TUGAS AKHIR

PENGEMBANGAN SISTEM INFORMASI MANAJEMEN DATA *STUDENT FEEDBACK QUESTIONNAIRE (SFQ)* BERBASIS WEB YANG DIGUNAKAN OLEH LP2MP DI UNIVERSITAS PELITA HARAPAN

Ditulis untuk memenuhi sebagian persyaratan akademik
guna memperoleh gelar Sarjana Komputer Strata Satu

Oleh:

NAMA	**:**	**ANDRE LAYARDI**
NPM	**:**	**00000011854**

PROGRAM STUDI SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER
UNIVERSITAS PELITA HARAPAN
TANGERANG

2019

PERNYATAAN KEASLIAN KARYA TUGAS AKHIR

Saya mahasiswa Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Pelita Harapan

Nama Mahasiswa	:	**Andre Layardi**
Nomor Pokok Mahasiswa	:	**00000011854**
Program Studi	:	**Sistem Informasi**

Dengan ini menyatakan bahwa karya tugas akhir yang saya buat dengan judul

PENGEMBANGAN SISTEM INFORMASI MANAJEMEN DATA *STUDENT FEEDBACK QUESTIONNAIRE (SFQ)* BERBASIS WEB YANG DIGUNAKAN OLEH LP2MP DI UNIVERSITAS PELITA HARAPAN

adalah :

1) Dibuat dan diselesaikan sendiri, dengan menggunakan hasil kuliah, tinjauan lapangan dan buku-buku serta jurnal acuan yang tertera di dalam referensi pada karya tugas akhir saya.
2) Bukan merupakan duplikasi karya tulis yang sudah dipublikasikan atau yang pernah dipakai untuk mendapatkan gelar sarjana di universitas lain, kecuali pada bagian-bagian sumber informasi dicantumkan dengan cara referensi yang semestinya.
3) Bukan merupakan karya terjemahan dari kumpulan buku atau jurnal acuan yang tertera di dalam referensi pada karya tugas akhir saya.

Kalau terbukti saya tidak memenuhi apa yang telah dinyatakan di atas, maka karya tugas akhir ini batal.

Tangerang, 11 Juni 2019
Yang membuat pernyataan

Materai
6000

Andre Layardi

UNIVERSITAS PELITA HARAPAN

FAKULTAS ILMU KOMPUTER

PERSETUJUAN DOSEN PEMIMBING TUGAS AKHIR

PENGEMBANGAN SISTEM INFORMASI MANAJEMEN DATA *STUDENT FEEDBACK QUESTIONNAIRE (SFQ)* BERBASIS WEB YANG DIGUNAKAN OLEH LP2MP DI UNIVERSITAS PELITA HARAPAN

Oleh :

Nama	**:**	**Andre Layardi**
NPM	**:**	**00000011854**
Program Studi	**:**	**Sistem Informasi**
Peminatan	**:**	**Business Information System**

Telah diperiksa dan disetujui untuk diajukan dan dipertahankan dalam Sidang Tugas Akhir guna memperoleh gelar Sarjana Komputer Strata Satu pada Program Studi Sistem Informasi, Fakultas Ilmu Komputer Universitas Pelita Harapan, Tangerang, Banten.

Tangerang, 11 Juni 2019

Menyetujui :

Pembimbing

(Alfa Satya Putra, B.Sc., M.Sc.)

Ketua Program Studi	**Dekan**
Sistem Informasi	**Fakultas Ilmu Komputer**
(Arnold Aribowo, S.T., M.T.)	**(Eric Jobiliong Ph.D)**

UNIVERSITAS PELITA HARAPAN

FAKULTAS ILMU KOMPUTER

PERSETUJUAN TIM PENGUJI TUGAS AKHIR

Pada hari senin, 8 Juli 2019, telah diselenggarakan Sidang Tugas Akhir untuk memenuhi sebagian persyaratan akademik guna memperoleh gelar Sarjana Komputer Strata Satu pada Program Studi Sistem Informasi, Fakultas Ilmu Komputer Universitas Pelita Harapan, atas nama:

Nama	**:**	**Andre Layardi**
NPM	**:**	**00000011854**
Program Studi	**:**	**Sistem Informasi**
Fakultas	**:**	**Fakultas Ilmu Komputer**

termasuk ujian Tugas Akhir yang berjudul "**PENGEMBANGAN SISTEM INFORMASI MANAJEMEN DATA *STUDENT FEEDBACK QUESTIONNAIRE (SFQ)* BERBASIS WEB YANG DIGUNAKAN OLEH LP2MP DI UNIVERSITAS PELITA HARAPAN"** oleh tim penguji yang terdiri dari:

Nama Penguji	**Jabatan dalam Tim Penguji**	**Tanda tangan**
1.Alfa Satya Putra, B.Sc., M.Sc.	, sebagai Ketua	_____
2. Suryasari, S.Kom., M.T.	, sebagai Anggota	_____
3. Hery, S.Kom, M.MSI	, sebagai Anggota	_____

Tangerang, 8 Juli 2019

ABSTRAK

Andre Layardi (00000011854)

PENGEMBANGAN SISTEM INFORMASI MANAJEMEN DATA *STUDENT FEEDBACK QUESTIONNAIRE (SFQ)* BERBASIS WEB YANG DIGUNAKAN OLEH LP2MP DI UNIVERSITAS PELITA HARAPAN

(xv+ 116 halaman: 83 gambar, 55 tabel, 4 lampiran)

Dalam kehidupan yang serba modern ini, sistem informasi menjadi kebutuhan manusia untuk membantu dalam menyelesaikan proses pekerjaan. UPH memiliki divisi yang berfungsi untuk memperhatikan kualitas dan mutu pendidikan, yaitu divisi Lembaga Penjamin dan Pengembangan Mutu Pendidikan (LP2MP). LP2MP memiliki tugas untuk melakukan evaluasi seluruh pengajar yang ada di UPH, yaitu dengan memberikan *Student Feedback Questionnaire* (SFQ) ke semua kelas di setiap semester. LP2MP menggunakan aplikasi *open source* untuk mengumpulkan SFQ. Selanjutnya, data hasil dari SFQ tersebut diolah menggunakan Microsoft Excel, Microsoft Word. Data yang diolah akan menghasilkan informasi yang akan diberikan kepada pihak-pihak yang berkepentingan.

Oleh sebab itu, LP2MP membutuhkan sistem informasi untuk mengelola data sehingga dapat menunjang proses evaluasi pengajar UPH. Penelitian ini berfokus pada pengembangan sistem informasi pengelolaan data SFQ dengan menggunakan metode *Prototyping*. Sistem dikembangkan menggunakan *framework CodeIgniter* dan *Grocery CRUD* dengan menggunakan bahasa pemrograman PHP. Sistem ini menggunakan pemodelan UML dengan *class diagram*, *use case diagram* dan *activity diagram*.

Hasil dari penelitian ini adalah sistem informasi berbasis web untuk mengelola data SFQ yang dapat membantu LP2MP dalam melakukan proses evaluasi seluruh pengajar UPH hingga penyebaran informasi hasil evaluasi kepada pihak-pihak yang berkepentingan.

Kata kunci: Kualitas Penjaminan Mutu Pendidikan, *Prototyping*, UML, *CodeIgniter*, *Grocery* CRUD

Referensi: 13 (2003-2018)

ABSTRACT

Andre Layardi (00000011854)

DEVELOPMENT OF WEB-BASED DATA MANAGEMENT INFORMATION SYSTEMS STUDENT FEEDBACK QUESTIONNAIRE (SFQ) USED BY LP2MP IN PELITA HARAPAN UNIVERSITY

(xv+ 116 pages: 83 figures, 55 tables, 4 appendices)

In this modern life, information system has become an essential component to assist human to perform their works. Universitas Pelita Harapan (UPH) is an education institution that has a division which functions is to evaluate and improve the quality of its education. The division's name is Lembaga Penjamin dan Pengembangan Mutu Pendidikan (LP2MP). In practice, LP2MP has responsibility to evaluate all the lecturers in UPH, by giving Student Feedback Questionnaire (SFQ) to all classes in every semester. LP2MP uses open source application to collect the SFQs. Next, the result data from SFQ is processed using Microsoft Excel and Microsoft Word. The processed data will produce information which will be given to related parties.

