TUGAS AKHIR – IF184802

# RANCANG BANGUN KAKAS PENGUKURAN KUALITAS PADA PRODUK MANAJEMEN PROYEK PERANGKAT LUNAK BERBASIS EXTENDED GOAL QUESTION METRIC

**NIDA REGITA FAUZIANTI**
**NRP 05111640000127**

Dosen Pembimbing
SARWOSRI, S.Kom., M.T.
Dr. UMI LAILI YUHANA, S.Kom., M.Sc.

DEPARTEMEN TEKNIK INFORMATIKA
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember
Surabaya 2020

**TUGAS AKHIR – IF184802**

# RANCANG BANGUN KAKAS PENGUKURAN KUALITAS PADA PRODUK MANAJEMEN PROYEK PERANGKAT LUNAK BERBASIS EXTENDED GOAL QUESTION METRIC

**NIDA REGITA FAUZIANTI**
**NRP 05111640000127**

Dosen Pembimbing
SARWOSRI, S.Kom., M.T.
UMI LAILI YUHANA, S.Kom., M.Sc.

DEPARTEMEN TEKNIK INFORMATIKA
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember
Surabaya 2020

***[Halaman ini sengaja dikosongkan]***

**UNDERGRADUATE THESIS – IF184802**

NIDA REGITA FAUZIANTI
NRP 05111640000127

Advisor
SARWOSRI, S.Kom., M.T.
UMI LAILI YUHANA, S.Kom., M.Sc.

INFORMATICS ENGINEERING DEPARTMENT
Faculty of Intelligent Electrical and Informatics Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2020

*[Halaman ini sengaja dikosongkan]*

**LEMBAR PENGESAHAN**

**RANCANG BANGUN KAKAS PENGUKURAN KUALITAS PADA PRODUK MANAJEMEN PROYEK PERANGKAT LUNAK BERBASIS EXTENDED GOAL QUESTION METRIC**

# TUGAS AKHIR

Diajukan Guna Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer pada
Rumpun Mata Kuliah Rekayasa Perangkat Lunak
Program Studi S-1 Departemen Teknik Informatika
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember

Oleh :
**NIDA REGITA FAUZIANTI**
NRP : 05111640000127

Disetujui oleh Dosen Pembimbing Tugas Akhir :

SARWOSRI, S.Kom., M.T. .....
NIP: 197608092001122001 (pembimbing 1)

Dr. UMI LAILI YUHANA, S.Kom., M.Sc. .....
NIP: 197906262005012002 (pembimbing 2)

**SURABAYA**
**JULI 2020**

*[Halaman ini sengaja dikosongkan]*

**RANCANG BANGUN KAKAS PENGUKURAN KUALITAS PADA PRODUK MANAJEMEN PROYEK PERANGKAT LUNAK BERBASIS EXTENDED GOAL QUESTION METRIC**


Nama Mahasiswa : Nida Regita Fauzianti
NRP : 05111640000127
Jurusan : Teknik Informatika ITS
Dosen Pembimbing I : Sarwosri, S.Kom., M.T.
Dosen Pembimbing II : Dr. Umi Laili Yuhana, S.Kom., M.Sc.


# ABSTRAK


*Suatu proyek pengembangan perangkat lunak dikatakan baik jika menghasilkan produk yang berkualitas. Pengukuran perangkat lunak merupakan salah satu cara untuk meningkatkan kualitas produk perangkat lunak. Berbagai metode telah dikembangkan untuk membantu aktivitas pengukuran produk perangkat lunak. Proses pengukuran dapat menjadi lebih efektif apabila berorientasi pada tujuan tertentu. Salah satu metode yang selaras dengan pemikiran ini adalah metode Extended Goal Question Metric (Extended GQM). Extended Goal Question Metric merupakan suatu pendekatan berbasis goal yang didasarkan pada Goal Question Metric yang banyak digunakan dan menjadi standard praktis dalam pendefinisian kerangka kerja pengukuran. Secara umum, pendekatan Extended Goal Question Metric tidak jauh berbeda dengan Origin Goal Question Metric.*

*Tahap pengukuran perangkat lunak berbasis GQM bekerja melalui empat tahapan yaitu tahap perencanaan, tahap definisi, tahap pengumpulan data dan tahap interpretasi. Dalam penelitian ini, tahap perencanaan merupakan tahap penentuan/pemilihan proyek yang akan diukur. Tahap definisi merupakan tahap pendefinisian pengukuran termasuk tujuan (goal), pertanyaan (question) dan metrik (metric). Masing-masing data goal, question dan metric yang tersimpan pada database kemudian direlasikan menggunakan suatu pembangkit GQM pada*

*kakas pengukuran. Pada tahap pengumpulan data, data jawaban dari pertanyaan yang telah didefinisikan pada tahap sebelumnya didapatkan dengan metode kuesioner. Tahap interpretasi merupakan tahap ketika data diproses menjadi hasil pengukuran. Kakas pengukuran melakukan pemrosesan data dari jawaban kuesioner yang diinputkan pada sistem, kemudian sistem mendokumentasikan hasil pengukuran produk perangkat lunak dalam bentuk laporan hasil analisis ukur baik secara kuantitatif maupun kualitatif.*

*Hasil dari penelitian ini memberikan kontribusi berupa kakas pengukuran kualitas pada produk manajemen proyek perangkat lunak yang dapat digunakan sebagai acuan dalam mengembangkan dan meningkatkan kualitas perangkat lunak.*

***Kata kunci: Extended Goal Question Metric, Kualitas, Perangkat Lunak, Produk, Pengukuran Perangkat Lunak***

**DESIGN AND IMPLEMENTATION OF SOFTWARE PRODUCT QUALITY MEASUREMENT TOOLS IN SOFTWARE PROJECT MANAGEMENT BASED ON EXTENDED GOAL QUESTION METRIC**


| | |
|---|---|
| Name | : NIDA REGITA FAUZIANTI |
| NRP | : 05111640000127 |
| Major | : Informatics Engineering ITS |
| Supervisor I | : SARWOSRI, S.Kom., M.T. |
| Supervisor II | : Dr. UMI LAILI YUHANA, S.Kom., M.Sc. |


## ABSTRACT


*A software development project is said to be good if it produces quality products. Software measurement is one way to improve the quality of software products. Various methods have been developed to assist software product measurement activities. The measurement process can be more effective if it is goal-oriented. One method that is in line with this thinking is the Extended Goal Question Metric (Extended GQM) method. The Extended Goal Question Metric is a goal-based approach that is based on the widely used Goal Question Metric and has become a practical standard in defining a measurement framework. In general, the Extended Goal Question Metric approach is not much different from the Origin Goal Question Metric.*

*The measurement stage of GQM-based software works through four stages, namely the planning stage, the definition stage, the data collection stage and the interpretation stage. In this study, the planning stage is the stage of determining / selecting the project to be measured. The definition stage is the stage of defining measurements including goals, questions and metrics. Each of the goal, question and metric data stored in the database is then correlated using a GQM generator on the measurement tool. At the data collection stage, the answer data from the questions that*

*had been defined in the previous stage were obtained by using the questionnaire method. The interpretation stage is the stage when the data is processed into measurement results. The measurement tool processes data from the answers to the questionnaires that are inputted into the system, then the system documents the measurement results of software products in the form of reports on the results of measuring analysis both quantitatively and qualitatively.*

*The results of this study contribute in the form of quality measurement tools in software project management products that can be used as a reference in developing and improving software quality.*

***Keywords: Extended Goal Question Metric, Quality, Software Measurement, Product, Software***

# KATA PENGANTAR

Segala puji dan syukur kehadirat Allah SWT yang telah memberikan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul :

**"RANCANG BANGUN KAKAS PENGUKURAN KUALITAS PADA PRODUK MANAJEMEN PROYEK PERANGKAT LUNAK BERBASIS EXTENDED GOAL QUESTION METRIC"**

Dalam pelaksanaan Tugas Akhir ini tentu penulis sebagai makhluk sosial tidak dapat menyelesaikannya tanpa bantuan dari pihak lain. Tanpa mengurangi rasa hormat, penulis memberikan penghargaan serta ucapan terima kasih yang sebesar-besarnya kepada:

1. Allah SWT atas segala nikmat dan rahmat yang telah diberikan selama ini serta senantiasa menemani dan memberi kemudahan kepada penulis dalam menyelesaikan Tugas Akhir.
2. Orang tua dan keluarga penulis, yang telah memberikan dukungan doa, moral, dan material kepada penulis sehingga penulis dapat menyelesaikan Tugas Akhir ini.
3. Sarwosri, S.Kom., M.T. dan Dr. Umi Laili Yuhana, S.Kom., M.Sc. selaku pembimbing I dan II yang telah membimbing dan memberikan motivasi, nasihat dan bimbingan dalam menyelesaikan Tugas Akhir ini.
4. Dr.Eng. Chastine Fatichah, S.Kom., M.Kom. selaku Ketua Departemen Teknik Informatika ITS dan seluruh dosen dan karyawan Departemen Teknik Informatika ITS yang telah memberikan ilmu dan pengalaman kepada penulis selama menjalani masa kuliah di Teknik Informatika ITS.

5. Maidina Nisa dan Ahmad Shidqi Firdaus yang menjadi teman seperjuangan, saling membantu dan menyemangati dalam mengerjakan Tugas Akhir ini.
6. Azkiatunnisa Rahma, Fariz Maulana, Fariz Putra, dan Nuzha Musyafira sebagai teman-teman *apple* yang telah memberi dukungan moral, menemani, dan menghibur penulis.
7. Seluruh mahasiswa Teknik Informatika ITS angkatan 2016 yang telah menjadi teman penulis selama menjalani masa kuliah di Teknik Informatika ITS.
8. Serta semua pihak yang yang telah turut membantu penulis dalam menyelesaikan Tugas Akhir ini.

Penulis sangat berharap bahwa apa yang dihasilkan dari tugas akhir ini bisa memberikan manfaat bagi semua pihak. Tidak ada manusia yang sempurna sekalipun penulis berusaha sebaik mungkin dalam menyelesaikan Tugas Akhir ini. Karena itu, penulis memohon maaf apabila terdapat kesalahan, kekurangan, maupun kelalaian yang telah penulis lakukan. Kritik dan saran yang membangun sangat diharapkan oleh penulis untuk dapat disampaikan untuk perbaikan selanjutnya.

Surabaya, Juli 2020

Nida Regita Fauzianti

# DAFTAR ISI

**ABSTRAK** ..... **vii**
**ABSTRACT** ..... **ix**
**KATA PENGANTAR** ..... **xi**
**DAFTAR ISI** ..... **xiii**
**DAFTAR GAMBAR** ..... **xvii**
**DAFTAR TABEL** ..... **xxii**
**DAFTAR KODE SUMBER** ..... **xxv**
**1 BAB I PENDAHULUAN** ..... **1**
1.1. Latar Belakang ..... 1
1.2. Rumusan Masalah ..... 2
1.3. Batasan Masalah ..... 3
1.4. Tujuan ..... 3
1.5. Manfaat ..... 3
1.6. Metodologi ..... 4
1.6.1. Penyusunan Proposal Tugas Akhir ..... 4
1.6.2. Studi Literatur ..... 4
1.6.3. Analisis dan Desain Perangkat Lunak ..... 4
1.6.4. Implementasi Perangkat Lunak ..... 5
1.6.5. Pengujian dan Evaluasi ..... 5
1.6.6. Penyusunan Buku ..... 5
1.7. Sistematika Penulisan Laporan ..... 5
**2 BAB II TINJAUAN PUSTAKA** ..... **9**
2.1. Penelitian Terkait ..... 9

2.2. Pengukuran Kualitas Produk DPTSI ..... 10
2.3. Goal Question Metric ..... 10
2.4. Extended Goal Question Metric ..... 12
2.5. Produk Perangkat Lunak ..... 14
2.6. Model Kualitas Produk Perangkat Lunak..... 14
2.7. Kerangka Kerja Evaluasi Kualitas Website Sistem Informasi Akademik..... 20
2.8. SQL & MySQL ..... 24
2.9. PHP..... 24
2.10. Laravel..... 25
2.11. Ajax Jquery..... 25
**3 BAB III ANALISIS DAN PERANCANGAN..... 27**
3.1. Analisis..... 27
3.1.1. Analisis Permasalahan..... 27
3.1.2. Deskripsi Umum Perangkat Lunak..... 28
3.1.3. Alur Penggunaan Aplikasi..... 30
3.1.4. Spesifikasi Kebutuhan Perangkat Lunak..... 31
3.2. Perancangan..... 52
3.2.1. Perancangan Arsitektur Perangkat Lunak ..... 52
3.2.2. Perancangan Basis Data..... 53
3.2.3. Perancangan Antarmuka Pengguna ..... 58
3.2.4. Perancangan Algoritma Melakukan Pengukuran 80
**4 BAB IV IMPLEMENTASI..... 81**
4.1. Lingkungan Implementasi ..... 81
4.2. Implementasi Antarmuka Sistem..... 81
4.2.1. Halaman Beranda ..... 81

4.2.2. Halaman Mendaftarkan Akun ..... 82
4.2.3. Halaman Membuat Proyek Pengukuran..... 83
4.2.4. Halaman Menjawab Pertanyaan..... 85
4.2.5. Halaman Melihat Analisis Tujuan ..... 86
4.2.6. Halaman Mengelola Ukuran ..... 86
4.2.7. Halaman Mengelola Pertanyaan ..... 88
4.2.8. Halaman Mengelola Data Proyek ..... 90
4.3. Implementasi Kasus Penggunaan..... 93
4.3.1. Implementasi Kasus Penggunaan Mendaftarkan Akun 93
4.3.2. Implementasi Kasus Penggunaan Membuat Proyek Pengukuran ..... 94
4.3.3. Implementasi Kasus Penggunaan Menjawab Pertanyaan..... 98
4.3.4. Implementasi Kasus Penggunaan Melihat Analisis Tujuan 103
4.3.5. Implementasi Kasus Penggunaan Mengelola Ukuran 104
4.3.6. Implementasi Kasus Penggunaan Mengelola Pertanyaan..... 107
4.3.7. Implementasi Kasus Penggunaan Mengelola Data Proyek 111
**5 BAB V UJI COBA DAN EVALUASI ..... 115**
5.1. Lingkungan Uji Coba..... 115
5.2. Material Pengujian ..... 115
5.3. Skenario Pengujian Kasus Penggunaan ..... 116
5.3.1. Pengujian Mendaftarkan Akun ..... 116

5.3.2. Pengujian Membuat Proyek Pengukuran..... 119
5.3.3. Pengujian Menjawab Pertanyaan..... 121
5.3.4. Pengujian Melihat Analisis Tujuan ..... 122
5.3.5. Pengujian Mengelola Ukuran ..... 124
5.3.6. Pengujian Mengelola Pertanyaan ..... 130
5.3.7. Pengujian Mengelola Data Proyek ..... 135
5.4. Evaluasi Pengujian ..... 142
5.5. Studi Kasus..... 143
5.5.1. Hasil Kuesioner ..... 144
5.5.2. Hasil Pengujian Studi Kasus..... 144
**BAB VI KESIMPULAN DAN SARAN ..... 147**
6.1. Kesimpulan..... 147
6.2. Saran..... 148
**6 DAFTAR PUSTAKA ..... 149**
**LAMPIRAN A ..... 150**
**LAMPIRAN B ..... 153**
**LAMPIRAN C ..... 156**
**LAMPIRAN D ..... 159**
**LAMPIRAN E ..... 161**
**LAMPIRAN F..... 163**
**LAMPIRAN G..... 165**
**BIODATA PENULIS ..... 167**

# DAFTAR GAMBAR

Gambar 2. 1 Pengukuran GQM (Basili;2001) .....11
Gambar 2. 2 Origin Goal Question Metric .....13
Gambar 2. 3 Extended Goal Question Metric.....13

Gambar 3. 1 Deskripsi Umum Sistem .....29
Gambar 3. 2 Alur Penggunaan Aplikasi .....31
Gambar 3. 3 Diagram Kasus Penggunaan Sistem Keseluruhan .35
Gambar 3. 4 Diagram aktivitas UC001.....38
Gambar 3. 5 Diagram aktivitas UC002.....41
Gambar 3. 6 Diagram aktivitas UC003.....44
Gambar 3. 7 Diagram aktivitas UC004.....46
Gambar 3. 8 Diagram aktivitas UC007 : Menambah data ukuran .....150
Gambar 3. 9 Diagram aktivitas UC007 : Menghapus data ukuran .....151
Gambar 3. 10 Diagram aktivitas UC007 : Mengubah Data Ukuran .....152
Gambar 3. 11 Diagram aktivitas UC008 : Menambah Data Pertanyaan.....154
Gambar 3. 12 Diagram aktivitas UC008 : Mengubah Data Pertanyaan.....155
Gambar 3. 13 Diagram aktivitas UC008 : Menghapus Data Pertanyaan.....155
Gambar 3. 14 Diagram aktivitas UC0009 : Menambah data proyek .....156
Gambar 3. 15 Diagram aktivitas UC0009 : Mengubah data proyek .....157
Gambar 3. 16 Diagram aktivitas UC0009 : Menghapus data proyek .....158
Gambar 3. 17 Ilustrasi Arsitektur Sistem Web .....53
Gambar 3. 18 CDM Sistem.....54
Gambar 3. 19 PDM Sistem .....54
Gambar 3. 20 Rancangan Antarmuka Beranda.....59

Gambar 3. 21 Rancangan Antarmuka Registrasi ..... 60
Gambar 3. 22 Rancangan Antarmuka Membuat Proyek Pengukuran : Langkah membuat proyek ..... 62
Gambar 3. 23 Rancangan Antarmuka Membuat Proyek Pengukuran : Langkah memilih goal ..... 62
Gambar 3. 24 Rancangan Antarmuka Membuat Proyek Pengukuran : Langkah pemetaan GQM ..... 63
Gambar 3. 25 Rancangan Antarmuka Menjawab Pertanyaan .... 65
Gambar 3. 26 Rancangan Antarmuka Analisis Tujuan ..... 66
Gambar 3. 27 Rancangan Antarmuka Melihat Daftar Ukuran/Metrik ..... 68
Gambar 3. 28 Rancangan Antarmuka Menambah Ukuran/Metrik ..... 69
Gambar 3. 29 Rancangan Antarmuka Mengubah Ukuran/Metrik ..... 70
Gambar 3. 30 Rancangan Antarmuka Menghapus Ukuran/Metrik ..... 71
Gambar 3. 31 Rancangan Antarmuka Melihat Daftar Pertanyaan ..... 72
Gambar 3. 32 Rancangan Antarmuka Menambah Pertanyaan ... 73
Gambar 3. 33 Rancangan Antarmuka Mengubah Pertanyaan .... 74
Gambar 3. 34 Rancangan Antarmuka Menghapus Ukuran/Metrik ..... 75
Gambar 3. 35 Rancangan Antarmuka Melihat Daftar Data Proyek ..... 76
Gambar 3. 36 Rancangan Antarmuka Menambah Data Proyek . 77
Gambar 3. 37 Rancangan Antarmuka Mengubah Data Proyek .. 78
Gambar 3. 38 Rancangan Antarmuka Menghapus Data Proyek 79

Gambar 4. 1 Implementasi Antarmuka Halaman Beranda ..... 82
Gambar 4. 2 Implementasi Antarmuka Mendaftarkan Akun ..... 83
Gambar 4. 3 Implementasi Antarmuka Halaman Membuat Proyek Pengukuran ( Langkah 1 ) ..... 84

Gambar 4. 4 Implementasi Antarmuka Halaman Membuat Proyek Pengukuran ( Langkah 2 ).....84
Gambar 4. 5 Implementasi Antarmuka Halaman Membuat Proyek Pengukuran ( Langkah 3 ).....85
Gambar 4.6 Implementasi Antarmuka Halaman Menjawab Pertanyaan.....86
Gambar 4. 7 Implementasi Antarmuka Halaman Analisis Tujuan .....86
Gambar 4. 8 Implementasi Antarmuka Melihat Daftar Data Ukuran .....87
Gambar 4. 9 Implementasi Antarmuka Menambah Data Ukuran .....87
Gambar 4. 10 Implementasi Antarmuka Menghapus Data Ukuran .....88
Gambar 4. 11 Implementasi Antarmuka Mengubah Data Ukuran .....88
Gambar 4. 12 Implementasi Antarmuka Melihat Daftar Data Pertanyaan.....89
Gambar 4. 13 Implementasi Antarmuka Menambah Data Pertanyaan.....89
Gambar 4. 14 Implementasi Antarmuka Menghapus Data Pertanyaan.....89
Gambar 4. 15 Implementasi Antarmuka Mengubah Data Pertanyaan.....90
Gambar 4. 16 Implementasi Antarmuka Melihat Daftar Data Proyek .....91
Gambar 4. 17 Implementasi Antarmuka Menambah Data Proyek .....91
Gambar 4. 18 Implementasi Antarmuka Menghapus Data Proyek .....92
Gambar 4. 19 Implementasi Antarmuka Mengubah Data Proyek .....92

*[Halaman ini sengaja dikosongkan]*

# DAFTAR TABEL

Tabel 2. 1 Perbandingan Metode GQM dan Extended GQM .... 13

Tabel 3. 1 Spesifikasi Karakteristik Pengguna pada Sistem..... 31
Tabel 3. 2 Kebutuhan Fungsional Sistem..... 33
Tabel 3. 3 Spesifikasi Kasus Penggunaan Mendaftarkan Akun . 37
Tabel 3. 4 Spesifikasi Kasus Penggunaan Membuat Proyek Pengukuran..... 39
Tabel 3. 5 Spesifikasi Kasus Penggunaan Menjawab Pertanyaan ..... 42
Tabel 3. 6 Spesifikasi Kasus Penggunaan Melihat Analisis Tujuan ..... 45
Tabel 3. 7 Spesifikasi Kasus Penggunaan Mengelola Ukuran ... 46
Tabel 3. 8 Spesifikasi Kasus Penggunaan Mengelola Pertanyaan ..... 48
Tabel 3. 9 Spesifikasi Kasus Penggunaan Mengelola Data Proyek ..... 50
Tabel 3. 10 Penjelasan Antarmuka Beranda..... 59
Tabel 3. 11 Penjelasan Antarmuka Registrasi ..... 61
Tabel 3. 12 Penjelasan Antarmuka Membuat Proyek Pengukuran ..... 63
Tabel 3. 13 Penjelasan Antarmuka Menjawab Pertanyaan ..... 65
Tabel 3. 14 Penjelasan Antarmuka Analisis Tujuan..... 66
Tabel 3. 15 Penjelasan Antarmuka Melihat Daftar Ukuran/Metrik ..... 68
Tabel 3. 16 Penjelasan Antarmuka Menambah Ukuran/Metrik . 69
Tabel 3. 17 Penjelasan Antarmuka Mengubah Ukuran/Metrik .. 70
Tabel 3. 18 Penjelasan Antarmuka Menghapus Ukuran/Metrik. 71
Tabel 3. 19 Penjelasan Antarmuka Melihat Daftar Pertanyaan.. 72
Tabel 3. 20 Penjelasan Antarmuka Menambah Pertanyaan ..... 73
Tabel 3. 21 Penjelasan Antarmuka Mengubah Pertanyaan ..... 74
Tabel 3. 22 Penjelasan Antarmuka Menghapus Ukuran/Metrik. 75

Tabel 3. 23 Penjelasan Antarmuka Melihat Daftar Data Proyek 76
Tabel 3. 24 Penjelasan Antarmuka Menambah Data Proyek.....77
Tabel 3. 25 Penjelasan Antarmuka Mengubah Data Proyek.....78
Tabel 3. 26 Penjelasan Antarmuka Menghapus Data Proyek.....79

Tabel 4. 1 Lingkungan Implementasi .....81
Tabel 4. 2 Penjelasan Melihat Halaman Registrasi.....93
Tabel 4. 3 Penjelasan Mendaftarkan Akun .....94
Tabel 4. 4 Penjelasan Melihat Daftar Daftar Goal.....95
Tabel 4. 5 Penjelasan Menampilkan Pertanyaan .....96
Tabel 4. 6 Penjelasan Menyimpan Data Proyek Pengukuran .....97
Tabel 4. 7 Penjelasan Menjawab Pertanyaan.....99
Tabel 4. 8 Penjelasan Menghitung Jawaban (1) .....101
Tabel 4. 9 Penjelasan Menghitung Jawaban (2) .....103
Tabel 4. 10 Penjelasan Melihat Analisis Tujuan.....104
Tabel 4. 11 Penjelasan Melihat Daftar Data Ukur.....105
Tabel 4. 12 Penjelasan Melihat Detail Data Ukur.....105
Tabel 4. 13 Penjelasan Menambah Data Ukur.....106
Tabel 4. 14 Penjelasan Mengubah Data Ukur.....106
Tabel 4. 15 Penjelasan Menghapus Data Ukur.....107
Tabel 4. 16 Penjelasan Melihat Daftar Pertanyaan .....108
Tabel 4. 17 Penjelasan Melihat Detail Pertanyaan.....108
Tabel 4. 18 Penjelasan Menambah Data Pertanyaan .....109
Tabel 4. 19 Penjelasan Mengubah Pertanyaan.....110
Tabel 4. 20 Penjelasan Menghapus Data Pertanyaan.....111
Tabel 4. 21 Penjelasan Melihat Daftar Data Proyek.....111
Tabel 4. 22 Penjelasan Melihat Detail Data Proyek .....112
Tabel 4. 23 Penjelasan Mengubah Data Proyek .....113
Tabel 4. 24 Penjelasan Menghapus Data Proyek.....113

Tabel 5. 1 Lingkungan Uji Coba.....115
Tabel 5. 2 Materi Pengujian Fungsional .....116
Tabel 5. 3 Pengujian Mendaftarkan Akun .....116

Tabel 5. 4 Pengujian Membuat Proyek Pengukuran ..... 120
Tabel 5. 5 Pengujian Menjawab Pertanyaan ..... 121
Tabel 5. 6 Pengujian Melihat Analisis Tujuan ..... 122
Tabel 5. 7 Pengujian Menambah Data Ukur ..... 124
Tabel 5. 8 Pengujian Mengubah Data Ukur ..... 127
Tabel 5. 9 Pengujian Menghapus Data Ukur..... 128
Tabel 5. 10 Pengujian Menambah Data Pertanyaan..... 130
Tabel 5. 11 Pengujian Mengubah Data Pertanyaan..... 133
Tabel 5. 12 Pengujian Menghapus Data Tujuan..... 134
Tabel 5. 13 Pengujian Menambah Data Proyek ..... 136
Tabel 5. 14 Pengujian Mengubah Data Proyek ..... 139
Tabel 5. 15 Pengujian Menghapus Data Pengguna ..... 140

# DAFTAR KODE SUMBER

Kode Sumber 4. 1 Fungsi Melihat Halaman Registrasi.....93
Kode Sumber 4. 2 Fungsi Mendaftarkan Akun .....94
Kode Sumber 4. 3 Fungsi Melihat Daftar Goal .....95
Kode Sumber 4. 4 Fungsi Menampilkan Pertanyaan.....96
Kode Sumber 4. 5 Fungsi Menyimpan Data Proyek Pengukuran .....97
Kode Sumber 4. 6 Fungsi Menjawab Pertanyaan.....99
Kode Sumber 4. 7 Fungsi Menghitung Jawaban (1).....101
Kode Sumber 4. 8 Fungsi Menghitung Jawaban (2).....102
Kode Sumber 4. 9 Fungsi Melihat Analisis Tujuan.....104
Kode Sumber 4. 10 Fungsi Melihat Daftar Data Ukur .....105
Kode Sumber 4. 11 Fungsi Melihat Detail Data Ukur.....105
Kode Sumber 4. 12 Fungsi Menambah Data Ukur.....106
Kode Sumber 4. 13 Fungsi Mengubah Data Ukur.....106
Kode Sumber 4. 14 Fungsi Menghapus Data Ukur .....107
Kode Sumber 4. 15 Fungsi Melihat Daftar Pertanyaan .....108
Kode Sumber 4. 16 Fungsi Melihat Detail Pertanyaan.....108
Kode Sumber 4. 17 Fungsi Menambah Data Pertanyaan .....109
Kode Sumber 4. 18 Fungsi Mengubah Pertanyaan.....110
Kode Sumber 4. 19 Fungsi Menghapus Data Pertanyaan.....111
Kode Sumber 4. 20 Fungsi Melihat Daftar Data Proyek .....111
Kode Sumber 4. 21 Fungsi Melihat Detail Data Proyek.....112
Kode Sumber 4. 22 Fungsi Mengubah Data Proyek.....113
Kode Sumber 4. 23 Fungsi Menghapus Data Proyek .....113

***[Halaman ini sengaja dikosongkan]***

# BAB I
# PENDAHULUAN

## 1.1. Latar Belakang

Seiring berkembangnya teknologi informasi, semakin banyak perangkat lunak yang dikembangkan untuk memenuhi kebutuhan dalam berbagai jenis bidang. Dalam suatu proyek pengembangan perangkat lunak, aspek kualitas perangkat lunak menjadi suatu hal yang penting. Kualitas perangkat lunak mencakup proses pengembangan dan produk. Untuk menjamin kualitas perangkat lunak maka perlu dilakukan pengukuran terhadap perangkat lunak yang dikembangkan. Pengukuran kualitas perangkat lunak dapat menjadi kompleks tergantung pada model evaluasi yang digunakan dan seberapa banyak faktor yang akan dinilai. Proses pengukuran akan efektif apabila fokus pada goal tertentu. Salah satu metode pengukuran yang cocok dengan pemikiran ini adalah metode Goal Question Metric (GQM).

Goal Question Metric memiliki kemampuan untuk menyelaraskan pertanyaan-pertanyaan yang mengerucut pada goal yang telah didefinisikan dan jawaban-jawabannya dituangkan dalam metric untuk selanjutnya diukur. Dari pengukuran ini dihasilkan suatu laporan kinerja yang memiliki validitas tinggi [1]. Sistem pengukuran kualitas perangkat lunak berbasis GQM bekerja melalui empat tahapan yaitu tahap perencanaan, tahap definisi, tahap pengumpulan data dan tahap interpretasi. Pada tahap definisi dilakukan pembentukan GQM. Tahap pengumpulan data adalah tahap memasukkan data proyek pengembangan perangkat lunak ke dalam basis data. Kemudian pada tahap interpretasi dilakukan perbandingan antara data yang telah dikumpulkan dengan baseline. Laporan hasil pengukuran menyatakan suatu goal tercapai atau tidak tercapai.

Extended Goal Question Metric menambahkan beberapa aspek untuk membatasi kemungkinan terjadi bias pada pengukuran, pengumpulan data dan Analisa yang mungkin terjadi. Aspek-aspek tersebut adalah *prioritization* dan *categorization*.

Aspek *prioritization* diharapkan dapat memfasilitasi pembatasan pengukuran dengan mereduksi jumlah goal dan pertanyaan sehingga tidak berpotensi menggunakan bias, sedangkan categorization memungkinkan penyeimbangan dimensi-dimensi yang berbeda. Dalam tugas akhir ini, objek pengukuran yang akan dibahas secara mendalam adalah produk. Produk meliputi *artifact*, *deliverables*, dan dokumen yang dibuat selama siklus hidup sistem. Menurut Lindstorm, pengukuran terhadap dokumen yang dihasilkan bersifat subjektif pada batas tertentu [3], sehingga dalam tugas akhir ini jenis produk yang diukur adalah versi release aplikasi.

Direktorat Pengembangan Teknologi dan Sistem Informasi (DPTSI) merupakan unit kerja di Institut Teknologi Sepuluh Nopember (ITS). DPTSI bertugas untuk menyediakan dan mengelola layanan Teknologi Informasi di lingkungan ITS. Sampai saat ini, belum ada divisi di DPTSI yang bertugas untuk melakukan pengukuran kualitas produk perangkat lunak. Oleh karena itu, DPTSI selama ini belum pernah melakukan pengukuran pada produk yang dikembangkan. Hal ini juga didukung dengan belum adanya kakas pengukuran khusus untuk melakukan pengukuran sesuai dengan tujuan/goal tertentu. Sehingga Tugas Akhir ini diharapkan dapat memberikan kontribusi melalui pembuatan kakas pengukuran kualitas berbasis web yang dapat membantu dalam mengukur kualitas produk manajemen proyek perangkat lunak. Dengan kakas pengukuran ini, diharapkan laporan informasi yg diperoleh dapat digunakan sebagai acuan untuk meningkatkan kualitas perangkat lunak yang dikembangkan.

## 1.2. Rumusan Masalah

Rumusan masalah yang diangkat dalam Tugas Akhir ini adalah sebagai berikut:

1. Bagaimana membuat kakas pengukuran perangkat lunak pada objek pengukuran produk manajemen proyek perangkat lunak berdasarkan metode Extended Goal Question Metric?

2. Bagaimana memetakan setiap goal yang sudah didefinisikan dengan question, serta memetakan question dengan metric yang tersedia dari aspek produk perangkat lunak?
3. Bagaimana menyajikan laporan hasil analisis pengukuran?

## 1.3. Batasan Masalah

Permasalahan yang dibahas pada Tugas Akhir ini memiliki beberapa batasan, yaitu sebagai berikut:

1. Kakas bantu pengukuran kualitas yang akan dibuat merupakan kakas bantu berbasis web, dengan menggunakan Bahasa pemrograman web (PHP, HMTL, CSS Javascript).
2. Studi kasus perangkat lunak yang akan diuji oleh kakas bantu sementara menggunakan data hasil kuesioner dengan sudut pandang pengguna akhir yaitu mahasiswa.
3. Model pengukuran kualitas yang dipakai adalah ISO 25010.

## 1.4. Tujuan

Tujuan dari pembuatan tugas akhir ini adalah sebagai berikut:

1. Membuat kakas pengukuran perangkat lunak pada objek pengukuran produk manajemen proyek perangkat lunak berdasarkan metode *Extended Goal Question Metric*.
2. Memetakan setiap *goal* yang sudah didefinisikan dengan *question*, serta memetakan *question* dengan *metric* yang tersedia dari aspek produk perangkat lunak
3. Menetapkan kesesuaian metrik dengan question menggunakan metode kuesioner.
4. Mendokumentasikan laporan hasil analisis pengukuran.

## 1.5. Manfaat

Manfaat yang diperoleh dari tugas akhir ini adalah sebagai berikut:

1. Memudahkan pengguna dalam melakukan pengukuran kualitas perangkat lunak.

2. Memberikan hasil yang dapat dijadikan sebagai acuan dalam menentukan kualitas suatu perangkat lunak.
3. Memberikan hasil yang dapat dijadikan sebagai acuan dalam perbaikan dan pengembangan suatu perangkat lunak.

## 1.6. Metodologi

Pembuatan Tugas Akhir ini dilakukan dengan menggunakan metodologi sebagai berikut:

### 1.6.1. Penyusunan Proposal Tugas Akhir

Tahap ini merupakan tahap awal dari penyusunan Tugas Akhir. Proposal Tugas Akhir memuat latar belakang diajukannya topik, rumusan masalah yang diangkat dari latar belakang, batasan masalah untuk membatasi penelitian agar tetap fokus pada rumusan masalah yang diangkat, tujuan pembuatan Tugas Akhir, serta manfaat yang diperoleh dari hasil Tugas Akhir. Proposal juga dilengkapi dengan tinjauan pustaka yang digunakan sebagai referensi pendukung penyusunan Tugas Akhir, metodologi yang berisi penjelasan mengenai tahapan penyusunan Tugas Akhir. Pada bagian akhir proposal dilengkapi dengan jadwal kegiatan penyusunan Tugas Akhir serta daftar pustaka untuk mencantumkan referensi yang digunakan dalam penyusunan Tugas Akhir.

### 1.6.2. Studi Literatur

Pada tahap ini dilakukan pencarian literatur yang menjadi acuan utama penyusunan tugas akhir yaitu metode *Extended Goal Question Metric*. Literatur yang dipelajari pada pengerjaan Tugas Akhir ini berasal dari jurnal ilmiah yang diambil dari berbagai sumber di internet, buku referensi, situs web serta berbagai macam literatur tentang konsep yang berkaitan dengan rumusan masalah.

### 1.6.3. Analisis dan Desain Perangkat Lunak

Analisis perangkat lunak dilakukan dengan aktivitas survey dan observasi di lapangan untuk menghasilkan kebutuhan

fungsional kakas yang akan dibangun. Pendefinisian kebutuhan fungsional dilengkapi dengan use case yang bertujuan memberikan gambaran lebih jelas mengenai apa yang dilakukan user baik dalam bentuk diagram maupun deskriptif. Perancangan struktur basis data dilakukan dengan pembuatan *Conceptual Data Model* dan *Physical Data Model*. Perancangan antar muka dimaksudkan untuk memberikan arahan tentang antar muka yang akan diimplementasikan pada pembangunan kakas pengukuran perangkat lunak.

### 1.6.4. Implementasi Perangkat Lunak

Pada tahap ini akan dilakukan pengembangan aplikasi pengukuran kualitas manajemen proyek perangkat lunak menggunakan bahasa pemrograman PHP pada IDE Visual Studio Code, laravel sebagai *framework*, serta *library* pendukung lainnya.

### 1.6.5. Pengujian dan Evaluasi

Pengujian perangkat lunak ini menggunakan metode pengujian black box yang berfokus pada persyaratan fungsional perangkat lunak yang dibuat. Pengujian kakas pengukuran perangkat lunak berbasis Extended Goal Question Metric ini terdiri dari pengisian data dan pengujian pembangkit GQM.

### 1.6.6. Penyusunan Buku

Pada tahap ini dilakukan penyusunan buku yang menjelaskan seluruh konsep, teori dasar dari metode yang digunakan, implementasi, serta hasil yang telah dikerjakan sebagai dokumentasi dari pelaksanaan Tugas Akhir.

## 1.7. Sistematika Penulisan Laporan

Sistematika penulisan laporan Tugas Akhir adalah sebagai berikut:

**Bab I Pendahuluan**

Bab ini berisikan penjelasan mengenai latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat, metodologi, dan sistematika penulisan dari pembuatan Tugas Akhir.

**Bab II Tinjauan Pustaka**

Bab ini berisi kajian teori dari metode yang digunakan dalam penyusunan Tugas Akhir ini. Secara garis besar, bab ini berisi tentang *Extended Goal Question Metric*, *Hierarchical Cumulative Voting* dan *library* yang digunakan.

