambarkan Functional Requirement (FR) tetapi juga Non-Functional Requirement (NFR).

D. SDD

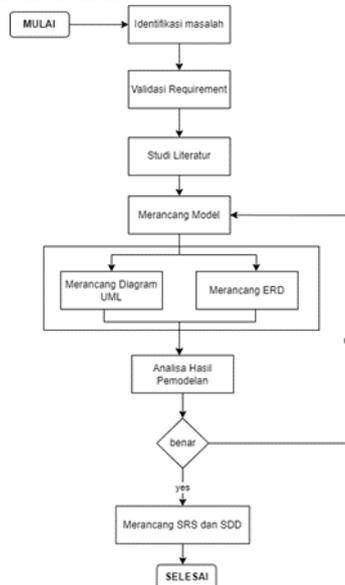
Software Design Document (SDD) adalah dokumen tertulis yang memberikan deskripsi produk dalam hal arsitektur perangkat lunak dengan berbagai komponen dengan fungsionalitas tertentu. Deskripsi desain ini juga digunakan sebagai media untuk

mengkomunikasikan desain perangkat lunak, dan dapat dianggap sebagai cetak biru atau model dari sistem yang akan dibangun.[19].

III. ALUR PEMODELAN

A. Diagram Alur Perancangan

Berikut merupakan alur dari tahap pemodelan yang akan dilakukan :



GAMBAR 3.1
Alur Pemodelan

B. Software Requirement Validation

Disebabkan proyek ini merupakan proyek alihan dari Fakultas Industri Kreatif kepada Fakultas Informatika, maka tahap planning dan requirement analysis sudah dilakukan sebelumnya oleh pihak FIK. Maka dari itu diperlukan adanya validasi ulang sebelum proyek dilanjutkan ke tahap development. Teknik validasi yang akan dilakukan yaitu:

1. Requirement review / inspections

Dilakukan dengan menganalisis dokumen yang dibuat oleh pihak FIK, menarik hipotesis, dan melakukan beberapa kali pertemuan dengan pihak FIK selaku client dan pemegang proyek sebelumnya untuk berkonsultasi, berdiskusi dan melakukan verifikasi bersama-sama terhadap aplikasi yang akan dibangun

Setelah tahap requirement validation ini selesai, maka akan dihasilkan requirement yang telah tervalidasi, dan dapat dijadikan acuan untuk memulai tahap pemodelan.

C. Pemodelan

Karena rancangan Software Requirement Specification (SRS) yang masih kurang maksimal saat dikerjakan oleh pihak FIK sebab baru menyajikan beberapa komponen SRS, dan masih belum berbentuk dokumen SRS standar, maka diperlukan adanya reverse engineering terhadap SRS yg ada sebelumnya. Selain itu, diperlukan juga pembuatan Software Design Document (SDD) untuk mendefinisikan lebih jelas mengenai arsitektural dari aplikasi.

Pada tahap ini, requirement specification yang sudah didapat akan dirubah menjadi model UML dan diagram basis data yang bisa menjabar requirement aplikasi. Model UML dan diagram basis data ini juga dapat membantu tahap pemrograman nantinya, karena developer jadi mempunyai pedoman dalam menulis kode programnya. Model atau struktur yang akan dirancang pada tahap ini adalah sebagai berikut:

- Use Case Diagram
- Activity Diagram
- Sequence Diagram
- Class Diagram
- Entity Relation Diagram (ERD)

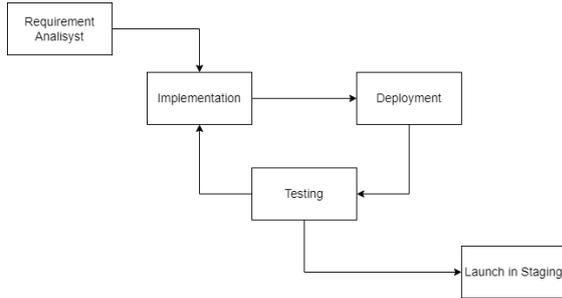
Output yang dihasilkan dari tahap ini yaitu dihasilkan dokumen final Software Requirement Specification (SRS) dan Software Design Document (SDD). Dokumen-dokumen tersebut akan membantu mendeskripsikan dan menentukan arsitektur sistem secara keseluruhan. Selanjutnya, dokumen-dokumen tersebut juga berguna sebagai identitas aplikasi yang sedang dibangun, dan sebagai tuntunan untuk tahap selanjutnya yaitu tahap pemrograman.

IV. IMPLEMENTASI

A. Software Development Life Cycle

Proyek pengembangan aplikasi mobile dan website dashboard ini dilakukan dengan proses Software Development Life Cycle (SDLC) Iterative Model. Proses ini melibatkan siklus pengembangan yang berulang-ulang dan setiap siklus yang dilakukan disebut dengan iterasi. Metode iteratif ini dipilih karena memungkinkannya pengembangan software yang lebih adaptif. Dengan menggunakan pendekatan ini, tim pengembang dapat membangun, menguji, dan mengintegrasikan bagian-bagian dari sistem secara bertahap, sehingga memungkinkan tim pengembang untuk mengenal komponen-komponen sistem dan mengidentifikasi potensi perbaikan lebih awal dalam siklus pengembangan. Berikut merupakan model iteratif yang diterapkan dalam proyek pengembangan aplikasi yang dilakukan oleh tim dan penulis :