Therefore, LP2MP needs an information system to manage the data to support the evaluation process of UPH lecturers. This research focuses on SFQ data management information system by using Prototyping method. The system is developed using framework Codeigniter and Grocery CRUD by using PHP as the programming language. This system is using UML modeling with class diagram, use case diagram and activity diagram.

The result of this research is a web based information system to manage SFQ data collection which can helps LP2MP to conduct evaluation process of all the lecturers in UPH. Finally, the system also helps with the distribution of information from evaluation results to the related parties.

Keywords**:** *Education Quality Assurance*, *Prototyping*, UML, *CodeIgniter*, *Grocery* CRUD

References**:** 13 (2003-2018)

KATA PENGANTAR

Puji syukur kepada Tuhan Yang Maha Esa atas rahmat yang telah diberikan sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul **"PENGEMBANGAN SISTEM INFORMASI MANAJEMEN DATA *STUDENT FEEDBACK QUESTIONNAIRE (SFQ)* BERBASIS WEB YANG DIGUNAKAN OLEH LP2MP DI UNIVERSITAS PELITA HARAPAN"**. Tuhan telah menyertai penulis dari awal hingga akhir proses penyusunan tugas akhir ini.

Tugas Akhir ini disusun untuk memenuhi sebagian persyaratan akademik guna memperolah gelar Sarjana Sistem Informasi Strata Satu Program Studi Sistem Informasi Universitas Pelita Harapan, Karawaci, Tangerang.

Penulis menyadari bahwa doa, bimbingan serta bantuan dari berbagai pihak sangat membantu penulis untuk menyelesaikan tugas akhir ini. Oleh karena itu, penulis mengucapkan terima kasih yang sebesar-besarnya kepada semua pihak terkait yang telah membantu dari proses awal hingga akhir pengerjaan tugas akhir ini yaitu kepada:

1. Bapak Eric Jobiliong, Ph.D., selaku *Dean* Fakultas Ilmu Komputer
2. Bapak Hendra Tjahyadi, S.T., M.T., Ph.D., selaku *Associate Dean* Fakultas Ilmu Komputer.
3. Bapak Arnold Aribowo, S.T., M.T., selaku Kepala Program Studi Jurusan Sistem Informasi.
4. Bapak Alfa Satya Putra, B.Sc., M.Sc. sebagai Dosen Pembimbing Tugas Akhir yang selalu memberikan dukungan, nasihat, motivasi untuk menyelesaikan tugas akhir ini serta telah meluangkan waktu dan tenaga dalam membimbing penulis.
5. Semua dosen yang telah mengajar dan memberi nasihat kepada penulis selama penulis menempuh pendidikan di program studi Sistem Informasi, Universitas Pelita Harapan.
6. Bapak Hendra Pranata Effendi, S.Kom., MBA, selaku *Head of Institutional Survey and System Improvement* yang telah bersedia membantu, memantau,

mengizinkan dan meluangkan waktu untuk penulis dalam memberikan informasi yang dibutuhkan penulis.

7. Semua karyawan divisi *Survey Center* LP2MP, khususnya untuk Pak Budi dan Pak Eldwin yang selama proses Tugas Akhir sudah memberikan dukungan dan semangat kepada penulis.
8. Orang tua dan kakak, serta keluarga besar penulis yang telah mendoakan, memberikan motivasi, dukungan tanpa henti, dan memberikan kasih sayang, serta selalu mendengarkan keluh kesah penulis.
9. Leona Alexandritya sebagai teman terkasih penulis yang selalu mendoakan, mendukung, memberikan semangat, memberi kritik, memberi saran, dan memberi bantuan dalam menyelesaikan Tugas Akhir ini.
10. Sahabat-sahabat penulis yang menempuh pendidikan bersama di program studi Sistem Informasi mulai dari awal perkuliahan hingga akhir masa perkuliahan: Nico Chandra, Yuda Desela, Ongka Karnaga, Andreas Putra Wijaya, Marvin Gow, Stefano William Abbas, Albert Clement, Yonathan Ong, Stephen Susanto, Bernard Baharudin, William Diki, Christopher Putra, Daniel Sebastian Chandra, Steven Hartono, Angela Irena, Vania Christiana, Tania Jovita, serta seluruh mahasiswa Sistem Informasi 2015 yang telah mendukung dan memberi semangat kepada penulis.
11. Para pejuang skripsi: Andreas Putra Wijaya, Ongka Karnaga, William Diki, Daniel Sebastian Chandra, Yuda Desela, Christopher Putra, serta para pejuang skripsi lainnya yang saling mendukung dan memberi semangat satu sama lain.
12. Semua pihak yang telah membantu penulis dalam menyelesaikan Tugas Akhir ini yang namanya tidak dapat disebutkan satu per satu.

DAFTAR ISI

PERNYATAAN KEASLIAN KARYA TUGAS AKHIR ii
PERSETUJUAN DOSEN PEMIMBING TUGAS AKHIR iii
PERSETUJUAN TIM PENGUJI TUGAS AKHIR iv
ABSTRAK v
ABSTRACT vi
KATA PENGANTAR vii
DAFTAR ISI ix
DAFTAR GAMBAR xi
DAFTAR TABEL xiii
DAFTAR LAMPIRAN xv
BAB I PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Perumusan Masalah 2
1.3 Pembatasan Masalah 2
1.4 Tujuan Penelitian 3
1.5 Metodologi Penelitian 3
1.5.1 Metode Pengumpulan Data 3
1.5.2 Metodologi Pengembangan Sistem 4
1.6 Sistematika Penulisan 4
BAB II LANDASAN TEORI 6
2.1 *System Development Life Cycle* 6
2.1.1 Tahap Perencanaan 6
2.1.2 Tahap Analisis 6
2.1.3 Tahap Perancangan 7
2.1.4 Tahap Implementasi 7
2.2 *Rapid Application Development* (RAD) 8
2.3 *Unified Modeling Language* (UML) 2.0 8
2.3.1 Pemodelan Fungsional 10
2.3.2 Pemodelan Struktural 13
2.4 *HyperText Prepocessor* 14
2.5 *CodeIgniter* 14
2.6 *Model – View – Controller* (MVC) 14
2.7 *Relational Database Management Systems* (RDBMS) 15
2.8 *Black Box Testing* 15
2.9 *Grocery* CRUD 16
2.10 Kualitas Penjaminan Mutu Pendidikan 16
BAB III SISTEM SAAT INI 18
3.1 Profil Organisasi 18
3.2 Tujuan Organisasi 18
3.3 Struktur Organisasi 18
3.4 Analisis Sistem Saat Ini 19
3.5 Kendala Sistem Saat Ini 20
BAB IV ANALISIS DAN PERANCANGAN SISTEM USULAN 22
4.1 Tahap Perencanaan 22

4.1.1 Identifikasi Sistem 22
4.1.2 Analisis Kelayakan Sistem 23
4.2 Tahap Analisis 25
4.2.1 *User Requirements* 26
4.2.2 Pemodelan Fungsional 28
4.2.3 Pemodelan Struktural 52
4.3 Tahap Perancangan 53
4.3.1 Perancangan Lapisan Manajemen Data 54
4.3.2 *Table Description* 55
4.4 Tahap Implementasi 60
4.4.1 Lapisan Antarmuka (*Interface*) 61
4.4.2 Pengujian Sistem 96
4.4.3 Analisis Pengujian dan Perbandingan Sistem 118
BAB V KESIMPULAN DAN SARAN 120
5.1 Kesimpulan 120
5.2 Saran 120
DAFTAR PUSTAKA 122