**Bab III Perancangan Sistem**

Bab ini berisi pembahasan mengenai analisis dan perancangan sistem secara rinci sebagai landasan bagi pembangunan sistem yang dibuat. Pembahasan meliputi analisis kebutuhan, perancangan *usecase*, perancangan struktur basis data, serta perancangan antarmuka sistem.

**Bab IV Implementasi**

Bab ini membahas implementasi dari perancangan yang telah dibuat pada bab sebelumnya. Penjelasan berupa kode sumber yang digunakan untuk proses implementasi.

**Bab V Uji Coba Dan Evaluasi**

Bab ini membahas rencana pengujian, kasus uji, tahapan uji coba, kemudian hasil uji coba dievaluasi terhadap kinerja dari sistem yang dibangun.

**Bab VI Kesimpulan dan Saran**

Bab ini merupakan bab yang menyampaikan kesimpulan dari hasil uji coba yang dilakukan, masalah-masalah yang dialami pada proses dan tertulis saat pengerjaan Tugas Akhir, dan saran untuk pengembangan solusi ke depannya.

**Bab VI. Kesimpulan dan Saran**

Bab ini merupakan bab terakhir yang menyampaikan kesimpulan dari hasil uji coba yang dilakukan dan saran untuk pengembangan perangkat lunak ke depannya.

***[Halaman ini sengaja dikosongkan]***

# BAB II
# TINJAUAN PUSTAKA

Bab tinjauan pustaka berisi mengenai penjelasan teori yang berkaitan dengan implementasi perangkat lunak. Penjelasan tersebut bertujuan untuk memberikan gambaran mengenai sistem yang akan dibangun dan berguna sebagai pendukung dalam pengembangan perangkat lunak.

## 2.1. Penelitian Terkait

Beberapa studi sebelumnya telah memperlihatkan efektifitas metode *Goal Question Metric* dalam pengukuran kualitas perangkat lunak. Penelitian tentang pengukuran menggunakan metode *Goal Question Metric* sudah pernah dilakukan baik di luar negeri maupun di Indonesia. Studi oleh Wiwin Kiswinardi menggunakan metode *Extended Goal Question Metric* dalam pengukuran kualitas layanan e-government. Penelitian ini menggunaakan kerangka kerja E-Gov *Performance Measurement.* Aspek yang diukur merupakan tingkat kepuasan pengguna, *response time, quality* dan *operation* [1]. Metode yang digunakan pada penelitian tersebut menggunakan pengumpulan data melalui kuesioner kemudian diukur melalui matrik tertentu. Kuesioner berisi pertanyaan yang dibuat berdasarkan goal yang telah didefinisikan pada tahap definisi. Kuesioner kemudian disebarkan pada partisipan terkait, jawaban dari kuesioner diukur dengan metrik.

Penulis sampai saat ini belum menemukan penelitian terkait pembuatan kakas pengukuran kualitas perangkat lunak berbasis metode *Goal Question Metric* dengan menggunakan kerangka kerja evaluasi kualitas website sistem informasi akademik. Oleh karena itu, dalam Tugas Akhir ini penulis akan mengintegrasikan pengukuran berbasis *Goal Question Metric* dengan kerangka kerja evaluasi kualitas website sistem informasi akademik dalam suatu aplikasi pengukuran kualitas.

## 2.2. Pengukuran Kualitas Produk DPTSI

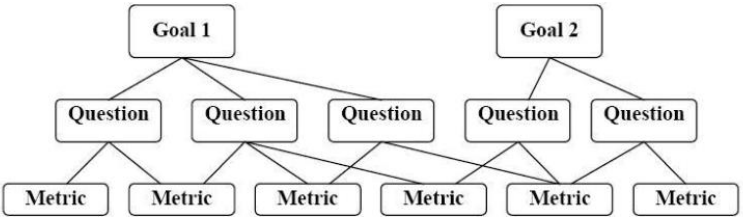
Direktorat Pengembangan Teknologi dan Sistem Informasi (DPTSI) merupakan unit kerja di Institut Teknologi Sepuluh Nopember (ITS). DPTSI bertugas untuk menyediakan dan mengelola layanan Teknologi Informasi di lingkungan ITS. Terkait peran, DPTSI berperan untuk mendukung aktivitas akademik, penelitian dan pengabdian masyarakat, serta manajerial di lingkungan ITS dalam rangka membantu ITS mencapai visi misinya.

Sampai saat ini, belum ada divisi di DPTSI yang bertugas untuk melakukan pengukuran kualitas produk perangkat lunak. Oleh karena itu, DPTSI selama ini belum pernah melakukan pengukuran pada produk yang dikembangkan. Dalam Tugas Akhir ini, akan dibuat kakas pengukuran kualitas untuk membantu DPTSI dalam melakukan pengukuran produk yang telah dikembangkan, yang hasilnya diharapkan dapat membantu perbaikan dalam pengembangan produk.

## 2.3. Goal Question Metric

*Goal Question Metric* adalah suatu pendekatan yang didasarkan pada asumsi bahwa untuk melakukan pengukuran yang berhasil harus terlebih dahulu menetapkan goal, baik terhadap organisasi maupun bagi project itu sendiri, kemudian harus pula ditelusuri data-data yang diharapkan untuk dapat mendefinisikan goal-goal tersebut secara operasional hingga akhirnya dapat menyediakan suatu kerangka kerja untuk menginterpretasikan data yang dinyatakan sebagai goal.

Model ini dikembangkan dengan mengidentifikasi seperangkat *productivity goal* yang terdapat pada perusahaan, divisi, maupun organisasi proyek. Goal ini mengacu pada model objek pengukurannya, kemudian dibuatlah pertanyaan yang lengkap dan komprehensif untuk mendefinisikan goal tersebut. Model pengukuran *Goal Question Metric* terdiri dari tiga tingkatan seperti diilustrasikan pada Gambar 2.1.




**Gambar 2. 1 Pengukuran GQM (Basili;2001)**

1. Tingkat Konseptual (Goal)

   Tujuan didefinisikan untuk sebuah objek, untuk berbagai alasan, sehubungan dengan berbagai model kualitas, dari berbagai sudut pandang, terkait pada lingkungan tertentu. Yang menjadi objek pengukurannya adalah produk, proses atau sumber daya.

2. Tingkat Operasional (*Question*)

   Seperangkat pertanyaan digunakan untuk mengkarakteristikan cara penilaian/pencapaian dari tujuan tertentu yang akan ditunjukkan berdasarkan terpenuhinya model pengkarakteristikan.

3. Tingkat Kuantitatif (*Metric*)

   Seperangkat data dikaitkan dengan setiap pertanyaan dengan tujuan untuk menjawabnya dengan cara kuantitatif. Data yang dimaksud bisa berupa data objektif atau data subjektif.

Model *Goal Question Metric* adalah metode struktur hirarki (Gambar 2.1) yang dimulai dengan menetapkan tujuan pengukuran, objek pengukuran, isu pengukuran dan sudut pandang mana yang akan diambil. Setelah dilakukan penetapan *goal*, dibentuklah beberapa pertanyaan yang kemudian dibentuk menjadi metrik.

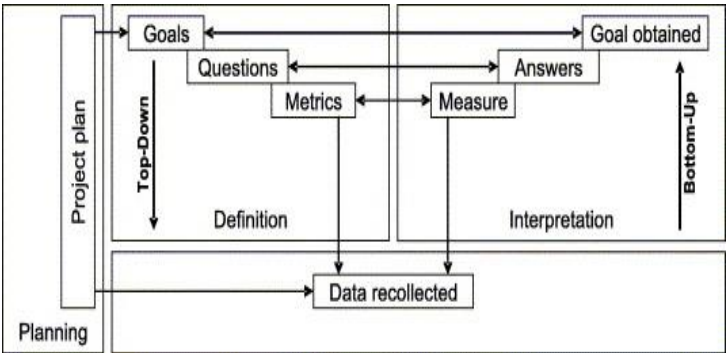
Metrik yang sama dapat digunakan untuk menjawab pertanyaan yang berbeda dengan tujuan yang sama. [2]

Dalam Tugas Akhir ini, pendekatan Goal Question Metric akan digunakan dalam pengembangan aplikasi. Goal Question Metric diterapkan dalam perancangan basis data aplikasi. Setiap Goal, Question, dan Metric masing-masing didefinisikan terlebih dulu dan dimasukkan ke dalam tabel master pada *database.* Kemudian dihubungkan dengan suatu pembangkit GQM yang merelasikan data goal, question dan metric yang telah didefinisikan sebelumnya ke dalam tabel pembangkit GQM. Jika relasi GQM dihapus, data pada tabel master tidak akan ikut terhapus dan akan tetap tersimpan, data tersebut dapat digunakan lagi jika diperlukan.

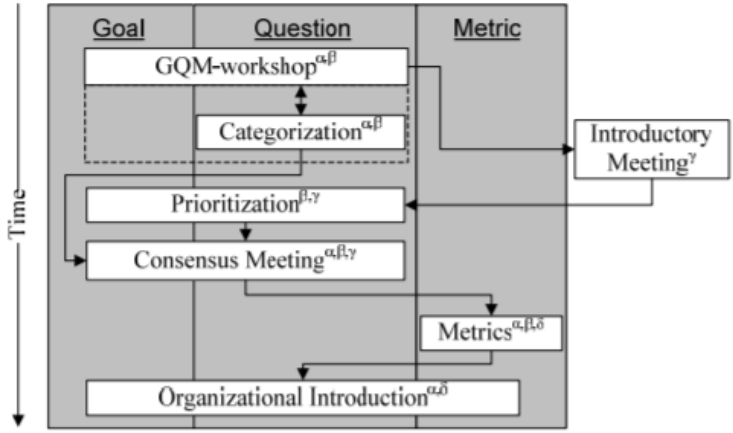
## 2.4. Extended Goal Question Metric

*Extended Goal Question Metric* merupakan suatu pendekatan yang didasarkan pada *Goal Question Metric* yang dikembangkan oleh Basili dan Weiss kemudian disempurnakan oleh Rombach merupakan salah satu pendekatan berbasis goal yang paling banyak digunakan dan menjadi standard praktis dalam pendefinisian kerangka kerja pengukuran. *Extended Goal Question Metric* menambahkan beberapa aspek untuk membatasi kemungkinan terjadi bias pada pengukuran, pengumpulan data dan analisa yang mungkin terjadi. Aspek-aspek tersebut adalah *prioritization* dan *categorization* [5].

Aspek *prioritization* diharapkan dapat memfasilitasi pembatasan pengukuran dengan mereduksi jumlah goal dan pertanyaan sehingga tidak berpotensi menggunakan bias, sedangkan *categorization* memungkinkan penyeimbangan dimensi-dimensi yang berbeda. Perbandingan metode *Origin Goal Question Metric* dan *Extended Goal Question Metric* dapat dilihat pada Gambar 2.2 dan Gambar 2.3.




**Gambar 2. 2 Origin Goal Question Metric**




**Gambar 2. 3 Extended Goal Question Metric**

Dalam Tugas Akhir ini, pendekatan *Extended Goal Question Metric* diterapkan pada kelas penggunaan membuat proyek pengukuran. Ketika membuat proyek pengukuran baru, pengguna tidak perlu mengukur semua *goal* yang tersedia pada sistem. Sistem menampilkan pilihan untuk memilih *goal* mana saja yang hendak diukur sesuai dengan prioritas pengguna dalam melakukan pengukuran.

## 2.5. Produk Perangkat Lunak

Objek pengukuran perangkat lunak dapat dilihat dari tiga jenis objek, yaitu produk, proses dan *resource* [1]. Objek pengukuran produk dilihat dari sudut pandang produk yang dihasilkan. Penilaian ini berorientasi pada bagaimana suatu perangkat lunak dapat dikembangkan sesuai dengan yang diharapkan oleh pengguna.

Produk meliputi seluruh *artifacts, deliverables*, dan dokumen yang dihasilkan selama siklus hidup suatu sistem. Artefak (*artifact*) merupakan sebuah cara komunikasi yang digunakan untuk menyajikan *view* tertentu dari arsitektur. Artefak diatur ke dalam bentuk katalog, matriks, dan diagram. Sesuatu yang dihasilkan dari eksekusi proyek dan diberikan kepada klien disebut dengan *deliverable*. Objek yang dihasilkan bisa dalam bentuk nyata atau tidak nyata. Ada dua jenis dari *deliverable*, yaitu *project deliverable* dan *product deliverable*.

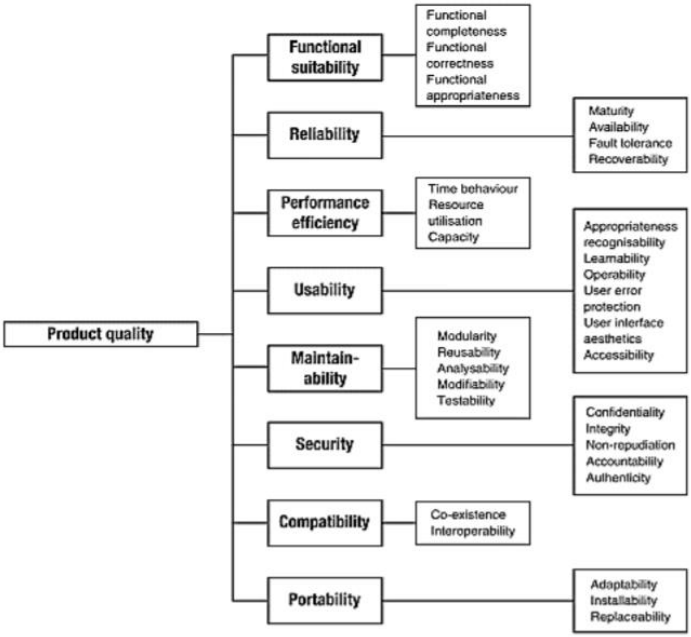
*Project deliverable* merupakan sesuatu yang dihasilkan dalam proyek. Umumnya contoh *project deliverable* berupa dokumen-dokumen yang dihasilkan di sepanjang proyek, seperti *project plan* dan laporan (reports). *Product deliverable* lebih menitikberatkan pada produk akhir yang dibuat dalam suatu proyek, seperti *hardware, software, apps*, kontrak dan hasil assessment.

Menurut Lindstorm, pengukuran terhadap dokumen yang dihasilkan bersifat subjektif pada batas tertentu [3], sehingga dalam tugas akhir ini jenis produk yang diukur adalah versi release aplikasi. Dari sudut pandang produk, pengukuran dapat dilakukan dengan cara mengintegrasikan metode GQM dengan kerangka kerja kualitas perangkat lunak seperti *McCall's Software Quality Framework*, ISO 9126 maupun ISO 25010.

## 2.6. Model Kualitas Produk Perangkat Lunak

Pengukuran perangkat lunak merupakan suatu hal yang terpenting dalam penjaminan kualitas perangkat lunak. Pengukuran perangkat lunak adalah suatu proses yang digunakan untuk mengidentifikasi ketepatan, kelengkapan dan mutu dari perangkat

lunak. Setiap pengukuran membutuhkan tersedianya suatu standar pengukuran. Salah satu parameter dalam pengukuran kualitas suatu perangkat lunak adalah dengan menggunakan ISO 25010 yang dijelaskan pada Gambar 2.4.




**Gambar 2. 4 ISO 25010**

Berikut merupakan penjelasan mengenai gambar 2.2 :

1. Functional Suitability

   *Functional Suitability* menilai sejauh mana produk atau sistem menyediakan fitur-fitur yang memenuhi kebutuhan dalam kondisi tertentu. Functional Suitability memiliki 3 sub-kualitas yaitu :

a) *Functional Completeness* yang mengukur sejauh mana sekumpulan fungsi dapat memenuhi sebuah tugas atau kebutuhan pengguna.

b) *Functional Correctness* yang mengukur sejauh mana sebuah sistem menyediakan hasil yang benar sesuai kebutuhan.

c) *Functional Appropriateness* yang mengukur sejauh mana fungsi dapat memfasilitasi pencapaian suatu kebutuhan.

2. Performance Efficiency

Performance menilai sejauh mana performa relatif terhadap jumlah sumber daya yang digunakan dalam kondisi tertentu. Performance memiliki 3 sub-kualitas yaitu:

a) *Time Behaviour* yang mengukur sejauh mana respon dan waktu ekseskusi proses dapat memenuhi suatu kebutuhan.
b) *Resource Utilization* yang mengukur sejauh mana jumlah dan tipe sumber daya yang digunakan sistem ketika memenuhi suatu kebutuhan.
c) *Capacity* yang mengukur sejauh mana batas maksimal suatu produk atau sistem memenuhi kebutuhan.

3. Compatibility

*Compatibility* menilai sejauh mana suatu produk, sistem, atau komponen dapat bertukar informasi ke produk, sistem, atau komponen lain dan menjalankan fungsi yang dibutuhkan dalam lingkup perangkat keras atau perangkat lunak yang sama. Compatibility memiliki 2
sub-kualitas yaitu:

a) *Co-existence* yang mengukur sejauh mana sebuah produk dapat menjalankan fungsi secara efisien sembari membagi lingkungan dan sumber daya dengan produk lain tanpa adanya konflik dengan produk tersebut.
b) *Interoperability* yang mengukur sejauh mana dua atau lebih sistem dapat saling bertukar informasi dan menggunakan informasi tersebut.

4. Usability

*Usability* menilai sejauh mana suatu produk atau sistem dapat digunakan oleh pengguna yang dituju untuk mencapai suatu tujuan dengan efisien. Usability memiliki 6 sub-kualitas yaitu:

a) *Appropriateness Recognisability* yang mengukur sejauh mana pengguna dapat mengenali apakah produk atau sistem dapat memenuhi kebutuhan.
b) *Learnability* yang megukur sejauh mana sebuah produk atau sistem dapat digunakan oleh pengguna untuk mencapai suatu tujuan dengan mempelajari produk tersebut dengan efektif dan efisien.
c) *Operability* yang mengukur sejauh mana produk atau sistem memiliki atribut yang memudahkan penggunaan.
d) *User Error Protection* yang mengukur sejauh mana sebuah sistem dapat melindungi pengguna dari *error*.
e) *User Interface Aesthetics* yang mengukur sejauh mana *User Interface* dapat menyediakan interaksi yang memuaskan pengguna.
f) *Accessibility* yang mengukur sejauh mana produk atau sistem dapat digunakan oleh berbagai macam karakteristik dan kapabilitas pengguna untuk memenuhi suatu kebutuhan.

5. Reliability

*Reliability* menilai sejauh mana suatu sistem, produk, atau komponen dapat mengerjakan sebuah fungsi dalam kondisi tertentu dan dalam waktu tertentu. Reliability memiliki 4 sub-kualitas yaitu:

a) *Maturity* yang mengukur sejauh mana kehandalan sebuah sistem dalam memenuhi kebutuhan secara normal.
b) *Availability* yang mengukur sejauh mana produk atau sistem dapat beroperasi dan dapat diakses ketika dibutuhkan.
c) *Fault Tolerance* yang mengukur sejauh mana sebuah sistem beroperasi ketika ada kesalahan pada perangkat lunak atau perangkat keras.
d) *Recoverability* yang mengukur sejauh mana sebuah sistem dapat mengembalikan data dan keadaan ke kondisi awal ketika terjadi sebuah kegagalan sistem.

6. Security

*Security* menilai sejauh mana suatu produk atau sistem dapat melindungi informasi dan data sehingga seseorang, sistem, atau produk lain dapat mengakses data atau informasi tersebut sesuai dengan hak akses mereka masing-masing. Security memiliki 5 sub-kualitas yaitu:

a) Confidentiality yang mengukur sejauh mana suatu data pada sistem atau produk hanya dapat diakses oleh pengguna yang memiliki hak akses.
b) Integrity yang mengukur sejauh mana sebuah sistem mencegah akses tak dikenal yang dapat memodifikasi sistem tersebut.
c) Non-repudiation yang mengukur sejauh mana setiap aksi atau peristiwa yang terjadi pada sistem dapat dibuktikan ketika dibutuhkan. Non-repudiaton memiliki

sebuah poin kualitas yaitu Utilization of Digital Signature yang mengukur proporsi kejadian yang diproses dengan *digital signature*.

d) Accountability yang mengukur sejauh mana aksi dari sebuah komponen dapat dilacak secara unik ke komponen tersebut.
e) Authenticity yang mengukur sejauh mana identitas suatu subjek atau sumber daya dapat dibuktikan oleh orang yang melakukan klaim terhadap identitas tersebut.

7. Maintainability

*Maintainability* menilai sejauh mana seberapa efektif dan efisien suatu produk atau sistem dapat dimodifikasi oleh pengelola. Maintainability memiliki 5 sub-kualitas yaitu:

a) *Modularity* yang mengukur sejauh mana perubahan sebuah komponen pada sistem hanya dapat berpengaruh kecil kepada komponen lain.
b) *Reusability* yang mengukur sejauh mana sebuah aset dapat digunakan lebih dari satu sistem.
c) *Analysability* yang mengukur apakah memungkinkan untuk menilai suatu dampak perubahan pada produk atau sistem, atau untuk mendiagnosis kekurangan dan kegagalan produk yang dapat terjadi, atau untuk mengidentifikasi bagian yang akan dimodifikasi.
d) *Modifiability* yang mengukur sejauh mana modifikasi dapat dilakukan dengan efektif dan efisien tanpa mengurangi kualitas sistem.
e) Testability yang mengukur sejauh mana kriteria sebuah tes dapat diterapkan ke sebuah sistem dan apakah beberapa tes bisa dilakukan untuk memenuhi kriteria-kriteria tersebut.

8) Portability

*Portability* menilai sseberapa efektif dan efisien suatu sistem, produk, atau komponen dapat dipindahkan dari suatu lingkungan operasional ke lingkungan operasional lain. Portability memiliki 3 sub-kualitas yaitu :

a) *Adaptability* yang mengukur sejauh mana sebuah sistem dapat menyesuaikan diri dengan perubahan perangkat keras atau perangkat lunak secara efektif dan efisien.
b) *Installability* yang mengukur sejauh mana sebuah sistem dapat ditempatkan atau dikeluarkan dari lingkungan tertentu.
c) *Replaceability* yang mengukur sejauh mana sebuah produk dapat tergantikan oleh produk lain yang sejenis.

Pada Tugas Akhir ini, penulis menggunakan empat faktor kualitas dari ISO 25010 sebagai parameter pengukuran. Empat faktor tersebut yaitu *Functional Suitability, Reliability, Performance Efficiency*, dan *Usability*.

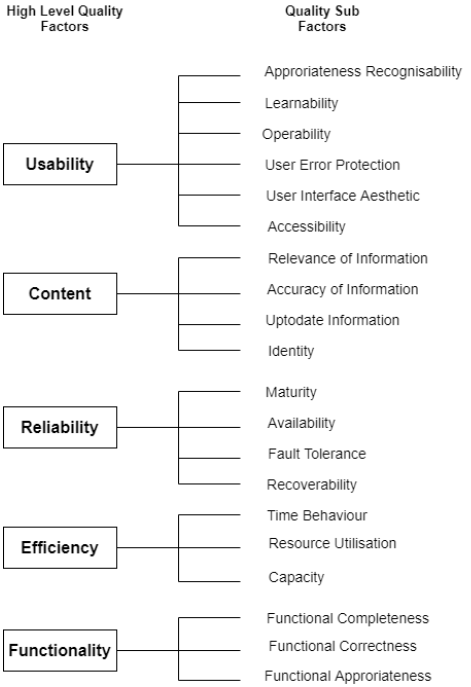
## 2.7. Kerangka Kerja Evaluasi Kualitas Website Sistem Informasi Akademik

Suatu kualitas dapat berbeda sesuai dengan sudut pandang penilaiannya. Sampai saat ini, telah banyak parameter penilaian kualitas perangkat lunak yang sering digunakan dalam pengukuran kualitas perangkat lunak. Tetapi belum ada kerangka kerja khusus yang membahas mengenai evaluasi kualitas website sistem informasi akademik. Tsigereda (2010) dalam penelitiannya melakukan studi dengan menggabungkan beberapa model kualitas untuk mengembangkan kerangka kerja baru khusus evaluasi kualitas website sistem informasi akademik [7]. Faktor kualitas yang diusulkan merupakan *usability, content, reliability, efficiency* dan *functionality*. Faktor kualitas tersebut digunakan sebagai masukkan pada tahap pendefinisian goal pada studi kasus pengukuran kualitas website sistem informasi akademik ITS. Berikut merupakan goal yang didefinisikan :

1. Meningkatkan kemudahan penggunaan perangkat lunak

2. Pengendalian kualitas informasi website
3. Meningkatkan keandalan website
4. Meningkatkan efisiensi sistem
5. Peningkatan fungsionalitas sistem dalam memenuhi kebutuhan pengguna

Pada Tugas Akhir ini dilakukan integrasi antara kelima faktor kualitas yang telah disebutkan sebelumnya dengan ISO 25010 sebagai parameter pengukuran kualitas. Gambar 2.5 merupakan faktor kualitas yang akan dilakukan pengukuran.




**Gambar 2. 5 Kerangka Kerja Evaluasi Kualitas Website Akademik**

Dalam kerangka kerja tersebut dijelaskan lima faktor kualitas yang dibutuhkan untuk mengevaluasi kualitas website sistem informasi akademik. Berikut penjelasan mengenai kelima faktor kualitas pada Gambar 2.5 :

1. Usability

   *Usability* secara umum didefinisikan sebagai karakteristik kualitas yang mengukur kemudahan pengguna dalam menggunakan sistem. Pada Tugas Akhir ini sub-kualitas pada *usability* menggunakan sub-kualitas pada karakteristik *usability* yang terdapat pada ISO 25010.

2. Content

   Karakteristik ini bukan merupakan bagian dari model ISO 25010, tetapi merupakan bagian dari penelitian model kualitas website dan sering digunakan untuk mengevaluasi website akademik. *Content* berkaitan dengan informasi yang disediakan pada website. Terdapat empat sub-kualitas pada karakteristik ini, yaitu :

   a. *Relevance of Information* mengukur sejauh mana website dapat menyediakan informasi yang relevan dan menarik bagi pengguna.
   b. *Accuracy of Information* mengukur sejauh mana website dapat memberikan informasi yang akurat yang berhubungan dengan organisasi/institusi.
   c. *Uptodate Information* mengukur sejauh mana website dapat menyampaikan informasi terbaru sesuai dengan keadaan terkini organisasi/institusi.
   d. *Identity* menilai apakah suatu website memiliki identitas organisasi pada setiap *page*. Hal ini menjadi penting untuk meyakinkan pengguna bahwa website beserta seluruh informasinya merupakan tanggung jawab organisasi.

3. Reliability

*Reability* menilai sejauh mana suatu sistem, produk, atau komponen dapat mengerjakan sebuah fungsi dalam kondisi tertentu dan dalam waktu tertentu. Pada Tugas Akhir ini sub-kualitas pada *reliability* menggunakan sub-kualitas pada karakteristik *reliability* yang terdapat pada ISO 25010.

4. Efficiency

   *Efficiency* menilai sejauh mana performa suatu sistem untuk melakukan suatu *task*. Pada Tugas Akhir ini sub-kualitas *efficiency* menggunakan sub-kualitas pada karakteristik *performance efficiency* yang terdapat pada ISO 25010.

5. Functionality

   *Functionality* menilai kemampuan sistem dalam menyediakan fitur yang sesuai dengan kebutuhan pengguna. Pada Tugas Akhir ini sub-kualitas *functionality* menggunakan sub-kualitas pada karakteristik *functional suitability* yang terdapat pada ISO 25010.

Masing-masing sub-kualitas pada kelima karakteristik tersebut didefinisikan menjadi metrik pengukuran. Metrik dapat memiliki lebih dari satu pertanyaan pembangun. Tingkat kelayakan perangkat lunak pada setiap metrik dihitung menggunakan persamaan sebagai berikut :

$$Kelayakan\ Metrik = \frac{\frac{\overline{p_1} + \overline{p_2} + \cdots \overline{p_n}}{5}}{n}$$

Keterangan :
n = Jumlah pertanyaan
$\overline{p_n}$ = Rata-rata jawaban pertanyaan ke n

## 2.8. SQL & MySQL

MySQL berkembang atas kebutuhan di awal 1990an untuk sebuah database yang cepat dan fleksibel dalam aplikasi berbasis web. MySQL adalah database open-source yang mudah untuk digunakan, stabil dan dapat diandalkan.

## 2.9. PHP

Tugas Akhir ini menggunakan PHP sebagai Bahasa pemrograman. PHP atau *Hypertext Preprocessor* merupakan sebuah bahasa *server-side scripting* yang didesain khusus untuk pengembangan Web. Di dalam sebuah halaman HTML, baris kode PHP dapat disertakan yang nantinya akan dieksekusi setiap kali halaman tersebut diakses. Kode PHP akan diinterpretasikan oleh web server dan menghasilkan HTML atau *output* lain yang nantinya akan dilihat oleh pengguna. PHP merupakan *open source product*, yang artinya siapapun dapat menggunakan, mengubah, dan mendistribusikan kembali secara gratis.

PHP disebut bahasa pemrograman server side karena PHP diproses pada komputer server. Hal ini berbeda dibandingkan dengan bahasa pemrograman client-side seperti JavaScript yang diproses pada web browser (client).

Pada awalnya PHP merupakan singkatan dari Personal Home Page. Sesuai dengan namanya, PHP digunakan untuk membuat website pribadi. Dalam beberapa tahun perkembangannya, PHP menjelma menjadi bahasa pemrograman web yang powerful dan tidak hanya digunakan untuk membuat halaman web sederhana, tetapi juga website populer yang digunakan oleh jutaan orang seperti wikipedia, wordpress, joomla, dll.

Saat ini PHP adalah singkatan dari PHP: Hypertext Preprocessor, sebuah kepanjangan rekursif, yakni permainan kata dimana kepanjangannya terdiri dari singkatan itu sendiri: PHP: Hypertext Preprocessor. PHP dapat digunakan dengan gratis (free) dan bersifat Open Source.

## 2.10. Laravel

Pada Tugas Akhir ini, aplikasi dirancang dengan arsitektur perangkat lunak Model-View-Controller. Oleh karena itu, digunakanlah Laravel sebagai *framework* PHP yang mengimplementasikan arsitektur model, view, controller atau MVC. Laravel didesain untuk mengurangi biaya dalam memulai pengembangan sebuah web dan biaya dalam *maintenance*, dan meningkatkan pengalaman dalam bekerja dengan cara menyediakan sintaks yang ekspresif dan jelas serta sekumpulan fungsionalitas-fungsionalitas inti yang dapat mepersingkat waktu implementasi. MVC(Model View Controller) adalah sebuah arsitektur yang memisahkan antara *business logic* dan antarmuka pengguna. Secara umum, arsitektur ini memisahkan sebuah aplikasi menjadi 3 bagian yaitu; Model, View and Controller.

1. Model merupakan bagian tingkat terendah yang bertanggung jawab untuk memelihara data dan bagian yang menangani hubungan antara controller dengan *database*
2. View merupakan bagian yang menangani presentation logic. View berfungsi untuk menerima dan merepresentasikan data kepada user. Bagian ini tidak memiliki akses langsung terhadap bagian model.
3. Controller merupakan bagian yang menangani kode perangkat lunak yang mengendalikan interaksi antara *model* dan *view.*

## 2.11. Ajax Jquery

*Asynchronous JavaScript and XMLHTTP*, atau disingkat AJaX, adalah suatu teknik pemrograman berbasis web untuk menciptakan aplikasi web interaktif. Tujuannya adalah untuk memindahkan sebagian besar interaksi pada komputer web surfer, melakukan pertukaran data dengan server di belakang layar, sehingga halaman web tidak harus dibaca ulang secara keseluruhan setiap kali seorang pengguna melakukan perubahan. Hal ini akan meningkatkan interaktivitas, kecepatan, dan usability.

Dalam Tugas Akhir ini, ajax digunakan pada kasus penggunaan membuat proyek pengukuran, menjawab pertanyaan serta melihat analisis tujuan. Ajax digunakan untuk mendapatkan data pertanyaan sesuai dengan goal yang terpilih pada tahap sebelumnya serta mendapatkan pemetaan GQM dari goal yang terpilih.

# BAB III
# ANALISIS DAN PERANCANGAN

Bab ini membahas analisis dan perancangan dari perangkat lunak yang dibangun pada Tugas Akhir ini. Perancangan sistem akan mencantumkan analisis kebutuhan yang diperlukan di perangkat lunak setelah mendapatkan proses bisnis. Perancangan sistem ini akan direpresentasikan dengan diagram *Unified Modelling Language* (disebut UML).

## 3.1. Analisis

Subbab ini membahas analisis kebutuhan perangkat lunak, yang meliputi analisis perancangan perangkat lunak secara garis besar, deskripsi umum, dan kebutuhan perangkat lunak.

### 3.1.1. Analisis Permasalahan

Permasalahan utama yang diangkat dalam pembuatan Tugas Akhir ini adalah bagaimana membangun sebuah aplikasi yang dapat membantu pengembang perangkat lunak untuk mengukur kualitas pada produk manajemen proyek perangkat lunak berdasarkan metode *Extended Goal Question Metric.*

Dewasa ini, semakin banyak perangkat lunak yang dikembangkan untuk memenuhi kebutuhan dalam berbagai bidang. Tentu saja hal ini menjadikan aspek kualitas perangkat lunak menjadi bagian yang penting dalam proses pengembangan. Telah banyak metode yang dikembangkan untuk menunjang pengukuran kualitas perangkat lunak, namun tak sedikit metodologi yang mengalami hambatan berupa munculnya kemungkinan terdapat bias terhadap pengukuran disebabkan luasnya lingkup identifikasi. Kondisi ini memungkinkan kurang fokusnya analisis karena dilakukannya pengukuran yang sebenarnya kurang penting.

Untuk mengatasi permasalahan tersebut, penulis akan mengimplementasikan pembuatan aplikasi pengukuran kualitas perangkat lunak dengan menggunakan metode *Extended Goal Question Metric* yang berfokus pada suatu goal tertentu. Sehingga dapat mengurangi kemungkinan terjadinya bias pada pengukuran.

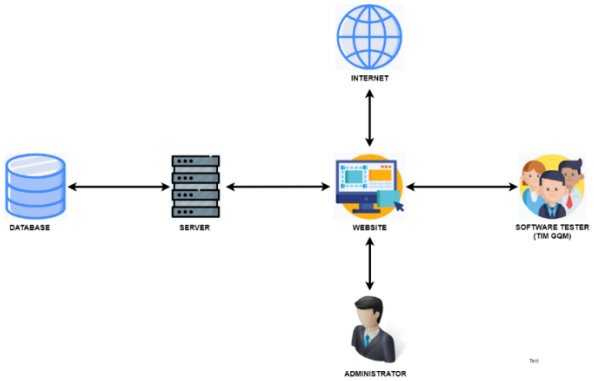
Objek pengukuran perangkat lunak dapat dilihat dari tiga jenis objek, yaitu produk, proses dan *resource.* Objek pengukuran produk dilihat dari sudut pandang produk yang dihasilkan. Penilaian ini berorientasi pada bagaimana suatu perangkat lunak dapat dikembangkan sesuai dengan yang diharapkan oleh pengguna.

### 3.1.2. Deskripsi Umum Perangkat Lunak

Pada Tugas Akhir ini akan dibuat sebuah perangkat lunak berupa aplikasi berbasis website yang bertujuan untuk memudahkan aktivitas pengukuran kualitas pada produk manajemen proyek perangkat lunak dengan metode *Extended Goal Question Metric*.

Pada Tugas Akhir ini akan dibuat sebuah perangkat lunak berupa aplikasi berbasis website yang bertujuan untuk memudahkan aktivitas pengukuran kualitas pada produk manajemen proyek perangkat lunak dengan metode *Extended Goal Question Metric*. *User* yang akan menggunakan aplikasi ini memiliki peran sebagai *Software Tester* dan Administrator.

Fitur-fitur yang ada disediakan dengan kebutuhan fungsional dari masing-masing *user* agar secara maksimal dapat menunjang aktivitas *user*. Dari sisi *Software Tester*, aplikasi akan menampilkan form pembuatan proyek pengukuran baru. *Software Tester* dapat memilih *goal* yang disajikan oleh sistem serta menjawab pertanyaan berdasarkan *goal* yang telah dipilih sebelumnya. Setelah pertanyaan terjawab, *Software Tester* dapat melihat laporan analisis *goal* maupun analisis *metric* dari proyek pengukuran serta mencetak laporan tersebut. Sedangkan dari sisi Administrator, aplikasi menyediakan fitur pengelolaan data serta pemetaan *goal*, *question* dan *metric*.




**Gambar 3. 1 Deskripsi Umum Sistem**

Gambar 3.1 menggambarkan keterkaitan antara *brainware*, *hardware*, dan *software*. Segala fitur dan informasi yang akan diakses oleh pengguna sepenuhnya dilakukan melalui website menggunakan internet. Seluruh informasi yang ditampilkan pada website disimpan pada basis data yang dihubungkan melalui bantuan *server*.

1. Admin
   Admin berperan mengelola data proyek, data tujuan, data pertanyaan, data ukuran, mengelola data pengguna dan memetakan *goal question metric* pada perangkat lunak. Penjelasan mengenai karakteristik Administrator dapat dilihat pada Subbab 3.1.4.1.

2. Software Tester
   *Software tester* berperan sebagai pengguna yang melakukan pengukuran kualitas GQM. *Software Tester* dapat melihat laporan hasil analisis pengukuran. Penjelasan mengenai karakteristik *Software Tester* dapat dilihat pada Subbab 3.1.4.1

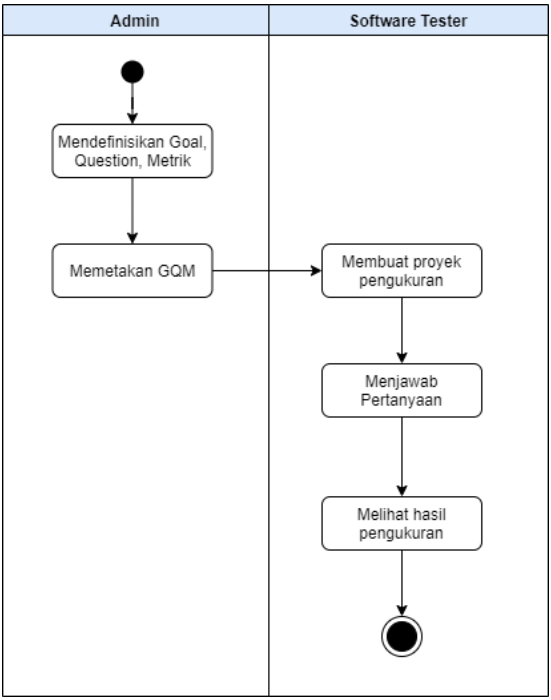
3. Database

Database digunakan untuk menyimpan goal, kuesioner, jawaban pertanyaan, data pengguna, hasil analisis *metric* dan *goal*.