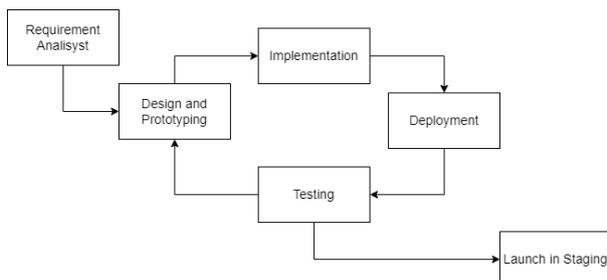
1. Model SDLC Mobile



Gambar 4.1.1. 1 Model SDLC Iterative Mobile

Pada proses pengembangan aplikasi mobile MI-FIK, desain dan prototype telah ditentukan oleh pihak client (FIK), sehingga kami ditugaskan untuk mengembangkan aplikasi, sesuai desain, style guide dan prototype yang telah ditentukan. Oleh karena itu pada proses Software Development Life Cycle (SDLC) Mobile, proses dimulai dengan melakukan requirement analyst terhadap desain, prototype, dan kebutuhan spesifik user (Dosen, Staff, dan Mahasiswa) terlebih dahulu. Kemudian iterasi terjadi pada tahap implementastion, deployment, dan testing, Saat iterasi yang yang ditetapkan akan berakhir, iterasi akan berhenti ditahap testing dan menuju ke tahap terakhir yaitu launch in staging. Pada proses Software Development Life Cycle (SDLC) iteratif ini, tahap perancangan model UML terjadi pada tahap implementation. Hal itu dikarenakan desain dan prototype telah di tentukan oleh pihak client (FIK), sehingga model diagram menyesuaikan dengan desain yang telah ditentukan

2. Model SDLC Web

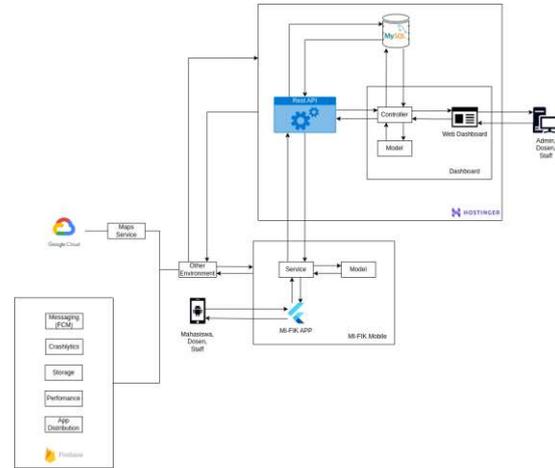


GAMBAR 4.1.2. 1 Model SDLC Iterative Web

Pada proses pengembangan website dashboard MI-FIK, desain belum ditentukan oleh pihak client (FIK). Oleh karena itu pad proses Software Development Life Cycle (SDLC) website dashboard, proses dimulai dengan melakukan requirement analyst terlebih dahulu terhadap kebutuhan user (Dosen, Admin). Kemudian iterasi terjadi pada tahap Design and Prototyping, implementastion, deployment, dan testing, Saat iterasi yang ditetapkan akan berakhir, iterasi akan berhenti ditahap testing

dan menuju ke tahap terakhir yaitu launch in staging. Pada proses Software Development Life Cycle (SDLC) iteratif ini, tahap perancangan model UML terjadi pada tahap Design and Prototyping. Perancangan model ini dibuat berdasarkan keseluruhan FR yang telah didapat dari hasil diskusi dengan client (FIK) dan tahap requirement analyst.

B. System Architecture



GAMBAR 4.2 System Architecture

Aplikasi ini berjalan pada smartphone dengan sistem operasi android dengan menggunakan framework flutter. Flutter dipilih karena kemampuannya dalam multiplatform, hal ini sesuai dengan keinginan client (FIK) yang menginginkan aplikasinya dapat berjalan di smartphone IOS nantinya. Kemudian untuk traffic data, aplikasi ini menggunakan koneksi data yang sepenuhnya melalui API, menyimpan data tertentu di penyimpanannya lokal, dan tidak mengakses database secara langsung. Hal ini dilakukan agar data di database tetap aman. Terkait pengelolaan data yang ada pada aplikasi mobile MI-FIK, hal dilakukan melalui web dashboard MI-FIK yang digunakan oleh Dosen, Staff, dan admin. Melalui website ini, dosen, terutama admin, dapat mengelola data yang berada pada RestAPI. Data yang dikelola pada web dashboard MI-FIK ini nantinya akan mempengaruhi data yang muncul pada aplikasi mobile MI-FIK.