DAFTAR GAMBAR

Gambar 2.1 Konsep *Model – View – Controller* Sumber: Buku Pro PHP MVC, 2012 15
Gambar 3.1. Struktur Organisasi LP2MP UPH Sumber : Data Internal Divisi Survey Center UPH 19
Gambar 4.1 *Use Case Diagram* Sistem Usulan 29
Gambar 4.2 *Activity Diagram Login Process* 30
Gambar 4.3 *Activity Diagram* Proses Mengubah *Profile* 31
Gambar 4.4 *Activity Diagram* Mengelola Data *User* 32
Gambar 4.5 *Activity Diagram* Mengelola Fakultas 34
Gambar 4.6 *Activity Diagram* Mengelola Program Studi 36
Gambar 4.7 *Activity Diagram* Mengelola *Course* 38
Gambar 4.8 *Activity Diagram* Mengubah Menu 40
Gambar 4.9 *Activity Diagram* Mengubah Sub Menu 41
Gambar 4.10 *Activity Diagram* Mengelola Hak Akses Menu 43
Gambar 4.11 *Activity Diagram* Melihat Laporan Dosen 44
Gambar 4.12 *Activity Diagram* Melihat Laporan Kaprodi 45
Gambar 4.13 *Activity Diagram* Melihat Laporan Dekan 46
Gambar 4.14 *Activity Diagram* Melihat Laporan VP 46
Gambar 4.15 *Activity Diagram* Melihat Laporan *Staff* 47
Gambar 4.16 *Activity Diagram* Mengelola *Question Set* 48
Gambar 4.17 *Activity Diagram* Proses Mengisi Survei *Staff* 49
Gambar 4.18 *Activity Diagram* Proses Mengisi Survei *Part Timer* 50
Gambar 4.19 *Activity Diagram* Mengelola Jadwal Survei 51
Gambar 4.20 *Activity Diagram* Mengubah Status Jadwal Survei 52
Gambar 4.21 *Class Diagram* Sistem Usulan 53
Gambar 4.22 *Entity Relationship Diagram* Sistem Usulan 54
Gambar 4.23 Tampilan Halaman *Login* 61
Gambar 4.24 Tampilan Halaman *Login Error* 62
Gambar 4.25 Tampilan Menu *Super Admin* dan *Staff* 63
Gambar 4.26 Tampilan Menu *Part Time* 63
Gambar 4.27 Tampilan Menu *Staff Akademik* 64
Gambar 4.28 Tampilan *Manage Users* 64
Gambar 4.29 Tampilan *Add Users* 65
Gambar 4.30 Tampilan *Edit User* 65
Gambar 4.31 Tampilan *Users Type* 66
Gambar 4.32 Tampilan *Add Users Type* 66
Gambar 4.33 Tampilan *Edit User Type* 67
Gambar 4.34 Tampilan *Manage* Akses Menu 68
Gambar 4.35 Tampilan *Add* Akses Menu 68
Gambar 4.36 Tampilan *Edit* Akses Menu 69
Gambar 4.37 Tampilan *Manage* Menu 69
Gambar 4.38 Tampilan *Edit* Menu 70
Gambar 4.39 Tampilan *Manage* Sub Menu 70
Gambar 4.40 Tampilan *Edit* Sub Menu 71

Gambar 4.41 Tampilan *Manage Faculty* 71
Gambar 4.42 Tampilan *Add Faculty* 72
Gambar 4.43 Tampilan *Edit Faculty* 72
Gambar 4.44 Tampilan *Manage Program Study* 73
Gambar 4.45 Tampilan *Add Program Study* 73
Gambar 4.46 Tampilan *Edit Program Study* 74
Gambar 4.47 Tampilan *Manage Course* 75
Gambar 4.48 Tampilan *Add Course* 75
Gambar 4.49 Tampilan *Edit Course* 76
Gambar 4.50 Tampilan *Calendar* 77
Gambar 4.51 Tampilan Detail *Calendar* 77
Gambar 4.52 Tampilan *Manage Question Set* 78
Gambar 4.53 Tampilan *Add Question Set* 78
Gambar 4.54 Tampilan *Edit Question Set* 79
Gambar 4.55 Tampilan *Manage Question* 80
Gambar 4.56 Tampilan *Add Question* (*Multiple Choice*) 81
Gambar 4.57 Tampilan *Add Question* (*Text*) 82
Gambar 4.58 Tampilan *Edit Question* 83
Gambar 4.59 Tampilan Survei 84
Gambar 4.60 Tampilan Selesai Survei 84
Gambar 4.61 Tampilan Penilaian *Super Admin* dan *Staff* 85
Gambar 4.62 Tampilan *Report Super Admin* dan *Staff* 86
Gambar 4.63 Tampilan Detail *Report Super Admin* dan *Staff* 87
Gambar 4.64 Tampilan Penilaian Dosen 88
Gambar 4.65 Tampilan *Report* Dosen 88
Gambar 4.66 Tampilan Detail *Report* Dosen 89
Gambar 4.67 Tampilan *Report* Kaprodi 90
Gambar 4.68 Tampilan *Report* Dekan 90
Gambar 4.69 Tampilan *Report* VP 91
Gambar 4.70 Tampilan *Dashboard Super Admin* dan *Staff* 91
Gambar 4.71 Tampilan *Dashboard* Detail *Super Admin* dan *Staff* 92
Gambar 4.72 Tampilan *Dashboard* Dosen 92
Gambar 4.73 Tampilan *Dashboard* Detail Dosen 93
Gambar 4.74 Tampilan *Dashboard* Kaprodi 93
Gambar 4.75 Tampilan *Dashboard* Detail Kaprodi 94
Gambar 4.76 Tampilan *Dashboard* Dekan 94
Gambar 4.77 Tampilan *Dashboard* Detail Dekan 95
Gambar 4.78 Tampilan *Dashboard Vice President* 95
Gambar 4.79 Tampilan *Dashboard* Detail *Vice President* 96

DAFTAR TABEL

Tabel 2.1 Elemen *Use Case Diagram* [3] 10
Tabel 2.2 Elemen *Activity Diagram* [3] 11
Tabel 2.3 Elemen *Class Diagram* [3] 13
Tabel 4.1 Daftar *Hardware* dan *Software* Minimum Sistem Usulan dari sisi *Server* 25
Tabel 4.2 Daftar *Hardware* dan *Software* Minimum dari Sisi *Client* 25
Tabel 4.3 *Table Description* dari Tabel *User* 55
Tabel 4.4 *Table Description* dari Tabel *User Type* 55
Tabel 4.5 *Table Description* dari Tabel Menu 56
Tabel 4.6 *Table Description* dari Tabel Sub Menu 56
Tabel 4.7 *Table Description* dari Tabel Akses Menu 56
Tabel 4.8 *Table Description* dari Tabel *Faculty* 57
Tabel 4.9 *Table Description* dari Tabel *Program Study* 57
Tabel 4.10 *Table Description* dari Tabel *Course* 57
Tabel 4.11 *Table Description* dari Tabel *Question Set* 58
Tabel 4.12 *Table Description* dari Tabel *Question Set Users* 59
Tabel 4.13 *Table Description* dari Tabel *Question Sessions* 59
Tabel 4.14 *Table Description* dari Tabel *Question Answers* 59
Tabel 4.15 *Table Description* dari Tabel *Questions* 60
Tabel 4.16 *Test Case* Melakukan *Login* 97
Tabel 4.17 *Test Case* Menambah Data *User* 98
Tabel 4.18 *Test Case* Mengubah Data *User* 98
Tabel 4.19 *Test Case* Mengelola *User Type* 99
Tabel 4.20 *Test Case* Mengubah *User Type* 100
Tabel 4.21 *Test Case* Menambah *Faculty* 101
Tabel 4.22 *Test Case* Mengubah *Faculty* 101
Tabel 4.23 *Test Case* Menambah *Program Study* 102
Tabel 4.24 *Test Case* Mengubah *Program Study* 102
Tabel 4.25 *Test Case* Menambah *Course* 103
Tabel 4.26 *Test Case* Mengubah *Course* 104
Tabel 4.27 *Test Case* Menambah Akses Menu 105
Tabel 4.28 *Test Case* Mengubah Akses Menu 105
Tabel 4.29 *Test Case* Mengubah Menu 106
Tabel 4.30 *Test Case* Mengubah Sub Menu 106
Tabel 4.31 *Test Case* Menambah *Question Set* 107
Tabel 4.32 *Test Case* Mengubah *Question Set* 108
Tabel 4.33 *Test Case* Menambah *Question* 109
Tabel 4.34 *Test Case* Mengubah *Question* 109
Tabel 4.35 *Test Case* Melihat *Question* 110
Tabel 4.36 *Test Case* Menpublish *Question* 110
Tabel 4.37 *Test Case* Melihat Hasil Survei 111
Tabel 4.38 *Test Case* Melihat Laporan Dosen 112
Tabel 4.39 *Test Case* Melihat Laporan Kaprodi 113