4. Website
   Website digunakan sebagai kakas bantu pengukuran dan menampilkan hasil uji perangkat lunak.

### 3.1.3. Alur Penggunaan Aplikasi

Kakas ini dioperasikan oleh admin dan software tester yang merupakan tim GQM. Admin dapat mengelola data pengguna dan mengelola data project. Sedangkan *software tester* dapat mengelola tujuan, mengelola pertanyaan, memasukkan hasil kuesioner, melihat analisis hasil ukur, dan melihat analisis tujuan. Input dari kakas ini adalah *goal* dan pertanyaan yang didefinisikan oleh tim GQM kemudian setelah diproses oleh kakas akan menghasilkan laporan tentang hasil pengukuran kualitas perangkat lunak yang diukur. Alur penggunaan aplikasi dapat dilihat pada Gambar 3.2.




**Gambar 3. 2 Alur Penggunaan Aplikasi**

### 3.1.4. Spesifikasi Kebutuhan Perangkat Lunak

Subbab ini membahas spesifikasi kebutuhan perangkat lunak dari hasil analisis yang telah dilakukan. Bagian ini berisi kebutuhan perangkat lunak yang direpresentasikan dalam bentuk kebutuhan fungsional, kebutuhan non-fungsional, diagram kasus penggunaan, dan diagram aktivitas.

#### 3.1.4.1. Karakteristik Pengguna

Pengguna/aktor adalah pihak-pihak yang terlibat dan berinteraksi langsung dengan sistem. Karakteristik pengguna yang berada pada sistem ini dibahas lebih rinci pada Tabel 3.1.

**Tabel 3. 1 Spesifikasi Karakteristik Pengguna pada Sistem**

| No | Kategori Pengguna | Tugas | Hak Akses |
|---|---|---|---|
| 1 | Administrator | • Mengelola data<br>• Memetakan Goal Question Metric | • Mengelola (melihat, menambah, mengubah, menghapus) data tujuan<br>• Mengelola (melihat, menambah, mengubah, menghapus) data pertanyaan<br>• Mengelola (melihat, menambah, mengubah, menghapus) data ukuran<br>• Mengelola (melihat, menambah, mengubah, menghapus) data pengguna<br>• Mengelola (melihat, menambah, mengubah, menghapus) data proyek.<br>• Memetakan Goal, Question dan Metric |
| 2 | Software Tester | • Melihat analisis hasil ukur<br>• Membuat proyek pengukuran<br>• Mencetak laporan pengukuran | • Melihat analisis hasil pengukuran tiap metric beserta detailnya<br>• Melihat analisis hasil pengukuran goal beserta detailnya<br>• Membuat proyek pengukuran sesuai goal yang diprioritaskan/dipilih<br>• Menjawab pertanyaan dari goal yang terpilih<br>• Mengunduh dalam bentuk pdf dan mencetak laporan pengukuran |

Administrator yang dimaksud dalam sistem ini merupakan super admin, yang merupakan pemilik dari website pengukuran yang

di bangun, dalam hal ini merupakan pihak DPTSI. Dikarenakan di DPTSI belum ada divisi khusus yang menangani pengukuran perangkat lunak, maka administrator dalam sistem ini merupakan pegawai DPTSI bagian pengembangan perangkat lunak. Sedangkan *Software Tester* merupakan pihak umum yang dapat menggunakan aplikasi pengukuran. *Software Tester* dapat merupakan admin dari suatu perusahaan/instansi yang hendak mengukur kualitas perangkat lunaknya dengan menggunakan sistem yang di bangun pada Tugas Akhir ini.

### 3.1.4.2. Kebutuhan Fungsional

Kebutuhan fungsional mendefinisikan layanan yang harus dimiliki oleh perangkat lunak, reaksi terhadap suatu masukan, hasil yang dilakukan pada situasi khusus. Kebutuhan fungsional dari perangkat lunak untuk aplikasi ini dijelaskan pada Tabel 3.2

**Tabel 3. 2 Kebutuhan Fungsional Sistem**

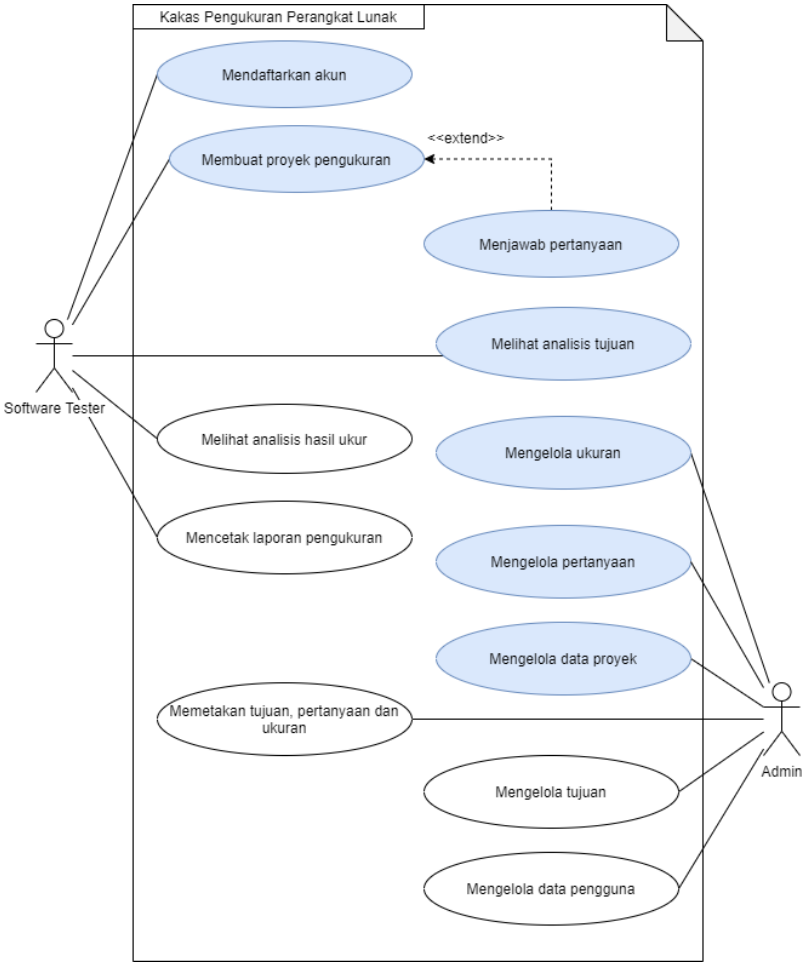
| Kode | Kebutuhan Fungsional |
|---|---|
| **F-001** | Software Tester dapat mendaftarkan akun |
| **F-002** | Software Tester dapat membuat proyek pengukuran |
| **F-003** | Software Tester dapat menjawab pertanyaan |
| **F-004** | Software Tester dapat melihat analisis tujuan |
| **F-005** | Software Tester dapat melihat analisis hasil ukur |
| **F-006** | Software Tester dapat mencetak laporan pengukuran |
| **F-007** | Admin dapat mengelola ukuran |
| **F-008** | Admin dapat mengelola pertanyaan |
| **F-009** | Admin dapat mengelola data proyek |
| **F-010** | Admin dapat memetakan tujuan, pertanyaan dan ukuran |
| **F-011** | Admin dapat mengelola tujuan |
| **F-012** | Admin dapat mengelola data pengguna |

### 3.1.4.3. Deskripsi Fungsional

Kebutuhan fungsional sistem akan dijelaskan secara rinci pada subbab ini. Deskripsi fungsional meliputi diagram kasus penggunaan, spesifikasi kasus penggunaan, serta diagram aktivitas kasus penggunaan.

#### 3.1.4.3.1 Diagram Kasus Penggunaan

Pada subbab ini, kasus penggunaan perangkat lunak akan dijelaskan secara rinci. Kasus penggunaan didasarkan pada hasil analisis kebutuhan fungsional dari perangkat lunak. Kasus penggunaan aplikasi secara umum dapat dilihat pada Gambar 3.3.




**Gambar 3. 3 Diagram Kasus Penggunaan Sistem Keseluruhan**

Berikut merupakan deskripsi dari kasus penggunaan kakas bantu pengukuran kualitas perangkat lunak yang akan dibahas pada Tugas Akhir ini :

1. Mendaftarkan akun

Software Tester harus mendaftar sebagai pengguna untuk dapat menggunakan aplikasi pengukuran. Mendaftarkan akun hanya dilakukan sekali ketika Tester belum memiliki akun yang terdaftar pada sistem.

2. Membuat proyek pengukuran

   Software Tester dapat membuat proyek pengukuran dengan mengisi form proyek terlebih dulu memilih goal yang akan diukur.

3. Menjawab pertanyaan

   Software Tester dapat menjawab pertanyaan yang telah dipilih ketika membuat proyek pengukuran.

4. Melihat analisis tujuan

   Software Tester dapat melihat resume pencapaian sebuah tujuan yang ditinjau dari hasil analisis metrik yang terkait dengan tujuan tersebut.

5. Mengelola data proyek

   Admin dapat mengelola (menambah, mengedit dan menghapus) data proyek dalam pengukuran perangkat lunak.

6. Mengelola pertanyaan

   Admin dapat menambah, mengedit, menghapus pertanyaan yang digunakan dalam pengukuran perangkat lunak.

7. Mengelola ukuran

   Admin dapat menambah, mengedit, menghapus ukuran (*metric*) yang digunakan dalam pengukuran perangkat lunak.

#### 3.1.4.3.2 F001 : Mendaftarkan Akun

##### 3.1.4.3.1.1 Kasus Penggunaan Mendaftarkan Akun

Kasus penggunaan mendaftarkan akun memungkinkan pengguna (Software Tester) untuk membuat akun yang akan digunakan untuk memulai melakukan pengukuran menggunakan

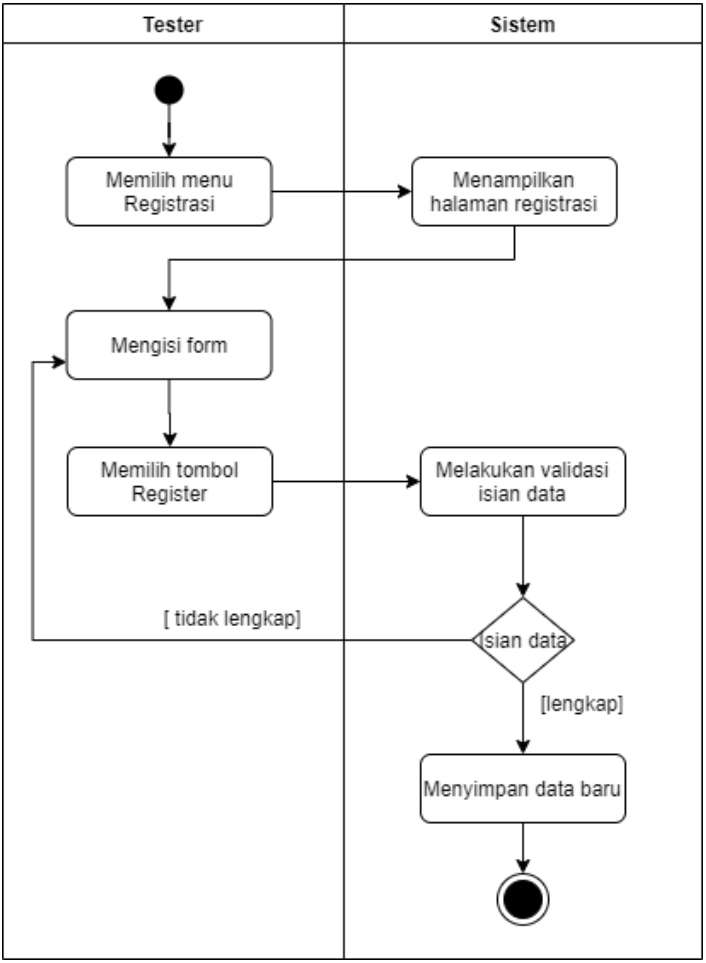
aplikasi. Spesifikasi kasus penggunaan mendaftarkan akun dapat dilihat pada Tabel 3.3.

**Tabel 3. 3 Spesifikasi Kasus Penggunaan Mendaftarkan Akun**

| | |
|---|---|
| **Nomor** | UC001 |
| **Nama** | Mendaftarkan akun |
| **Skenario** | Mendaftarkan akun baru |
| **Use Case Terkait** | - |
| **Deskripsi** | Pengguna dapat mendaftarkan diri sebagai Software Tester untuk menggunakan aplikasi |
| **Aktor** | Software Tester/Tamu |
| **Kondisi Awal** | Pengguna belum terdaftar sebagai Software Tester pada sistem. |
| **Kondisi Akhir** | Pengguna terdaftar sebagai Software Tester pada sistem. |
| **Alur Normal** | |
| Aktor | Sistem |
| 1. Pengguna memilih menu ‘registrasi’ | 1.1 Sistem menampilkan halaman registrasi beserta form pendaftaran |
| 2. Pengguna mengisi form pendaftaran | 4.1 Sistem melakukan validasi isian data |
| 3. Pengguna memilih tombol ‘register’ | 3.1 Sistem menyimpan data pengguna baru |
| **Alur Alternatif** | |
| - | |

### 3.1.4.3.1.2 Diagram Aktivitas :Mendaftarkan Akun

Berdasarkan scenario kasus penggunaan pada Tabel 3.3 selanjutnya scenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan mendaftarkan akun dapat dilihat pada Gambar 3.4.




**Gambar 3. 4 Diagram aktivitas UC001**

#### 3.1.4.3.3 F002 : Membuat Proyek Pengukuran

##### 3.1.4.3.1.3 Kasus Penggunaan Membuat Proyek Pengukuran

Kasus penggunaan membuat proyek pengukuran memungkinkan Software Tester untuk membuat proyek pengukuran baru. Spesifikasi kasus penggunaan membuat proyek pengukuran dapat dilihat pada Tabel 3.4.

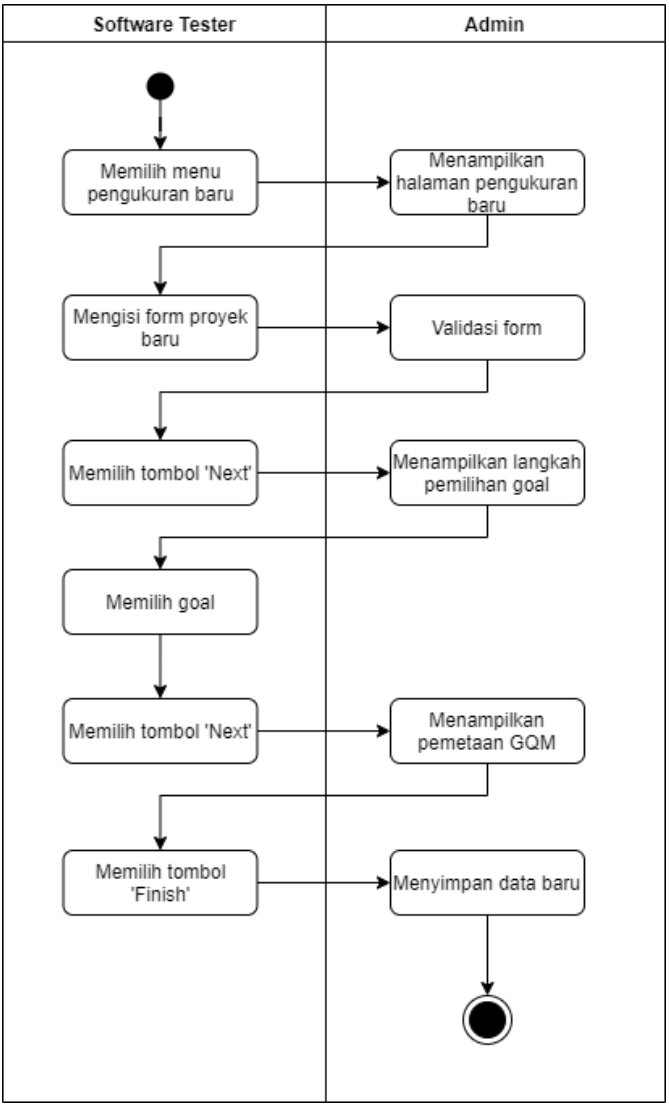
**Tabel 3. 4 Spesifikasi Kasus Penggunaan Membuat Proyek Pengukuran**

| | |
|---|---|
| **Nomor** | UC002 |
| **Nama** | Membuat proyek pengukuran |
| **Skenario** | Membuat proyek pengukuran baru |
| **Use Case Terkait** | Menjawab Pertanyaan |
| **Deskripsi** | Software Tester dapat membuat proyek pengukuran baru |
| **Aktor** | Software Tester |
| **Kondisi Awal** | Data proyek pengukuran belum tersimpan |
| **Kondisi Akhir** | Data proyek pengukuran tersimpan dalam sistem |
| **Alur Normal** | |
| Aktor | Sistem |
| 1. Tester memilih menu ‘Pengukuran Baru’ | 8.1 Sistem menampilkan halaman ‘Pengukuran Baru’ |
| 2. Mengisi form proyek baru | 2.1 Sistem melakukan validasi form |
| 3. Memilih tombol ‘Next’ | 3.1 Sistem menampilkan halaman memilih goal |
| 4. Memilih goal | |
| 5. Memilih tombol ‘Next’ | 5.1 Sistem menampilkan halaman pemetaan goal-question-metric |
| 6. Memilih tombol ‘Finish’ | |

| | |
|---|---|
| | 6.1 Sistem menyimpan data proyek baru |
| **Alur Alternatif** | |
| - | |

#### 3.1.4.3.1.4 Diagram Aktivitas : Membuat Proyek Pengukuran

Berdasarkan skenario kasus penggunaan pada Tabel 3.4 selanjutnya skenario tersebut gambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan membuat proyek pengukuran dapat dilihat pada Gambar 3.5.




**Gambar 3. 5 Diagram aktivitas UC002**

#### 3.1.4.3.4 F003 : Menjawab Pertanyaan

##### 3.1.4.3.1.5 Kasus Penggunaan Menjawab Pertanyaan

Kasus penggunaan menjawab pertanyaan memungkinkan Software Tester untuk memasukkan jawaban pertanyaan yang disediakan pada sistem. Spesifikasi kasus penggunaan menjawab pertanyaan dapat dilihat pada Tabel 3.4.

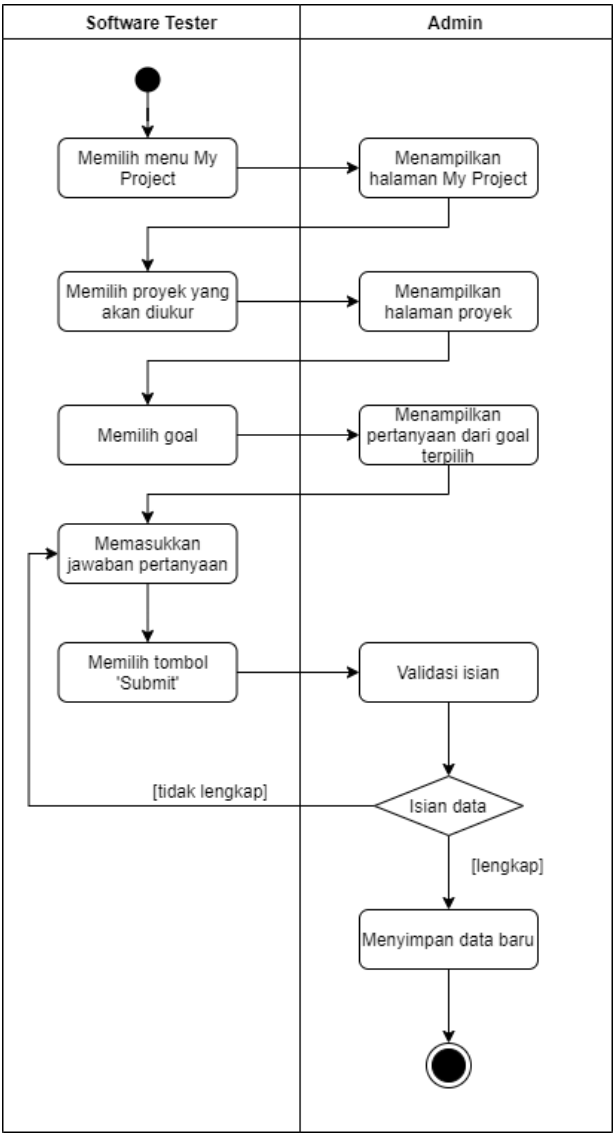
**Tabel 3. 5 Spesifikasi Kasus Penggunaan Menjawab Pertanyaan**

| | |
|---|---|
| **Nomor** | UC003 |
| **Nama** | Menjawab pertanyaan |
| **Skenario** | Menjawab pertanyaan sesuai goal |
| **Use Case Terkait** | Membuat proyek pengukuran |
| **Deskripsi** | Software Tester dapat menjawab pertanyaan yang disediakan sistem sesuai dengan goal yang dipilih |
| **Aktor** | Software Tester |
| **Kondisi Awal** | • Data jawaban belum tersimpan pada sistem<br>• Data proyek sudah tersimpan pada sistem |
| **Kondisi Akhir** | Data jawaban ditambahkan dan tersimpan pada sistem |
| **Alur Normal** | |
| Aktor | Sistem |
| 1. Tester memilih menu ‘My Project | 8.2 Sistem menampilkan halaman ‘My Project’ |
| 2. Tester memilih project yang akan diukur | 3.1 Sistem menampilkan halaman project |
| 3. Tester memilih goal | 3.1 Sistem menampilkan pertanyaan dari goal yang terpilih |
| 4. Tester menjawab pertanyaan | |

| | |
|---|---|
| 5. Memilih tombol ‘Submit’ | 4.1 Sistem memvalidasi jawaban<br>5.1 Sistem menyimpan data jawaban |
| **Alur Alternatif** | |
| - | |

#### 3.1.4.3.1.6 Diagram Aktivitas : Menjawab Pertanyaan

Berdasarkan skenario kasus penggunaan pada Tabel 3.4 selanjutnya skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan menjawab dapat dilihat pada Gambar 3.6.




**Gambar 3. 6 Diagram aktivitas UC003**

#### 3.1.4.3.5 F004 : Melihat Analisis Tujuan

##### 3.1.4.3.1.7 Kasus Penggunaan Melihat Analisis Tujuan

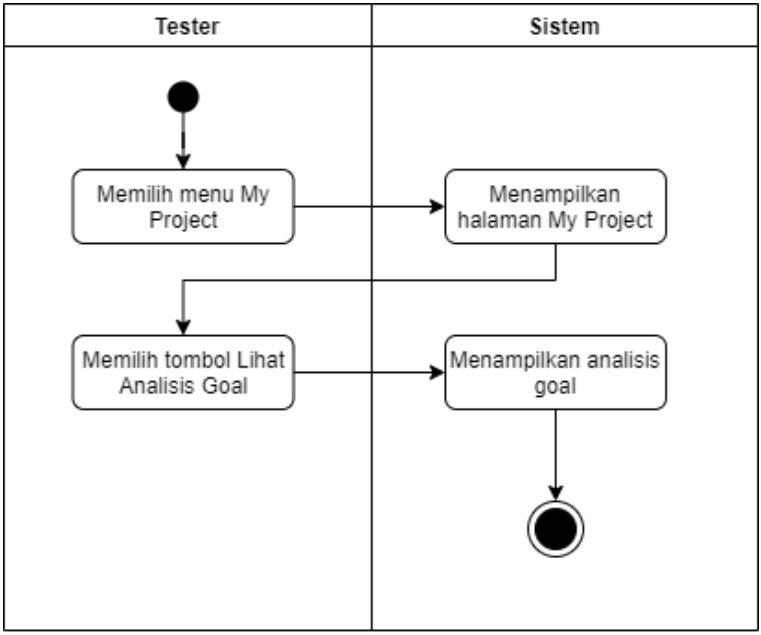
Kasus penggunaan melihat analisis tujuan memungkinkan Software Tester untuk melihat analisis dari tujuan (goal) yang telah dipilih pada proyek yang diukur. Spesifikasi kasus penggunaan melihat analisis tujuan dapat dilihat pada Tabel 3.6.

**Tabel 3. 6 Spesifikasi Kasus Penggunaan Melihat Analisis Tujuan**

| | |
|---|---|
| **Nomor** | UC004 |
| **Nama** | Melihat analisis tujuan |
| **Skenario** | Melihat analisis tujuan pada pengukuran proyek |
| **Use Case Terkait** | - |
| **Deskripsi** | Software Tester dapat melihat analisis tujuan yang telah dipilih pada pengukuran proyek |
| **Aktor** | Software Tester |
| **Kondisi Awal** | Data analisis tujuan telah tersimpan pada sistem |
| **Kondisi Akhir** | Software Tester dapat melihat hasil analisis tujuan |
| **Alur Normal** | |
| Aktor | Sistem |
| 1.1.1 Tester memilih menu ‘My Project’<br>1.1.2 Tester memilih data proyek yang akan dilihat hasil analisis tujuan proyek | 3.2 Sistem menampilkan halaman ‘My Project’<br>3.3 Sistem menampilkan hasil analisis tujuan proyek yang dipilih |
| **Alur Alternatif** | |
| - | |

##### 3.1.4.3.1.8 Diagram Aktivitas : Melihat Analisis Tujuan

Berdasarkan skenario kasus penggunaan pada Tabel 3.6 selanjutnya skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan mendaftarkan akun dapat dilihat pada Gambar 3.7.




**Gambar 3. 7 Diagram aktivitas UC004**

### 3.1.4.3.6 F007 : Mengelola Ukuran

#### 3.1.4.3.1.9 Kasus Penggunaan Mengelola Ukuran

Kasus penggunaan mengelola ukuran memungkinkan Admin untuk mengelola data ukuran (metric), termasuk menambah, mengubah dan menghapus data. Spesifikasi kasus penggunaan mengelola ukuran dapat dilihat pada Tabel 3.7,

**Tabel 3. 7 Spesifikasi Kasus Penggunaan Mengelola Ukuran**

| | |
|---|---|
| **Nomor** | UC007 |
| **Nama** | Mengelola ukuran |
| **Skenario** | Mengelola data ukuran |
| **Trigger** | Admin memilih menu ‘Metric’ |
| **Deskripsi** | Admin dapat mengelola data ukuran dengan menambah, mengubah dan menghapus data ukuran. |
| **Aktor** | Admin |
| **Kondisi Awal** | Data ukuran belum tersimpan atau sudah tersimpan pada sistem |
| **Kondisi Akhir** | Data ukuran berhasil ditambahkan, diubah atau dihapus dari sistem |

| **Alur Normal** | |
|---|---|
| Aktor | Sistem |
| .1.1.1 Admin memilih menu ‘Metric’ | 1.1 Sistem menampilkan halaman ‘Metric’ |
| 3.4 Admin memilih tombol ‘Tambah Metric’ | 1.1.1 Sistem menampilkan form untuk menambah data ukur |
| 3.4.1 Admin mengisi form | 1.1.1.1 Sistem melakukan validasi data isian |
| 3.4.2 Admin memilih tombol ‘sumbit’ | 1.1.2.1 Sistem menyimpan data ukur |
| 3.5 Admin memilih tombol‘Edit’ pada salah satu data ukuran | 1.2.1 Sistem menampilkan halaman untuk mengubah data ukur |
| 3.5.1 Admin mengisi form | 1.2.1.1 Sistem melakukan validasi data isian |
| 3.5.2 Admin memilih tombol ‘sumbit’ | 1.2.2.1 Sistem menyimpan data ukur yang telah diubah |
| 3.6 Admin memilih tombol ‘Hapus’ pada salah satu data ukuran | 1.3.1 Sistem menampilkan modal konfirmasi penghapusan data yang dipilih |

| | |
|---|---|
| 3.6.1 Admin mengkonfirmasi penghapusan data | 1.3.1.1 Data ukur berhasil dihapus dari sistem |
| **Alur Alternatif** | |
| 1.1.1 Jika isian data pada formtidak lengkap, sistem menampilkan pesan error bahwa terdapat isian data yang belum lengkap, lalu kembali ke langkah 1.1.1<br>1.2.2 Jika admin memilih tombol ‘Cancel’, maka perubahan data tidak tersimpan pada sistem<br>1.3.1 Jika admin memilih tombol ‘Batal’, maka data tidak terhapus dari sistem | |

#### 3.1.4.3.1.10 Diagram Aktivitas : Mengelola Ukuran

Berdasarkan skenario kasus penggunaan pada Tabel 3.8 selanjutnya skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan mengelola ukuran dapat dilihat pada Gambar 3.8, Gambar 3.9 dan Gambar 3.10.

### 3.1.4.3.7 F008 : Mengelola Pertanyaan

#### 3.1.4.3.1.11 Kasus Penggunaan Mengelola Pertanyaan

Kasus penggunaan mengelola pertanyaan memungkinkan Admin untuk mengelola pertanyaan (question), termasuk menambah, mengubah dan menghapus data pertanyaan. Spesifikasi kasus penggunaan mengelola pertanyaan dapat dilihat pada Tabel 3.8.

**Tabel 3. 8 Spesifikasi Kasus Penggunaan Mengelola Pertanyaan**

| **Nomor** | UC008 |
|---|---|
| **Nama** | Mengelola pertanyaan |
| **Skenario** | Mengelola data pertanyaan |
| **Trigger** | Admin memilih menu ‘Question’ |
| **Deskripsi** | Admin dapat mengelola data pertanyaan dengan menambah, mengubah dan menghapus data pertanyaan. |
| **Aktor** | Admin |
| **Kondisi Awal** | Data pertanyaan belum tersimpan atau sudah tersimpan pada sistem |
| **Kondisi Akhir** | Data pertanyaan berhasil ditambahkan, diubah atau dihapus dari sistem |
| **Alur Normal** | |
| Aktor | Sistem |
| 1. Admin memilih menu ‘Question’<br>1.1 Admin memilih tombol ‘Tambah Pertanyaan’<br>1.1.1 Admin mengisi form<br>1.1.2 Admin memilih tombol ‘sumbit’<br>1.2 Admin memilih tombol ‘Edit’ pada salah satu data pertanyaan<br>1.2.1 Admin mengisi form<br>1.2.2 Admin memilih tombol ‘sumbit’ | 1.1 Sistem menampilkan halaman ‘Question’<br>1.1.1 Sistem menampilkan form untuk menambah data pertanyaan<br>1.1.1.1 Sistem melakukan validasi data isian<br>1.1.2.1 Sistem menyimpan data pertanyaan<br>1.2.1 Sistem menampilkan halaman untuk mengubah data pertanyaan<br>1.2.1.1 Sistem melakukan validasi data isian<br>1.2.2.1 Sistem menyimpan data pertanyaan yang telah diubah |

| | |
|---|---|
| 1.3 Admin memilih tombol 'Hapus' pada salah satu data pertanyaan<br><br>1.3.1 Admin mengkonfirmasi penghapusan data | 1.3.1 Sistem menampilkan modal konfirmasi penghapusan data yang dipilih<br><br>1.3.1.1 Data pertanyaan berhasil dihapus dari sistem |
| **Alur Alternatif** | |
| 1.1.1 Jika isian data pada form tidak lengkap, sistem menampilkan pesan error bahwa terdapat isian data yang belum lengkap, lalu kembali ke langkah 1.1.1<br>1.2.2 Jika admin memilih tombol 'Cancel', maka perubahan data tidak tersimpan pada sistem<br>1.3.1 Jika admin memilih tombol 'Batal', maka data tidak terhapus dari sistem | |

#### 3.1.4.3.1.12 Diagram Aktivitas : Mengelola Pertanyaan

Berdasarkan skenario kasus penggunaan pada Tabel 3.10 selanjutnya skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan mengelola pertanyaan dapat dilihat pada Gambar 3.11, Gambar 3.12 dan Gambar 3.13.

### 3.1.4.3.8 F0009 : Mengelola Data Proyek

#### 3.1.4.3.1.13 Kasus Penggunaan Mengelola Data Proyek

Kasus penggunaan mengelola data proyek memungkinkan Admin untuk mengelola data proyek, termasuk menambah, mengubah dan menghapus Spesifikasi kasus penggunaan mengelola data proyek dapat dilihat pada Tabel 3.9.

**Tabel 3. 9 Spesifikasi Kasus Penggunaan Mengelola Data Proyek**

| **Nomor** | UC0009 |
|---|---|
| **Nama** | Mengelola data proyek |
| **Skenario** | Mengelola data proyek pengguna pada sistem |
| **Trigger** | Admin memilih menu ‘Project’ |
| **Deskripsi** | Admin dapat mengelola data proyek dengan menambah, mengubah dan menghapus data proyek. |
| **Aktor** | Admin |
| **Kondisi Awal** | Data proyek belum tersimpan atau sudah tersimpan pada sistem |
| **Kondisi Akhir** | Data proyek berhasil ditambahkan, diubah atau dihapus dari sistem |
| **Alur Normal** | |
| Aktor | Sistem |
| 1. Admin memilih menu ‘Project’ | 1.1 Sistem menampilkan halaman ‘Goal’ |
| 1.1 Admin memilih tombol ‘Tambah Proyek’ | 1.1.1 Sistem menampilkan form untuk menambah data proyek |
| 1.1.1 Admin mengisi form | 1.1.1.1 Sistem melakukan validasi data isian |
| 1.1.2 Admin memilih tombol ‘sumbit’ | 1.1.2.1 Sistem menyimpan data proyek |
| 1.2 Admin memilih tombol ‘Edit’ pada salah satu data proyek | 1.2.1 Sistem menampilkan halaman untuk mengubah data proyek |
| 1.2.1 Admin mengisi form | 1.2.1.1 Sistem melakukan validasi data isian |
| 1.2.2 Admin memilih tombol ‘sumbit’ | 1.2.2.1 Sistem menyimpan data proyek yang telah diubah |

| | |
|---|---|
| 1.3 Admin memilih tombol ‘Hapus’ pada salah satu data proyek<br><br>1.3.1 Admin mengkonfirmasi penghapusan data | 1.3.1 Sistem menampilkan modal konfirmasi penghapusan data yang dipilih<br><br>1.3.1.1 Data proyek berhasil dihapus dari sistem |
| **Alur Alternatif** | |
| 1.1.1 Jika isian data pada form tidak lengkap, sistem menampilkan pesan error bahwa terdapat isian data yang belum lengkap, lalu kembali ke langkah 1.1.1 | |
| 1.2.2 Jika admin memilih tombol ‘Cancel’, maka perubahan data tidak tersimpan pada sistem | |
| 1.3.1 Jika admin memilih tombol ‘Batal’, maka data tidak terhapus dari sistem | |

#### 3.1.4.3.1.14 Diagram Aktivitas : Mengelola Data Proyek

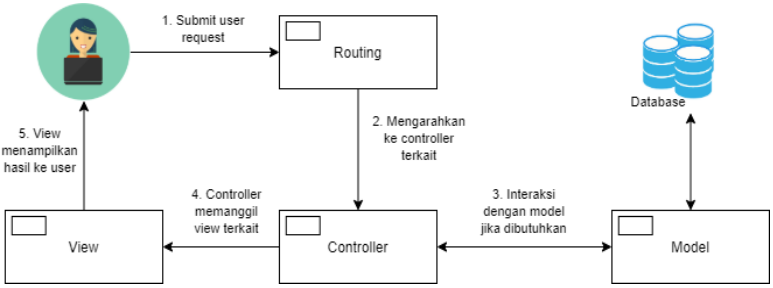
Berdasarkan skenario kasus penggunaan pada Tabel 3.12 selanjutnya skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan mengelola data proyek dapat dilihat pada Gambar 3.14, Gambar 3.15 dan Gambar 3.16.

## 3.2. Perancangan

Pada subbab perancangan akan dijelaskan mengenai arsitektur sistem yang digunakan, perancangan diagram kelas, perancangan basis data, dan perancangan antarmuka.

### 3.2.1. Perancangan Arsitektur Perangkat Lunak

Arsitektur sistem yang digunakan pada platform web pada perangkat lunak ini menggunakan kerangka kerja (*framework*) Laravel dengan pola perancangan *Model-View-Controller* (MVC). Ilustrasi arsitektur sistem dapat dilihat pada Gambar 3.17.

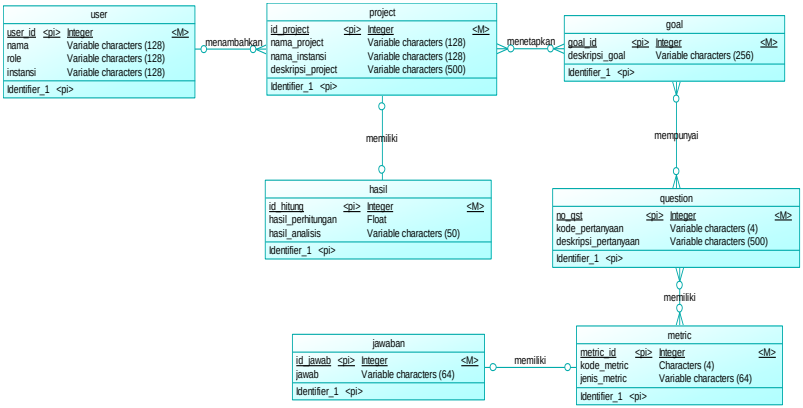



**Gambar 3. 8 Ilustrasi Arsitektur Sistem Web**

Ketika client melakukan request data, Routing akan menerima request tersebut dan mengarahkan request tersebut kepada controller yang terkait dengan request yang diberikan. Controller akan berinteraksi dengan model yang dibutuhkan, dan model akan melakukan query terhadap database. Model mengembalikan data yang dibutuhkan kepada controller. Kemudian controller memanggil view terkait, untuk menampilkan hasil yang diminta oleh user melalui browser.