Untuk kebutuhan layanan notifikasi, penyimpanan media, dan distribusi aplikasi, aplikasi ini menggunakan Firebase Service. Melalui Firebase Service ini, performa dan error aplikasi juga dapat dipantau dengan baik. Kemudian untuk fitur maps pada setiap event, aplikasi ini menggunakan layanan Google Cloud Maps Service. Pada arsitektur diagram, RestAPI dan database diperlukan untuk memenuhi hampir keseluruhan funtional requirement mobile dan web. Firebase service diperlukan untuk memenuhi FR-M-013, FR-M-010, FR-M-005, FR-W-014, FR-W-020. Google maps service diperlukan untuk memenuhi FR-M-006, FR-W-006, FR-W-007,

dan FR-W-028. Kemudian, hampir keseluruhan functional requirement mobile berjalan pada aplikasi MI-FIK yang berjalan pada smartphone android, aplikasi ini digunakan oleh mahasiswa, dosen, dan staff. Untuk functional requirement web berjalan pada website dashboard, yang dapat diakses melalui komputer dekstop. Website dashboard ini digunakan oleh Dosen, Staff, dan Admin.

C. Requirement

Dalam proses pembangunan software, requirement sangatlah diperlukan untuk memenuhi kebutuhan client, memberikan developer pedoman serta kejelasan dalam pembangunan berbagai aspek dalam aplikasi, dan menjadi dasar bagi perencanaan yang lebih kompleks kedepannya. Kemudian, berdasarkan hasil penulis melakukan interview dan berdiskusi dengan pihak client serta developer, sebagai mana tertera pada Minutes of Meeting (MOM), maka didapatkanlah dokumentasi Functional Requirement (FR) sebagai berikut :

1. Functional Requirement Mobile

ID	Functional Requirement	Functionalit y	Priorit y
FR - M-001	Dosen, Staff, dan Mahasiswa dapat melakukan registrasi	Autentikasi	High
FR - M-002	Dosen, Staff, dan Mahasiswa dapat melakukan login	Autentikasi	High
FR - M-003	Dosen, Staff, dan Mahasiswa dapat melakukan logout	Autentikasi	Mediu m
FR - M-004	Dosen, Staff, dan Mahasiswa dapat melihat dan request role baru	Roles	High
FR - M-005	Dosen, Staff, dan Mahasiswa dapat mengelola data profile	User	High

FR - M-006	Dosen, Staff, dan Mahasiswa dapat melihat informasi event sesuai role yang dimiliki	Event	High
FR - M-007	Dosen, Staff, dan Mahasiswa dapat melakukan pencarian informasi event	Event	Mediu m
FR - M-008	Dosen, Staff, dan Mahasiswa dapat mengelola folder archive untuk menyimpan informasi	Archive	High
FR - M-009	Dosen, Staff, dan Mahasiswa dapat menyimpan dan mengelola event/informasi pada folder archive yang dipilih	Archive	High
FR - M-010	Dosen, Staff, dan Mahasiswa mendapat pengingat(reminder) ketika kegiatan akan berlangsung	Reminder	High
FR - M-011	Dosen, Staff, dan Mahasiswa dapat mengelola task pribadi	Task	High
FR - M-012	Dosen, Staff, dan Mahasiswa dapat melihat daftar event dalam rentang satu bulan	Calendar	Low
FR - M-013	Dosen, Staff, dan Mahasiswa dapat melihat kumpulan pengumuman sesuai role dan tag yang dimiliki	Announceme nt	High
FR - M-014	Dosen dan Staff dapat membuat event baru	Event	High

FR - M-015	Dosen, Staff, dan Mahasiswa dapat melihat riwayat kegiatan yang pernah dilakukan oleh masing-masing pengguna	History	Mediu
FR - M-016	Dosen, Staff, dan Mahasiswa dapat melihat role yang dimiliki	User	High
FR - M-017	Dosen, Staff, dan Mahasiswa dapat melihat dan mengirimkan pertanyaan	FAQ	Mediu m
FR - M-018	Dosen, Staff, dan Mahasiswa dapat mengirimkan feedback	Feedback	Mediu m
FR - M-019	Dosen, Staff, dan Mahasiswa dapat melihat panduan penggunaan aplikasi	Help	Mediu m
FR - M-020	Dosen, Staff, dan Mahasiswa dapat melihat informasi tentang aplikasi	About us	Low
FR - M-021	Dosen, Staff, dan Mahasiswa dapat melihat agenda task atau event pada suatu hari selama satu minggu kedepan	Schedule	High

2. Functional Requirement Web Dashboard

ID	Functional Requirement	Fungsionalitas	Prioritas
FR-W-001	Dosen dan staf harus mendaftar akun terlebih dahulu	Autentikasi	High

FR-W-002	Admin, Dosen, dan Staf harus melakukan login untuk mengakses keseluruhan web dashboard	Autentikasi	High
FR-W-003	Admin, Dosen, dan Staf dapat melakukan sign out	Autentikasi	Medium
FR-W-004	Admin, Dosen, dan Staf dapat melihat daftar event yang akan berlangsung, sedang berlangsung, dan sudah selesai	Event	High
FR-W-005	Admin, Dosen, dan Staf dapat melihat detail event	Event	High
FR-W-006	Admin, Dosen, dan Staf dapat melakukan pencarian event dan melakukan filter event yang dicari	Event	Medium
FR-W-007	Admin, Dosen, dan Staf dapat membuat dan mengelola event	Event	High
FR-W-008	Admin dapat mengelola Tag	Tag	High
FR-W-009	Dosen, dan Staf dapat melihat dan melakukan	Tag	Low

	pencarian Tag		
FR-W-010	Admin, Dosen, dan Staf dapat melihat event pada rentang hari, minggu, dan bulan tertentu	Calendar	Medium
FR-W-011	Admin, Dosen, dan Staf dapat melihat event yang telah selesai	Calendar	Medium
FR-W-012	Admin dapat membuat dan mengelola data Info, serta dapat mencari dan filter berdasarkan type	Info	Medium
FR-W-013	Admin dapat melihat data dictionary dan mengelola, serta melakukan pencarian data	Dictionary	High
FR-W-014	Admin dapat melihat pengumuman dan mengelola, serta melakukan pencarian data	Notification	High
FR-W-015	Admin dapat melihat daftar feedback dan destroy feedback, serta melakukan filter berdasarkan Suggestion	Feedback	Medium