Tabel 4.40 *Test Case* Melihat Laporan Dekan 113
Tabel 4.41 *Test Case* Melihat Laporan VP 114
Tabel 4.42 *Test Case* Mengubah *Profile* 114
Tabel 4.43 *Test Case* Melihat Dashboard *Super Admin* 115
Tabel 4.44 *Test Case* Melihat Dashboard *Staff* 116
Tabel 4.45 *Test Case* Melihat Dashboard *Part Timer* 116
Tabel 4.46 *Test Case* Melihat Dashboard Dosen 117
Tabel 4.47 *Test Case* Melihat Dashboard Kaprodi 117
Tabel 4.48 *Test Case* Melihat Dashboard Dekan 117
Tabel 4.49 *Test Case* Melihat Dashboard *Vice President* 118
Tabel 4.50 Tabel Perbandingan Sistem Usulan Dengan Sistem Saat Ini 119

DAFTAR LAMPIRAN

LAMPIRAN A HASIL WAWANCARA.....A-1
LAMPIRAN B HASIL *USE CASE DIAGRAM*.....B-1
LAMPIRAN C *TEST OF CASE*.....C-1
LAMPIRAN D LEMBAR *MONITORING*.....D-1

BAB I
PENDAHULUAN

1.1 Latar Belakang

Dalam kehidupan saat ini, sangat sulit untuk hidup tanpa teknologi. Manusia sangat membutuhkan teknologi dalam kehidupan sehari-harinya untuk membantu memudahkan pekerjaan mereka. Sistem informasi adalah sistem yang digunakan untuk mengambil, memproses, menganalisis dan menyebarkan data, sehingga dapat mendukung pembuatan keputusan dan mengontrol jalannya suatu organisasi [1]. Sistem informasi sangat dibutuhkan di industri-industri saat ini, salah satu dari industri tersebut adalah industri pendidikan.

Universitas Pelita Harapan (UPH) adalah salah satu institusi pendidikan yang telah menggunakan sistem informasi dalam menunjang kebutuhannya untuk menjaga kualitas fasilitas pengajar dan fasilitas untuk belajar agar proses kegiatan di UPH dapat berjalan dengan baik. Dalam kegiatan belajar mengajar, kualitas pengajar merupakan salah satu faktor penting untuk dapat menghasilkan mahasiswa yang berkualitas maka diperlukan pengajar yang berkualitas.

UPH memiliki sebuah divisi yang bertugas untuk menjaga dan mengembangkan kualitas dan mutu pendidikan, yaitu divisi Lembaga Penjamin dan Pengembangan Mutu Pendidikan (LP2MP). Tindakan yang dilakukan oleh LP2MP UPH adalah dengan mengumpulkan data melalui *Student Feedback Questionnaire* (SFQ) di mana LP2MP dapat menggunakan data dari SFQ tersebut untuk menjalankan proses evaluasi dosen. SFQ tersebut dilakukan ke semua kelas di setiap semester.

Pada saat ini, LP2MP menggunakan aplikasi *open source* untuk mengumpulkan SFQ. Namun, dalam pengolahan data hasil dari SFQ tersebut harus dilakukan secara terpisah dengan menggunakan Microsoft Excel, Microsoft Word dan PDF. Data yang diolah akan menghasilkan sebuah informasi mengenai evaluasi para pengajar di UPH. Informasi tersebut lalu diberikan kepada pihak-pihak yang berkepentingan di UPH seperti *Vice President,* Dekan, Ketua Program Studi

(Kaprodi), Dosen dan pihak-pihak yang bersangkutan. LP2MP telah memiliki aplikasi berbasis *web* namun tidak berperan banyak dalam hal ini.

Berdasarkan permasalahan tersebut, tugas akhir ini mengenai sebuah sistem usulan untuk mengelola data *Student Feedback Questionnaire*. Dalam sistem ini mahasiswa dapat melakukan pengisian kuesioner secara *online* dan *staff* LP2MP dapat memperoleh hasil kuesioner dengan mudah, melakukan pengolahan, pengubahan dan penambahan data, hingga penyebaran informasi dari hasil evaluasi kepada seluruh pihak yang berkepentingan.

1.2 Perumusan Masalah

Masalah yang dihadapi oleh LP2MP antara lain:

1. Pihak LP2MP harus mengganti nomor *barcode*, nama dosen, dan jurusan di lembar kuesioner per satu kelas. Apabila terdapat 1000 kelas maka *staff* harus membuat 1000 kuesioner.
2. Pengolahan data yang dilakukan oleh pihak LP2MP masih manual sehingga orang – orang yang membutuhkan hasil survei tersebut baru bisa melihat hasil survei sebulan kemudian.
3. Hak akses yang masih terganggu untuk fakultas yang memiliki lebih dari satu program studi. Di mana dekan hanya dapat melihat satu program studi fakultasnya.
4. *Dashboard* masih dilakukan dengan cara manual, yaitu dengan menggunakan papan tulis. *Dashboard* digunakan untuk melihat proses pengerjaan LP2MP.

1.3 Pembatasan Masalah

Berdasarkan perumusan masalah di atas, pembatasan masalah yang ada adalah sebagai berikut:

1. Sistem informasi yang akan dibuat adalah berbasis *web*.
2. Sistem ini akan memiliki *dashboard* dalam bentuk grafik dan tabel penilaian.

3. Sistem ini hanya berada pada ruang lingkup UPH dan sampel data yang akan digunakan adalah Fakultas Ilmu Komputer (FIK).

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah membangun sebuah sistem informasi berbasis *web* agar memudahkan *staff* LP2MP dalam mengumpulkan data hasil kuesioner, mengelola data, dan menyebarkan informasi mengenai hasil evaluasi dosen kepada pihak yang berkepentingan.

1.5 Metodologi Penelitian

Metodologi penelitian yang digunakan dalam tugas akhir ini terdiri dari dua bagian, yaitu metode pengumpulan data dan metodologi pengembangan sistem.

1.5.1 Metode Pengumpulan Data

Metode pengumpulan data yang digunakan dalam tugas akhir ini adalah sebagai berikut:

1. Observasi

   Observasi dilakukan dengan cara mengamati setiap proses yang dilakukan oleh LP2MP yang terkait dengan fitur yang akan dikembangkan pada sistem ini, yaitu seputar pengumpulan data, pengecekan data, pengolahan data, dan penyebaran informasi

2. Wawancara

   Wawancara dilakukan dengan narasumber dari LP2MP yang memiliki peran dalam sistem pengelolaan data *SFQ*, yakni *manager* dan *staff* divisi LP2MP.

3. Studi Literatur

   Studi literatur digunakan sebagai metode untuk mencari referensi teori yang terkait dengan pengembangan sistem informasi ini. Dengan tujuan untuk mendalami dan memahami konsep yang akan digunakan, sehingga dapat mendukung penelitian dan pengembangan sistem pengelolaan data *SFQ* ini. Studi literatur dilakukan dengan mempelajari buku, jurnal, makalah,

maupun sumber lainnya yang terkait dengan penelitian dalam membangun sistem.

1.5.2 Metodologi Pengembangan Sistem

Dalam tahap pengembangan sistem ini menggunakan metodologi *System Development Life Cycle* (SDLC). Pada SDLC ini terdapat beberapa jenis metodologi untuk pengembangan sistem. Salah satu metodologi yang akan digunakan dalam sistem ini adalah metodologi *Rapid Application Development* (RAD) dengan pendekatan *prototyping* karena pengguna telah mengetahui kebutuhan apa saja yang diperlukan dalam pengembangan sistem ini.