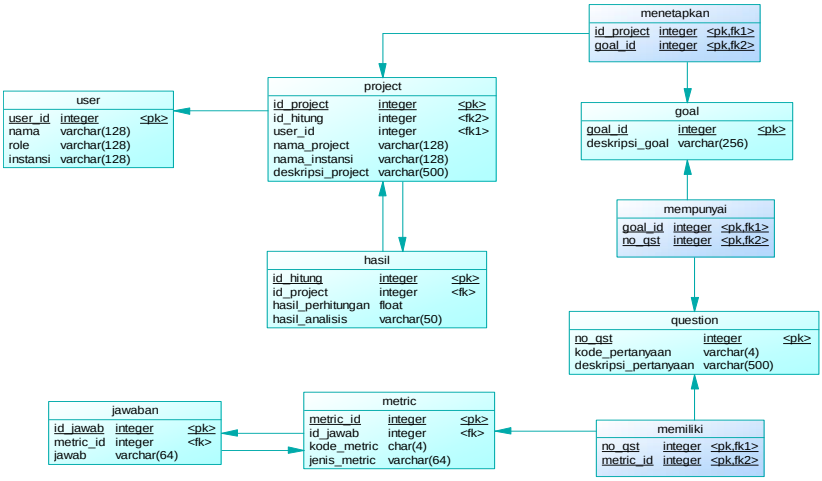
### 3.2.2. Perancangan Basis Data

Dalam membuat suatu aplikasi berupa sistem informasi, diperlukan sebuah analisis berupa perancangan basis data. MySQL dipilih menjadi aplikasi basis data karena penggunaannya yang mudah.




**Gambar 3. 9 CDM Sistem**

Rancangan basis data ditampilkan dalam bentuk *Conceptual Data Model* (CDM) dan *Physical Data Model* (PDM). Secara umum, struktur CDM dan PDM dapat dilihat pada **Error! Reference source n ot found.** 3.18 dan Gambar 3.19.




**Gambar 3. 10 PDM Sistem**

#### 3.2.2.1. Tabel User

Tabel user digunakan untuk menyimpan data pengguna yang akan melakukan pengukuran kualitas perangkat lunak. Atribut yang terdapat dalam entitas ini adalah user_id sebagai primary key, nama, role, dan instansi. Deskripsi rinci tabel user dapat dilihat pada Tabel 3.1.

**Tabel 3.1 Deskripsi Rinci Tabel User**

| Id Field | Deskripsi | Tipe & length | Keterangan |
|---|---|---|---|
| user_id | Variabel untuk ID user. | Integer | Primary Key |
| nama | Variabel untuk nama user. | Varchar(128) | |
| role | Variabel untuk peran/hak akses user. | Varchar(128) | |
| instansi | Variabel untuk instansi user. | Varchar(128) | |

#### 3.2.2.2. Tabel Project

Tabel project digunakan untuk menyimpan data proyek atau perangkat lunak yang akan diuji. Atribut yang terdapat dalam entitas ini adalah id_project sebagai primary key, nama projek, dan nama instansi. Deskripsi rinci tabel project dapat dilihat pada Tabel 3.2.

**Tabel 3.2 Deskripsi Rinci Tabel Project**

| Id Field | Deskripsi | Tipe & length | Keterangan |
|---|---|---|---|
| id_project | Variabel untuk ID proyek | Integer | Primary Key |

| Id Field | Deskripsi | Tipe & length | Keterangan |
|---|---|---|---|
| id_hitung | Variabel utuk ID hasil perhitungan | Integer | Foreign Key |
| user_id | Variabel untuk ID user | Integer | Foreign Key |
| nama_project | Variabel untuk menyimpan nama proyek | Varchar(128) | |
| nama_instansi | Variabel untuk menyimpan nama instansi proyek | Varchar(128) | |

### 3.2.2.3. Tabel Goal

Tabel goal digunakan untuk menyimpan data goal yang telah dibuat. Atribut yang terdapat dalam entitas ini adalah id sebagai primary key dan deskripsi goal. Deskripsi rinci tabel goal dapat dilihat pada Tabel 3.3.

**Tabel 3.3 Deskripsi Rinci Tabel Goal**

| Id Field | Deskripsi | Tipe & length | Keterangan |
|---|---|---|---|
| goal_id | Variabel untuk ID goal/tujuan | Integer | Primary Key |
| id_project | Variabel untuk ID proyek | Integer | Foreign Key |
| deskripsi_goal | Variabel untuk menyimpan deskripsi tujuan | Varchar(255) | |

### 3.2.2.4. Tabel Question

Tabel question digunakan untuk menyimpan data pertanyaan yang telah dibuat. Atribut yang terdapat dalam entitas ini adalah no_qst sebagai primary key, kode pertanyaan, dan deskripsi pertanyaan. Deskripsi rinci tabel question dapat dilihat pada Tabel 3.4.

**Tabel 3.4 Deskripsi Rinci Tabel Question (font kecilkan di dalam tabel)**

| Id Field | Deskripsi | Tipe & length | Keterangan |
|---|---|---|---|
| no_qst | Variabel untuk ID question/pertanyaan | Integer | Primary Key |
| kode_pertanyaan | Variabel untuk menyimpan kode pertanyaan | Varchar(4) | |
| deskripsi_pertanyaan | Variabel untuk menyimpan deksripsi pertanyaan | Varchar(500) | |

### 3.2.2.5. Tabel Metric

Tabel metric digunakan untuk menyimpan data metric. Atribut yang terdapat pada entitas ini adalah metric_id sebagai primary key, kode metric dan jenis metric. Deskripsi rinci tabel metric dapat dilihat pada Tabel 3.5.

**Tabel 3.5 Deskripsi Rinci Tabel Metric**

| Id Field | Deskripsi | Tipe & length | Keterangan |
|---|---|---|---|
| metric_id | Variabel untuk ID metric/ukur | Integer | Primary Key |
| id_jawab | Variabel untuk ID jawaban | Integer | Foreign Key |
| kode_metric | Variabel untuk menyimpan kode metric | Char(4) | |
| jenis_metric | Variabel untuk menyimpan nama jenis metric | Varchar(64) | |

### 3.2.2.6. Tabel Jawaban

Tabel jawaban digunakan untuk menyimpan jawaban dari pertanyaan yang ada, Atribut yang terdapat pada entitas ini adalah id_jawab sebagai primary key dan jawab. Deskripsi rinci tabel jawaban dapat dilihat pada Tabel 3.5.

**Tabel 3.6 Deskripsi Rinci Tabel Jawaban**

| Id Field | Deskripsi | Tipe & length | Keterangan |
|---|---|---|---|
| id_jawab | Variabel untuk ID jawaban | Integer | Primary Key |
| metric_id | Variabel untuk ID metric/ukur | Integer | Foreign Key |
| jawab | Variabel untuk menyimpan jawaban dari pertanyaan | Varchar(64) | |

### 3.2.2.7. Tabel Hasil

Tabel hasil digunakan untuk menyimpan data hasil pengukuran. Atribut yang terdapat pada entitas analisis metric adalah id_hitung sebagai primary key, hasil perhitungan dan hasil analisis. Deskripsi rinci tabel hasil dapat dilihat pada Tabel 3.5.

**Tabel 3.7 Deskripsi Rinci Tabel Metric**

| Id Field | Deskripsi | Tipe & length | Keterangan |
|---|---|---|---|
| id_hitung | Variabel untuk ID hasil perhitungan | Integer | Primary Key |
| id_project | Variabel untuk ID proyek | Integer | Foreign Key |
| hasil_perhitungan | Variabel untuk menyimpan hasil perhitunan | Float | |
| hasil_analisis | Variabel untuk menyimpan hasil analisis | Varchar(50) | |

### 3.2.3. Perancangan Antarmuka Pengguna

Pada bagian ini dijelaskan mengenai rancangan tampilan antarmuka pengguna dari sistem. Antarmuka pengguna yang berhubungan langsung dengan aktor, harus memiliki tampilan yang

menarik bagi penggunanya. Aplikasi ini menggunakan antarmuka yang indah dilihat. Pengguna dapat mengakses menggunakan semua *web* browser dan hanya menggunakan keyboard dan mouse. Aplikasi ini memiliki dua hak akses yaitu hak akses untuk Administrator dan Software Tester. Berikut akan dijelaskan masing rancangan antarmuka.

### 3.2.3.1. Halaman Beranda

Halaman beranda merupakan halaman utama ketika aplikasi web dibuka. Pada halaman ini terdapat menu registrasi dan login. Tester yang belum melakukan pendaftaran akun tidak dapat melakukan pengukuran.

Antarmuka beranda dan penjelasan terkait antarmuka beranda dapat dilihat pada Gambar 3.20 dan Tabel 3.10.

**Gambar 3. 11 Rancangan Antarmuka Beranda**

**Tabel 3. 10 Penjelasan Antarmuka Beranda**

| No | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Jenis Masukan/ Keluaran |
|---|---|---|---|---|
| **1** | *txtTitle* | *Text* | *Field* untuk menampilkan nama aplikasi | *String* |
| **2** | *txtDeskripsi* | *Text* | *Field* untuk menampilkan deskripsi aplikasi | *String* |

### 3.2.3.2. Halaman Mendaftarkan Akun

Antarmuka registrasi/mendaftarkan akun akan digunakan pada kasus penggunaan UC001. Halaman melihat antarmuka registrasi menampilkan form pendaftaran akun.

Antarmuka registrasi/mendaftarkan akun dan penjelasan terkait antarmuka registrasi/mendaftarkan akun dapat dilihat pada Gambar 3. 21 dan Tabel 3.11.

**Gambar 3. 12 Rancangan Antarmuka Registrasi**

**Tabel 3. 11 Penjelasan Antarmuka Registrasi**

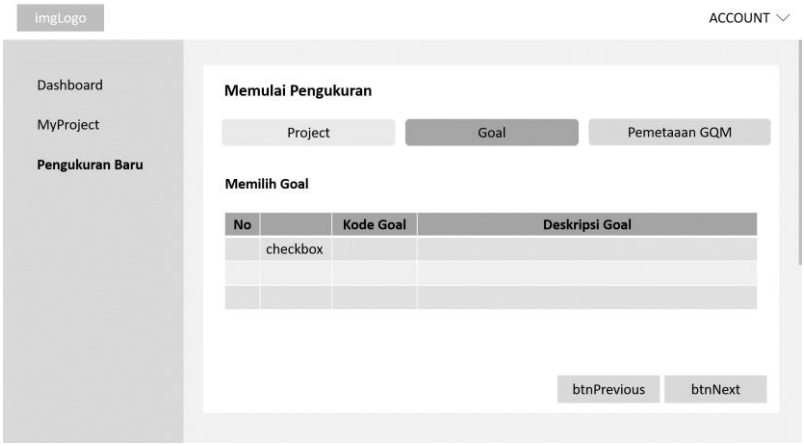
| No | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Jenis Masukan/ Keluaran |
|---|---|---|---|---|
| **1** | *txtFieldNama* | *Textfield* | *Field* untuk menambah nama tester | *String* |
| **2** | *txtFieldInstansi* | *Textfield* | *Field* untuk menambah instansi tester | *String* |
| **3** | *txtFieldEmail* | *Textfield* | *Field* untuk menambah email tester | *String* |
| **4** | *txtFieldPassword* | *Textfield* | *Field* untuk menambah password tester | *String* |
| **5** | *btnRegister* | *Button* | *Menyimpan data akun baru* | *Button* |

### 3.2.3.3. Halaman Membuat Proyek Pengukuran

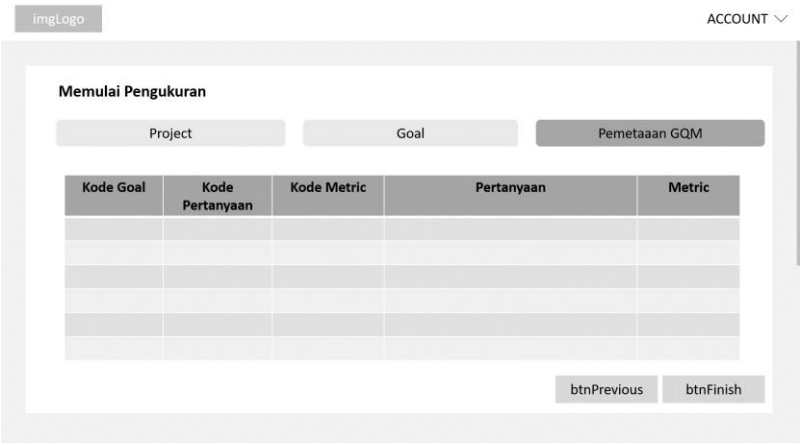
Antarmuka membuat proyek pengukuran akan digunakan pada kasus penggunaan UC002. Halaman antarmuka membuat proyek pengukuran menampilkan form pembuatan proyek pengukuran baru.

Antarmuka membuat proyek pengukuran dan penjelasan terkait antarmuka membuat proyek pengukuran dapat dilihat pada Gambar 3. 22, Gambar 3.23, Gambar 3.24 dan Tabel 3.12.

**Gambar 3. 13 Rancangan Antarmuka Membuat Proyek Pengukuran : Langkah membuat proyek**




**Gambar 3. 14 Rancangan Antarmuka Membuat Proyek Pengukuran : Langkah memilih goal**




**Gambar 3. 15 Rancangan Antarmuka Membuat Proyek Pengukuran : Langkah pemetaan GQM**

**Tabel 3. 12 Penjelasan Antarmuka Membuat Proyek Pengukuran**

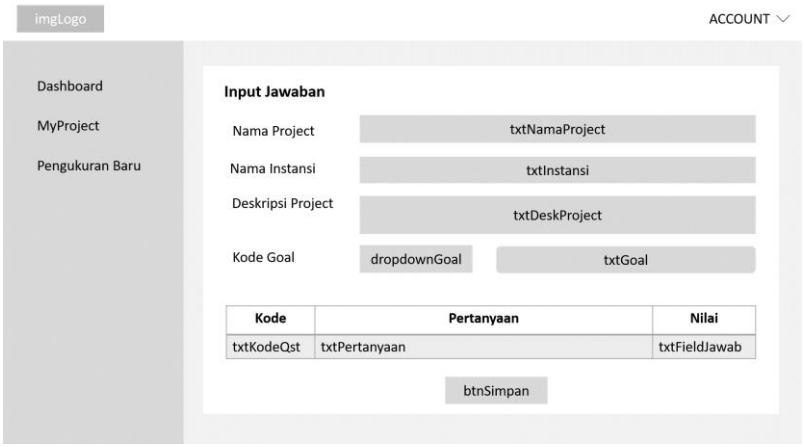
| No | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Jenis Masukan/ Keluaran |
|---|---|---|---|---|
| **1** | *txtFieldNamaProyek* | *Textfield* | *Field* untuk menambah nama proyek | *String* |
| **2** | *txtFieldInstansi* | *Textfield* | *Field* untuk menambah instansi tester | *String* |
| **3** | *txtFieldDeskripsi* | *Textfield* | *Field* untuk menambah deskripsi proyek | *String* |
| **4** | *checkbox* | *Checkbox* | *Checkbox* untuk memilih goal | *String* |

| | | | | |
|---|---|---|---|---|
| **5** | *btnPrevious* | *Button* | *Button* untuk kembali ke langkah sebelumnya | *Button* |
| **6** | *btnNext* | *Button* | *Button* untuk ke langkah selenjutnya | *Button* |
| **7** | *btnFinish* | *Button* | *Button* untuk menyimpan data proyek pengukuran baru | *Button* |

### 3.2.3.4. Halaman Menjawab Pertanyaan

Antarmuka menjawab pertanyaan akan digunakan pada kasus penggunaan UC003. Halaman antarmuka menjawab pertanyaan menampilkan form untuk menjawab pertanyaan sesuai goal yang dipilih.

Antarmuka menjawab pertanyaan dan penjelasan terkait antarmuka menjawab pertanyaan dapat dilihat pada Gambar 3.25 dan Tabel 3.13.



**Gambar 3. 16 Rancangan Antarmuka Menjawab Pertanyaan**

**Tabel 3. 13 Penjelasan Antarmuka Menjawab Pertanyaan**

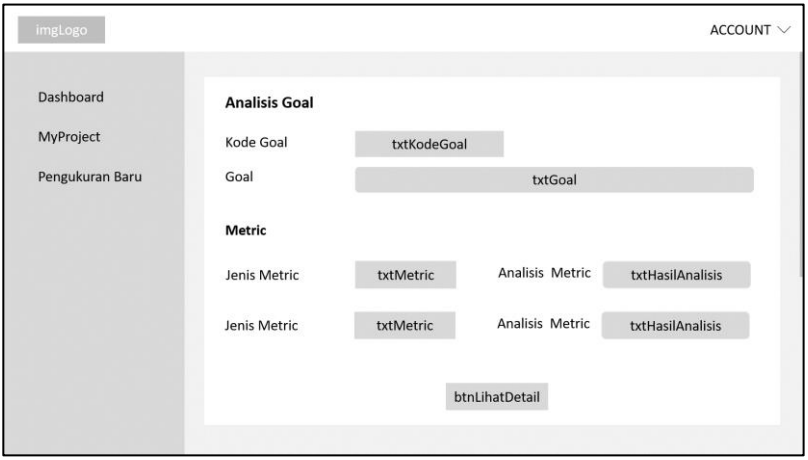
| No | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Jenis Masukan/ Keluaran |
|---|---|---|---|---|
| **1** | *txtNamaProject* | *Text* | *Field* untuk menampilkan nama proyek | *String* |
| **2** | *txtInstansi* | *Text* | *Field* untuk menampilkan instansi tester | *String* |
| **3** | *txtFieldDeskProject* | *Text* | *Field* untuk menampilkan deskripsi proyek | *String* |
| **4** | *dropdownGoal* | *Dropdown* | *Dropdown* untuk menampilkan goal yang terpilih saat membuat proyek pengukuran | *Button* |
| **5** | *txtGoal* | *Text* | *Field* untuk menampilkan deskripsi goal terpilih | *String* |
| **6** | *txtKodeQst* | *Text* | *Field* untuk menampilkan kode pertanyaan | *String* |
| **7** | *txtPertanyaan* | *Text* | *Field* untuk menampilkan pertanyaan | *String* |
| **8** | *txtFieldJawab* | *Text* | *Field* untuk memasukkan jawaban pertanyaan | *String* |

| | | | | |
|---|---|---|---|---|
| **9** | *btnSimpan* | *Button* | *Button* untuk menyimpan data jawaban pertyanyaan | *Button* |

### 3.2.3.5. Halaman Melihat Analisis Tujuan

Antarmuka melihat analisis tujuan akan digunakan pada kasus penggunaan UC004. Halaman melihat antarmuka analisis tujuan menampilkan laporan pencapaian tujuan pada pada suatu proyek.

Antarmuka melihat analisis tujuan dan penjelasan terkait antarmuka melihat analisis tujuan dapat dilihat pada Gambar 3.26 dan Tabel.




**Gambar 3. 17 Rancangan Antarmuka Analisis Tujuan**

**Tabel 3. 14 Penjelasan Antarmuka Analisis Tujuan**

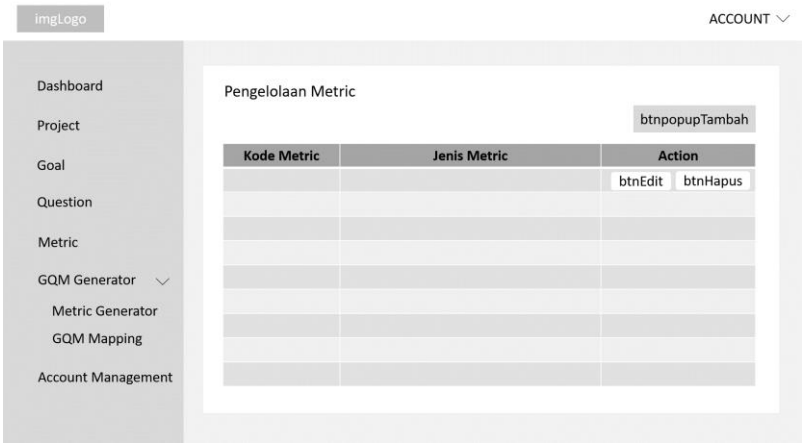
| No | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Jenis Masukan/ Keluaran |
|---|---|---|---|---|

| | | | | |
|---|---|---|---|---|
| **1** | *txtKodeGoal* | *Text* | *Field* untuk menampilkan kode goal | *String* |
| **2** | *txtGoal* | *Text* | *Field* untuk menampilkan deskripsi goal | *String* |
| **3** | *txtPencapaianGoal* | *Text* | *Field* untuk menampilkan pencapaian goal | *String* |
| **4** | *txtMetric* | *Text* | *Field* untuk menampilkan jenis metrik | *String* |
| **5** | *txtHasilPerhitungan* | *Button* | *Field* untuk menampilkan hasil perhitungan metrik | *Integer* |

### 3.2.3.6. Halaman Pengelolaan Ukuran

Antarmuka pengelolaan ukuran akan digunakan pada kasus penggunaan UC007. Halaman antarmuka pengelolaan ukuran terdiri dari beberapa antarmuka, yaitu melihat daftar ukuran, halaman mengubah metrik/ukuran, form menambah metrik dan dialog konfirmasi menghapus metrik.

Antarmuka melihat daftar ukuran/metrik dan penjelasan terkait antarmuka melihat daftar metrik dapat dilihat pada Gambar 3.27 dan Tabel 3.15.

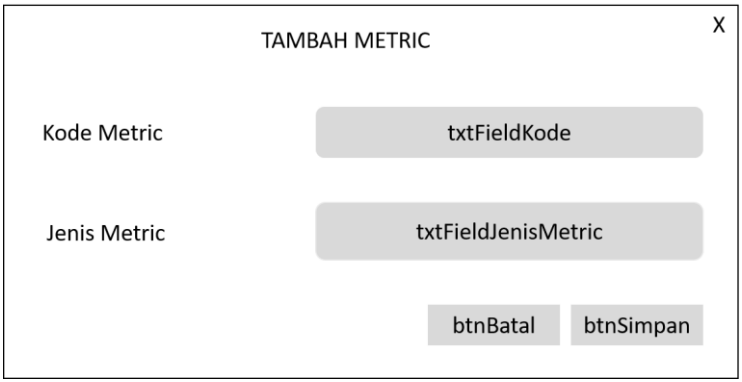



**Gambar 3. 18 Rancangan Antarmuka Melihat Daftar Ukuran/Metrik**

**Tabel 3. 15 Penjelasan Antarmuka Melihat Daftar Ukuran/Metrik**

| No | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Jenis Masukan/ Keluaran |
|---|---|---|---|---|
| **1** | *btnpopupTambah* | *Button* | Menampilkan modal form untuk menambah metrik | *Button Clicked* |
| **2** | *btnEdit* | *Button* | Menampilkan halaman detail metrik serta form untuk mengubah detail metrik | *Button Clicked* |
| **3** | *btnHapus* | *Button* | Menampilkan modal konfirmasi untuk menghapus metrik | *Button Clicked* |

Antarmuka menambah ukuran/metrik dan penjelasan terkait antarmuka menambah ukuran/metrik dapat dilihat pada Gambar 3.28 dan Tabel 3.16.

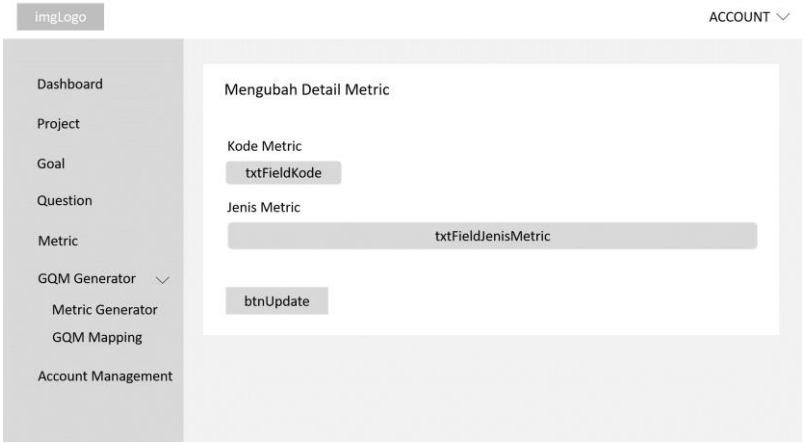



**Gambar 3. 19 Rancangan Antarmuka Menambah Ukuran/Metrik**

**Tabel 3. 16 Penjelasan Antarmuka Menambah Ukuran/Metrik**

| No | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Jenis Masukan/ Keluaran |
|---|---|---|---|---|
| **1** | *txtFieldKode* | *Textfield* | *Field* untuk menambah kode metrik | *String* |
| **2** | *btnFieldJenisMetric* | *Textfield* | *Field* untuk menambah nama jenis metrik | *String* |
| **3** | *btnSimpan* | *Button* | Menyimpan data ukur/metrik baru | *Button Clicked* |
| **4** | *btnBatal* | *Button* | Membatalkan penambahan data ukur | *Button Clicked* |

Antarmuka mengubah data ukuran/metrik dan penjelasan terkait antarmuka mengubah ukuran/metrik dapat dilihat pada Gambar 3.29 dan Tabel 3.17.

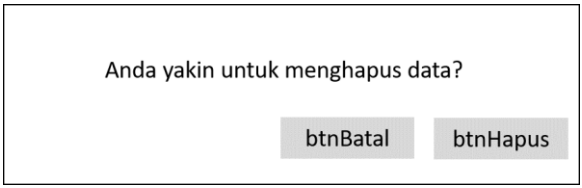



**Gambar 3. 20 Rancangan Antarmuka Mengubah Ukuran/Metrik**

**Tabel 3. 17 Penjelasan Antarmuka Mengubah Ukuran/Metrik**

| No | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Jenis Masukan/ Keluaran |
|---|---|---|---|---|
| **1** | *txtFieldKode* | *Textfield* | *Field* untuk mengubah kode metrik | *String* |
| **2** | *btnFieldJenisMetric* | *Textfield* | *Field* untuk mengubah nama jenis metrik | *String* |
| **3** | *btnUpdate* | *Button* | Menyimpan data perubahan | *Button Clicked* |

Antarmuka menghapus data ukuran/metrik dan penjelasan terkait antarmuka menghapus data ukuran/metrik dapat dilihat pada Gambar 3.30 dan Tabel 3.18.




**Gambar 3. 21 Rancangan Antarmuka Menghapus Ukuran/Metrik**

**Tabel 3. 18 Penjelasan Antarmuka Menghapus Ukuran/Metrik**

| No | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Jenis Masukan/ Keluaran |
|---|---|---|---|---|
| **1** | *btnBatal* | *Button* | Membatalkan penghapusan data ukuran/metrik | *Button Clicked* |
| **2** | *btnHapus* | *Button* | Menghapus data ukuran/metrik | *Button Clicked* |

### 3.2.3.7. Halaman Pengelolaan Pertanyaan

Antarmuka pengelolaan pertanyaan akan digunakan pada kasus penggunaan UC008. Halaman antarmuka pengelolaan pertanyaan terdiri dari beberapa antarmuka, yaitu melihat daftar pertanyaan, halaman mengubah pertanyaan, form menambah pertanyaan dan dialog konfirmasi menghapus pertanyaan.

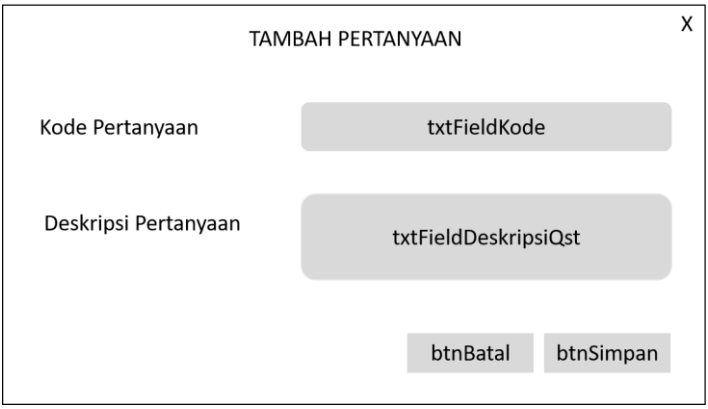
Antarmuka melihat daftar pertanyaan dan penjelasan terkait antarmuka melihat daftar pertanyaan dapat dilihat pada Gambar 3.31 dan Tabel 3.19.

**Gambar 3. 22 Rancangan Antarmuka Melihat Daftar Pertanyaan**

**Tabel 3. 19 Penjelasan Antarmuka Melihat Daftar Pertanyaan**

| No | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Jenis Masukan/ Keluaran |
|---|---|---|---|---|
| **1** | *btnpopupTambah* | *Button* | Menampilkan modal form untuk menambah pertanyaan | *Button Clicked* |
| **2** | *btnEdit* | *Button* | Menampilkan halaman detail metrik serta form untuk mengubah detail pertanyaan | *Button Clicked* |
| **3** | *btnHapus* | *Button* | Menampilkan modal konfirmasi untuk menghapus pertanyaan | *Button Clicked* |

Antarmuka menambah pertanyaan dan penjelasan terkait antarmuka menambah pertanyaan dapat dilihat pada Gambar 3.32 dan Tabel 3.20.




**Gambar 3. 23 Rancangan Antarmuka Menambah Pertanyaan**

**Tabel 3. 20 Penjelasan Antarmuka Menambah Pertanyaan**

| No | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Jenis Masukan/ Keluaran |
|---|---|---|---|---|
| **1** | *txtFieldKode* | *Textfield* | *Field* untuk menambah kode pertanyaan | *String* |
| **2** | *txtFieldDeskripsiQst* | *Textfield* | *Field* untuk menambah nama deskripsi | *String* |
| **3** | *btnSimpan* | *Button* | Menyimpan data pertanyaan baru | *Button Clicked* |
| **3** | *btnBatal* | *Button* | Membatalkan penambahan pertanyaan | *Button Clicked* |

Antarmuka mengubah data pertanyaan dan penjelasan terkait antarmuka mengubah pertanyaan dapat dilihat pada Gambar 3.33 dan Tabel 3.21.

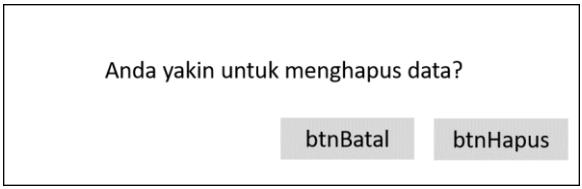
**Gambar 3. 24 Rancangan Antarmuka Mengubah Pertanyaan**

**Tabel 3. 21 Penjelasan Antarmuka Mengubah Pertanyaan**

| No | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Jenis Masukan/ Keluaran |
|---|---|---|---|---|
| **1** | *txtFieldKode* | *Textfield* | *Field* untuk mengubah kode pertanyaan | *String* |
| **2** | *btnFieldDeskripsiPertanyaan* | *Textfield* | *Field* untuk mengubah deskripsi pertanyaan | *String* |
| **3** | *dropdownTambahGoal* | *Dropdown* | *Field* untuk menambah relasi goal | *Dropdown Selected* |

| | | | | |
|---|---|---|---|---|
| **4** | *btnHapus* | *Button* | *Menghapus data relasi goal* | *Button Clicked* |
| **5** | *btnUpdate* | *Button* | Menyimpan data perubahan | *Button Clicked* |

Antarmuka menghapus data pertanyaan dan penjelasan terkait antarmuka menghapus data pertanyaan dapat dilihat pada Gambar 3.34 dan Tabel 3.32.




**Gambar 3. 25 Rancangan Antarmuka Menghapus Ukuran/Metrik**

**Tabel 3. 22 Penjelasan Antarmuka Menghapus Ukuran/Metrik**

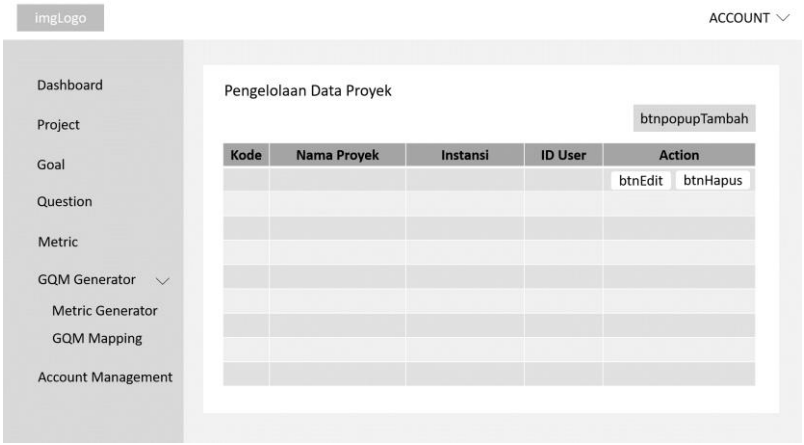
| No | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Jenis Masukan/ Keluaran |
|---|---|---|---|---|
| **1** | *btnBatal* | *Button* | Membatalkan penghapusan data pertanyaan | *Button Clicked* |
| **2** | *btnHapus* | *Button* | Menghapus data pertanyaan | *Button Clicked* |

### 3.2.3.8. Halaman Pengelolaan Data Proyek

Antarmuka pengelolaan data proyek akan digunakan pada kasus penggunaan UC009. Halaman antarmuka pengelolaan data proyek terdiri dari beberapa antarmuka, yaitu melihat daftar data

proyek, halaman mengubah data proyek, form menambah data proyek dan dialog konfirmasi menghapus data proyek.

Antarmuka melihat daftar data proyek dan penjelasan terkait antarmuka melihat daftar data proyek dapat dilihat pada Gambar 3.35 dan Tabel 3.23.

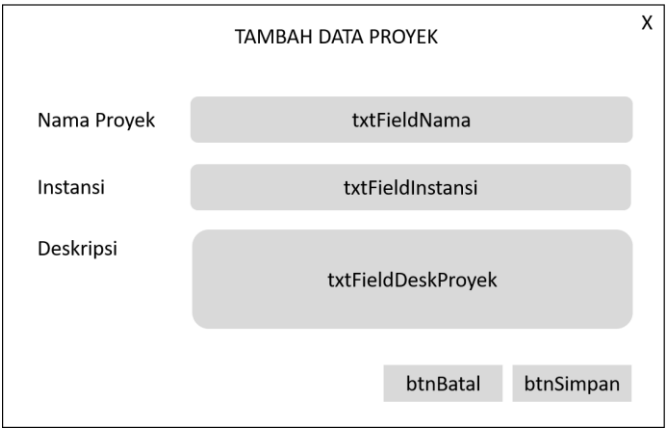



**Gambar 3. 26 Rancangan Antarmuka Melihat Daftar Data Proyek**

**Tabel 3. 23 Penjelasan Antarmuka Melihat Daftar Data Proyek**

| No | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Jenis Masukan/ Keluaran |
|---|---|---|---|---|
| **1** | *btnpopupTambah* | *Button* | Menampilkan modal form untuk menambah data proyek | *Button Clicked* |
| **2** | *btnEdit* | *Button* | Menampilkan halaman detail metrik serta form untuk mengubah detail data proyek | *Button Clicked* |
| **3** | *btnHapus* | *Button* | Menampilkan modal konfirmasi untuk menghapus data proyek | *Button Clicked* |

Antarmuka menambah data proyek dan penjelasan terkait antarmuka menambah data proyek dapat dilihat pada Gambar 3.36 dan Tabel 3.24.



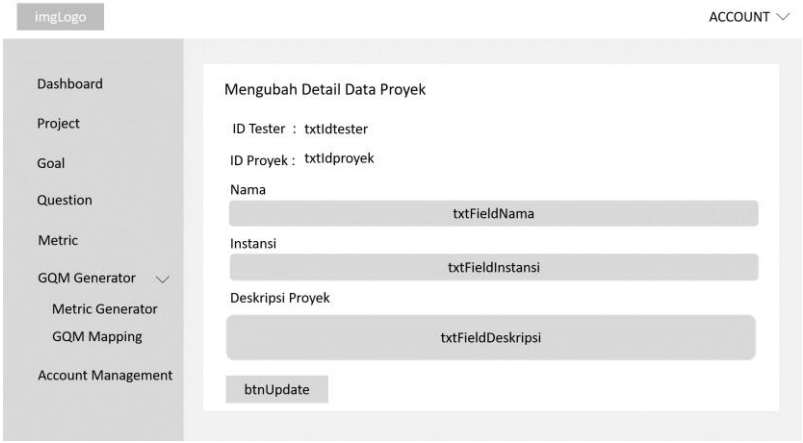

**Gambar 3. 27 Rancangan Antarmuka Menambah Data Proyek**

**Tabel 3. 24 Penjelasan Antarmuka Menambah Data Proyek**

| No | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Jenis Masukan/ Keluaran |
|---|---|---|---|---|
| **1** | *txtFieldNama* | *Textfield* | *Field* untuk menambah kode pertanyaan | *String* |
| **2** | *txtFieldInstansi* | *Textfield* | *Field* untuk menambah nama instansi | *String* |
| **3** | *txtFieldDeskProyek* | *Textfield* | *Field* untuk menambah nama deskripsi proyek | *String* |
| **4** | *btnSimpan* | *Button* | Menyimpan data pertanyaan baru | *Button Clicked* |

| | | | | |
|---|---|---|---|---|
| **5** | *btnBatal* | *Button* | Membatalkan penambahan pertanyaan | *Button Clicked* |

Antarmuka mengubah data data proyek dan penjelasan terkait antarmuka mengubah data proyek dapat dilihat pada Gambar 3.37 dan Tabel 3.25.