FR-W-016	Admin dapat melihat dan menjawab pertanyaan	FAQ	High
FR-W-017	Admin dapat melihat statistik data	Statistic	Medium
FR-W-018	Admin dapat mengatur range data	Setting	Low
FR-W-019	Seluruh pengunjung dapat melihat informasi sekilas tentang MI-FIK, random FAQ dan mengirim Feedback	Landing	Medium
FR-W-020	Admin, Dosen, dan Staf dapat mengelola profile	Profile	High
FR-W-021	Admin dapat mengelola about us, help editor, dan contact us	About	Medium
FR-W-022	Dosen dan Staf dapat melihat about us, help editor, dan contact us	About	Medium
FR-W-023	Admin dapat melihat login history seluruh user dan admin	Access	High
FR-W-024	Dosen dan Staff dapat melakukan request add dan remove tag	Profile	High
FR-W-025	Admin dapat mengelola Group dan User Group	Grouping	Medium

FR-W-026	Admin dapat mengelola role Pengguna	User	High
FR-W-027	Admin, Dosen, dan Staf dapat melihat dan mengelola Trash	Trash	High
FR-W-028	Admin, Dosen, dan Staf dapat melihat seluruh lokasi event	Location	High
FR-W-029	Dosen dan Staf dapat melihat dan bertanya	Profile	High

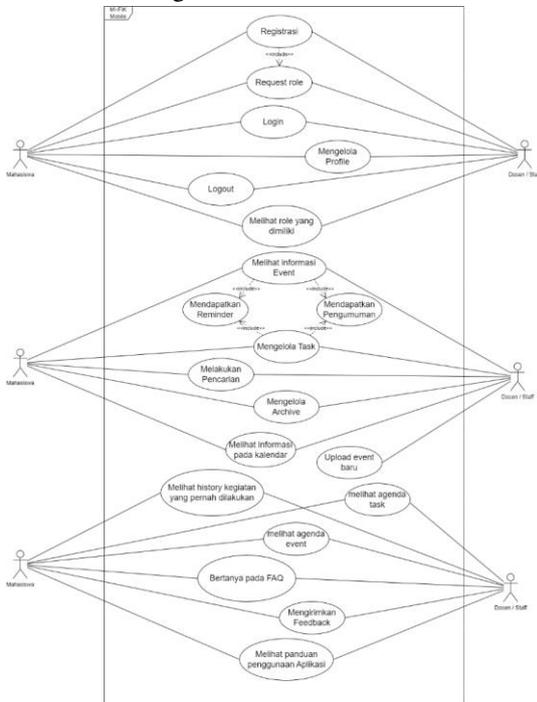


GAMBAR 4.4.2 Use Case Diagram Web

D. Use Case Diagram

Berdasarkan hasil analisis FR dan NFR yang didapatkan, serta diskusi dengan pihak client (FIK), maka dapat dirancangkan use case diagram sebagai berikut :