1.6 Sistematika Penulisan

Berikut ini merupakan sistematika penulisan dari masing-masing bab dalam tugas akhir ini:

BAB I PENDAHULUAN

Bab I membahas mengenai Latar Belakang, Rumusan Masalah, Tujuan Penelitian, Batasan Masalah, Metodologi Penelitian, dan Sistematika Penulisan.

BAB II LANDASAN TEORI

Bab II menjelaskan berbagai teori yang telah diperoleh sehingga dapat digunakan sebagai landasan dalam proyek tugas akhir ini.

BAB III SISTEM SAAT INI

Bab III menjelaskan mengenai profil organisasi LP2MP, struktur organisasi LP2MP, analisa sistem saat ini, dan kendala yang dihadapi saat ini yang akan digunakan sebagai petunjuk dalam proses perancangan sistem usulan.

BAB IV ANALISIS DAN PERANCANGAN SISTEM USULAN

Bab IV akan menjelaskan mengenai kegiatan perencanaan, analisis, dan perancangan sistem, beserta metode – metode yang digunakan untuk sistem usulan.

BAB V KESIMPULAN DAN SARAN

Bab ini merupakan bab terakhir yang memberikan kesimpulan yang telah diperoleh dari hasil analisis dan perancangan sistem, serta saran – saran untuk penelitian selanjutnya.

BAB II
LANDASAN TEORI

2.1 *System Development Life Cycle*

System Development Life Cycle (SDLC) adalah proses *model* yang menggambarkan dan menentukan fase yang harus dilakukan terlebih dahulu dalam proses pengembangan sistem informasi. SDLC terdiri dari 4 fase, yaitu *planning* (tahap perencanaan), *analysis* (tahap analisis), *design* (tahap perancangan), dan *implementation* (tahap implementasi) [2].

2.1.1 Tahap Perencanaan

Tahap perencanaan sebagai tahap pertama dalam siklus SDLC merupakan proses fundamental untuk memahami mengapa sistem informasi harus dibangun dan menentukan bagaimana tim proyek akan melaksanakan tugasnya untuk membangun sistem tersebut. Dalam tahap ini terdapat inisiasi proyek yang memiliki tujuan untuk mengidentifikasi *system requirements, business need,* dan *business value.* Identifikasi dilakukan untuk mengetahui kebutuhan dalam membangun sistem dan manfaat apa saja yang akan dihasilkan oleh sistem tersebut.

Dalam tahap ini juga dilakukan berbagai analisis kelayakan (*feasibility analysis*) dalam berbagai aspek, yaitu analisa kelayakan teknis (*technical feasibility*), kelayakan ekonomi (*economic feasibility*), dan kelayakan organisasi (*organizational feasibility*). Analisa ini bertujuan untuk menentukan bagaimana sistem akan dibangun.

2.1.2 Tahap Analisis

Tahap analisis merupakan tahap yang menjelaskan mengenai siapa yang akan menggunakan sistem, apa yang akan sistem lakukan, di mana dan kapan sistem dapat digunakan. Terdapat tiga tahap dalam tahap analisis:

1) Strategi analisis (*analysis strategy*) dikembangkan untuk menganalisis strategi untuk membangun sistem.

2) Pengumpulan kebutuhan (*requirements gathering*) dapat dilakukan dengan cara wawancara atau penyebaran kuesioner.
3) Usulan sistem (*system proposal*) merupakan kombinasi dari analisa, konsep sistem dan *model.*

2.1.3 Tahap Perancangan

Tahap perancangan akan menentukan bagaimana sistem nantinya akan beroperasi. Operasi yang dimaksud melingkupi perangkat keras (*hardware*), perangkat lunak (*software*), infrastruktur jaringan (*network infrastructure*), antarmuka pengguna (*user interface*), formulir (*form*), laporan (*reports*), program, struktur data (*database*), dan *files*. Tahap perancangan terdiri dari empat tahap, yaitu:

1. *Design Strategy*

   Tahap ini menjelaskan bagaimana strategi yang tepat untuk menerapkan sistem yang akan dibangun.
2. *Architecture Design*

   Tahap ini menentukan perangkat keras, perangkat lunak, dan antarmuka pengguna yang akan digunakan.
3. *Database and File Specifications*

   Tahap ini menentukan data apa yang akan disimpan dan di mana data tersebut disimpan.
4. *Program Design*

   Tahap ini mendefinisikan program yang akan dibuat dan fungsi dari program tersebut.

2.1.4 Tahap Implementasi

Tahap terakhir dari SDLC adalah tahap implementasi. Dalam tahap ini sistem yang sebenarnya sudah dibuat. Tahap ini merupakan tahap yang mendapat perhatian terbesar, karena tahap ini merupakan tahap yang memakan waktu yang lama serta membutuhkan biaya yang besar. Tahap ini memilih tiga langkah, yaitu:

1. *System Construction*

Sistem usulan dibangun dan diuji agar dapat dipastikan bahwa sistem melakukan sesuai dengan yang sudah dirancang. Kebanyakan organisasi memberikan perhatian dan waktu yang banyak untuk melakukan *testing*. Organisasi melakukan hal ini dikarenakan biaya yang dibutuhkan akan jauh lebih besar jika terjadi *bug* (kesalahan), sehingga perlu dilakukan tahap pengujian untuk mengurangi dan mengetahui kesalahan yang ada di dalam sistem.

2. *Installation*

   Proses penggantian sistem dari sistem yang lama ke sistem yang baru. Langkah ini merupakan sarana pelatihan bagi pengguna agar dapat terbiasa dalam menggunakan sistem baru dan berguna untuk menangani perubahan yang disebabkan oleh sistem baru.

3. *Support Plan*

   Review secara formal atau informal mengenai sistem yang telah diimplementasikan untuk mengidentifikasi perubahan yang diperlukan oleh sistem.

2.2 *Rapid Application Development* (RAD)

Metodologi yang digunakan dalam pengembangan sistem ini adalah *Rapid Application Development* (RAD). Metodologi RAD dirancang menggunakan teknik khusus dan berbagai alat komputer untuk meningkatkan kecepatan dan kualitas dalam pengembangan sistem. Dalam pengembangan metodologi RAD ini akan menggunakan pendekatan *prototyping*, di mana fase analisa, desain dan implementasi dilakukan secara berulang hingga kebutuhan dari sistem terpenuhi. Oleh karena itu, dalam pengembangan sistem ini membutuhkan saran dari *user* dan pihak-pihak terkait [2].

2.3 *Unified Modeling Language* (UML) 2.0

Unified Modeling Language (UML) merupakan standardisasi bahasa pemodelan untuk memodelkan perangkat lunak pengembangan sistem. Dalam mendesain sistem, model menjadi salah satu hal penting untuk mengatur hal yang

kompleks. Pemodelan bertujuan untuk mendokumentasikan dan mengkomunikasikan aspek penting dari sistem yang didesain. Bahasa pemodelan dapat berupa notasi yang merupakan cara untuk mengekspresikan model tersebut dan berupa deskripsi yang menjelaskan arti dari notasi [3].

Terdapat empat fungsi dari UML, yaitu:

1. *Visualizing*

   UML menjadi sebuah bahasa grafis yang membantu *developer* melakukan pemodelan terhadap sistem yang akan dibangun sehingga mudah untuk dipahami dan divisualisasi.

2. *Specifying*

   Pada hal ini, UML dapat membangun sebuah model yang tepat, tidak ambigu dan menyeluruh. UML juga dapat melakukan spesifikasi pada seluruh analisa, desain dan implementasi dari sistem.

3. *Constructing*

   UML dapat menjadi bahasa pemodelan dalam membangun suatu perangkat lunak yang kemudian diterjemahkan ke dalam bahasa pemrograman seperti Java, C++ atau Visual Basic, dan bahasa program lainnya. Dalam hal ini dapat diartikan bahwa *developer* dapat membangun program berdasarkan model yang telah dibangun menggunakan UML.

4. *Documenting*

   Terdapat beberapa hal yang harus diperhatikan dalam membangun sebuah sistem, seperti *requirements*, arsitektur, desain, *source code*, *prototype*, dan lainnya. Hal-hal tersebut perlu didokumentasikan sehingga dapat digunakan sebagai pengontrol, pengukur dan mengkomunikasikan sistem selama dibangun dan setelah diimplementasikan.