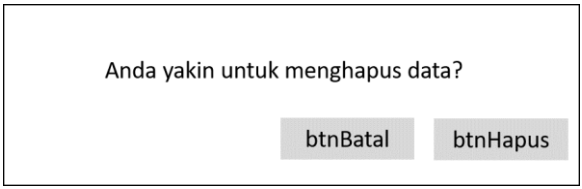

**Gambar 3. 28 Rancangan Antarmuka Mengubah Data Proyek**

**Tabel 3. 25 Penjelasan Antarmuka Mengubah Data Proyek**

| No | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Jenis Masukan/ Keluaran |
|---|---|---|---|---|
| **1** | txtIdtester | *Text* | *Field* untuk menampilkan ID tester | *Integer* |
| **2** | txtIdproyek | *Text* | *Field* untuk menampilkan ID proyek | *Integer* |

| | | | | |
|---|---|---|---|---|
| **3** | *txtFieldNama* | *Textfield* | *Field* untuk mengubah kode pertanyaan | *String* |
| **4** | *txtFieldInstansi* | *Textfield* | *Field* untuk mengubah deskripsi pertanyaan | *String* |
| **5** | *txtFieldDeskripsi* | *Dropdown* | *Field* untuk menambah relasi goal | *Dropdown Selected* |
| **6** | *btnUpdate* | *Button* | Menyimpan data perubahan | *Button Clicked* |

Antarmuka menghapus data data proyek dan penjelasan terkait antarmuka menghapus data data proyek dapat dilihat pada Gambar 3.38 dan Tabel 3.26.



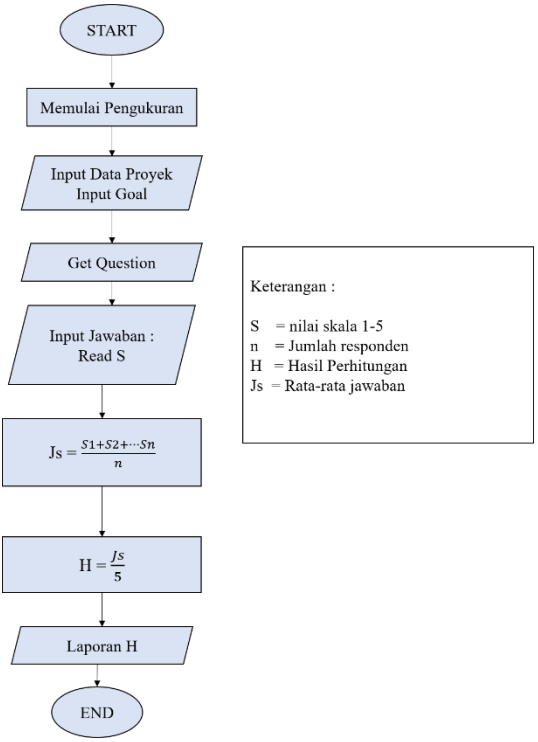

**Gambar 3. 29 Rancangan Antarmuka Menghapus Data Proyek**

**Tabel 3. 26 Penjelasan Antarmuka Menghapus Data Proyek**

| No | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Jenis Masukan/ Keluaran |
|---|---|---|---|---|
| **1** | *btnBatal* | *Button* | Membatalkan penghapusan data proyek | *Button Clicked* |
| **2** | *btnHapus* | *Button* | Menghapus data proyek | *Button Clicked* |

### 3.2.4. Perancangan Algoritma Melakukan Pengukuran

Pada Subbab ini akan dijelaskan perancangan algoritma melakukan pengukuran. Nantinya, pengguna dengan hak akses Tester akan melakukan pengukuran dengan membuat proyek data pengukuran baru dengan goal yang terpilih saat pengisian data proyek. Kemudian Tester diharuskan untuk menjawab pertanyaan yang disediakan sistem sesuai dengan goal yang dipilih. Jawaban akan disimpan pada sistem dan dihitung saat melakukan analisis hasil pengukuran. Proses melakukan pengukuran dapat dilihat pada Gambar 3.39.




**Gambar 3. 30 Rancangan Algoritma Melakukan Pengukuran**

# BAB IV
# IMPLEMENTASI

Bab ini membahas implementasi yang dilakukan berdasarkan rancangan yang telah dijabarkan pada bab sebelumnya. Sebelum penjelasan implementasi akan ditunjukkan terlebih dahulu lingkungan untuk melakukan implementasi.

Pada bagian implementasi ini juga akan dijelaskan mengenai fungsi-fungsi yang digunakan dalam program Tugas Akhir ini dan disertai dengan kode semu masing-masing fungsi utama.

## 4.1. Lingkungan Implementasi

Lingkungan implementasi sistem yang digunakan untuk mengembangkan Tugas Akhir memiliki spesifikasi perangkat keras dan perangkat lunak seperti yang ditampilkan pada Tabel 4.1.

**Tabel 4. 1 Lingkungan Implementasi**

| Perangkat | Spesifikasi | Deskripsi |
|---|---|---|
| Keras | CPU | Intel(R) Core(TM) i5-8265U CPU @ 1.60GHz 1.80 GHz |
| | RAM | 4.00 GB |
| Lunak | Sistem Operasi | Windows 10 Home Single Language 64-bit |
| | Browser | Google Chrome Version 83.0.4103.106 (Official Build) (64-bit) |
| | Web Server | Apache |
| | Server Basis Data | MySql |

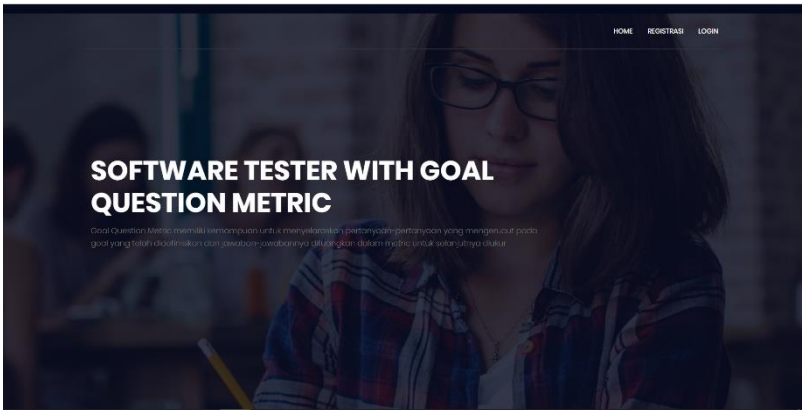
## 4.2. Implementasi Antarmuka Sistem

Berikut ini akan dijelaskan mengenai implementasi antarmuka sistem yang sudah direalisasikan.

### 4.2.1. Halaman Beranda

Halaman beranda merupakan halaman utama ketika aplikasi web dibuka. Pada halaman ini terdapat menu registrasi dan login.

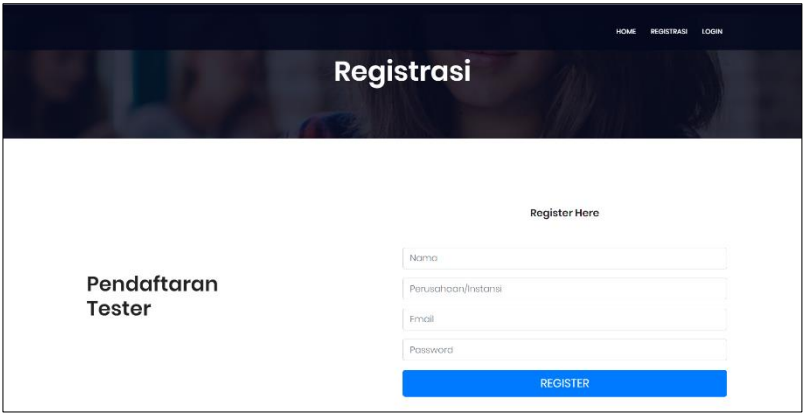
Tester yang belum melakukan pendaftaran akun tidak dapat melakukan pengukuran. Aktor yang dapat mengakses ini adalah Tester dan Administrator. Halaman beranda dapat dilihat pada Gambar 4.1.




**Gambar 4. 1 Implementasi Antarmuka Halaman Beranda**

### 4.2.2. Halaman Mendaftarkan Akun

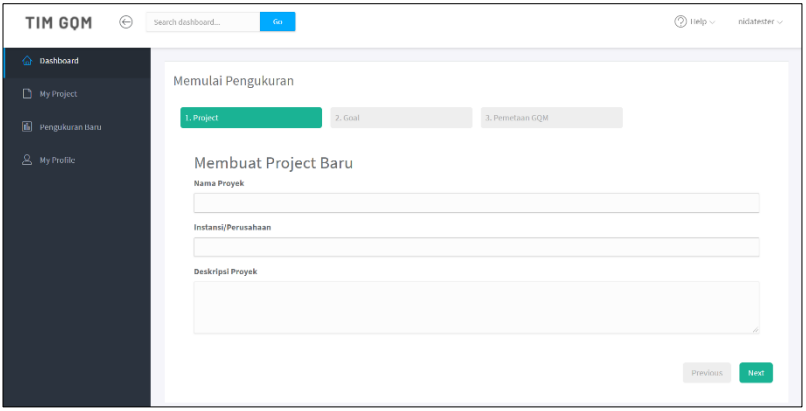
Halaman mendaftarkan akun merupakan halaman yang digunakan untuk melakukan kasus penggunaan UC001 yaitu mendaftarkan akun. Aktor yang mendapat hak akses pada halaman ini adalah Tester/Calon Tester. Pengguna yang belum mempunyai akun pada sistem dapat mendaftarkan diri sebagai tester. Halaman antarmuka mendaftarkan dapat dilihat pada Gambar 4.2.



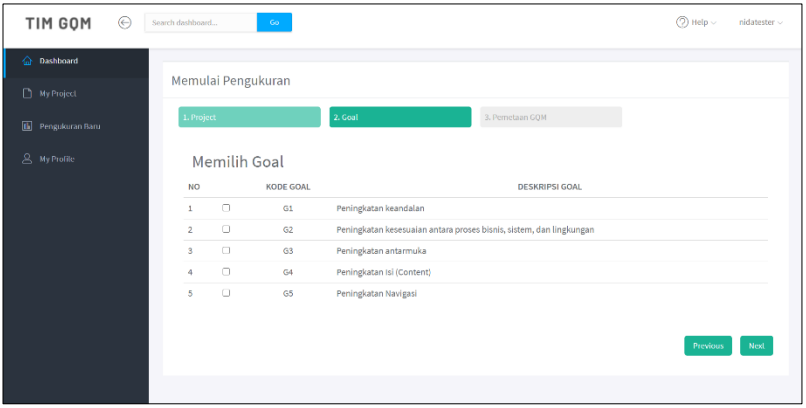

**Gambar 4. 2 Implementasi Antarmuka Mendaftarkan Akun**

### 4.2.3. Halaman Membuat Proyek Pengukuran

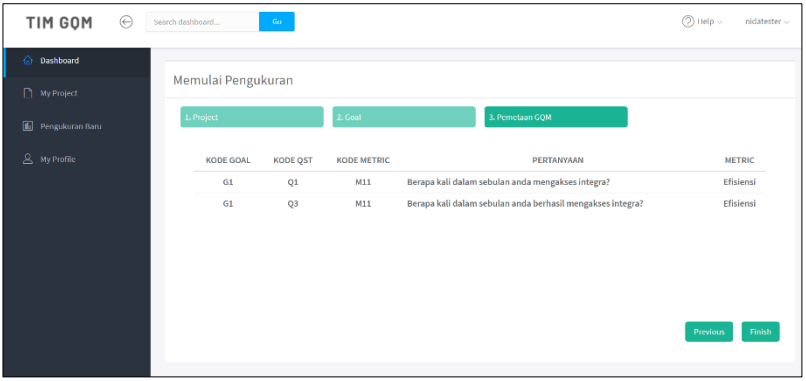
Halaman membuat proyek pengukuran merupakan halaman yang digunakan untuk melakukan kasus penggunaan UC002 yaitu membuat proyek pengukuran. Aktor yang mendapat hak akses pada halaman ini adalah Tester. Tester dapat membuat proyek pengukuran baru dengan mengisi form pembuatan proyek pengukuran dan memilih goal yang hendak diukur. Halaman antarmuka membuat proyek pengukuran dapat dilihat pada Gambar 4.3, Gambar 4.4 dan 4.5.




**Gambar 4. 3 Implementasi Antarmuka Halaman Membuat Proyek Pengukuran ( Langkah 1 )**



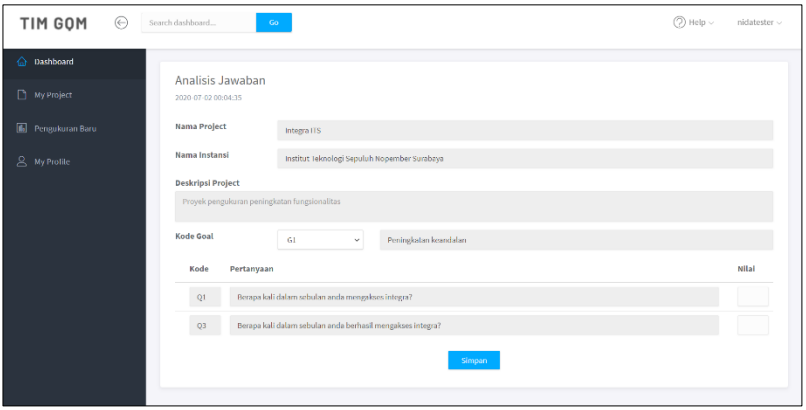

**Gambar 4. 4 Implementasi Antarmuka Halaman Membuat Proyek Pengukuran ( Langkah 2 )**




**Gambar 4. 5 Implementasi Antarmuka Halaman Membuat Proyek Pengukuran ( Langkah 3 )**

### 4.2.4. Halaman Menjawab Pertanyaan

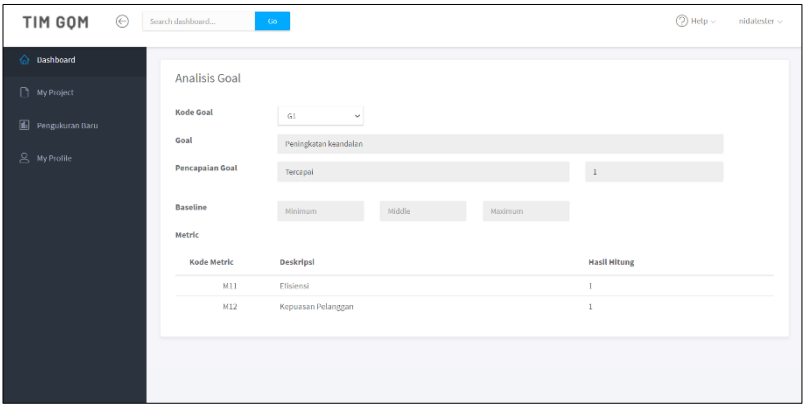
Halaman menjawab pertanyaan merupakan halaman yang digunakan untuk melakukan kasus penggunaan UC003 yaitu menjawab pertanyaan. Aktor yang mendapat hak akses pada halaman ini adalah Tester. Tester hanya dapat menjawab pertanyaan sesuai goal yang dipilih saat pembuatan proyek pengukuran baru. Halaman antarmuka menjawab pertanyaan dapat dilihat pada Gambar 4.6.



**Gambar 4.6 Implementasi Antarmuka Halaman Menjawab Pertanyaan**

### 4.2.5. Halaman Melihat Analisis Tujuan

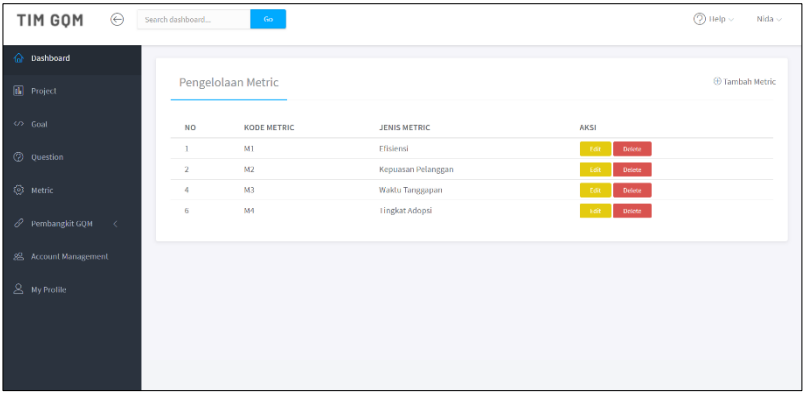
Halaman melihat analisis tujuan merupakan halaman yang digunakan untuk melakukan kasus penggunaan UC004 yaitu melihat analisis tujuan. Aktor yang mendapat hak akses pada halaman ini adalah Tester. Tester dapat melihat analisis goal dari proyek yang telah diukur. Halaman antarmuka melihat analisis tujuan dapat dilihat pada Gambar 4.7.




**Gambar 4. 7 Implementasi Antarmuka Halaman Analisis Tujuan**

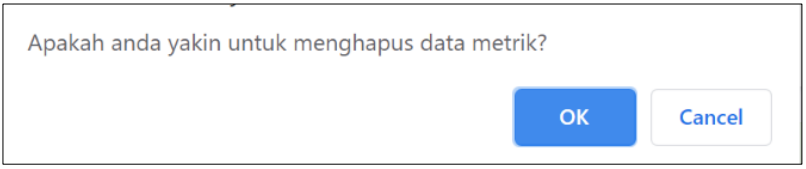
### 4.2.6. Halaman Mengelola Ukuran

Halaman mengelola ukuran merupakan halaman yang digunakan untuk melakukan kasus penggunaan UC007 yaitu mengelola data ukuran/metric. Aktor yang mendapat hak akses pada halaman ini hanyalah Admin. Admin dapat melihat, menambah, mengubah dan menghapus data ukuran pada kasus penggunaan ini. Halaman antarmuka mengelola data ukuran dapat dilihat pada Gambar 4.8, Gambar 4.9, Gambar 4.10 dan Gambar 4.11.

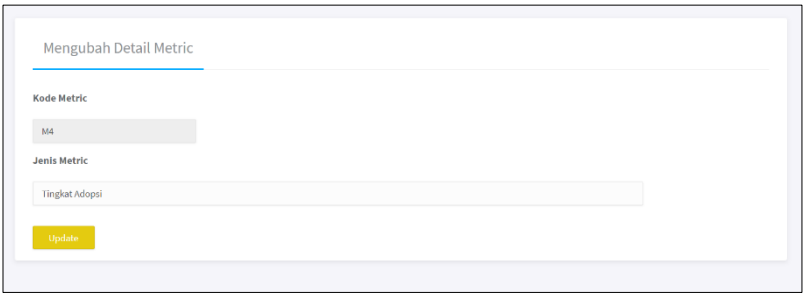



**Gambar 4. 8 Implementasi Antarmuka Melihat Daftar Data Ukuran**

**Gambar 4. 9 Implementasi Antarmuka Menambah Data Ukuran**



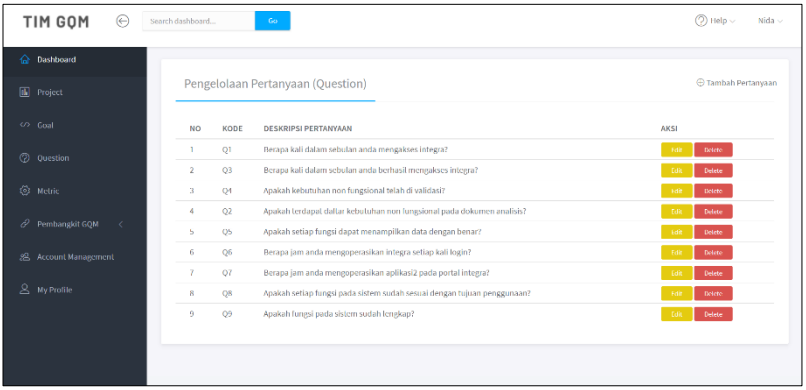
**Gambar 4. 10 Implementasi Antarmuka Menghapus Data Ukuran**



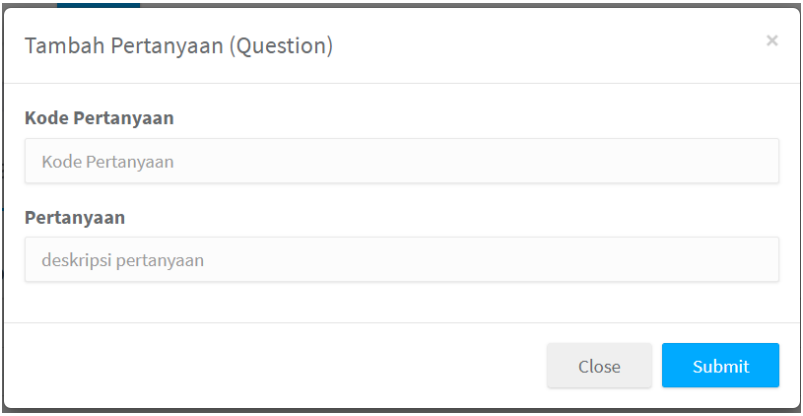

**Gambar 4. 11 Implementasi Antarmuka Mengubah Data Ukuran**

### 4.2.7. Halaman Mengelola Pertanyaan

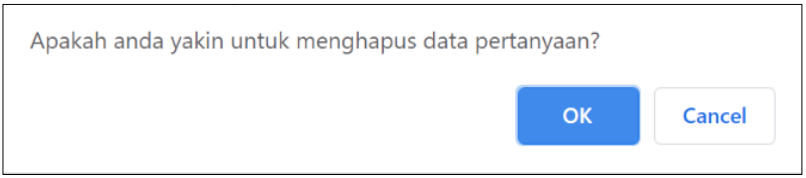
Halaman mengelola pertanyaan merupakan halaman yang digunakan untuk melakukan kasus penggunaan UC008 yaitu mengelola data pertanyaan. Aktor yang mendapat hak akses pada halaman ini hanyalah Admin. Admin dapat melihat, menambah, mengubah dan menghapus data pertanyaan pada kasus penggunaan ini. Halaman antarmuka mengelola data pertanyaan dapat dilihat pada Gambar 4.12, Gambar 4.13, Gambar 4.14 dan Gambar 4.15.



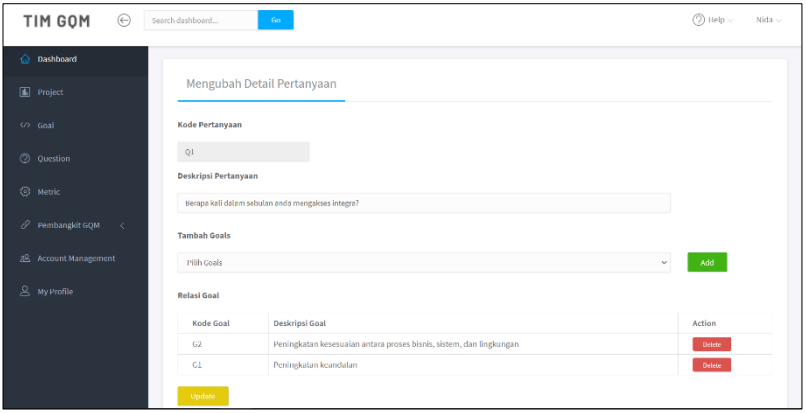
**Gambar 4. 12 Implementasi Antarmuka Melihat Daftar Data Pertanyaan**




**Gambar 4. 13 Implementasi Antarmuka Menambah Data Pertanyaan**



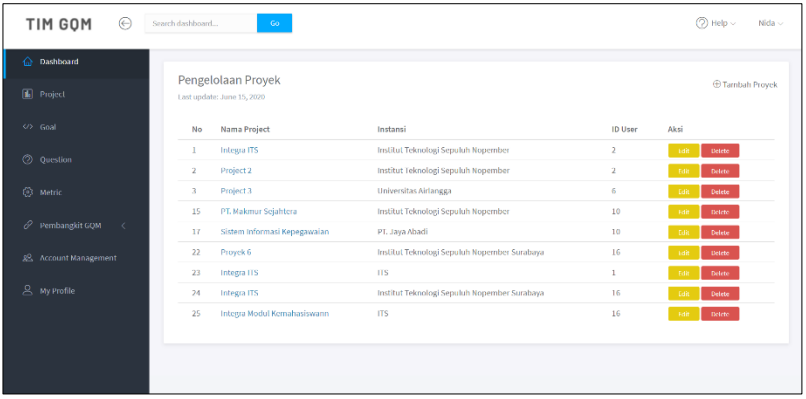

**Gambar 4. 14 Implementasi Antarmuka Menghapus Data Pertanyaan**



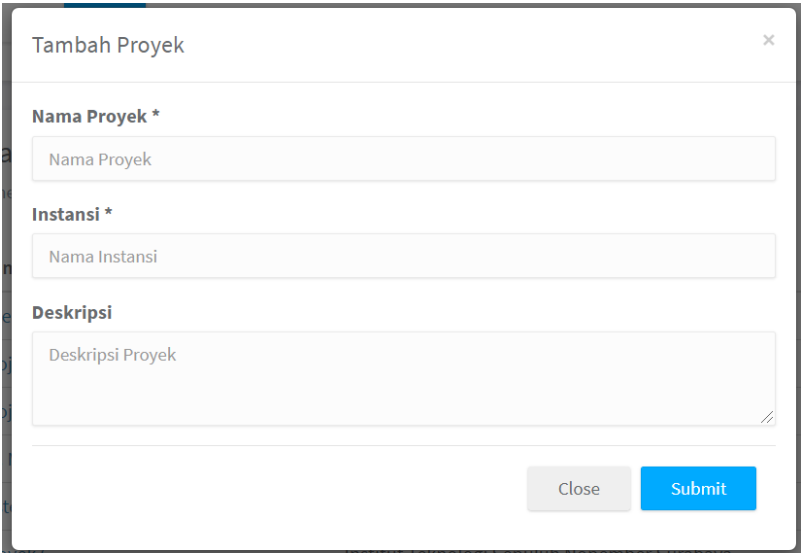

**Gambar 4. 15 Implementasi Antarmuka Mengubah Data Pertanyaan**

### 4.2.8. Halaman Mengelola Data Proyek

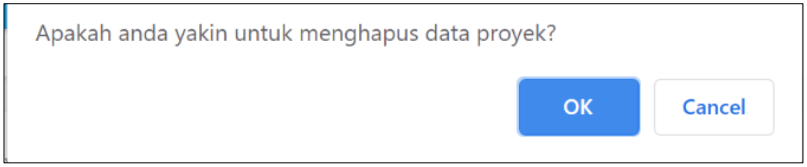
Halaman mengelola data proyek merupakan halaman yang digunakan untuk melakukan kasus penggunaan UC009 yaitu mengelola data proyek. Aktor yang mendapat hak akses pada halaman ini hanyalah Admin. Admin dapat melihat, menambah, mengubah dan menghapus data proyek pada kasus penggunaan ini. Halaman antarmuka mengelola data proyek dapat dilihat pada Gambar 4.16, Gambar 4.17, Gambar 4.18 dan Gambar 4.19.



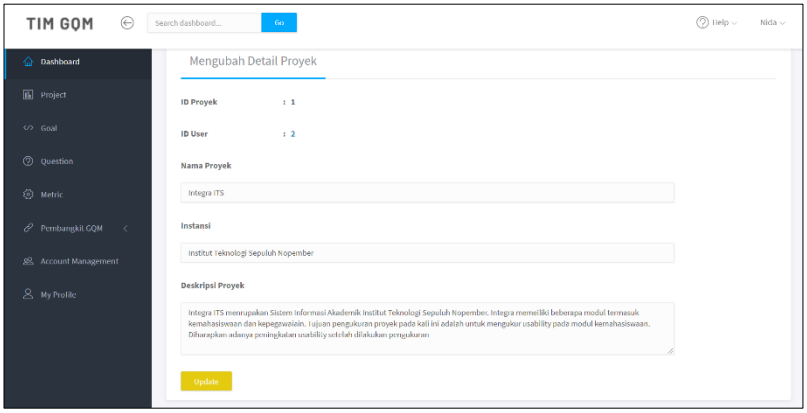

**Gambar 4. 16 Implementasi Antarmuka Melihat Daftar Data Proyek**




**Gambar 4. 17 Implementasi Antarmuka Menambah Data Proyek**




**Gambar 4. 18 Implementasi Antarmuka Menghapus Data Proyek**




**Gambar 4. 19 Implementasi Antarmuka Mengubah Data Proyek**

## 4.3. Implementasi Kasus Penggunaan

Pada subbab ini akan menampilkan dan menjelaskan kode sumber pada kasus penggunaan. Implementasi kasus penggunaan menggunakan *framework* Laravel 7.0 dengan Bahasa pemrograman php dan dibangun pada lingkungan pengembangan Visual Studio Code.

### 4.3.1. Implementasi Kasus Penggunaan Mendaftarkan Akun

Halaman ini merupakan halaman yang digunakan untuk melakukan kasus penggunaan UC001 yaitu mendaftarkan akun. Seorang calon Software Tester harus terlebih dulu mendaftar sebagai tester sebelum memulai pengukuran. Implementasi kasus penggunaan melihat halaman registrasi dapat dilihat pada Kode Sumber 4.1 dengan penjelasan pada Tabel 4.2. Sedangkan kasus penggunaan mendaftarkan akun dapat dilihat pada Kode Sumber 4.2 dan penjelasannya dapat dilihat pada Tabel 4.3.

```
1.  public function register()
2.  {
3.      return view('sites.register');
4.  }
```

**Kode Sumber 4. 1 Fungsi Melihat Halaman Registrasi**

**Tabel 4. 2 Penjelasan Melihat Halaman Registrasi**

| No. Baris | Kegunaan |
|---|---|
| 3 | Menampilkan view halaman registrasi |

```
1.  public function postreg(Request $request)
2.  {
3.      $this->validate($request,[
4.          'nama' =>'required',
5.          'perusahaan' => 'required',
6.          'email' => 'required|email|unique:users',
7.          'password' => 'required|min:5'
```

```
8.      ]);
9.
10.     $user = new \App\User;
11.     $user->role = 'tester';
12.     $user->name = $request->nama;
13.     $user->email = $request->email;
14.     $user->password = bcrypt($request->password);
15.     $user->remember_token = str::random(60);
16.     $user->save();
17.
18.     return redirect('/register')-
    >with('success','Pendaftaran berhasil, silahkan login
    !');
19. }
```

**Kode Sumber 4. 2 Fungsi Mendaftarkan Akun**

**Tabel 4. 3 Penjelasan Mendaftarkan Akun**

| No. Baris | Kegunaan |
|---|---|
| 3-7 | Melakukan validasi data isian pada form pendaftaran, semua data wajib diisi dengan ketentuan yang berlaku |
| 10-16 | Menambahkan dan menyimpan data pengguna baru pada database |
| 18 | Mengembalikan tampilan ke halaman registrasi |

### 4.3.2. Implementasi Kasus Penggunaan Membuat Proyek Pengukuran

Halaman ini merupakan halaman yang digunakan untuk melakukan kasus penggunaan UC002 yaitu membuat proyek pengukuran. Seorang Tester dapat membuat proyek pengukuran untuk mengukur perangkat lunak. Ada tiga langkah dalam membuat proyek pengukuran. Pertama, Tester diharuskan mengisi data proyek terlebih dulu. Kedua, Tester dipersilakan untuk memilih goal pengukuran. Ketiga, Tester dapat melihat pertanyaan serta pemetaan GQM pada goal yang terpilih. Implementasi kasus penggunaan membuat proyek pengukuran langkah menampilkan goal yang tersedia dapat dilihat pada Kode Sumber 4.3 dan penjelasannya dapat dilihat pada Tabel 4.4.

```
1.  public function testergoal()
2.  {
3.      $data_goal = Goal::all()->sortBy('kode_goal');
4.
5.      return view('goal.testergoal',['data_goal'=> $dat
    a_goal]);
6.  }
```

**Kode Sumber 4. 3 Fungsi Melihat Daftar Goal**

**Tabel 4. 4 Penjelasan Melihat Daftar Daftar Goal**

| No. Baris | Kegunaan |
|---|---|
| 3 | Mengambil seluruh data goal yang tersimpan pada database |
| 5 | Mengembalikan tampilan ke halaman melihat daftar goal pada halaman membuat pengukuran baru |

Implementasi kasus penggunaan membuat proyek pengukuran langkah menampilkan pertanyaan/pemetaan GQM dapat dilihat pada Kode Sumber 4.4 dengan penjelasan pada Tabel 4.5.

```
1.  function ajaxGetQuestion(Request $request)
2.  {
3.     $output = [];
4.     if($request->get('id')){
5.       foreach($request->get('id') as $item)
6.       {
7.         $getQuestionId = PembangkitGqm::where('pembang
    kit_gqm.goal_id', '=', $item)->get();
8.
9.       foreach($getQuestionId as $item2){
10.
11.        $menu = ...;
12.
13.        $data = array(
14.        'kode_goal' => $menu->kode_goal,
```

```
15.         'kode_qst' => $menu->kode_qst,
16.         'kode_metric' => $menu->kode_metric,
17.           'deskripsi_qst' => $menu-
    >deskripsi_qst,
18.           'jenis_metric' => $menu-
    >jenis_metric,
19.           );
20.
21.         $output[] = $data;
22.         }
23.     }
24.   }
25.         return $output;
26. }
```

**Kode Sumber 4. 4 Fungsi Menampilkan Pertanyaan**

**Tabel 4. 5 Penjelasan Menampilkan Pertanyaan**

| No. Baris | Kegunaan |
|---|---|
| 4-7 | Mengambil data dari pembangkit GQM berdasarkan id goal yang dipilih |
| 9-11 | Query join tabel goal, question, metric dan pembangkit GQM |
| 13-19 | Array yang berisi hasil dari query pada baris 9-11 yang dimasukkan ke dalam variabel data |
| 21 | $output berisi array dari variabel data |
| 25 | Mengembalikan nilai output |

Implementasi kasus penggunaan membuat proyek pengukuran langkah menyimpan data proyek pengukuran baru dapat dilihat pada Kode Sumber 4.5 dengan penjelasan pada Tabel 4.6.

```
1.  public function testergoal_store(Request $request)
2.  {
3.
4.      DB::BeginTransaction();
5.      try {
6.          $project = new Project;
```

```
7.          $project->nama_project = $request-
    >get('nama_proyek');
8.          $project->instansi = $request-
    >get('instansi');
9.          $project->deskripsi_project = $request-
    >get('deskripsi_proyek');
10.         $project->user_id = Auth::user()->id;
11.         $project->save();
12.
13.         foreach($request-
    >get('kode_goal') as $item){
14.             $projectDetail = new ProjectGoal;
15.             $projectDetail->project_id = $project-
    >id;
16.             $projectDetail->goal_id = $item;
17.             $projectDetail->save();
18.         }
19.
20.        DB::commit();
21.        return redirect()->route('goalTest')-
    >with('success', 'Data berhasil ditambah.');
22.        } catch (\Exception $e) {
23.            DB::rollback();
24.            echo $e->getmessage();
25.        }
26. }
```

**Kode Sumber 4. 5 Fungsi Menyimpan Data Proyek Pengukuran**

**Tabel 4. 6 Penjelasan Menyimpan Data Proyek Pengukuran**

| No. Baris | Kegunaan |
|---|---|
| 4 | Menginisiasi proses transaksi data |
| 6-11 | Mendeklarasikan variabel nama_proyek, instansi, deskripsi_proyek dan user_id yang diambil dari request parameter yang telah diinputkan user dan menambahkan data ke dalam database |
| 13-17 | Menambah dan menyimpan detail proyek pengukuran yang diambil dari request parameter berupa id proyek |

| | |
|---|---|
| | dan id goal yang terpilih serta menambahkan data ke dalam database |
| 20-21 | Menyimpan data yang diinputkan ke dalam database dan mengembalikan tampilan |
| 22-24 | Melakukan *rollback* untuk mengembalikan data seperti sebelumnya jika pemrosesan query belum selesai |

### 4.3.3. Implementasi Kasus Penggunaan Menjawab Pertanyaan

Halaman ini merupakan halaman yang digunakan untuk melakukan kasus penggunaan UC003 yaitu menjawab pertanyaan. Setelah seorang Tester membuat proyek pengukuran baru, ia diharuskan untuk menjawab pertanyaan sesuai goal yang dipilih sebelumnya saat membuat proyek baru. Implementasi kasus penggunaan menjawab pertanyaan dapat dilihat pada Kode Sumber 4.6 dan penjelasannya dapat dilihat pada Tabel 4.7.

```
1.  public function goal_question_answer_store_user(Reque
    st $request, $id)
2.  {
3.      $goals = $request->get('goals');
4.      $kode_qst = $request->get('kode_qst');
5.      $nilai = $request->get('jawaban_qst');
6.      $question_id = $request->get('question_id');
7.      $ans_opt = $request->get('pembilang_penyebut');
8.
9.      DB::BeginTransaction();
10.      try {
11.         $val_rata_penyebut = 0;
12.         $val_rata_pembilang = 0;
13.         $val_range = 0;
14.
15.         for ($i=0; $i<count($request-
    >get('kode_qst')); $i++) {
16.             $val_kode = $kode_qst[$i];
17.
18.             $val_nilai = $nilai[$i];
19.             $val_question = $question_id[$i];
20.             if($ans_opt[$i] == 'Range'){
```

```
21.                 $pembangkit_id = ...;
22.                 $jawabanQst = new JawabanGqm;
23.                 ...
24.                 $jawabanQst -> is_range = 1;
25.                 $jawabanQst->save();
26.
27.             } else {
28.                 $pembangkit_id = ...;
29.                 $jawabanQst = new JawabanGqm;
30.                 ...
31.                 $jawabanQst->save();
32.         }
33. }
```

**Kode Sumber 4. 6 Fungsi Menjawab Pertanyaan**

**Tabel 4. 7 Penjelasan Menjawab Pertanyaan**

| No. Baris | Kegunaan |
|---|---|
| 3-7 | Mendeklarasikan variabel goals, kode_qst, nilai, question_id dan ans_opt yang diambil dari request parameter pada halaman menjawab pertanyaan. Variabel ans_opt berisi status dari pertanyaan yang tersedia, status pertanyaan dapat menjadi pembilang, penyebut maupun range. Range merupakan status untuk pertanyaan yang membutuhkan jawaban menggunakan skala *likert*. Status pertanyaan dibuat ketika membuat pertanyaan pada pembangkit metric. |
| 11-13 | Inisiasi rata-rata untuk rata-rata jawaban penyebut, rata-rata jawaban pembilang dan rata-rata jawaban range. |
| 15 | Melakukan looping untuk setiap pertanyaan yang tersedia. Setiap jawaban masuk ke dalam database secara berurutan. |
| 16-19 | Mendeklarasikan variabel val_kode, val_nilai dan val_question yang merupakan kode_qst, nilai dan question_id yg diambil dari tahap sebelumnya pada baris 3-7. |
| 20-25 | Jika ans_opt berstatus ‘range’, maka sistem akan mendeklarasikan variabel pembangkit_id yang |

| | |
|---|---|
| | merupakan array dari data pembangkit gqm pada pertanyaan tersebut, kemudian mendeklarasikan variabel baru untuk memasukkan jawaban pada database. Untuk membedakan jawaban dari pertanyaan yang berstatus range, diinputkan angka 1 pada atribut is_range. Hal ini dilakukan karena perhitungan pada setiap status pertanyaan berbeda. Sehingga status pertanyaan menjadi sangat penting. Pada line 25, jawaban pertanyaan disimpan ke dalam database. |
| 27-31 | Jika ans_opt berstatus pembilang atau penyebut, maka sistem akan mendeklarasikan variabel pembangkit_id yang merupakan array dari data pembangkit gqm pada pertanyaan tersebut, kemudian mendeklarasikan variabel baru unruk memasukkan jawaban pada database. Jika sebelumnya untuk range menambahkan angka 1 pada atribut is_range, pada bagian ini tidak dibutuhkan penambahan pda atribut is_range. Pada line 31, jawaban pertanyaan disimpan ke dalam database. |