1. Use Case Diagram Mobile



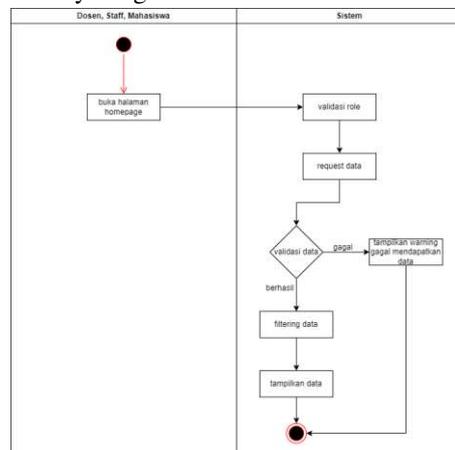
GAMBAR 4.4.1 Use Case Diagram Mobile

2. Use Case Diagram Dashboard Web

E. Activity Diagram

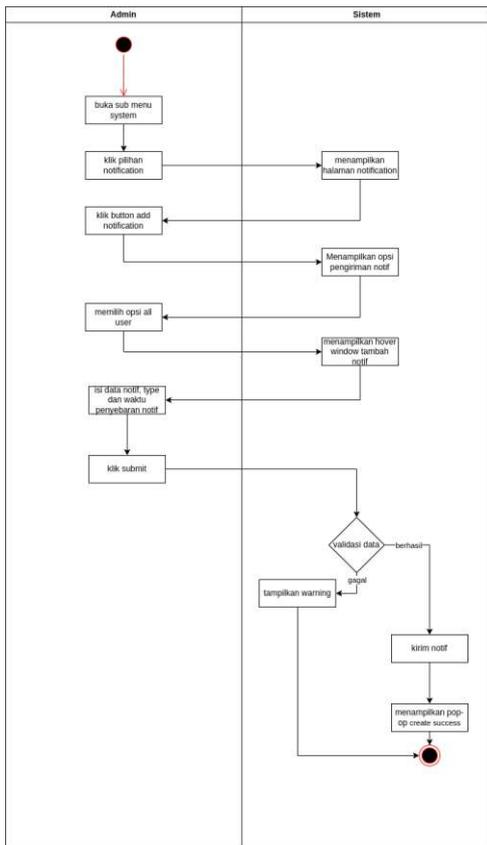
Berdasarkan rancangan use case dan use case description yang telah dibuat, maka dirancanglah activity diagram untuk menjelaskan lebih jelas mengenai alur aplikasi, serta aktivitas yg terjadi antara user (Mahasiswa, Dosen, staff, Admin) dengan sistem. Dari proses perancangan activity diagram ini, dihasilkan 38 activity diagram mobile dan 109 activity diagram web. Berikut merupakan salah satunya :

1. Activity Diagram Mobile



GAMBAR 4.5.1.1 Activity Diagram Show Event

2. Activity Diagram Dashboard Web

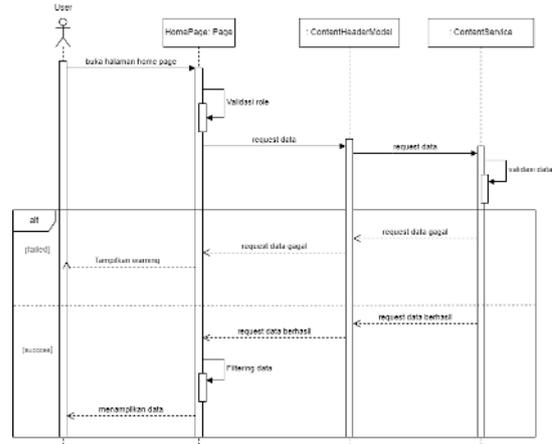


GAMBAR 4.5.2. 1
Activity Diagram Send Notification to All

F. Sequence Diagram

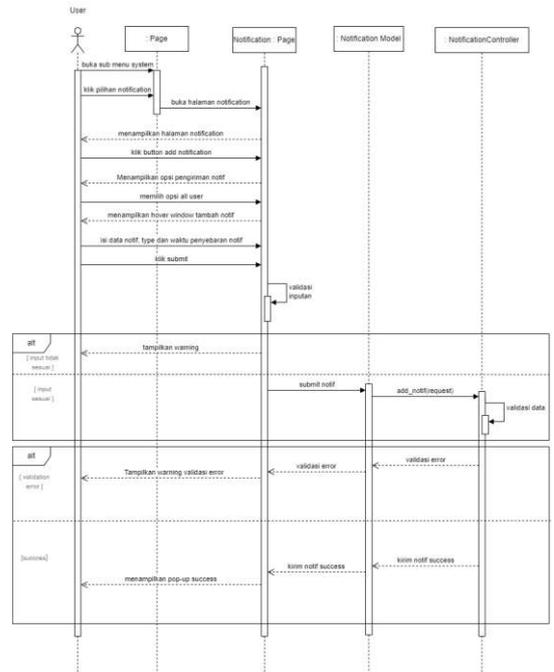
Berdasarkan rancangan use case, use case description, dan activity diagram yang telah dibuat, maka perancangan sequence diagram pun dapat dimulai. Sequence diagram ini dapat menjelaskan lebih detail mengenai urutan pesan dan berbagai kondisi yang terjadi di dalam sistem. Sequence diagram yang penulis buat untuk proyek pengembangan software ini, selain menggambarkan method dan urutan pesan, diagram ini juga menggambarkan interaksi UI dalam sistem. Dari proses perancangan sequence diagram ini, dihasilkan 38 sequence diagram mobile dan 109 sequence diagram web. Berikut merupakan salah satunya :