Pada Tugas Akhir ini akan dibangun dengan menggunakan pemodelan UML versi 2.0 dan menggunakan dua kelompok pemodelan, yaitu pemodelan fungsional dan pemodelan struktural. Pada pemodelan fungsional, diagram yang akan digunakan antara lain adalah *use case diagram* dan *activity diagram*. Sedangkan pada pemodelan struktural akan menggunakan *class diagram*.

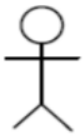
2.3.1 Pemodelan Fungsional

Pemodelan fungsional menggambarkan proses di dalam sistem dan gambaran mengenai apa yang perlu dilakukan oleh sistem. Tugas akhir ini menggunakan *use case diagram* dan *activity diagram* untuk pemodelan fungsional.

1. *Use case diagram*

   Use case diagram merupakan salah satu diagram fungsional yang menggambarkan fungsi dasar dari sistem. Diagram ini menggambarkan apa yang pengguna dapat lakukan dan bagaimana sistem merespon yang pengguna lakukan [4]. Berikut ini merupakan elemen-elemen dari *use case diagram*:

Tabel 2.1 Elemen *Use Case Diagram [3]*

Notasi	Nama Elemen	Keterangan
	*Actor*	Menggambarkan entitas yang memiliki peran dalam sistem
	Use Case	Merepresentasikan bagaimana sistem berinteraksi dan menggambarkan kegiatan yang dilakukan oleh pengguna dan respons yang diberikan sistem.
	System Boundary	Merepresentasikan cakupan dalam sistem
	Association	Garis *association* ini menghubungkan antara pengguna dan sistem untuk berinteraksi

Tabel 2.1 Elemen *Use Case Diagram* Lanjut [3]

------<<include>>----->	*Include Relationship*	Hubungan penyertaan fungsi dari satu *use case* ke *use case* lainnya.
-------<<extend>>------>	*Extend Relationship*	Merepresentasikan perluasan fungsionalitas dari sebuah *use case*
——————▷	*Generalization Relationship*	Hubungan antara *use case* yang umum dengan *use case* yang lebih khusus

Sumber: A. Dennis, B. H. Wixom, and R. M. Roth, *Systems Analysis & Design,* 5th ed. John Wiley & Sons, Inc, 2012

2. *Activity diagram*

   Activity diagram menggambarkan sebuah aktivitas utama dan hubungan antar aktivitas dalam suatu proses yang terdapat pada sistem [5]. Pada dasarnya, *Activity diagram* merupakan *flowchart* yang menunjukan alur dari satu aktivitas ke aktivitas lainnya [6]. *Activity diagram* memiliki elemen – elemen sebagai berikut:

Tabel 2.2 Elemen *Activity Diagram [3]*

Notasi	**Nama Elemen**	**Keterangan**
●	*Initial State*	Menggambarkan awal dari *Activity diagram*
◉	*Final State*	Menggambarkan akhir dari *Activity diagram*

Tabel 2.2 Elemen *Activity Diagram* Lanjut *[3]*

	Activity	Merepresentasikan sebuah tindakan atau aktivitas dalam *Activity diagram*
	Action Flow	Garis yang menghubungkan dua aktivitas dalam *Activity diagram*
Activity 1 Activity 2 Activity 3	*Decision Node*	Merepresentasikan sebuah kondisi untuk memastikan bahwa alur objek dapat menuju lebih dari satu *path,* tergantung kepada kondisi yang ada.
Activity 2 Activity 3 Activity 1	*Merge Node*	Penggabungan kembali dari beberapa *path* yang pada awalnya terbagi oleh *decision mode*
Activity 1 Activity 2 Activity 3	*Fork Node*	Kondisi di mana satu alur terbagi menjadi beberapa alur aktivitas yang berjalan bersamaan
Activity 2 Activity 3 Activity 1	*Join Node*	Kondisi di mana penggabungan kembali alur yang berjalan bersamaan menjadi satu

Sumber: A. Dennis, B. H. Wixom, and R. M. Roth, *Systems Analysis & Design,* 5th ed. John Wiley & Sons, Inc, 2012

2.3.2 Pemodelan Struktural

Pemodelan struktural digunakan untuk merepresentasikan struktur dan elemen – elemen yang terdapat dalam sistem. *Class diagram* akan digunakan sebagai pemodelan struktural dalam tugas akhir ini. *Class diagram* merupakan model statis yang menunjukkan *class* dan hubungan antar *class* [5]. *Class diagram* banyak digunakan karena menjadi kekuatan dasar dari sistem yang dibangun bersifat berorientasi pada obyek [7]. Berikut ini merupakan elemen – elemen yang terdapat pada *class diagram*:

Tabel 2.3 Elemen *Class Diagram [3]*

Notasi	Nama Elemen	Keterangan
Class 1 -Attribute 1: Type -Attribute 2: Type -Attribute 3: Type -Operation 1 () -Operation 2 ()	*Class*	Merepresentasikan objek yang dibutuhkan oleh sistem untuk menyimpan informasi
-Attribute 2: *Type*	*Attribute*	Merepresentasikan sifat atau karakter yang dimiliki oleh obyek
+Method Name ()	*Method*	Merepresentasikan tindakan atau fungsi yang dapat dilakukan oleh *class*.
0..* 1..*	*Association Relationship*	Merepresentasikan hubungan antara dua *class* atau lebih dengan *class* itu sendiri
	Generalization Relationship	Merepresentasikan hubungan antar *class* yang umum ke *class* yang lebih spesifik.

Sumber: A. Dennis, B. H. Wixom, and R. M. Roth, *Systems Analysis & Design,* 5th ed. John Wiley & Sons, Inc, 2012

2.4 *HyperText Prepocessor*

HyperText Preprocessor (PHP) merupakan sebuah bahasa pemrograman yang digunakan untuk pembuatan dan pengembangan web dan dapat digunakan bersamaan dengan *HyperText Markup Language* (HTML) [7]. PHP dapat digunakan untuk berbagai sistem operasi, seperti Linux, Microsoft Windows, Mac OS X. PHP berfokus pada *server-side scripting* dan salah satu fitur yang paling signifikan dalam PHP yaitu mendukung berbagai macam *database*.

2.5 *CodeIgniter*

CodeIgniter merupakan sebuah *framework* yang digunakan untuk membangun web menggunakan PHP sehingga dalam penulisan kode lebih terstruktur. Penggunaan *framework* ini bertujuan agar pengembangan proyek dapat lebih mudah dan lebih cepat serta lebih terstruktur. *Codeigniter* menggunakan desain pola *Model – View – Controller* (MVC) [5].

2.6 *Model – View – Controller* (MVC)

Model – View – Controller (MVC) merupakan pola perancangan perangkat lunak yang terdiri dari tiga komponen, yaitu model, *view* dan *controller* [8]. Pola ini dapat mendefinisikan cara obyek berkomunikasi dengan yang lainnya. Berikut ini merupakan ketiga bagian dari MVC, yaitu:

1) *Model*

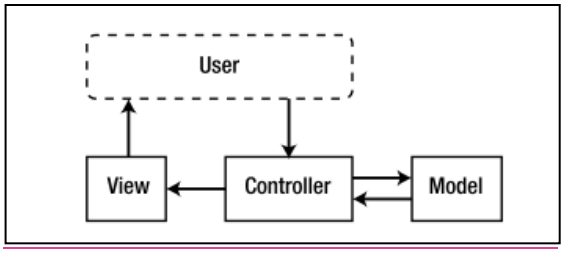
   Model merupakan representasi struktur data yang biasanya berhubungan dengan *database*. *Model* berisi fungsi yang membantu untuk mengambil, memasukkan dan memperbarui data di *database*.

2) *View*

   View merupakan aplikasi atau tampilan *user interface* yang memberikan informasi dan dapat dilihat langsung oleh pengguna. *View* terhubung dengan *model* untuk mendapatkan sebuah data dan kemudian ditampilkan kepada pengguna.