Implementasi kasus penggunaan menghitung jawaban pertanyaan dapat dilihat pada Kode Sumber 4.7, dan penjelasan kode dapat dilihat pada Tabel 4.8.

```
1.  $kode_analisa = ...;
2.  if($pembangkit_id-
    >pembilang_penyebut == "Pembilang"){
3.     $val_rata_pembilang = ...;
4.     $val_rata_pembilang = $val_rata_pembilang['rata_pe
    mbilang'];
5.   }
6.
7.  else if ($pembangkit_id-
    >pembilang_penyebut == "Penyebut"){
8.     $val_rata_penyebut = ...;
9.     $val_rata_penyebut = $val_rata_penyebut['rata_peny
    ebut'];
10.  }
11.
12. else if ($pembangkit_id-
    >pembilang_penyebut == "Range"){
```

```
13.     $val_range = ...;
14.     $val_range = $val_range['rata_range'];
15.  }
```

**Kode Sumber 4. 7 Fungsi Menghitung Jawaban (1)**

**Tabel 4. 8 Penjelasan Menghitung Jawaban (1)**

| No. Baris | Kegunaan |
|---|---|
| 1 | Setelah memasukkan jawaban, sistem kemudian menghitung rata-rata jawaban dan menyimpannya pada tabel Analisa GQM sesuai dengan metric yang diukur. |
| 2-4 | Sebelumnya, data pembangkit gqm telah dideklarasikan pada variabel pembangkit_id. Jika status pertanyaan merupakan pembilang, maka dihitung rata-rata pembilang pada val_rata_pembilang. Rata-rata dihitung dengan menjumlahkan jawaban dari pertanyaan yang sama, kemudian dibagi dengan jumlah responden (berapa kali jawaban diinput). Setelah itu, rata-rata pembilang disimpan pada array rata_pembilang. |
| 7-9 | Sebelumnya, data pembangkit gqm telah dideklarasikan pada variabel pembangkit_id. Jika status pertanyaan merupakan penyebut, maka dihitung rata-rata penyebut pada val_rata_penyebut. Rata-rata dihitung dengan menjumlahkan jawaban dari pertanyaan yang sama, kemudian dibagi dengan jumlah responden (berapa kali jawaban diinput). Setelah itu, rata-rata penyebut disimpan pada array rata_penyebut. |
| 12-14 | Sebelumnya, data pembangkit gqm telah dideklarasikan pada variabel pembangkit_id. Jika status pertanyaan merupakan range, maka dihitung rata-rata range pada val_range. Setiap skala dimuai dari ‘sangat tidak setuju’ sampai ‘sangat setuju’ diberi nilai 1-5. Rata-rata dihitung dengan menjumlahkan jawaban/nilai dari pertanyaan pada setiap metric, kemudian dibagi dengan jumlah responden (berapa kali jawaban diinput). Setelah itu, rata-rata range disimpan pada array rata_range. |

Implementasi menghitung jawaban bagian 2 dapat dilihat pada Kode Sumber 4.8, dan penjelasannya dapat dlilhat pada tabel 4.9.

```
$analisaProject = ...;

if($analisaProject){
  if($val_rata_penyebut != 0 && $val_rata_pembilang != 0){
     ...
  } else if($val_range != 0){
     $hasil_hitung = $val_range/5;

     if($hasil_hitung > $kode_analisa['baseline_med']) {
        $hasil_analisa = "Tercapai";
     } else $hasil_analisa = "Tidak Tercapai";
        $AnalisaProject = AnalisaProject::find($analisaProject->id);
        ...
  }

} else {
  if($val_rata_penyebut != 0 && $val_rata_pembilang != 0){
     ...
   } else if($val_range != 0){
     $hasil_hitung = $val_range/5;
        if($hasil_hitung > $kode_analisa['baseline_med']) {
           $hasil_analisa = "Tercapai";
        } else $hasil_analisa = "Tidak Tercapai";
           $AnalisaProject = new AnalisaProject;
           ...
           $AnalisaProject->save();
         }
    }
}
```

**Kode Sumber 4. 8 Fungsi Menghitung Jawaban (2)**

**Tabel 4. 9 Penjelasan Menghitung Jawaban (2)**

| No. Baris | Kegunaan |
|---|---|
| 1 | Mengambil data Analisa project |
| 3-5 | Jika rata pembilang serta rata penyebut pada Analisa project tidak sama dengan 0, maka hasil perhitungannya adalah rata pembilang dibagi rata penyebut. |
| 6-7 | Jika rata range tidak sama dengan 0, maka hasil perhitungan adalah rata range dibagi 5, karena skala yang digunakan menggunakan 5 nilai. |
| 9-11 | Jika hasil perhitungan lebih dari nilai batas tengah, maka hasil Analisa adalah 'tercapai', jika kurang dari nilai batas tengah maka hasil Analisa 'tidak tercapai'. |
| 12-13 | Jika data Analisa project sudah ada sebelumnya, maka sistem akan mencari data tersebut dan mengubahnya dengan data terbaru. |
| 17-18 | Jika rata pembilang dan penyebut tidak sama dengan 0, dan data Analisa project belum tersimpan pada sistem, maka sistem akan menyimpan data Analisa project baru ke dalam database. |
| 24-26 | Jika data Analisa project belum ada sebelumnya, maka sistem akan menyimpan data Analisa project baru. |

### 4.3.4. Implementasi Kasus Penggunaan Melihat Analisis Tujuan

Halaman ini merupakan halaman yang digunakan untuk melakukan kasus penggunaan UC004 yaitu melihat analisis tujuan. Seorang Tester dapat melihat analisis tujuan jika ia sudah menjawab pertanyaan yang disediakan sistem setelah memilih goal. Implementasi kasus penggunaan melihat analisis tujan dapat dilihat pada Kode Sumber 4.9 dan penjelasannya dapat dilihat pada Tabel 4.10.

```
1.  public function laporan_goal($id)
2.  {
3.      $data = [
4.          'metric' => ...,
5.          'goal' => ....,
```

```
6.          ];
7.
8.      return view('project.laporangoal',compact('data')
    );
9.  }
```

**Kode Sumber 4. 9 Fungsi Melihat Analisis Tujuan**

**Tabel 4. 10 Penjelasan Melihat Analisis Tujuan**

| No. Baris | Kegunaan |
|---|---|
| 3-6 | Array yang berisi data hasil analisis metric dan data analisis goal |
| 4 | Query untuk mengambil data hasil analisa metric berdasarkan goal yang dipilih pada saat pembuatan proyek pengukuran |
| 5 | Query untuk mengambil data proyek beserta goal yang dipilih pada proyek |
| 8 | Mengembalikan tampilan ke halaman melihat analisis tujuan dan passing data array pada baris 3-6 |

### 4.3.5. Implementasi Kasus Penggunaan Mengelola Ukuran

Halaman ini merupakan halaman yang digunakan untuk melakukan kasus penggunaan UC007 yaitu mengelola ukuran. Seorang Administrator dapat melakukan pengelolaan data ukur/metric, termasuk melihat daftar data ukur, menambah data ukur, mengubah data ukur serta melakukan penghapusan data ukuran. Implementasi kasus penggunaan melihat daftar data ukuran dapat dilihat pada Kode Sumber 4.10 dan penjelasannya dapat dilihat pada Tabel 4.11.

```
1.  public function index()
2.  {
3.          $data_metric = Metric::all();
4.
5.          return view('metric.index',['data_metric'=> $
    data_metric]);
6.  }
```

**Kode Sumber 4. 10 Fungsi Melihat Daftar Data Ukur**

**Tabel 4. 11 Penjelasan Melihat Daftar Data Ukur**

| No. Baris | Kegunaan |
|---|---|
| 3 | Mengambil seluruh data ukur yang tersimpan pada database |
| 5 | Mengembalikan tampilan ke halaman melihat daftar data ukuran/metric |

```
1.  public function detail($id)
2.  {
3.      $metric= Metric::find($id);
4.
5.      return view('metric/edit', ['metric' => $metric]);

6.  }
```

**Kode Sumber 4. 11 Fungsi Melihat Detail Data Ukur**

**Tabel 4. 12 Penjelasan Melihat Detail Data Ukur**

| No. Baris | Kegunaan |
|---|---|
| 3 | Mengambil detail data ukur yang tersimpan pada database |
| 5 | Mengembalikan tampilan ke halaman melihat detail data ukuran/metric |

Implementasi kasus penggunaan menambah data ukuran dapat dilihat pada Kode Sumber 4.12 Dan penjelasannya dapat dilihat pada Tabel 4.13.

```
1.  public function create(Request $request)
2.  {
3.      $this->validate($request,[
4.          'kode_metric' =>'required',
```

```
5.          'jenis_metric' => 'required'
6.      ]);
7.      Metric::create($request->all());
8.
9.      return redirect ('/metric')-
    >with('success','Berhasil menyimpan data metrik');
10. }
```

**Kode Sumber 4. 12 Fungsi Menambah Data Ukur**

**Tabel 4. 13 Penjelasan Menambah Data Ukur**

| No. Baris | Kegunaan |
|---|---|
| 3-6 | Memvalidasi isian untuk data ukur baru, semua *field* pada form penambahan harus terisi untuk menambah data ukur baru |
| 7 | Menyimpan data yang diinputkan ke dalam database |
| 9 | Mengembalikan tampilan ke halaman melihat daftar data ukuran/metric |

Implementasi kasus penggunaan mengubah data ukuran dapat dilihat pada Kode Sumber 4.13 Dan penjelasannya dapat dilihat pada Tabel 4.14.

```
1.  public function update(Request $request,$id)
2.  {
3.      $metric= Metric::find($id);
4.      $metric->update($request->all());
5.
6.      return back()-
    >with('success','Berhasil menyimpan data perubahan');

7.  }
```

**Kode Sumber 4. 13 Fungsi Mengubah Data Ukur**

**Tabel 4. 14 Penjelasan Mengubah Data Ukur**

| No. Baris | Kegunaan |
|---|---|
| 3 | Mengambil detail data ukur yang terpilih pada database |
| 4 | Menyimpan perubahan data ke dalam database |
| 6 | Mengembalikan tampilan ke halaman melihat detail data ukuran/metric |

Implementasi kasus penggunaan menghapus data ukuran dapat dilihat pada Kode Sumber 4.14 Dan penjelasannya dapat dilihat pada Tabel 4.15.

```
1.  public function delete($id)
2.  {
3.      $metric= Metric::find($id);
4.      $metric->delete();
5.
6.      return redirect ('/metric')-
    >with('success','Berhasil menghapus data metrik');
7.  }
```

**Kode Sumber 4. 14 Fungsi Menghapus Data Ukur**

**Tabel 4. 15 Penjelasan Menghapus Data Ukur**

| No. Baris | Kegunaan |
|---|---|
| 3 | Mengambil detail data ukur yang terpilih pada database |
| 4 | Menghapus data yang terpilih dari database |
| 6 | Mengembalikan tampilan ke halaman melihat daftar data ukuran/metric |

### 4.3.6. Implementasi Kasus Penggunaan Mengelola Pertanyaan

Halaman ini merupakan halaman yang digunakan untuk melakukan kasus penggunaan UC008 yaitu mengelola pertanyaan. Seorang Administrator dapat melakukan pengelolaan data pertanyaan, termasuk melihat daftar data pertanyaan, melihat detail pertanyaan, menambah data pertanyaan, mengubah data pertanyaan serta melakukan penghapusan data pertanyaan. Implementasi kasus

penggunaan melihat daftar pertanyaan dapat dilihat pada Kode Sumber 4. dan penjelasannya dapat dilihat pada Tabel 4..

```
1.  public function index(Request $request)
2.  {
3.      $data_question = Question::all();
4.
5.      return view('question.index',['data_qst'=> $data_
    question]);
6.  }
```

**Kode Sumber 4. 15 Fungsi Melihat Daftar Pertanyaan**

**Tabel 4. 16 Penjelasan Melihat Daftar Pertanyaan**

| No. Baris | Kegunaan |
|---|---|
| 3 | Mengambil seluruh data pertanyaan yang tersimpan pada database |
| 5 | Mengembalikan tampilan ke halaman melihat daftar data pertanyaan |

```
1.  public function detail($id)
2.  {
3.      $qst= Question::find($id);
4.      $goal_qst = GoalQuestion::all();
5.
6.      return view('question/edit', ['qst' => $qst], ['g
    oal_qst' => $goal_qst]);
7.  }
```

**Kode Sumber 4. 16 Fungsi Melihat Detail Pertanyaan**

**Tabel 4. 17 Penjelasan Melihat Detail Pertanyaan**

| No. Baris | Kegunaan |
|---|---|
| 3 | Mengambil detail data pertanyaan yang tersimpan pada database |

| | |
|---|---|
| 4 | Mengambil data relasi antara *goal* dengan *question* |
| 5 | Mengembalikan tampilan ke halaman melihat detail data pertanyaan |

Implementasi kasus penggunaan menambah data pertanyaan dapat dilihat pada Kode Sumber 4.17 Dan penjelasannya dapat dilihat pada Tabel 4.18.

```
1.  public function create(Request $request)
2.  {
3.       $this->validate($request,[
4.      'kode_qst' =>'required',
5.      'deskripsi_qst' => 'required'
6.      ]);
7.
8.       $question = new Question;
9.       $question->kode_qst = $request->kode_qst;
10.      $question->deskripsi_qst = $request-
    >deskripsi_qst;
11.      $question->save();
12.
13.
14.     return redirect ('/question')-
    >with('success','Berhasil menyimpan data pertanyaan')
    ;
15. }
```

**Kode Sumber 4. 17 Fungsi Menambah Data Pertanyaan**

**Tabel 4. 18 Penjelasan Menambah Data Pertanyaan**

| No. Baris | Kegunaan |
|---|---|
| 3-6 | Memvalidasi isian untuk data pertanyaan baru, semua *field* pada form penambahan harus terisi untuk menambah data pertanyaan baru |
| 8-11 | Menambahkan data pertanyaaan baru dan menyimpan data yang diinputkan ke dalam database |
| 14 | Mengembalikan tampilan ke halaman melihat daftar data pertanyaan |

Implementasi kasus penggunaan mengubah data pertanyaan dapat dilihat pada Kode Sumber 4.18 Dan penjelasannya dapat dilihat pada Tabel 4.19.

```
1.  public function update(Request $request,$id)
2.  {
3.      $qst= Question::find($id);
4.      $qst->update($request->all());
5.
6.      return back()-
    >with('success','Data berhasil diupdate');
7.  }
```

**Kode Sumber 4. 18 Fungsi Mengubah Pertanyaan**

**Tabel 4. 19 Penjelasan Mengubah Pertanyaan**

| No. Baris | Kegunaan |
|---|---|
| 3 | Mengambil detail data pertanyaan yang terpilih pada database |
| 4 | Menyimpan perubahan data ke dalam database |
| 6 | Mengembalikan tampilan ke halaman melihat detail data pertanyaan |

Implementasi kasus penggunaan menghapus data pertanyaan dapat dilihat pada Kode Sumber 4.19 Dan penjelasannya dapat dilihat pada Tabel 4.20.

```
1.  public function delete($id)
2.  {
3.      $qst= Question::find($id);
4.      $qst->delete();
5.
6.      return redirect ('/question')-
    >with('success','Berhasil menghapus data pertanyaan')
    ;
7.  }
```

**Kode Sumber 4. 19 Fungsi Menghapus Data Pertanyaan**

**Tabel 4. 20 Penjelasan Menghapus Data Pertanyaan**

| No. Baris | Kegunaan |
|---|---|
| 3 | Mengambil detail data pertanyaan yang terpilih pada database |
| 4 | Menghapus data yang terpilih dari database |
| 6 | Mengembalikan tampilan ke halaman melihat daftar data pertanyaan |

### 4.3.7. Implementasi Kasus Penggunaan Mengelola Data Proyek

Halaman ini merupakan halaman yang digunakan untuk melakukan kasus penggunaan UC009 yaitu mengelola data proyek. Seorang Administrator dapat melakukan pengelolaan data proyek, termasuk melihat daftar data proyek, melihat detail proyek, mengubah data proyek serta melakukan penghapusan data proyek. Implementasi kasus penggunaan melihat daftar data proyek dapat dilihat pada Kode Sumber 4.20 dan penjelasannya dapat dilihat pada Tabel 4.21.

```
1.  public function project(Request $request)
2.  {
3.      $data_project= \App\Project::all();
4.
5.      return view('project.index',['data_project'=> $da
    ta_project]);
6.  }
```

**Kode Sumber 4. 20 Fungsi Melihat Daftar Data Proyek**

**Tabel 4. 21 Penjelasan Melihat Daftar Data Proyek**

| No. Baris | Kegunaan |
|---|---|

| | |
|---|---|
| 3 | Mengambil seluruh data proyek yang tersimpan pada database |
| 5 | Mengembalikan tampilan ke halaman melihat daftar data proyek |

```
1.  public function detail($id)
2.  {
3.      $project= \App\Project::find($id);
4.
5.      return view('project/edit', ['project' => $projec
    t]);
6.  }
```

**Kode Sumber 4. 21 Fungsi Melihat Detail Data Proyek**

**Tabel 4. 22 Penjelasan Melihat Detail Data Proyek**

| No. Baris | Kegunaan |
|---|---|
| 3 | Mengambil detail data proyek yang tersimpan pada database |
| 5 | Mengembalikan tampilan ke halaman melihat detail data proyek |

Implementasi kasus penggunaan mengubah data proyek dapat dilihat pada Kode Sumber 4.22 Dan penjelasannya dapat dilihat pada Tabel 4.23.

```
1.  public function update(Request $request,$id)
2.  {
3.      $project= \App\Project::find($id);
4.      $project->update($request->all());
5.
6.      return back()-
    >with('success','Berhasil menyimpan data perubahan');

7.  }
```

**Kode Sumber 4. 22 Fungsi Mengubah Data Proyek**

**Tabel 4. 23 Penjelasan Mengubah Data Proyek**

| No. Baris | Kegunaan |
|---|---|
| 3 | Mengambil detail data proyek yang terpilih pada database |
| 4 | Menyimpan perubahan data ke dalam database |
| 6 | Mengembalikan tampilan ke halaman melihat detail data proyek |

Implementasi kasus penggunaan menghapus data proyek dapat dilihat pada Kode Sumber 4.23 Dan penjelasannya dapat dilihat pada Tabel 4.24.

```
1.  public function delete($id)
2.  {
3.      $project= \App\Project::find($id);
4.      $project->delete();
5.
6.      return redirect ('/project')-
    >with('success','Berhasil menghapus data proyek');
7.  }
```

**Kode Sumber 4. 23 Fungsi Menghapus Data Proyek**

**Tabel 4. 24 Penjelasan Menghapus Data Proyek**

| No. Baris | Kegunaan |
|---|---|
| 3 | Mengambil detail data proyek yang terpilih pada database |
| 4 | Menghapus data yang terpilih dari database |
| 6 | Mengembalikan tampilan ke halaman melihat daftar data proyek |

***[Halaman ini sengaja dikosongkan]***

# BAB V
# UJI COBA DAN EVALUASI

Bab ini membahas uji coba dan evaluasi terhadap perangkat lunak yang telah dikembangkan dari implementasi kasus penggunaan.

## 5.1. Lingkungan Uji Coba

Pada proses pengujian perangkat lunak, lingkungan uji coba meliputi lingkungan perangkat keras dan perangkat lunak yang digunakan untuk melakukan uji coba. Lingkungan pengujian dalam Tugas Akhir ini dilakukan pada setiap kasus penggunaan. Spesifikasi masing-masing lingkungan pengujian dijelaskan secara rinci pada Tabel 5.1.

**Tabel 5. 1 Lingkungan Uji Coba**

| Perangkat | Spesifikasi | Deskripsi |
|---|---|---|
| Keras | CPU | Intel(R) Core(TM) i5-8265U CPU @ 1.60GHz 1.80 GHz |
| | RAM | 4.00 GB |
| Lunak | Sistem Operasi | Windows 10 Home Single Language 64-bit |
| | Browser | Google Chrome Version 83.0.4103.106 (Official Build) (64-bit) |
| | Web Server | Apache |
| | Server Basis Data | MySql |

## 5.2. Material Pengujian

Pada subbab material pengujian dijelaskan mengenai materi yang hendak diuji. Materi yang diujikan pada tahap pengujian aplikasi mencakup materi fungsional. Adapun poin-poin yang termasuk dalam kebutuhan fungsional disajikan pada Tabel 5.2.

**Tabel 5. 2 Materi Pengujian Fungsional**

| Aktor | Kode | Deskripsi |
|---|---|---|
| Software Tester | F001 | Mendaftarkan akun |
| | F002 | Membuat proyek pengukuran |
| | F003 | Melihat analisis hasil ukur |
| | F004 | Melihat analisis tujuan |
| Administrator | F007 | Mengelola ukuran |
| | F008 | Mengelola pertanyaan |
| | F009 | Mengelola data proyek |

## 5.3. Skenario Pengujian Kasus Penggunaan

Pada bagian ini akan dibahas mengenai proses uji coba pada kasus penggunaan yang digunakan. Pengujian dilakukan dengan metode *black box* untuk menguji masing-masing fungsionalitas yang sudah dirancang pada sistem. Metode *black box* merupakan metode pengujian perangkat lunak yang memeriksa fungsionalitas dari suatu perangkat lunak tanpa memandang struktur internalnya.

Pada proses uji coba, pengujian dilakukan dengan menjalankan serangkaian perintah terhadap sistem yang selanjutnya akan disebut sebagai kasus pengujian. Kasus pengujian ini berkorelasi dengan kasus-kasus penggunaan dan kebutuhan fungsional yang sebelumnya sudah dirancang dan dijelaskan pada Bab III.

### 5.3.1. Pengujian Mendaftarkan Akun

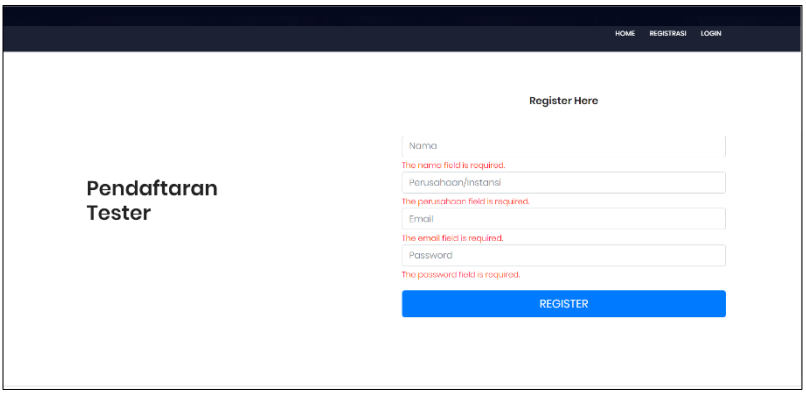
Pengujian mendaftarkan akun merupakan pengujian terhadap kemampuan aplikasi untuk mendaftarkan/menambahkan pengguna sebagai Software Tester. Rincian kasus pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.3. Gambar contoh scenario dapat dilihat pada Gambar 5.1, Gambar 5.2, Gambar 5.3 dam Gambar 5.4.

**Tabel 5. 3 Pengujian Mendaftarkan Akun**

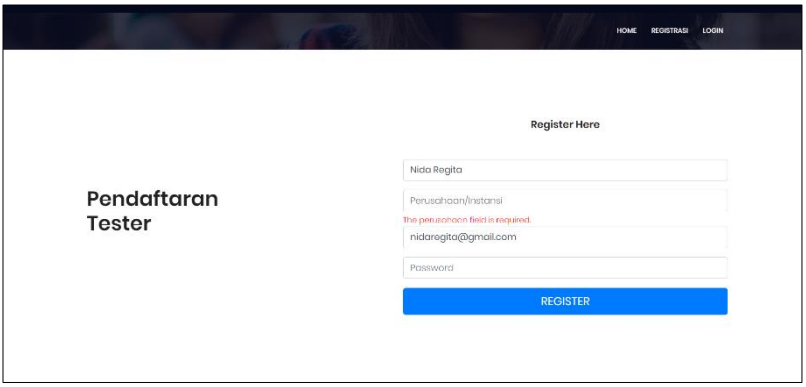
| | |
|---|---|
| **ID Pengujian** | UJ001 |

| | |
|---|---|
| **Nama** | Pengujian mendaftakan akun |
| **Tujuan Pengujian** | Menguji apakah sistem sudah mampu menambahkan data pengguna baru sebagai Software Tester |
| **Kondisi Awal** | Pengguna belum terdaftar sebagai Software Tester |
| **Prosedur Pengujian** | 1. Aktor memilih menu Registrasi<br>2. Masukkan data uji<br>3. Klik tombol “Submit” |
| **Skenario 1** | |
| **Masukan** | Data kosong |
| **Data Uji** | - |
| **Hasil yang diharapkan** | Pesan kesalahan data kosong muncul |
| **Hasil yang diperoleh** | Sistem menampilkan pesan bahwa semua data harus diisi |
| **Hasil Pengujian** | Berhasil |
| **Skenario 2** | |
| **Masukan** | Data tidak lengkap |
| **Data Uji** | Data Test-A01 |
| **Hasil yang diharapkan** | Pesan kesalahan data tidak lengkap muncul |
| **Hasil yang diperoleh** | Sistem menampilkan pesan data harus diisi pada kolom yang tidak terisi |
| **Hasil Pengujian** | Berhasil |
| **Skenario 3** | |
| **Masukan** | Data lengkap |
| **Data Uji** | Data Test-A02 |
| **Hasil yang diharapkan** | Pesan berhasil menambahkan data pengguna |
| **Hasil yang diperoleh** | Sistem menampilkan pesan bahwa pengguna berhasil mendaftarkan diri sebagai Tester |
| **Hasil Pengujian** | Berhasil |
| **Skenario 4** | |
| **Masukan** | Data salah |
| **Data Uji** | Data Test-A03 |
| **Hasil yang diharapkan** | Pesan kesalahan format pengisian data salah |

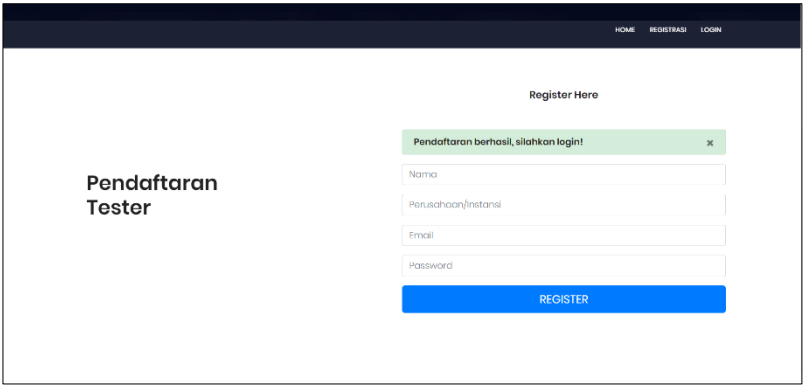
| | |
|---|---|
| **Hasil yang diperoleh** | Sistem menampilkan pesan kesalahan format dan sugesti perbaikan masukan |
| **Hasil Pengujian** | Berhasil |



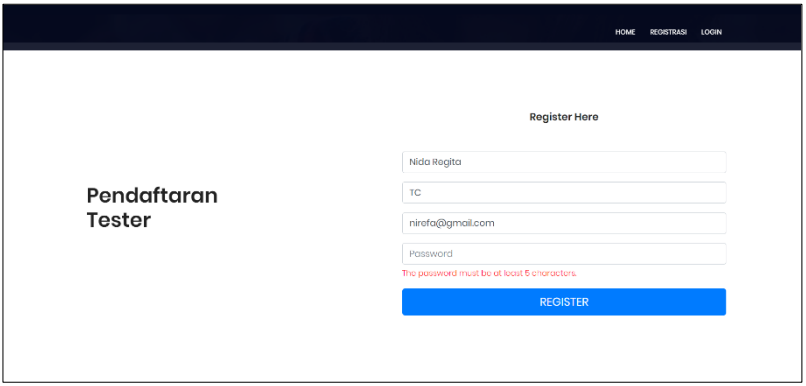

**Gambar 5. 1 Mendaftarkan Akun : Skenario 1**




**Gambar 5. 2 Mendaftarkan Akun : Skenario 2**




**Gambar 5. 3 Mendaftarkan Akun : Skenario 3**



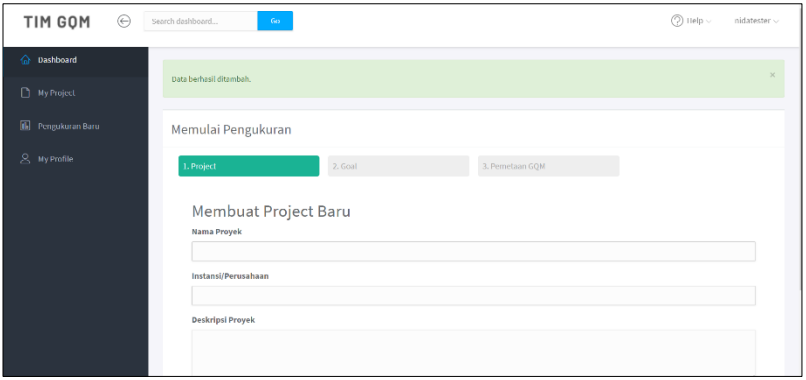

**Gambar 5. 4 Mendaftarkan Akun : Skenario 4**

### 5.3.2. Pengujian Membuat Proyek Pengukuran

Pengujian membuat proyek pengukuran merupakan pengujian terhadap kemampuan aplikasi untuk menerima masukan berupa pembuatan proyek pengukuran baru dan pemilihan goal pengukuran. Rincian kasus pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.4. Gambar contoh skenario dapat dilihat pada Gambar 5.5.

**Tabel 5. 4 Pengujian Membuat Proyek Pengukuran**

| | |
|---|---|
| **ID Pengujian** | UJ002 |
| **Nama** | Pengujian membuat proyek pengukuran |
| **Tujuan Pengujian** | Menguji apakah sistem sudah mampu menambahkan data proyek pengukuran baru pada sistem |
| **Kondisi Awal** | Pengguna telah login sebagai Software Tester |
| **Prosedur Pengujian** | 1. Aktor memilih objek pengukuran produk<br>2. Aktor mengisi form pembuatan proyek pengukuran<br>3. Memilih goal yang akan diukur<br>4. Melihat pemetaan GQM<br>5. Menyimpan data pengukuran baru |
| **Skenario 1** | |
| **Masukan** | Data lengkap |
| **Data Uji** | Data Test-B01 |
| **Hasil yang diharapkan** | Pesan berhasil menambahkan data proyek pengukuran baru |
| **Hasil yang diperoleh** | Sistem menampilkan pesan data proyek pengukuran berhasil ditambahkan |
| **Hasil Pengujian** | Berhasil |



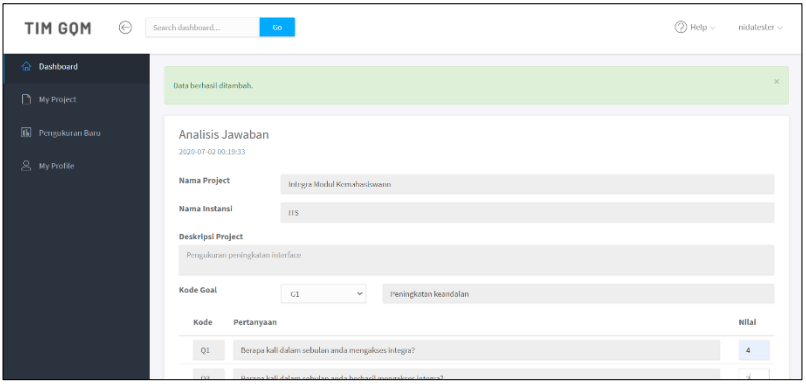
**Gambar 5. 5 Membuat Proyek Pengukuran : Skenario 1**

### 5.3.3. Pengujian Menjawab Pertanyaan

Pengujian menjawab pertanyaan merupakan pengujian terhadap kemampuan aplikasi untuk menerima masukan berupa jawaban dari pertanyaan yang telah dipilih saat pembuatan proyek pengukuran baru. Rincian kasus pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.5. Gambar contoh scenario dapat dilihat pada Gambar 5.6.

**Tabel 5. 5 Pengujian Menjawab Pertanyaan**

| | |
|---|---|
| **ID Pengujian** | UJ003 |
| **Nama** | Pengujian menjawab pertanyaan |
| **Tujuan Pengujian** | Menguji apakah sistem sudah mampu menambahkan data jawaban pertanyaan pada sistem |
| **Kondisi Awal** | Pengguna telah login sebagai Software Tester |
| **Prosedur Pengujian** | 1. Aktor memilih proyek yang akan diukur pada halaman My Project<br>2. Memilih goal<br>3. Menjawab pertanyaan yang ditampilkan sesuai goal yang terpilih<br>4. Memilih tombol ‘Submit’ |
| **Skenario 1** | |
| **Masukan** | Data lengkap |
| **Data Uji** | Data Test-C01 |
| **Hasil yang diharapkan** | Pesan berhasil menambahkan data jawaban pertanyaan |
| **Hasil yang diperoleh** | Sistem menampilkan pesan data jawaban pertanyaan berhasil ditambahkan |
| **Hasil Pengujian** | Berhasil |




**Gambar 5. 6 Menjawab Pertanyaan : Skenario 1**

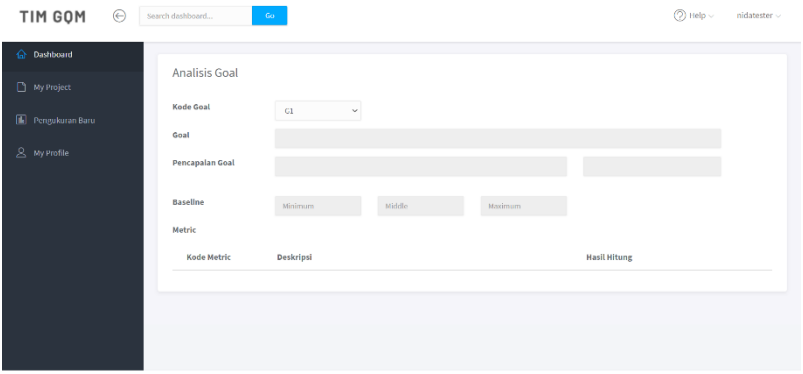
### 5.3.4. Pengujian Melihat Analisis Tujuan

Pengujian melihat analisis tujuan merupakan pengujian terhadap kemampuan aplikasi untuk melihat laporan analisis tujuan (goal) secara umum. Rincian kasus pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.6. Gambar contoh skenario dapat dilihat pada Gambar 5.7 dan 5.8.

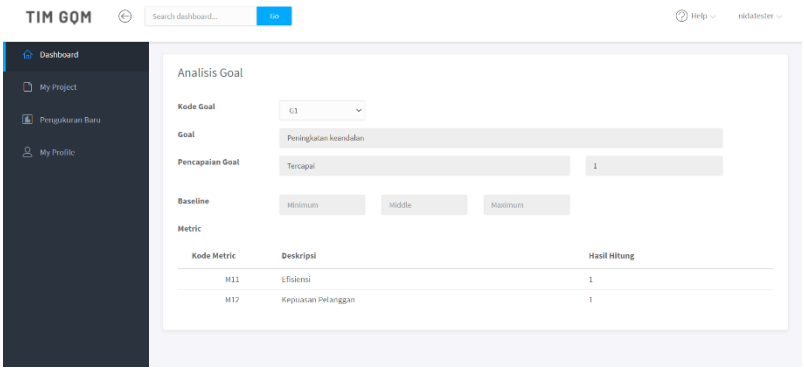
**Tabel 5. 6 Pengujian Melihat Analisis Tujuan**

| | |
|---|---|
| **ID Pengujian** | UJ004 |
| **Nama** | Pengujian melihat analisis tujuan |
| **Tujuan Pengujian** | Menguji apakah sistem sudah mampu menampilkan laporan analisis tujuan |
| **Kondisi Awal** | • Pengguna telah login sebagai Software Tester<br>• Pengguna telah membuat proyek pengukuran |
| **Prosedur Pengujian** | 1. Aktor memilih proyek yang hendak dilihat laporan analisis tujuannya<br>2. Pilih lihat detail proyek |

| Skenario 1 | |
|---|---|
| **Masukan** | Data jawaban belum terisi |
| **Data Uji** | - |
| **Hasil yang diharapkan** | Sistem tidak dapat menampilkan data tujuan |
| **Hasil yang diperoleh** | Sistem menampilkan data kosong |
| **Hasil Pengujian** | Berhasil |
| **Skenario 2** | |
| **Masukan** | Data jawaban terisi |
| **Data Uji** | Data Test-D01 |
| **Hasil yang diharapkan** | Sistem menampilkan data analisis tujuan |
| **Hasil yang diperoleh** | Sistem menampilkan data analisis tujuan |
| **Hasil Pengujian** | Berhasil |




**Gambar 5. 7 Melihat Analisis Tujuan : Skenario 1**




**Gambar 5. 8 Melihat Analisis Tujuan : Skenario 2**

### 5.3.5. Pengujian Mengelola Ukuran

Pengujian mengelola ukuran merupakan pengujian terhadap kemampuan aplikasi untuk mengelola data ukur. Mengelola data ukur meliputi menambah, mengubah dan menghapus data ukur. Rincian kasus pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.8, Tabel 5.9 dan Tabel 5.10. Gambar contoh skenario dapat dilihat pada Gambar 5.9, Gambar 5.10, Gambar 5.11 dan Gambar 5.12.