1. Sequence Diagram Mobile



GAMBAR 4.5.1. 1
Sequence Diagram Show Event

2. Sequence Diagram Web

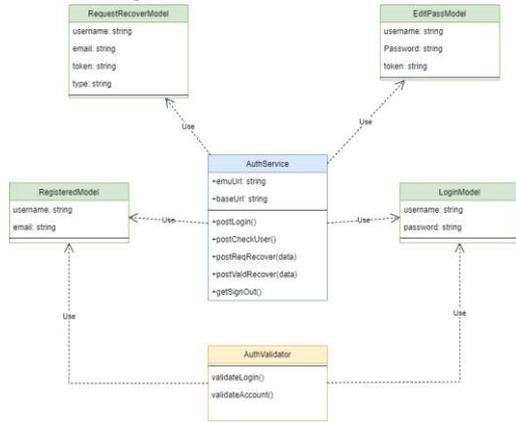


GAMBAR 4.6.2. 1
Sequence Diagram Send Notification to All

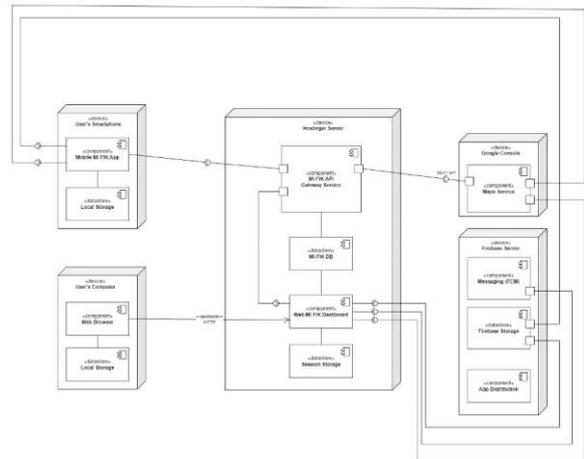
G. Class Diagram

Setelah use case, use case description, dan sequence diagram telah dibuat, maka perancangan class diagram pun dapat dimulai. Class diagram ini merupakan salah satu diagram yang paling dekat hubungannya dengan programming, karena class diagram ini menunjukkan kelas, objek, dan attribute yang dapat mewakili struktur dari code yang akan dibuat oleh developer nantinya. Dalam perancangan class diagram ini, penulis berdiskusi banyak dengan pihak developer untuk merancang struktur class yang baik. Dari proses perancangan class diagram ini, dihasilkan 13 class diagram mobile dan 28 class diagram web. Berikut merupakan salah satunya:

1. Class Diagram Mobile

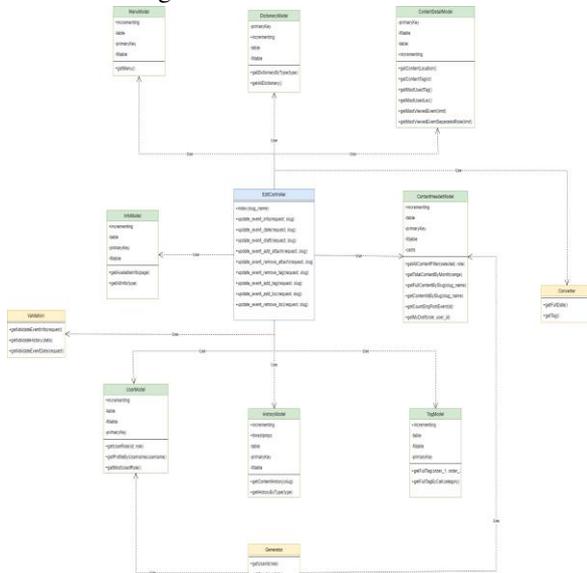


GAMBAR 4.7.1. 1
Class Diagram Auth



GAMBAR 4.8. 1
Deployment Diagram MI-FIK

2. Class Diagram Web



GAMBAR 4.7.2. 1
Class Diagram Event - Edit

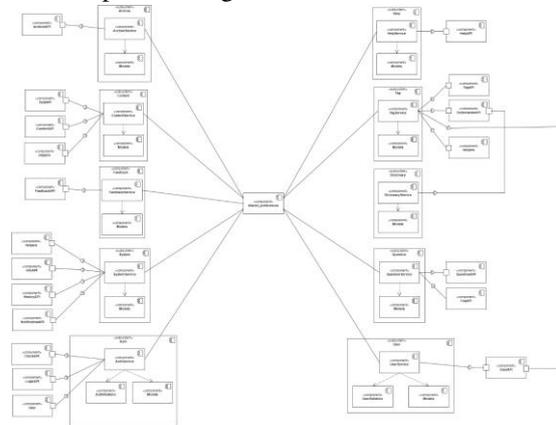
H. Deployment Diagram

Deployment diagram digunakan untuk menggambarkan bagaimana komponen-komponen software dan hardware fisik berinteraksi dalam environment sebenarnya. Diagram ini memberikan pandangan visual tentang bagaimana sistem software diimplementasikan pada infrastruktur fisik, termasuk server, perangkat jaringan, dan hardware lainnya. Dengan menggunakan diagram penyebaran, tim pengembang dapat memahami dengan lebih baik bagaimana komponen-komponen berhubungan dengan hardware dan menunjukkan komunikasi antara komponen terjadi melalui jaringan. Berikut merupakan deployment diagram dari sistem software MI-FIK :

I. Component Diagram

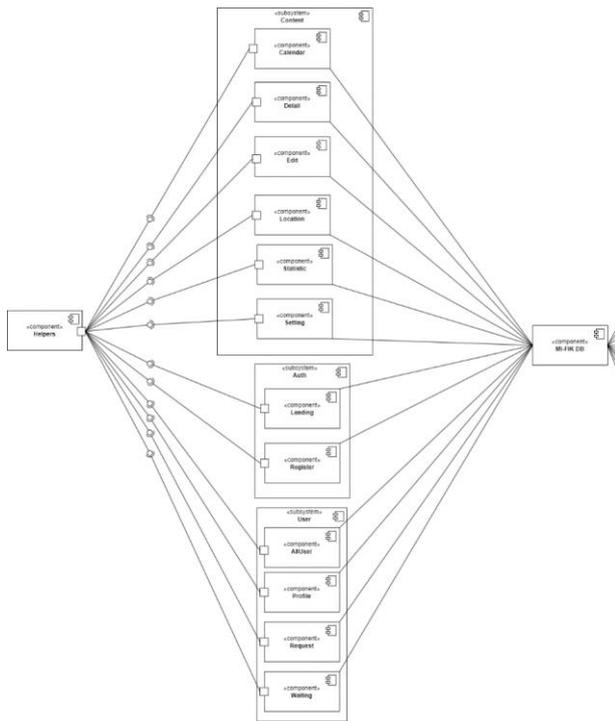
Diagram komponen ini digunakan untuk menggambarkan struktur dan hubungan antara komponen-komponen utama dalam suatu sistem perangkat lunak. Diagram ini memvisualisasikan bagaimana komponen-komponen berinteraksi satu sama lain, serta bagaimana informasi berinteraksi dalam sistem, dan bagaimana komponen-komponen tersebut menjadi unit-unit yang terpisah namun saling tergantung. Berikut merupakan component diagram dari aplikasi MI-FIK dan web dashboard MI-FIK :