3) *Controller*

Controller merupakan komponen yang mengatur hubungan antara *model*, *view* dan berfungsi untuk memproses permintaan dari pengguna sehingga dapat menghasilkan sebuah halaman *web*.



Gambar 2.1 Konsep *Model – View – Controller*
Sumber: Buku Pro PHP MVC, 2012

2.7 *Relational Database Management Systems* (RDBMS)

Database Management Systems (DBMS) adalah sekumpulan data dan sekumpulan program yang memungkinkan pengguna dapat membuat dan mengelola *database* tersebut [9]. Sedangkan *Relational Database Management System* (RDBMS) merupakan DBMS yang menerapkan model *relational* [9]. RDBMS menyimpan data menjadi sekumpulan tabel yang terkait dengan tabel lainnya. Dalam RDBMS biasanya menggunakan *Structured Query Language* (SQL) sebagai bahasa standar yang digunakan untuk mengakses *database* [9]. SQL secara spesifik dirancang untuk memungkinkan seseorang dapat membuat *database*, menambahkan data baru ke dalam *database*, mengelola data dan dapat mengambil data yang diinginkan [10].

2.8 *Black Box Testing*

Untuk memastikan sistem dapat berjalan dengan baik sesuai dengan fungsinya, maka perlu dilakukan pengujian sistem. Salah satu bentuk pengujian yang akan digunakan dalam tugas akhir ini yaitu *black box testing*. *Black box testing* merupakan pengujian untuk perangkat lunak yang hanya melihat hasil yang dikeluarkan oleh sistem tanpa mengetahui proses pembuatannya [11]. Pengujian *black box* ini akan dilakukan jika semua *requirements* dari *user* telah terpenuhi.

Black box lebih berfokus pada *input* dan *output* pada setiap fungsi yang terdapat dalam sistem.

2.9 *Grocery* CRUD

Grocery CRUD merupakan *framework library* untuk *codeigniter* bersifat *open source* yang digunakan untuk mempermudah dalam pembuatan fungsi *create*, *read*, *update* dan *delete* (CRUD) untuk aplikasi berbasis *web*. *Framework* ini memiliki banyak fitur yang dapat digunakan untuk pengembangan sistem. Berikut ini merupakan fitur- fitur yang telah disediakan *grocery* CRUD [12]:

1. Tabel data yang dilengkapi dengan *paging*, menyortir dan mencari di setiap *field* kolom dengan menggunakan *ajax* atau *javascript* secara instan.
2. Relasi *database* 1-1, 1-n, dan n-n secara otomatis
3. *Form* validasi dari sisi *client* maupun server.
4. Dapat menambahkan aturan validasi sendiri.
5. Fleksibilitas dalam memilih kolom dan *fields* yang ingin ditambahkan dalam tabel.
6. Dapat mengubah tipe data dengan mudah.
7. Berfungsi di berbagai *browser*, seperti Mozilla Firefox, Google Chrome, Opera, Safari, Internet Explorer 8.
8. Selain itu dapat berfungsi dengan baik pada *mobile browser*, seperti Android, Windows, dan Apple Mobiles.
9. Telah diterjemahkan ke dalam 34 bahasa.

2.10 Kualitas Penjaminan Mutu Pendidikan

Kualitas penjamin mutu pendidikan menilai kualitas pengajaran dan pembelajaran yang terdapat di dalam institusi pendidikan. Pengajar merupakan sumber yang paling signifikan untuk meningkatkan standar pendidikan. Kualitas penjaminan mutu memiliki tujuan untuk meningkatkan kinerja di masa yang akan datang dengan membuat penilaian dari kinerja sebelumnya.

Dalam kualitas penjaminan terdapat tiga pendekatan yang dapat diterapkan, yakni akreditasi, penilaian dan audit. Akreditasi dan penilaian dilakukan dengan

mengawasi kualitas pengajaran. Sedangkan, audit berfokus dalam hal prosedur internal [13]. Terdapat mekanisme pengumpulan data dalam kualitas penjaminan mutu yang biasa digunakan yaitu dengan menggunakan survei. Survei yang dilakukan dapat berbentuk kuesioner maupun wawancara. Dari mekanisme pengumpulan data selanjutnya akan menghasilkan sebuah laporan atau prosedur yang ditindak lanjuti.

BAB III
SISTEM SAAT INI

3.1 Profil Organisasi

Lembaga Pengembangan dan Penjaminan Mutu Pendidikan atau (LP2MP) atau *Institutional Quality Enhancement and Assurance* UPH merupakan lembaga yang bertugas untuk menjamin mutu pendidikan yang terdapat di UPH. Lembaga ini telah berdiri sejak 2 Februari 2015. Terbentuknya lembaga ini dikarenakan UPH sedang mengalami pertumbuhan yang pesat, baik dalam hal jumlah mahasiswa baru maupun jumlah dan ragam program studi. Saat ini UPH memiliki 13 fakultas, 34 program studi dan kurang lebih 15000 mahasiswa. LP2MP terbentuk mengingat adanya Undang – Undang Nomor 12 Tahun 2012 tentang Pendidikan Tinggi, Peraturan Menteri Pendidikan dan Kebudayaan Republik Indonesia Nomor 49 Tahun 2014 tentang Standar Pendidikan Tinggi, Peraturan Menteri Pendidikan dan Kebudayaan Republik Indonesia Nomor 50 Tahun 2014 tentang Sistem Penjaminan Mutu Pendidikan Tinggi dan Peraturan Menteri Pendidikan dan Kebudayaan Republik Indonesia Nomor 87 Tahun 2014 tentang Akreditasi Program Studi dan Perguruan Tinggi, serta Statuta Universitas Pelita Harapan Tahun 2007.

3.2 Tujuan Organisasi

LP2MP memiliki tujuan yaitu menjamin mutu pendidikan yang terdapat di UPH secara berkelanjutan sehingga kualitas dan mutu pendidikan dapat terjaga dengan baik. Selain itu, LP2MP bertugas melakukan perencanaan, pelaksanaan, pengendalian dan pengembangan Sistem Penjamin Mutu Internal serta memimpin dan mengkoordinasikan Fakultas dan Program Studi dalam seluruh proses akreditasi Program Studi dan Institusi.

3.3 Struktur Organisasi

Pada Gambar 3.1 menunjukkan struktur organisasi LP2MP

Gambar 3.1. Struktur Organisasi LP2MP UPH

Sumber : Data Internal Divisi Survey Center UPH

3.4 Analisis Sistem Saat Ini

Sebelum melakukan pendistribusian SFQ, divisi *Survey Center* akan meminta atau mengunduh data melalui Oracle *Peoplesoft Campus Solution* (OPCS) sebagai *website* yang disediakan oleh UPH untuk mahasiswa dan *staff* dengan berbagai kegunaan seperti melihat jadwal mata kuliah, melihat nilai mata kuliah, melihat jumlah tagihan, dan lain-lain atau dengan meminta pada *admin* program studi. Data yang diterima akan diperiksa kembali oleh divisi *Survey Center* dan meminta verifikasi Kaprodi untuk memastikan bahwa data yang diterima adalah benar. Setelah data terkonfirmasi maka divisi akan mempersiapkan segala keperluan untuk survei ke kelas.

Pada saat ini LP2MP telah menggunakan aplikasi *open source* untuk mendistribusikan SFQ ke setiap kelas. Aplikasi ini berfungsi untuk membantu para *staff* dan *part-timer* membuat dan mengumpulkan kuesioner evaluasi dosen. Akan tetapi, aplikasi tersebut tidak dapat melakukan pengolahan data, sehingga dalam

proses pengolahannya harus dilakukan secara terpisah dengan menggunakan Microsoft Excel dan PDF.

Setelah proses pengolahan data, maka akan menghasilkan informasi berupa penilaian untuk setiap dosen. Hasil dari penilaian ini akan diunggah ke web LP2MP oleh para *staff*, sehingga para dosen dan pihak-pihak yang berkepentingan dapat mengakses dan melihat sendiri hasil dari penilaian yang diberikan oleh para mahasiswa.