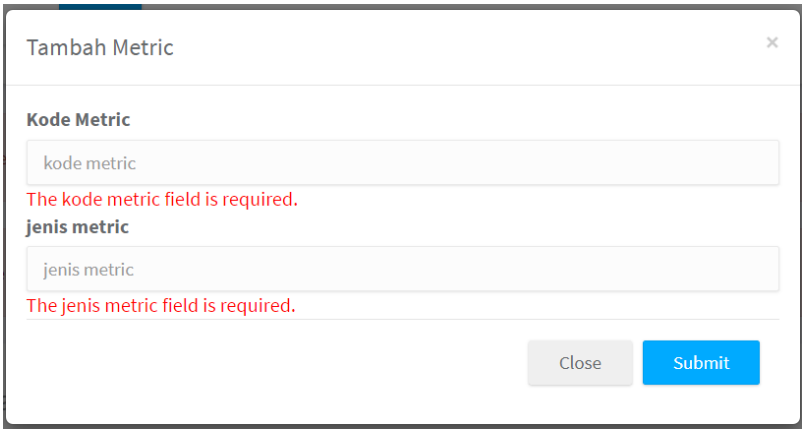
#### 5.3.5.1. Pengujian Menambah Data Ukur

Pengujian menambah data ukur merupakan pengujian terhadap kemampuan aplikasi untuk menambahkan data ukur/metric baru. Rincian pengujian dapat dilihat pada tabel 5.7.

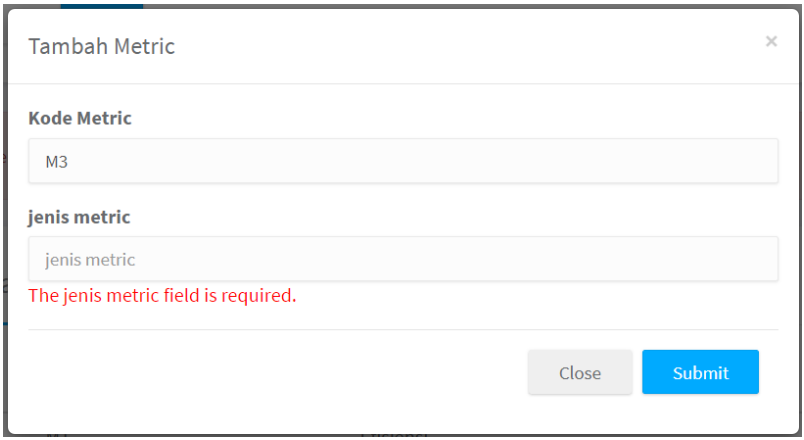
**Tabel 5. 7 Pengujian Menambah Data Ukur**

| | |
|---|---|
| **ID Pengujian** | UJ006 |
| **Nama** | Pengujian menambah data ukur |
| **Tujuan Pengujian** | Menguji apakah sistem sudah mampu melakukan penambahan data ukur (metric) |
| **Kondisi Awal** | Pengguna telah login sebagai Admin |

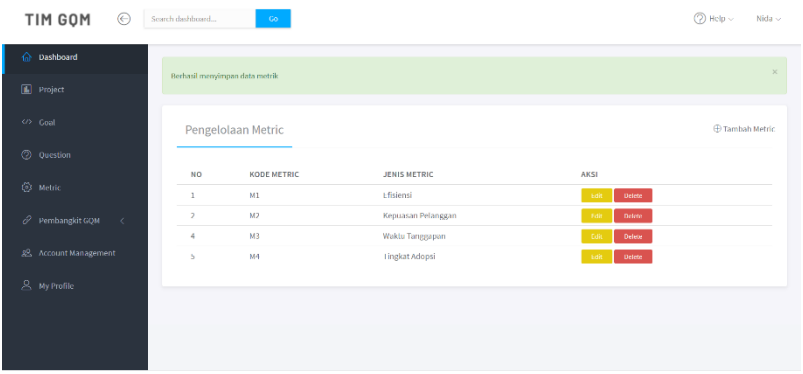
| | |
|---|---|
| **Prosedur Pengujian** | 1. Aktor memilih menu ‘Metric’<br>2. Aktor memilih opsi ‘Tambah Metric’<br>3. Aktor mengisi form<br>4. Klik tombol ‘Submit’ |
| **Skenario 1** | |
| **Masukan** | Data kosong |
| **Data Uji** | - |
| **Hasil yang diharapkan** | Pesan kesalahan data kosong muncul |
| **Hasil yang diperoleh** | Sistem menampilkan pesan bahwa semua data harus diisi |
| **Hasil Pengujian** | Berhasil |
| **Skenario 2** | |
| **Masukan** | Data tidak lengkap |
| **Data Uji** | Data Test-E01 |
| **Hasil yang diharapkan** | Pesan kesalahan data tidak lengkap muncul |
| **Hasil yang diperoleh** | Sistem menampilkan pesan bahwa data wajib diisi |
| **Hasil Pengujian** | Berhasil |
| **Skenario 3** | |
| **Masukan** | Data lengkap |
| **Data Uji** | Data Test-E02 |
| **Hasil yang diharapkan** | Pesan data ukuran berhasil disimpan |
| **Hasil yang diperoleh** | Sistem menampilkan pesan data ukuran berhasil disimpan |
| **Hasil Pengujian** | Berhasil |




**Gambar 5. 9 Menambah Data Ukur : Skenario 1**




**Gambar 5. 10 Menambah Data Ukur : Skenario 2**




**Gambar 5. 11 Menambah Data Ukur : Skenario 3**

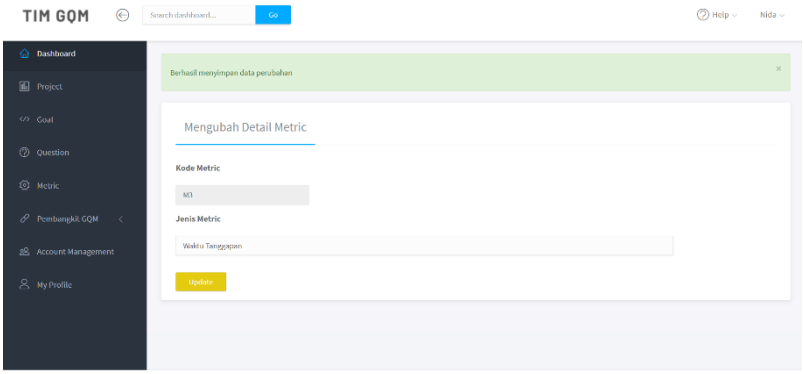
### 5.3.5.2. Pengujian Mengubah Data Ukur

Pengujian mengubah data ukur merupakan pengujian terhadap kemampuan aplikasi untuk mengubah data ukur/metric yang telah tersimpan pada sistem. Rincian pengujian dapat dilihat pada tabel 5.8.

**Tabel 5. 8 Pengujian Mengubah Data Ukur**

| | |
|---|---|
| **ID Pengujian** | UJ007 |
| **Nama** | Pengujian mengubah data ukur |
| **Tujuan Pengujian** | Menguji apakah sistem sudah mampu melakukan pengelolaan ukuran untuk mengubah data ukur (metric) |
| **Kondisi Awal** | • Pengguna telah login sebagai Admin<br>• Data ukur telah tersimpan pada sistem |
| **Prosedur Pengujian** | 1. Aktor melilih menu ‘Metric’<br>2. Aktor memilih opsi data ukur yang akan diubah<br>3. Klik tombol ‘Edit’<br>4. Aktor mengubah detil ukuran dengan mengisi form perubahan |

| | |
|---|---|
| | 5. Klik tombol ‘Submit’ |
| **Skenario 1** | |
| **Masukan** | Data perubahan |
| **Data Uji** | Data Test-E03 |
| **Hasil yang diharapkan** | Pesan data berhasil diubah |
| **Hasil yang diperoleh** | Sistem menampilkan pesan berhasil menyimpan data perubahan |
| **Hasil Pengujian** | Berhasil |




**Gambar 5. 12 Mengubah Data Ukur : Skenario 1**

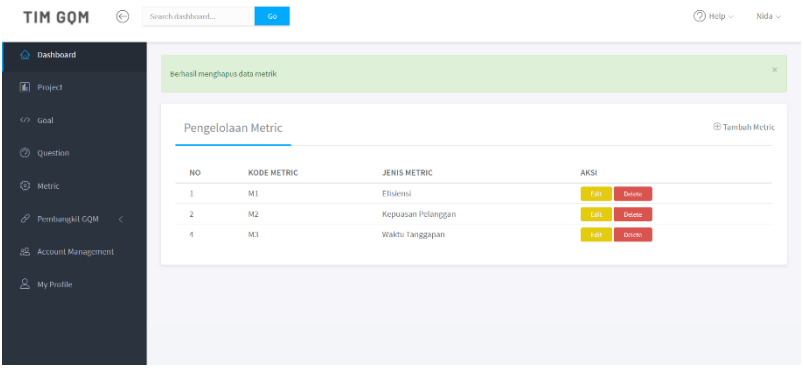
### 5.3.5.3. Pengujian Menghapus Data Ukur

Pengujian menghapus data ukur merupakan pengujian terhadap kemampuan aplikasi untuk menghapus data ukur/metric yang telah tersimpan pada sistem. Rincian pengujian dapat dilihat pada tabel 5.9.

**Tabel 5. 9 Pengujian Menghapus Data Ukur**

| | |
|---|---|
| **ID Pengujian** | UJ008 |
| **Nama** | Pengujian menghapus data ukur |

| | |
|---|---|
| **Tujuan Pengujian** | Menguji apakah sistem sudah mampu melakukan pengelolaan ukuran untuk menghapus data ukur (metric) |
| **Kondisi Awal** | • Pengguna telah login sebagai Admin<br>• Data ukur telah tersimpan pada sistem |
| **Prosedur Pengujian** | 1. Aktor melilih menu ‘Metric’<br>2. Aktor memilih opsi data ukur yang akan dihapus<br>3. Klik tombol ‘Hapus’ |
| **Skenario 1** | |
| **Masukan** | Aktor mengonfirmasi penghapusan dengan menekan tombol ‘Ok’ |
| **Data Uji** | - |
| **Hasil yang diharapkan** | Pesan berhasil menghapus data muncul |
| **Hasil yang diperoleh** | Sistem menampilkan data metrik berhasil dihapus |
| **Hasil Pengujian** | Berhasil |




**Gambar 5. 13 Menghapus Data Ukur: Skenario 1**

### 5.3.6. Pengujian Mengelola Pertanyaan

Pengujian mengelola pertanyaan merupakan pengujian terhadap kemampuan aplikasi untuk mengelola data pertanyaan. Mengelola data pertanyaan meliputi menambah, mengubah dan menghapus data pertanyaan. Rincian kasus pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.10. Gambar contoh scenario dapat dilihat pada Gambar 5.14, Gambar 5.15, Gambar 5.16, Gambar 5.17 dan Gambar 5.18.

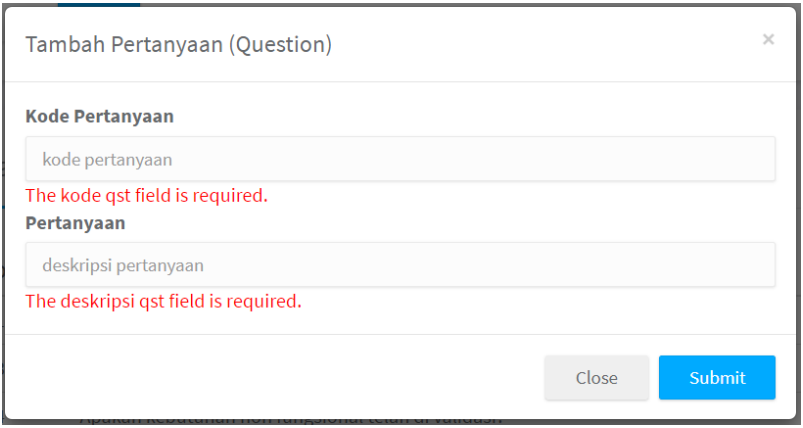
#### 5.3.6.1. Pengujian Menambah Data Pertanyaan

Pengujian menambah data tujuan merupakan pengujian terhadap kemampuan aplikasi untuk menambahkan data pertanyaan baru. Rincian pengujian dapat dilihat pada tabel 5.10.

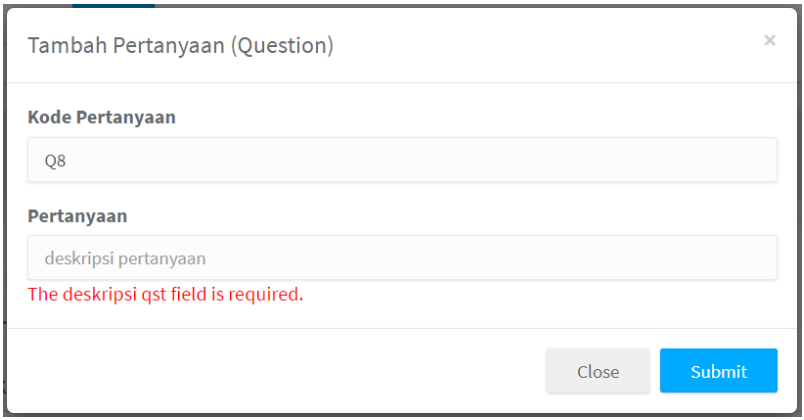
**Tabel 5. 10 Pengujian Menambah Data Pertanyaan**

| | |
|---|---|
| **ID Pengujian** | UJ012 |
| **Nama** | Pengujian menambah data pertanyaan |
| **Tujuan Pengujian** | Menguji apakah sistem sudah mampu melakukan pengelolaan pertanyaan untuk menambah data pertanyaan (question) |
| **Kondisi Awal** | • Pengguna telah login sebagai Admin<br>• Data pertanyaan belum tersimpan pada sistem |
| **Prosedur Pengujian** | 1. Aktor melilih menu ‘Question’<br>2. Aktor memilih opsi ‘Tambah Pertanyaan ’<br>3. Aktor mengisi form<br>4. Klik tombol ‘Submit’ |
| **Skenario 1** | |
| **Masukan** | Data kosong |
| **Data Uji** | - |
| **Hasil yang diharapkan** | Pesan kesalahan data kosong muncul |
| **Hasil yang diperoleh** | Sistem menampilkan pesan bahwa semua data harus diisi |

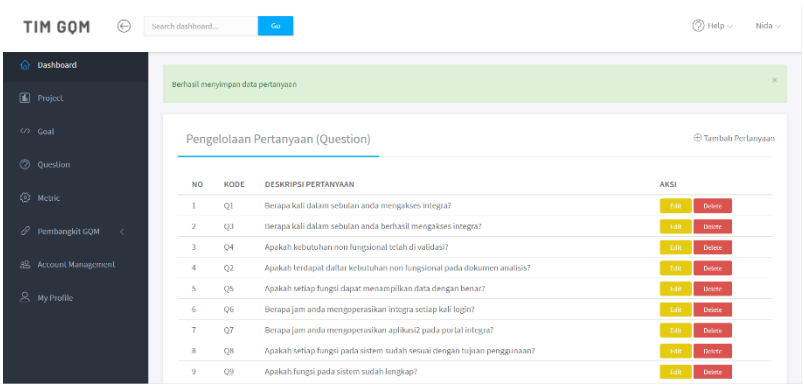
| | |
|---|---|
| **Hasil Pengujian** | Berhasil |
| **Skenario 2** | |
| **Masukan** | Data tidak lengkap |
| **Data Uji** | Data Test-F01 |
| **Hasil yang diharapkan** | Pesan kesalahan data tidak lengkap muncul |
| **Hasil yang diperoleh** | Sistem menampilkan pesan bahwa data wajib diisi |
| **Hasil Pengujian** | Berhasil |
| **Skenario 3** | |
| **Masukan** | Data lengkap |
| **Data Uji** | Data Test-F02 |
| **Hasil yang diharapkan** | Pesan data ukuran berhasil disimpan |
| **Hasil yang diperoleh** | Sistem menampilkan pesan data pertanyaan berhasil disimpan |
| **Hasil Pengujian** | Berhasil |




**Gambar 5. 14 Menambah Data Pertanyaan: Skenario 1**




**Gambar 5. 15 Menambah Data Pertanyaan : Skenario 2**



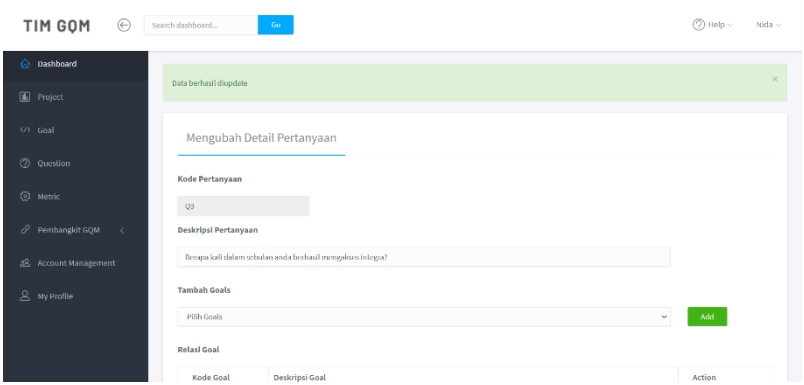

**Gambar 5. 16 Menambah Data Pertanyaan : Skenario 3**

### 5.3.6.2. Pengujian Mengubah Data Pertanyaan

Pengujian mengubah data pertanyaan merupakan pengujian terhadap kemampuan aplikasi untuk mengubah data pertanyaan yang telah tersimpan pada sistem. Rincian pengujian dapat dilihat pada tabel 5.12.

**Tabel 5. 11 Pengujian Mengubah Data Pertanyaan**

| | |
|---|---|
| **ID Pengujian** | UJ013 |
| **Nama** | Pengujian mengubah data pertanyaan |
| **Tujuan Pengujian** | Menguji apakah sistem sudah mampu melakukan pengelolaan data pertanyaan untuk mengubah data pertanyaan |
| **Kondisi Awal** | • Pengguna telah login sebagai Admin<br>• Data pertanyaan telah tersimpan pada sistem |
| **Prosedur Pengujian** | 1. Aktor melilih menu ‘Question’<br>2. Aktor memilih opsi data pertanyaan yang akan diubah<br>3. Klik tombol ‘Edit’<br>4. Aktor mengubah detil pertanyaandengan mengisi form perubahan<br>5. Klik tombol ‘Submit’ |
| **Skenario 1** | |
| **Masukan** | Data perubahan |
| **Data Uji** | Data Test-F03 |
| **Hasil yang diharapkan** | Pesan data berhasil diubah |
| **Hasil yang diperoleh** | Sistem menampilkan pesan berhasil menyimpan data perubahan |
| **Hasil Pengujian** | Berhasil |




**Gambar 5. 17 Mengubah Data Pertanyaan : Skenario 1**

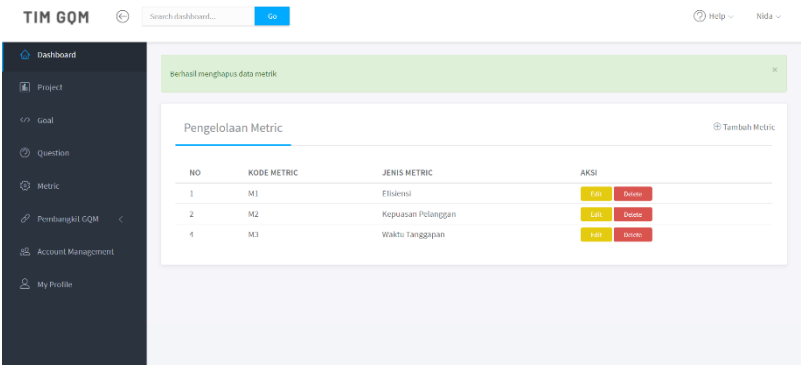
### 5.3.6.3. Pengujian Menghapus Data Pertanyaan

Pengujian menghapus data pertanyaan merupakan pengujian terhadap kemampuan aplikasi untuk menghapus data pertanyaan yang telah tersimpan pada sistem. Rincian pengujian dapat dilihat pada tabel 5.12.

**Tabel 5. 12 Pengujian Menghapus Data Tujuan**

| | |
|---|---|
| **ID Pengujian** | UJ014 |
| **Nama** | Pengujian menghapus data pertanyaan |
| **Tujuan Pengujian** | Menguji apakah sistem sudah mampu melakukan pengelolaan data pertanyaan untuk menghapus data pertanyaan |
| **Kondisi Awal** | • Pengguna telah login sebagai Admin<br>• Data pertanyaan telah tersimpan pada sistem |
| **Prosedur Pengujian** | 1. Aktor melilih menu ‘Question’<br>2. Aktor memilih opsi data pertanyaan yang akan dihapus<br>3. Klik tombol ‘Hapus’ |

| Skenario 1 | |
|---|---|
| **Masukan** | Aktor mengonfirmasi penghapusan dengan menekan tombol ‘Ok’ |
| **Data Uji** | - |
| **Hasil yang diharapkan** | Pesan berhasil menghapus data muncul |
| **Hasil yang diperoleh** | Sistem menampilkan data pertanyaan berhasil dihapus |
| **Hasil Pengujian** | Berhasil |




**Gambar 5. 18 Menghapus Data Pertanyaan: Skenario 1**

### 5.3.7. Pengujian Mengelola Data Proyek

Pengujian mengelola data proyek merupakan pengujian terhadap kemampuan aplikasi untuk mengelola data proyek. Mengelola data proyek meliputi menambah, mengubah dan menghapus data proyek. Rincian kasus pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.13. Gambar contoh skenario dapat dilihat pada Gambar 5.19.

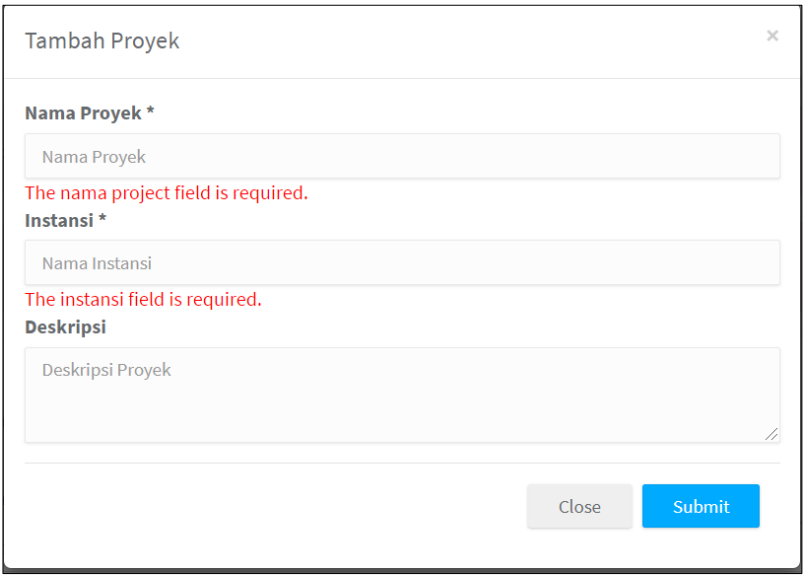
#### 5.3.7.1. Pengujian Menambah Data Proyek

Pengujian menambah data proyek merupakan pengujian terhadap kemampuan aplikasi untuk menambahkan data proyek baru. Rincian pengujian dapat dilihat pada tabel 5.11.

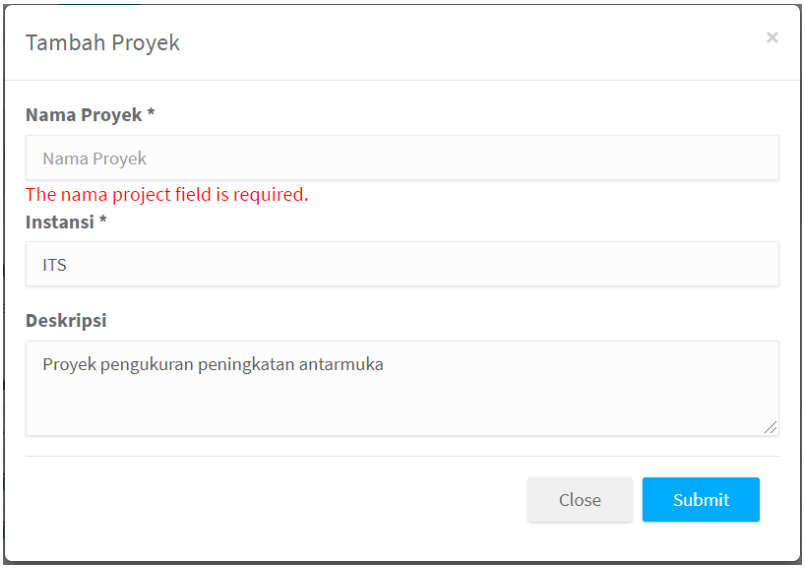
**Tabel 5. 13 Pengujian Menambah Data Proyek**

| | |
|---|---|
| **ID Pengujian** | UJ018 |
| **Nama** | Pengujian menambah data proyek |
| **Tujuan Pengujian** | Menguji apakah sistem sudah mampu melakukan pengelolaan data proyek untuk menambah data proyek baru |
| **Kondisi Awal** | • Pengguna telah login sebagai Admin<br>• Data proyek belum tersimpan pada sistem |
| **Prosedur Pengujian** | 1. Aktor melilih menu ‘Project’<br>2. Aktor memilih opsi ‘Tambah Proyek’<br>3. Aktor mengisi form<br>4. Klik tombol ‘Submit’ |
| **Skenario 1** | |
| **Masukan** | Data kosong |
| **Data Uji** | - |
| **Hasil yang diharapkan** | Pesan kesalahan data kosong muncul |
| **Hasil yang diperoleh** | Sistem menampilkan pesan bahwa semua data harus diisi |
| **Hasil Pengujian** | Berhasil |
| **Skenario 2** | |
| **Masukan** | Data tidak lengkap |
| **Data Uji** | Data Test-G01 |
| **Hasil yang diharapkan** | Pesan kesalahan data tidak lengkap muncul |
| **Hasil yang diperoleh** | Sistem menampilkan pesan bahwa data wajib diisi |
| **Hasil Pengujian** | Berhasil |
| **Skenario 3** | |

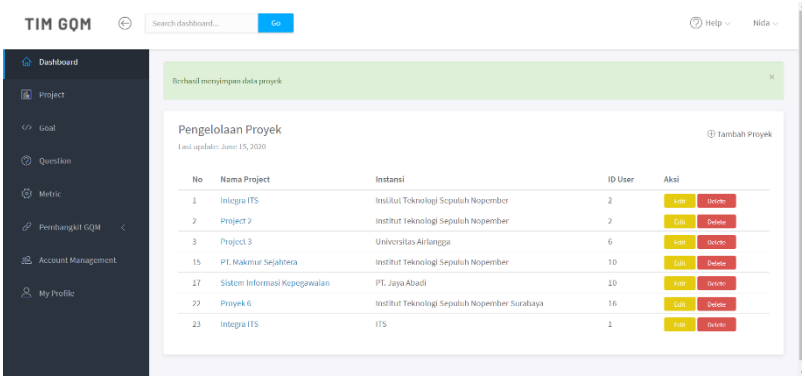
| | |
|---|---|
| **Masukan** | Data lengkap |
| **Data Uji** | Data Test-G02 |
| **Hasil yang diharapkan** | Pesan data ukuran berhasil disimpan |
| **Hasil yang diperoleh** | Sistem menampilkan pesan data proyek berhasil disimpan |
| **Hasil Pengujian** | Berhasil |




**Gambar 5. 19 Menambah Data Proyek: Skenario 1**




**Gambar 5. 20 Menambah Data Proyek : Skenario 2**



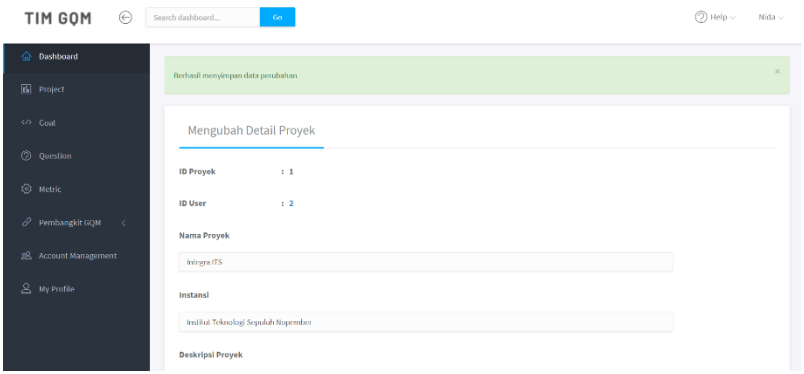

**Gambar 5. 21 Menambah Data Proyek : Skenario 3**

### 5.3.7.2. Pengujian Mengubah Data Proyek

Pengujian mengubah data proyek merupakan pengujian terhadap kemampuan aplikasi untuk mengubah data proyek yang telah tersimpan pada sistem. Rincian pengujian dapat dilihat pada Tabel 5.14.

**Tabel 5. 14 Pengujian Mengubah Data Proyek**

| | |
|---|---|
| **ID Pengujian** | UJ019 |
| **Nama** | Pengujian mengubah data proyek |
| **Tujuan Pengujian** | Menguji apakah sistem sudah mampu melakukan pengelolaan data proyek untuk mengubah data proyek |
| **Kondisi Awal** | • Pengguna telah login sebagai Admin<br>• Data proyek telah tersimpan pada sistem |
| **Prosedur Pengujian** | 1. Aktor melilih menu ‘Project’<br>2. Aktor memilih opsi data proyek yang akan diubah<br>3. Klik tombol ‘Edit’<br>4. Aktor mengubah detil proyek dengan mengisi form perubahan<br>5. Klik tombol ‘Submit’ |
| **Skenario 1** | |
| **Masukan** | Data perubahan |
| **Data Uji** | Data Test-G03 |
| **Hasil yang diharapkan** | Pesan data berhasil diubah |
| **Hasil yang diperoleh** | Sistem menampilkan pesan berhasil menyimpan data perubahan |
| **Hasil Pengujian** | Berhasil |




**Gambar 5. 22 Mengubah Data Proyek : Skenario 1**

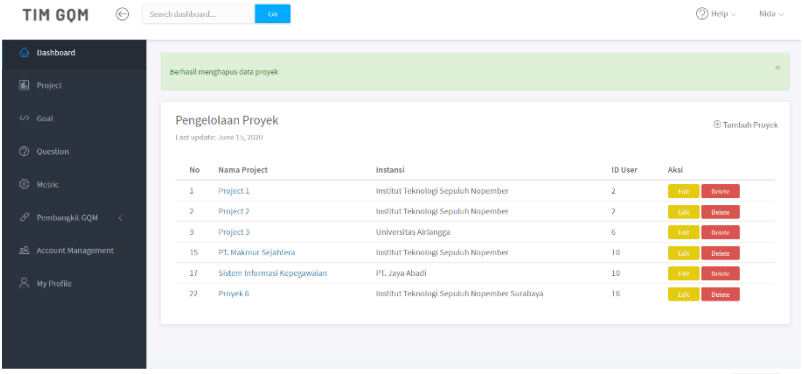
### 5.3.7.3. Pengujian Menghapus Data Proyek

Pengujian menghapus data proyek merupakan pengujian terhadap kemampuan aplikasi untuk menghapus data proyek yang telah tersimpan pada sistem. Rincian pengujian dapat dilihat pada Tabel 5.15.

**Tabel 5. 15 Pengujian Menghapus Data Pengguna**

| | |
|---|---|
| **ID Pengujian** | UJ014 |
| **Nama** | Pengujian menghapus data proyek |
| **Tujuan Pengujian** | Menguji apakah sistem sudah mampu melakukan pengelolaan data proyek untuk menghapus data proyek |
| **Kondisi Awal** | • Pengguna telah login sebagai Admin<br>• Data proyek telah tersimpan pada sistem |
| **Prosedur Pengujian** | 1. Aktor melilih menu ‘Project’<br>2. Aktor memilih opsi data pengguna yang akan dihapus<br>3. Klik tombol ‘Hapus’ |
| **Skenario 1** | |

| | |
|---|---|
| **Masukan** | Aktor mengonfirmasi penghapusan dengan menekan tombol ‘Ok’ |
| **Data Uji** | - |
| **Hasil yang diharapkan** | Pesan berhasil menghapus data muncul |
| **Hasil yang diperoleh** | Sistem menampilkan data proyek berhasil dihapus |
| **Hasil Pengujian** | Berhasil |




**Gambar 5. 23 Menghapus Data Proyek: Skenario 1**

## 5.4. Evaluasi Pengujian

Berikut dijelaskan pada Tabel 5.16 merupakan rangkuman dari hasil pengujian fungsional kakas pengukuran.

**Tabel 5. 16 Hasil Pengujian**

| Fungsionalitas | Skenario | Hasil Uji |
|---|---|---|
| **Mendaftarkan akun** | Data kosong | Terpenuhi |
| | Data tidak lengkap | Terpenuhi |
| | Data lengkap | Terpenuhi |
| | Data salah | Terpenuhi |
| **Membuat proyek pengukuran** | Data lengkap | Terpenuhi |
| **Menjawab pertanyaan** | Data lengkap | Terpenuhi |
| **Melihat analisis tujuan** | Data jawaban terisi | Terpenuhi |
| | Data jawaban belum terisi | Terpenuhi |
| **Menambah data ukur** | Data kosong | Terpenuhi |
| | Data tidak lengkap | Terpenuhi |
| | Data lengkap | Terpenuhi |
| **Mengubah data ukur** | Data perubahan | Terpenuhi |
| **Menghapus data ukur** | Menghapus data | Terpenuhi |
| **Menambah data pertanyaan** | Data kosong | Terpenuhi |
| | Data tidak lengkap | Terpenuhi |
| | Data lengkap | Terpenuhi |
| **Mengubah data pertanyaan** | Data perubahan | Terpenuhi |
| **Menghapus data pertanyaan** | Menghapus data | Terpenuhi |
| **Menambah data pertanyaan** | Data kosong | Terpenuhi |
| | Data tidak lengkap | Terpenuhi |
| | Data lengkap | Terpenuhi |
| **Mengubah data proyek** | Data perubahan | Terpenuhi |
| **Menghapus data proyek** | Menghapus data | Terpenuhi |

## 5.5. Studi Kasus

Pada Tugas Akhir ini, pengujian studi kasus menggunakan data penilaian terhadap Sistem Informasi Akademik Institut Teknologi Sepuluh Nopember yaitu Integra. Karena keterbatasan, penilaian hanya dilakukan dari sudut pandang mahasiswa. Mahasiswa berperan sebagai salah satu dari pengguna akhir aplikasi. Pengukuran ini bertujuan untuk menilai kualitas integra sesuai dengan tujuan yang didefinisikan pada proyek pengukuran. Berikut merupakan langkah-langkah untuk memperoleh data hingga pengolahan data :

1. Karena keterbatasan dalam kondisi pandemi, tahap pendefinisian goal dan pertanyaan dilakukan dengan studi literatur terlebih dahulu. Studi literatur yang ditelaah adalah mengenai kerangka kerja evaluasi kualitas website akademik.
2. Studi literatur pada tahap sebelumnya menghasilkan beberapa goal yang dideskripsikan pada Tabel 5.17 sebagai berikut :

**Tabel 5. 17 Pendefinisian Goal**

| Kode Goal | Deskripsi Goal |
|---|---|
| G1 | Meningkatkan kemudahan penggunaan perangkat lunak |
| G2 | Pengendalian kualitas informasi website |
| G3 | Meningkatkan keandalan website |
| G4 | Meningkatkan efisiensi sistem |
| G5 | Peningkatan fungsionalitas sistem dalam memenuhi kebutuhan pengguna |

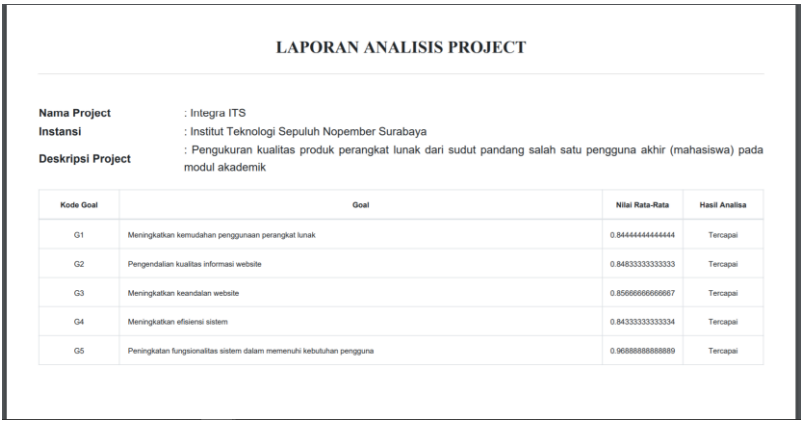
3. Goal yang telah didefinisikan kemudian mendapatkan konfirmasi persetujuan dari *stakeholder*, maka dilakukanlah pendefinisian pertanyaan.
4. Pertanyaan disebarkan dengan metode kuesioner menggunakan *google form*, kuesioner dibuat dengan menggunakan skala *likert*[6] dengan skor 1-5 untuk memudahkan responden dalam melakukan pengisian jawaban. Responden yang telah mengisi kuesioner berjumlah 15 orang dari mahasiswa ITS.
5. Setelah mendapatkan data hasil kuesioner, maka dilakukan pembuatan proyek pengukuran baru dengan memasukkan jawaban hasil kuesioner yang sudah di dapat pada tahap sebelumnya.
6. Hasil analisis tiap metrik dapat dilihat pada halaman analisis metrik jika jawaban pertanyaan yang tersedia pada sistem sudah terisi.
7. Hasil pencapaian goal secara keseluruhan dapat dilihat pada laporan yang dapat di cetak secara langsung.

### 5.5.1. Hasil Kuesioner

Hasil kuesioner diperoleh dari 15 orang responden mahasiswa, kuesioner ini digunakan untuk mengumpulkan data dari salah satu sudut pandang pengguna akhir integra. Hasil kuesioner dapat dilihat pada tabel 5.18.