1. Component Diagram Mobile



GAMBAR 4.9.1. 1
Component Diagram Mobile

2. Component Diagram Web



GAMBAR 4.9.2. 1
Component Diagram Web

J. List Table

Berdasarkan hasil analisis FR, serta diskusi dengan pihak client dan developer. Untuk memenuhi kebutuhan penyimpanan data pada aplikasi, maka didapatkan rancangan list table sebagai berikut :

No	Table Name	Description
1	admins	Menyimpan data admin
2	archives	Menyimpan data archive
3	relations	Menyimpan relasi archive, user, dan event
4	contents_details	Menyimpan data detail dari event (content attachment, location, content tag)
5	contents_headers	Menyimpan data utama dari event (nama event, deskripsi, waktu mulai dan selesainya event, dll)
6	contents_views	Menyimpan jumlah user yang melihat info event tertentu
7	dictionaries	Menyimpan variable-variable yang dibutuhkan system
8	ties_types	Menyimpan kategori dari setiap variable yang ada pada dictionaries
9	feedbacks	Menyimpan data feedback dari pengguna
10	groups_relations	Menyimpan data kumpulan user dalam group (group dosen, group staff)
11	helps	Menyimpan informasi petunjuk
12	histories	Menyimpan history segala hal yang dilakukan dosen, staff dan admin
13	infos	Menyimpan informasi petunjuk penggunaan setiap fitur / komponennya
14	menus	Menyimpan data list menu pada dashboard
15	notifications	Menyimpan data notifikasi
16	settings	Menyimpan data pengaturan secara umum
17	settings_systems	Menyimpan data pengaturan secara khusus
18	tags	Menyimpan data tag dan role

19	tasks	Menyimpan data task yang dibuat user
20	users	Menyimpan data user
21	questions	Menyimpan data pertanyaan beserta jawaban antara pengguna dan dosen atau admin
22	ds_resets	Menyimpan data perbaruan password, jika user melakukan forget password
23	personal_access_tokens	Menyimpan token user, ketika user melakukan login pada aplikasi
24	users_groups	Menyimpan data user yang terdaftar pada group
25	users_requests	Menyimpan data request add role dan remove role yang berasal dari user

K. ERD Diagram

Berdasarkan hasil analisis FR, use case, dan rancangan tabel database, serta diskusi dengan pihak client dan developer, maka dihasilkan Entity Relationship Diagram (ERD) sebagai berikut :

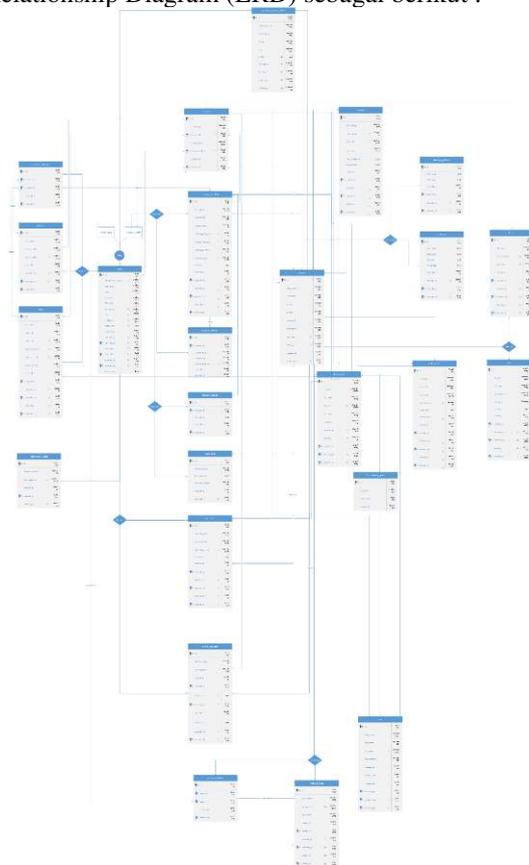


Diagram Link
: <https://bit.ly/ModelSystem-MIFIK>

V. KESIMPULAN

A. Kesimpulan

Kesimpulannya, tujuan yang ditetapkan untuk dokumen ini telah berhasil dicapai. Melalui hasil analisis dan evaluasi mendalam, penulis telah merancang dan menerapkan model UML dan ERD yang dapat mendukung proses pengembangan aplikasi MI-FIK, Model diagram ini dapat memastikan bahwa proses pengembangan aplikasi tetap konsisten dan sesuai dengan apa yang pengguna butuhkan.

Penulis juga telah memberikan gambaran komprehensif tentang alur kerja aplikasi Manajemen Informasi Fakultas Industri Kreatif (MI-FIK), serta menjelaskan berbagai tahapan, proses, dan berbagai fungsi dalam sistem. Dengan rancangan model UML ini, para stakeholder juga dapat membuat keputusan lebih lanjut nantinya, berdasarkan informasi model diagram yang sudah ada. Sehingga, proses pengoptimalkan kinerja aplikasi dan maintenance dapat lebih mudah untuk dilakukan. Namun dalam tugas akhir ini, penulis tidak menyertakan statechart diagram. Hal itu disebabkan oleh proyek ini yang tidak terlalu kompleks, waktu yg terbatas, dan batasan tugas akhir ini yang hanya memberi gambaran umum mengenai fungsi aplikasi.

B. Saran

Saran untuk penelitian kedepannya, diharapkan fitur pada aplikasi ini dapat di kembangkan kembali agar lebih baik dan lebih efektif dan lebih efisien dalam memenuhi kebutuhan aktivitas diruang lingkung Fakultas Indutri Kreatif (FIK)

REFERENSI

- [1] "Software Development Life Cycle (SDLC) Phases & Models." <https://www.guru99.com/software-development-life-cycle-tutorial.html#6> (accessed Dec. 02, 2022).
- [2] "SDLC - Overview." https://www.tutorialspoint.com/sdlc/sdlc_overview.htm (accessed Dec. 02, 2022).
- [3] James. Rumbaugh, Ivar. Jacobson, and Grady. Booch, The unified modeling language reference manual. Addison-Wesley, 1999.
- [4] "Entity Relationship Diagram (ERD) - What is an ER Diagram?" <https://www.smartdraw.com/entity-relationship-diagram/> (accessed Dec. 02, 2022).
- [5] "What is User Acceptance Testing (UAT): A Complete Guide." <https://www.softwaretestinghelp.com/what-is-user-acceptance-testing-uat/> (accessed Dec. 01, 2022).
- [6] D. Sasa, G. Darko, K. Mario, and Darko Huljenic, "Advantages of UML-based object-oriented system development".
- [7] "Software Management - Google Buku." https://books.google.co.id/books?hl=id&lr=&id=9yrdvfZX3CoC&oi=fnd&pg=PA21&dq=software+project+failure&ots=y5yDfdXkAU&sig=jQVlcGizpq7yYoDcOKV2-MYkFx4&redir_esc=y#v=onepage&q=software%20project%20failure&f=false (accessed Jan. 10, 2023).
- [8] "UML models and diagrams - IBM Documentation." <https://www.ibm.com/docs/en/rational-soft-arch/9.6.1?topic=diagrams-uml-models> (accessed Jan. 10, 2023).
- [9] "Why UML Modeling?" <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/why-uml-modeling/> (accessed Jan. 10, 2023).
- [10] "UML 2.5 Diagrams Overview." <https://www.uml-diagrams.org/uml-25-diagrams.html> (accessed Dec. 01, 2022).
- [11] "Use case diagrams are UML diagrams describing units of useful functionality (use cases) performed by a system in collaboration with external users (actors)." <https://www.uml-diagrams.org/use-case-diagrams.html> (accessed Dec. 02, 2022).
- [12] "UML activity diagrams are UML behavior diagrams which show flow of control or object flow with emphasis on the sequence and conditions of the flow." <https://www.uml-diagrams.org/activity-diagrams.html> (accessed Dec. 02, 2022).
- [13] "UML sequence diagrams overview of graphical notation - lifeline, message, execution specification, interaction use, etc." <https://www.uml-diagrams.org/sequence-diagrams.html> (accessed Dec. 02, 2022).
- [14] "UML Class and Object Diagrams Overview - common types of UML structure diagrams." <https://www.uml-diagrams.org/class-diagrams-overview.html> (accessed Dec. 02, 2022).
- [15] "What Is a Database | Oracle." <https://www.oracle.com/database/what-is-database/#WhatIsDBMS> (accessed Dec. 01, 2022).
- [16] "Entity Relationship (ER) Diagram Model with DBMS Example." <https://www.guru99.com/er-diagram-tutorial-dbms.html> (accessed Dec. 02, 2022).
- [17] I.-Y. Song, M. Evans, and U. E. K. Park, "A Comparative Analysis of Entity-Relationship Diagrams 1," 1995.
- [18] M. Asif, I. Ali, M. S. A. Malik, M. H. Chaudary, S. Tayyaba, and M. T. Mahmood, "Annotation of Software Requirements Specification (SRS), Extractions of Nonfunctional Requirements, and Measurement of Their Tradeoff," IEEE Access, vol. 7, pp. 36164–36176, 2019, doi: 10.1109/ACCESS.2019.2903133.
- [19] "Design Documentation in Software Engineering - GeeksforGeeks." <https://www.geeksforgeeks.org/design-documentation-in-software-engineering/> (accessed Dec. 02, 2022).
- [20] "What is User Acceptance Testing (UAT)? Examples." <https://www.guru99.com/user-acceptance-testing.html> (accessed Dec. 01, 2022).
- [21] "Software Engineering | Requirement Engineering - javatpoint." <https://www.javatpoint.com/software-engineering-requirement-engineering> (accessed Dec. 02, 2022).