### 5.5.2. Hasil Pengujian Studi Kasus

Pengujian pada studi kasus bertujuan untuk mendapatkan hasil analisisis kualitas dari sudut pandang pengguna akhir. Setelah semua data hasil kuesioner diinputkan, maka berikut hasil analisisnya dapat dilihat pada Gambar 5.24.



LAPORAN ANALISIS PROJECT

**Nama Project** : Integra ITS
**Instansi** : Institut Teknologi Sepuluh Nopember Surabaya
**Deskripsi Project** : Pengukuran kualitas produk perangkat lunak dari sudut pandang salah satu pengguna akhir (mahasiswa) pada modul akademik

| Kode Goal | Goal | Nilai Rata-Rata | Hasil Analisa |
|---|---|---|---|
| G1 | Meningkatkan kemudahan penggunaan perangkat lunak | 0.844444444444444 | Tercapai |
| G2 | Pengendalian kualitas informasi website | 0.84833333333333 | Tercapai |
| G3 | Meningkatkan keandalan website | 0.85666666666667 | Tercapai |
| G4 | Meningkatkan efisiensi sistem | 0.84333333333334 | Tercapai |
| G5 | Peningkatan fungsionalitas sistem dalam memenuhi kebutuhan pengguna | 0.96888888888889 | Tercapai |

**Gambar 5. 24 Laporan Hasil Analisis**

Hasil pengujian pada goal pertama yaitu meningkatkan kemudahan penggunaan perangkat lunak mendapatkan nilai rata-rata 0.844. Goal kedua yaitu pengendalian kualitas informasi website mendapat rata-rata 0,848. Goal ketiga yaitu meningkatkan keandalan website mendapatkan nilai rata-rata 0,856. Goal keempat yaitu meningkatkan efisiensi sistem mendapat rata-rata nilai 0,843. Serta goal kelima yaitu peningkatan fungsionalitas sistem dalam memenuhi kebutuhan pengguna mendapatkan nilai tertinggi dengan nilai rata-rata 0,968.

Secara kesuluruhan, hasil pengujian dengan goal yang telah didefinisikan telah tercapai. Kelima *goal* memiliki nilai rata-rata diatas 0,5 yang berarti hasil analisa proyek pengukuran website Integra ITS dari salah satu sudut pandang pengguna akhir (mahasiswa) telah tercapai sepenuhnya.

*[Halaman ini sengaja dikosongkan]*

# BAB VI
# KESIMPULAN DAN SARAN

Bab ini berisi tentang kesimpulan yang diperoleh selama pengerjaan Tugas Akhir ini berdasarkan hasil pengujian dan hal lainnya yang telah dilakukan. Selain itu, juga terdapat beberapa saran terhadap Tugas Akhir yang penulis ajukan terhadap pengembangan Tugas Akhir ini untuk pengembangan kedepannya.

## 6.1. Kesimpulan

Berdasarkan pengerjaan Tugas Akhir ini dan hasil yang didapatkan pada tahap uji coba aplikasi, penulis mengambil beberapa kesimpulan sebagai berikut :

1. Pembuatan kakas pengukuran perangkat lunak berbasis Goal Question Metric telah berhasil diimplementasikan. Hasil uji fungsionalitas aplikasi pada bab pengujian sudah tercapai sepenuhnya.
2. Kakas pengukuran dapat memetakan setiap goal dengan question, serta memetakan question dengan metric pada aspek produk perangkat lunak. Goal serta pertanyaan didefinisikan terlebih dulu kemudian diidentifikasi dan di definisikan metrik yang dibutuhkan. Menetapkan kesesuaian metrik dengan question dilakukan dengan cara menyelaraskan rumusan pertanyaan dengan karakteristik yang terdapat pada ISO 25010. Pemetaan GQM dilakukan menggunakan pembangkit GQM. Data goal yang telah didefinisikan direlasikan dengan pertanyaan terlebih dahulu. Pada pembangkit GQM, administrator memilih metric yang akan direlasikan dengan goal. Pada setiap goal yang terpilih, sistem menampilkan pertanyaan yang sudah terelasi dengan goal tersebut. Pertanyaan tersebut akan menjadi pertanyaan penyusun metric pada metric yang terpilih. Penamaan kode pembangkit metric tersusun dari kode goal dan kode metric yang terelasi.

3. Kakas pengukuran dapat menyajikan laporan hasil analisis pengukuran. Laporan analisis dapat ditampilkan sistem jika pengguna sudah memasukkan jawaban dari pertanyaan berdasarkan goal yang terpilih. Jawaban pertanyaan dihitung sesuai dengan metric yang terelasi dengan goal. Hasil pengukuran metric dapat dilihat pada halaman analisa metric, sedangkan hasil pengukuran goal dapat dilihat pada halaman analisa goal. Laporan hasil analisa goal dapat dicetak dan diunduh dalam bentuk pdf.

## 6.2. Saran

Terdapat beberapa saran terkait Tugas Akhir ini yang diharapkan bisa membuat Tugas Akhir ini menjadi lebih baik. Saran-saran tersebut antara lain:

1. Penelitian lanjutan terhadap *framework* pengukuran kualitas pada Sistem Informasi Akademik.
2. Menambahkan rekomendasi/saran pada laporan hasil analisis ukur.
3. Menambahkan fitur mengajukan usulan goal/diskusi dengan Tim GQM pengembang aplikasi.

# DAFTAR PUSTAKA

[1] Kiswinardi, W., 2011.” Pembuatan Kakas Pengukuran Kinerja Aplikasi Layanan E- Government Dengan Metode Extended Goal Question Metric”, Intsitut Teknologi Sepuluh Nopember, Surabaya, 2011.

[2] V.R. Basili, H.D. Rombach, G. Caldiera, “The Goal Question Metric Approach.” Institute for Advanced Computer StudiesDepartment of Computer ScienceUniversity Of MarylandCollege Park, Maryland,1994.

[3] Mebrate, W. Tsigereda, 2010, “A Framework for Evaluating Academic Website Quality From Students Perspective”, Delf University of Technology, Netherlands.

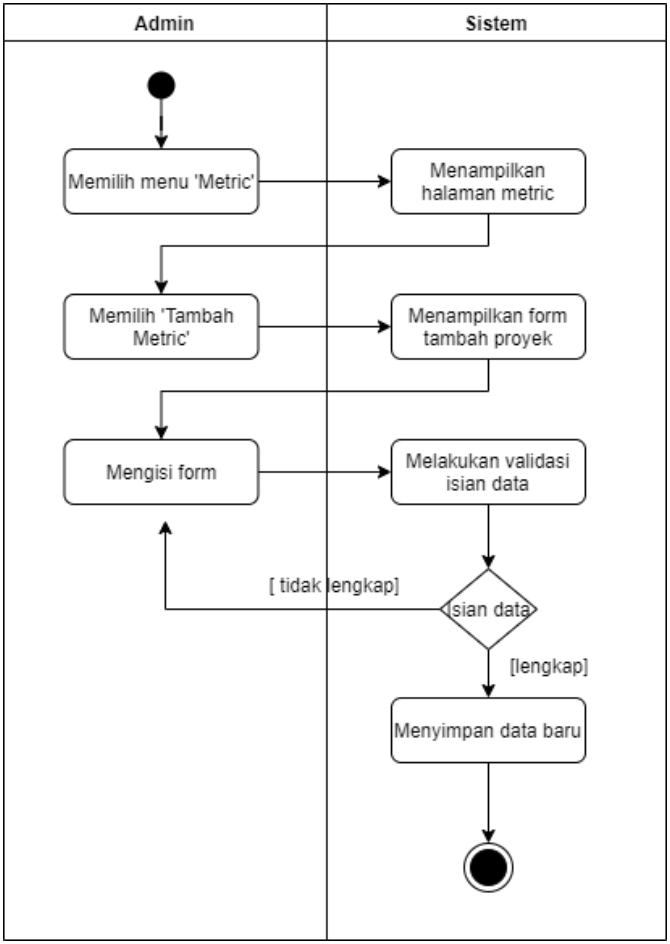
[4] Lindstorm, B., 2004. “A Software Measurement Case Study Using GQM.”, J. Lund University, USA,2004.

[5] Berander, Patrik and Jonsson, Per. 2006, “Goal Question Metric Based Approach for Efficient Measurement Framework Definition. Proceedings of ISESE ’06”, Rio de Janeiro, Brazil.

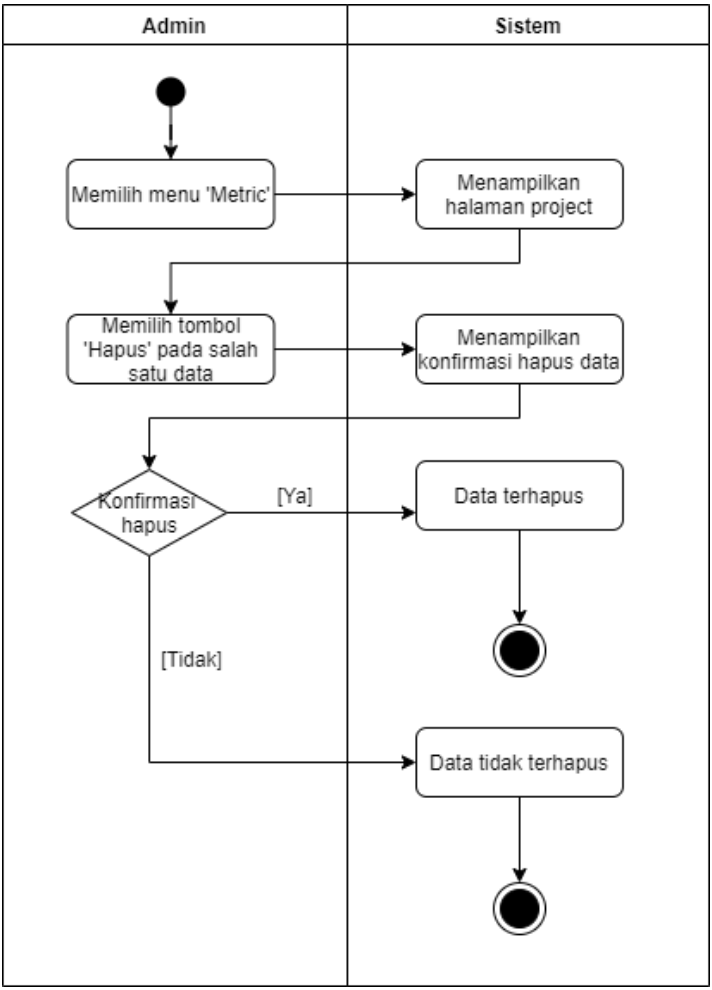
[6] Sugiyono, 2006. “Metode Penelitian Kuantitatif Kualitatif dan R&D”, Alfabeta, Bandung.

# LAMPIRAN A

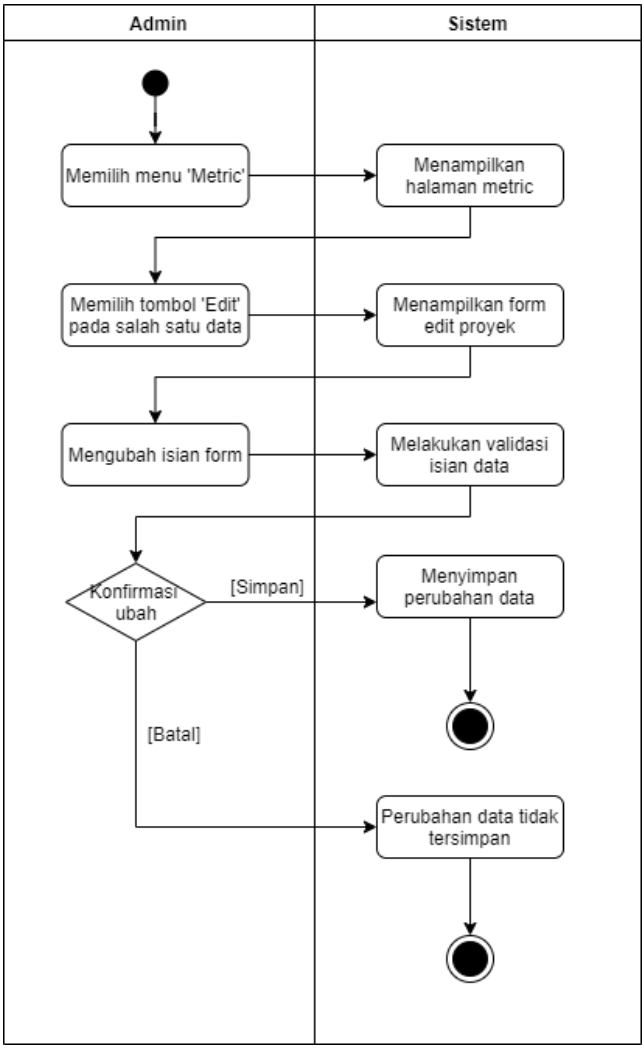
## Diagram Aktivitas Mengelola Ukuran




**Gambar 3. 31 Diagram aktivitas UC007 : Menambah data ukuran**



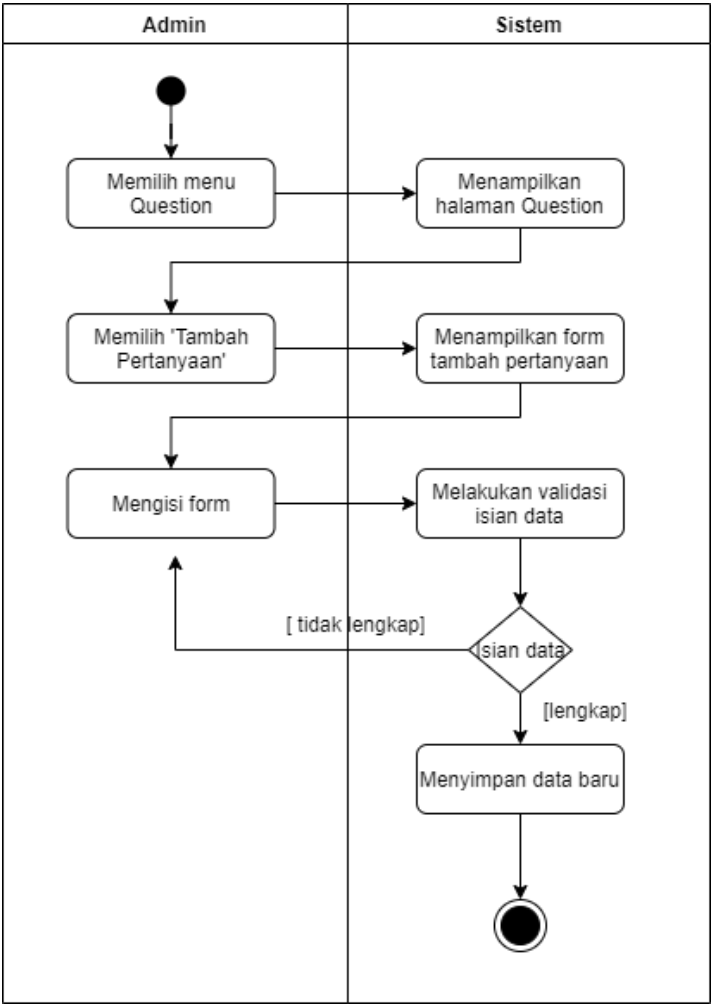

**Gambar 3. 32 Diagram aktivitas UC007 : Menghapus data ukuran**



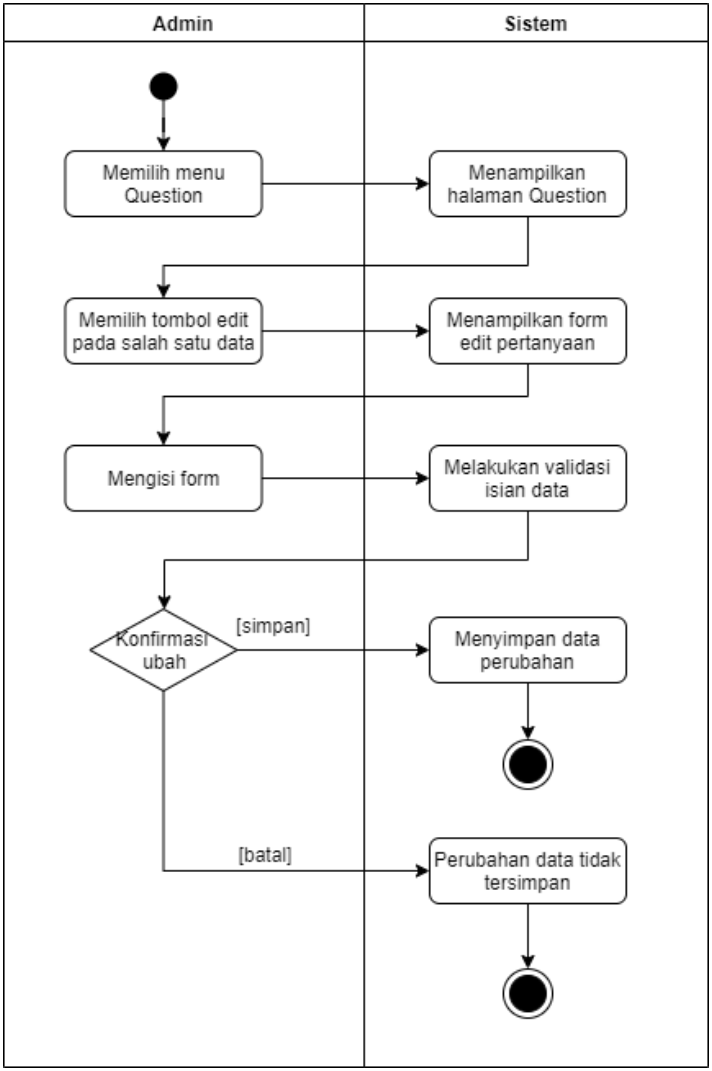

**Gambar 3. 33 Diagram aktivitas UC007 : Mengubah Data Ukuran**

# LAMPIRAN B

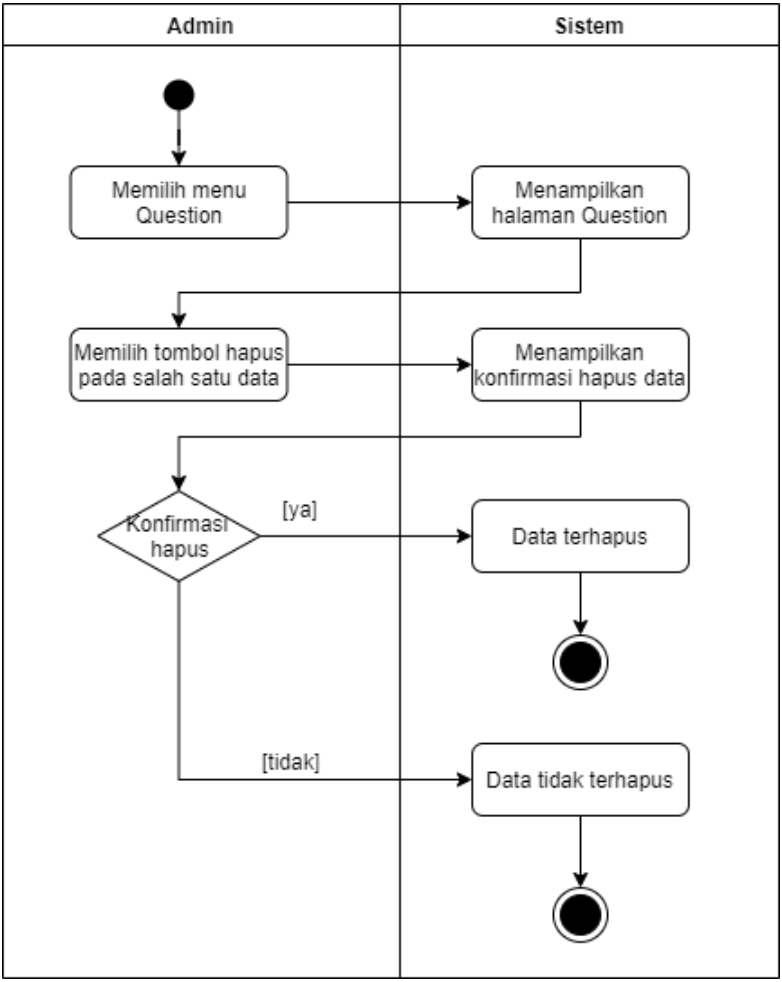
## Diagram Aktivitas Mengelola Pertanyaan



**Gambar 3. 34 Diagram aktivitas UC008 : Menambah Data Pertanyaan**



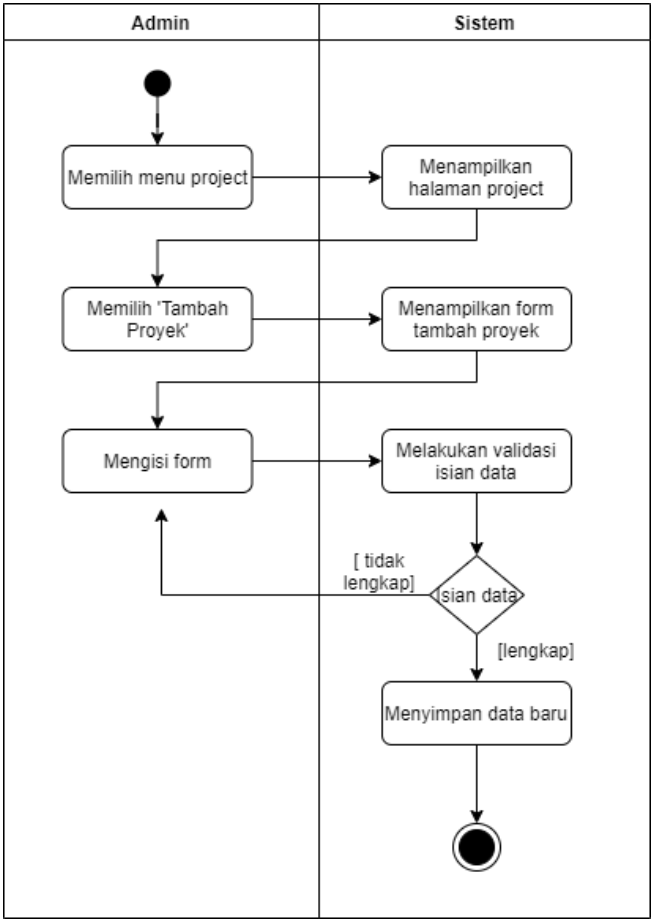
**Gambar 3. 35 Diagram aktivitas UC008 : Mengubah Data Pertanyaan**



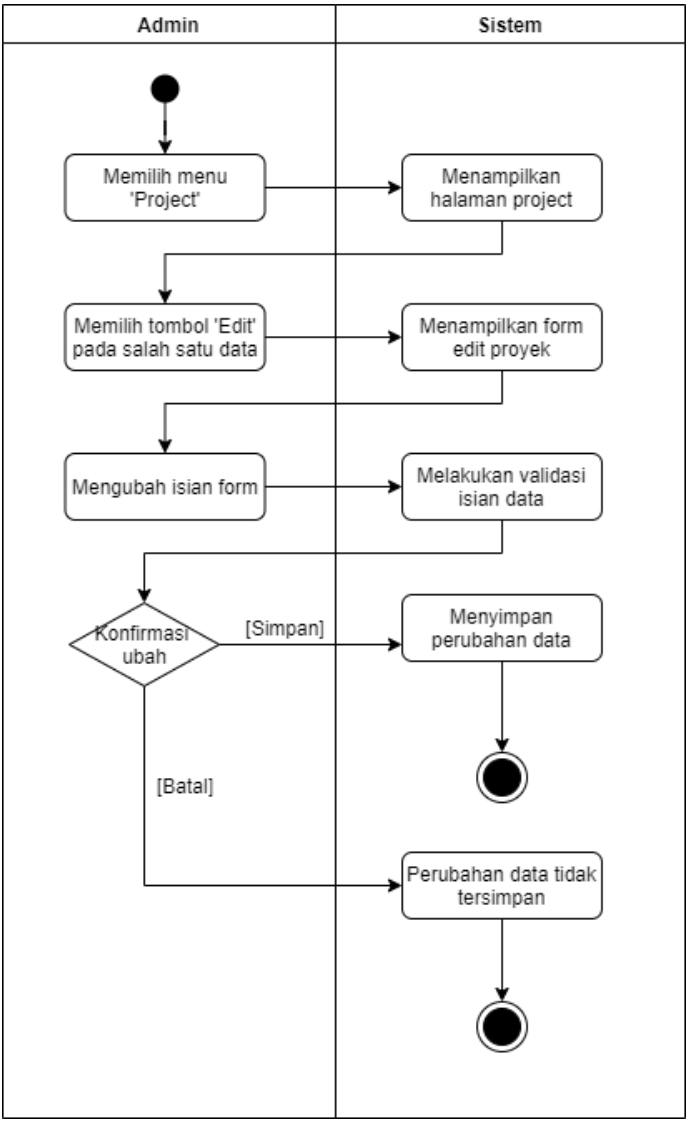

**Gambar 3. 36 Diagram aktivitas UC008 : Menghapus Data Pertanyaan**

# LAMPIRAN C

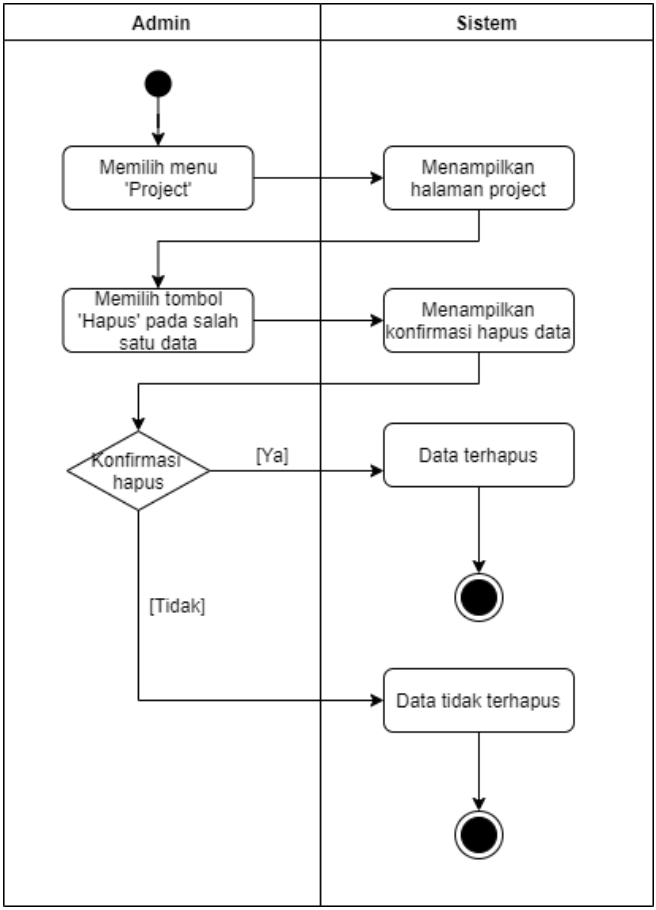
## Diagram Aktivitas Mengelola Data Proyek




**Gambar 3. 37 Diagram aktivitas UC0009 : Menambah data proyek**




**Gambar 3. 38 Diagram aktivitas UC0009 : Mengubah data proyek**




**Gambar 3. 39 Diagram aktivitas UC0009 : Menghapus data proyek**

# LAMPIRAN D

## Hasil Kuesioner Studi Kasus Integra ITS

| Goal | Question | Sangat Tidak Setuju | Tidak Setuju | Ragu-ragu | Setuju | Sangat Setuju |
|---|---|---|---|---|---|---|
| **G1** | Q1 | 0 | 0 | 0 | 6 | 9 |
| | Q2 | 0 | 0 | 0 | 0 | 15 |
| | Q3 | 0 | 0 | 0 | 8 | 7 |
| | Q4 | 1 | 7 | 7 | 0 | 0 |
| | Q5 | 0 | 0 | 0 | 5 | 10 |
| | Q6 | 0 | 0 | 1 | 7 | 7 |
| | Q7 | 0 | 3 | 8 | 4 | 0 |
| | Q8 | 0 | 0 | 0 | 7 | 8 |
| | Q9 | 0 | 3 | 11 | 1 | 0 |
| | Q10 | 0 | 0 | 0 | 0 | 15 |
| **G2** | Q11 | 0 | 0 | 0 | 4 | 11 |
| | Q12 | 1 | 2 | 0 | 12 | 0 |
| | Q13 | 0 | 0 | 0 | 0 | 15 |
| | Q14 | 0 | 0 | 1 | 8 | 6 |

| | | | | | | |
|---|---|---|---|---|---|---|
| | Q15 | 0 | 0 | 1 | 9 | 5 |
| | Q16 | 0 | 0 | 0 | 0 | 15 |
| | Q17 | 1 | 8 | 6 | 0 | 0 |
| **G3** | Q18 | 0 | 0 | 1 | 2 | 12 |
| | Q19 | 0 | 0 | 2 | 9 | 4 |
| | Q20 | 0 | 0 | 3 | 9 | 3 |
| | Q21 | 0 | 0 | 1 | 12 | 2 |
| | Q22 | 0 | 0 | 1 | 5 | 9 |
| | Q23 | 0 | 0 | 0 | 8 | 7 |
| **G4** | Q24 | 0 | 0 | 0 | 0 | 15 |
| | Q25 | 0 | 0 | 0 | 0 | 15 |
| | Q26 | 3 | 4 | 7 | 1 | 0 |
| | Q27 | 0 | 0 | 0 | 8 | 7 |
| **G5** | Q28 | 0 | 0 | 0 | 3 | 12 |
| | Q29 | 0 | 0 | 0 | 0 | 15 |
| | Q30 | 0 | 0 | 0 | 3 | 12 |
| | Q31 | 0 | 0 | 0 | 0 | 15 |
| | Q32 | 0 | 0 | 0 | 4 | 11 |

# LAMPIRAN E

## Rumusan Pertanyaan pada Kuesioner

| Kode Pertanyaan | Deskripsi Pertanyaan |
|---|---|
| Q1 | Apakah cara penggunaan website mudah diingat? |
| Q2 | Apakah fungsionalitas sistem sudah sesuai dengan kebutuhan pengguna? |
| Q3 | Apakah penggunaan website mudah untuk dipelajari? |
| Q4 | Apakah terdapat user manual atau cara penggunaan sistem? (baik berupa dokumen maupun FAQ) |
| Q5 | Apakah website mudah dioperasikan? |
| Q6 | Apakah muncul pesan error jika terjadi kesalahan saat menggunakan website? |
| Q7 | Apakah terdapat pesan pencegahan error pada sistem? |
| Q8 | Apakah tampilan website mudah dipahami? |
| Q9 | Apakah tampilan yang ditampilkan website dapat menyenangkan atau memuaskan pengguna? |
| Q10 | Apakah website dapat digunakan oleh beberapa pengguna dalam waktu yang sama? |
| Q11 | Apakah sistem memiliki kesesuaian fungsi yang ditampilkan dengan kebutuhan pengguna? |
| Q12 | Apakah sistem dapat memenuhi kebutuhan pengguna? |
| Q13 | Apakah hasil keluaran setiap fungsi sesuai dengan yang diharapkan pengguna? |
| Q14 | Apakah semua fungsi dapat digunakan dengan baik sesuai dengan yang diharapkan? |
| Q15 | Apakah fungsi yang tersedia pada website sudah lengkap? |
| Q16 | Apakah informasi yang ditampilkan website selalu relevan dengan kebutuhan pengguna? |
| Q17 | Apakah informasi pada website ambigu? |

| | |
|---|---|
| Q18 | Apakah informasi yang ditampilkan pada setiap fungsi sesuai dengan tujuan fungsionalitas? |
| Q19 | Apakah website selalu menampilkan informasi terbaru? |
| Q20 | Apakah informasi terbaru selalu ada di halaman utama website? |
| Q21 | Apakah identitas instansi tersedia pada website? |
| Q22 | Apakah terdapat kontak yang dapat di hubungi pada website? |
| Q23 | Apakah website dapat diakses setiap waktu? |
| Q24 | Apakah website dapat diakses pada waktu yang ditentukan? |
| Q25 | Apakah seluruh fungsionalitas selalu dapat diakses jika diperlukan? |
| Q26 | Apakah terdapat banyak kesalahan/error ketika menjalankan website? |
| Q27 | Apakah sering terjadi kegagalan ketika menggunakan fungsionalitas? |
| Q28 | Jika tejadi kesalahan/error dalam menggunakan website, apakah sistem dapat kembali normal ketika digunakan kembali? |
| Q29 | Apakah website cepat merespon ketika menampilkan informasi? |
| Q30 | Apakah website cepat merespon ketika melakukan pengolahan data? |
| Q31 | Apakah website menjadi down dan tidak dapat diakses ketika jam sibuk? |
| Q32 | Apakah sistem memiliki keterbatasan saat meng-upload data? |

# LAMPIRAN F

## Penentuan Metrik Pengukuran

| Kode Metrik | Metrik | Perhitungan |
|---|---|---|
| M1 | Appropriateness Recognisability | $\frac{(\text{Rata} - \text{rata Q1}/5) + (\text{Rata} - \text{rata Q2}/5)}{2}$ |
| M2 | Learnability | $\frac{(\text{Rata} - \text{rata Q3}/5) + (\text{Rata} - \text{rata Q4}/5)}{2}$ |
| M3 | Operability | Rata-rata Q5/5 |
| M4 | User Error Protection | $\frac{(\text{Rata} - \text{rata Q6}/5) + (\text{Rata} - \text{rata Q7}/5)}{2}$ |
| M5 | User Interface Aesthetic | $\frac{(\text{Rata} - \text{rata Q8}/5) + (\text{Rata} - \text{rata Q9}/5)}{2}$ |
| M6 | Accesibility | Rata-rata Q10/5 |
| M19 | Functional Appropriateness | $\frac{(\text{Rata} - \text{rata Q11}/5) + (\text{Rata} - \text{rata Q12}/5)}{2}$ |
| M18 | Functional Correctness | $\frac{(\text{Rata} - \text{rata Q13}/5) + (\text{Rata} - \text{rata Q14}/5)}{2}$ |
| M17 | Functional Completeness | Rata-rata Q15/5 |
| M7 | Relevance of Information | Rata-rata Q16/5 |
| M8 | Accuracy of Information | $\frac{(\text{Rata} - \text{rata Q17}/5) + (\text{Rata} - \text{rata Q18}/5)}{2}$ |
| M9 | Uptodate Information | $\frac{(\text{Rata} - \text{rata Q19}/5) + (\text{Rata} - \text{rata Q20}/5)}{2}$ |
| M10 | Identity | $\frac{(\text{Rata} - \text{rata Q21}/5) + (\text{Rata} - \text{rata Q22}/5)}{2}$ |
| M11 | Maturity | $\frac{(\text{Rata} - \text{rata Q23}/5) + (\text{Rata} - \text{rata Q24}/5)}{2}$ |
| M12 | Availability | Rata-rata Q25/5 |

| | | |
|---|---|---|
| M13 | Fault Tolerance | $\frac{(\text{Rata} - \text{rata Q26}/5) + (\text{Rata} - \text{rata Q27}/5)}{2}$ |
| M14 | Recoverability | Rata-rata Q28/5 |
| M15 | Time Behaviour | $\frac{(\text{Rata} - \text{rata Q29}/5) + (\text{Rata} - \text{rata Q30}/5)}{2}$ |
| M16 | Resource Utilisation & Capacity | $\frac{(\text{Rata} - \text{rata Q31}/5) + (\text{Rata} - \text{rata Q32}/5)}{2}$ |

# LAMPIRAN G

## Pemetaan Goal Question Metric

| Kode Goal | Metrik | Kode Metrik | Kode Pertanyaan |
|---|---|---|---|
| G1 | Appropriateness Recognisability | M1 | Q1 |
| | | | Q2 |
| | Learnability | M2 | Q3 |
| | | | Q4 |
| | Operability | M3 | Q5 |
| | User Error Protection | M4 | Q6 |
| | | | Q7 |
| | User Interface Aesthetic | M5 | Q8 |
| | | | Q9 |
| | Accesibility | M6 | Q10 |
| G5 | Functional Appropriateness | M19 | Q11 |
| | | | Q12 |
| | Functional Correctness | M18 | Q13 |
| | | | Q14 |
| | Functional Completeness | M17 | Q15 |
| G2 | Relevance of Information | M7 | Q16 |
| | Accuracy of Information | M8 | Q17 |
| | | | Q18 |
| | Uptodate Information | M9 | Q19 |
| | | | Q20 |
| | Identity | M10 | Q21 |
| | | | Q22 |
| G3 | Maturity | M11 | Q23 |
| | | | Q24 |
| | Availability | M12 | Q25 |
| | Fault Tolerance | M13 | Q26 |
| | | | Q27 |

| | | | |
|---|---|---|---|
| | Recoverability | M14 | Q28 |
| G4 | Time Behaviour | M15 | Q29 |
| | | | Q30 |
| | Resource Utilisation & Capacity | M16 | Q31 |
| | | | Q32 |

# BIODATA PENULIS

Nida Regita Fauzianti, si sulung dari empat bersaudara yang lahir di Bandung pada tanggal 20 Januari 1997. Penulis telah menempuh pendidikan formal mulai dari SDN Percobaan Cileunyi, SMP Negeri 8 Bandung, SMA Negeri 24 Bandung dan terakhir sebagai mahasiswa Departemen Informatika Institut Teknologi Sepuluh Nopember dengan rumpun mata kuliah Rekayasa Perangkat Lunak (2016-2020). Lulus dari SMA, penulis melanjutkan pendidikan di Departemen Informatika.

Selama perkuliahan, penulis aktif dalam organisasi kemahasiswaan, antara lain sebagai Staff Divisi Acara FTIf Festival 2016, Staff Departemen *External Affair* BEM FTIf ITS 2017-2018, Staff Biro National Logic Competition 2017 dan 2018. Selama kuliah di Departemen Informatika ITS, penulis mengambil bidang minat Rekayasa Perangkat Lunak (RPL) dengan ketertarikan penulis terdapat pada rancang bangun perangkat lunak dan *website*. Penulis dapat dihubungi melalui surel nidaregita@gmail.com.