Sistem yang dibangun mencakup fungsi-fungsi dalam sistem LP2MP saat ini, yakni sebagai berikut:

1. Melihat jadwal survei kelas secara manual Saat ini apabila *staff* ingin melakukan survei ke setiap kelas, maka *staff* harus membawa jadwal kelas yang telah didapat dari OPCS atau *admin* jurusan. Jadwal kelas yang dibawa oleh para staff masih berupa kertas.
2. Dalam pembuatan kuesioner saat ini, *staff* harus mempersiapkannya dengan menggunakan aplikasi bernama Limesurvey. *Staff* mengubah keterangan, seperti nama dosen, mata kuliah, dan jurusan dengan cara satu per satu di setiap kuesioner. Hal ini dilakukan agar memudahkan mahasiswa dalam pengisian kuesioner.
3. Mengolah data secara manual Saat ini pengolahan data dilakukan dengan cara terpisah dari aplikasi kuesioner. Data yang sudah terkumpul akan di *copy* dan *paste* ke dalam Microsoft Excel dan Microsoft Word kemudian diolah dan hasil pengolahan akan diubah ke dalam bentuk PDF.

3.5 Kendala Sistem Saat Ini

Sistem saat ini yang digunakan oleh LP2MP memiliki beberapa kendala dalam proses pelaksanaan. Beberapa kendala yang dimiliki sistem saat ini adalah

1. *Staff* LP2MP mengalami kendala dalam melihat jadwal kelas yang akan dikunjungi. Kendala ini disebabkan jadwal kelas yang masih dicetak dalam bentuk kertas sehingga untuk melihat atau ingin mencari data sebuah kelas harus dilihat satu per satu.

2. *Staff* LP2MP mengalami kendala dalam pembuatan kuesioner yang masih menggunakan aplikasi *open source*. Dalam pembuatan kuesioner *staff* harus membuat satu kuesioner untuk satu kelas. Dan jika ada 2000 kelas maka *staff* harus membuat 2000 kuesioner dengan mengubah keterangan kuesioner seperti *barcode*, nama dosen, dan jurusan satu per satu.
3. *Staff* LP2MP2 memiliki kendala dalam melakukan pengolahan data. Kendala ini disebabkan proses untuk pengolahan data tersebut dilakukan dengan cara terpisah dari aplikasi survei. *Staff* harus *export* data untuk memindahkan data dari aplikasi survei ke Microsoft excel dan Microsoft word. Setelah data terolah maka akan menghasilkan informasi yang berbentuk PDF.
4. LP2MP memiliki sistem informasi berbasis web yang hanya berfungsi untuk mengunggah hasil laporan dan dosen serta para pihak berkepentingan dapat mengakses dengan cara mengunduh hasil laporan tersebut.

BAB IV
ANALISIS DAN PERANCANGAN SISTEM USULAN

4.1 Tahap Perencanaan

Tahap perencaan ini terdiri dari identifikasi sistem dan analisis kelayakan sistem untuk membantu menganalisa permasalahan saat mengembangkan sistem usulan.

4.1.1 Identifikasi Sistem

Identifikasi sistem terdiri dari tiga aspek, yaitu *business needs, business requirement, business value.*

1) *Business Needs*

   Sistem yang dibuat untuk divisi LP2MP bertujuan untuk dapat meningkatkan perfoma kegiatan evaluasi seluruh dosen UPH, seperti memberikan SFQ tanpa harus menggunakan aplikasi *open source* sehingga memudahkan para *staff* dalam membuat survei. Kemudian, pengolahan data secara otomatis dan dapat menampilkan informasi mengenai hasil dari evaluasi tersebut kepada pihak-pihak terkait. Sistem ini diharapkan dapat menampilkan jadwal dalam melakukan kunjungan ke setiap kelas dan menampilkan *dashboard* yang digunakan untuk mengetahui banyaknya jumlah kelas yang telah dikunjungi.

2) *Business Requirement*

   Sistem usulan ini memiliki *business requirement* sebagai berikut:

   a) Sistem akan menjadi sarana untuk melihat jadwal, sehingga akan mempermudah *staff* dan *part timer* dalam melihat jadwal kunjungan setiap kelas.

   b) Sistem dapat membantu para *staff* dan *part timer* dalam membuat survei SFQ. Dengan adanya sistem ini, proses pembuatan survei akan menjadi lebih mudah dan dapat menghemat waktu.

c) Sistem ini akan membantu *Super Admin* dan *Staff* dalam melakukan pengolahan data survei. Dengan begitu, *Super Admin* dan *Staff* tidak perlu melakukan pengolahan data secara manual menggunakan Microsoft Excel dan Microsoft Word.

d) Sistem akan menjadi sarana laporan penilaian dosen, seperti melihat laporan penilaian dari hasil evaluasi SFQ dan memberikan serta menerima komentar.

3) *Business Value*

Penerapan sistem ini dapat memberikan nilai bisnis pada LP2MP, yaitu:

a) Mempermudah dan mengurangi waktu yang diperlukan oleh *Super admin* dan *Staff* dalam membuat survei.

b) Mempermudah para dosen serta pihak-pihak yang terkait dalam melihat laporan penilaian.

c) Mempermudah *Super Admin, Staff* dan *Part Timer* dalam melihat jadwal visitasi ke setiap kelas.

4.1.2 Analisis Kelayakan Sistem

Dalam analisis kelayakan sistem terdapat tiga perspektif, yaitu kelayakan teknis (*technical feasibility*), kelayakan ekonomi (*economic feasibility*), dan kelayakan organisasi (*organizational feasibility*).

4.1.2.1 Kelayakan Teknis (*Technical Feasibility*)

Pada kelayakan teknis, terdapat risiko-risiko yang mungkin dihadapi dalam penggunaannya. Risiko tersebut antara lain:

1. Risiko *familiarity* terhadap aplikasi adalah sedang. Sistem yang dibuat secara garis besar merupakan aplikasi yang mudah dimengerti sehingga mudah untuk digunakan oleh para penggunanya. Akan tetapi, seluruh pengguna yang terbiasa menggunakan sistem sebelumnya perlu beradapatasi dalam menggunakan sistem ini. Pada sistem sebelumnya, para *staff* LP2MP perlu membuat banyak kuesioner untuk para pengajar,

sedangkan sistem usulan tidak perlu membuat banyak kuesioner. Kemudian, para *staff* LP2MP tidak perlu mengolah data secara manual dan melakukan pengunggahan laporan hasil evaluasi dosen seperti sistem sebelumnya, karena pada sistem usulan semua hasil kuesioner telah terolah dan langsung dapat menampilkan laporan evaluasi dosen. Dengan begitu, para dosen dan pihak-pihak yang berkepentingan dapat melihat hasil evaluasi dosen. Namun, diperlukan sosialisasi yang cukup mengenai sistem usulan ini kepada seluruh pengguna, agar dapat mempercepat proses adapatasi.

2. Risiko *familiarity* terhadap teknologi adalah rendah. Sebagian besar pengguna sistem usulan ini memiliki latar belakang pendidikan yang tinggi, yaitu strata satu (S1), sehingga sudah terbiasa dengan menggunakan teknologi yang berbasis *web*.
3. Risiko mengenai ukuran proyek adalah tinggi. Sistem usulan yang dikembangkan ini cukup kompleks, karena banyak fitur yang harus dipenuhi. Selain itu, waktu yang ditetapkan untuk pembangunan sistem ini cukup singkat, yaitu lima bulan.

4.1.2.2 Kelayakan Ekonomi (*Economic Feasibility*)

Sistem usulan yang dibangun dapat menjawab kebutuhan dari *Super admin,* para *staff* dalam menyediakan sarana pembuatan survei SFQ tanpa harus membuat banyak survei. Selain itu, sistem usulan ini diharapkan dapat memberikan informasi mengenai laporan hasil evaluasi dosen, sehingga pihak-pihak terkait dapat melihat dengan mudah.

Pada sistem usulan ini, membutuhkan *hardware* dan *software* untuk menerapkan sistem. Kebutuhan *hardware* dan *software* dari sisi *server* dan *client* dapat dilihat pada tabel 4.1 dan tabel 4.2. Kemampuan minimum *hardware* dan *software* pada sistem usulan telah disesuaikan dengan spesifikasi yang telah digunakan oleh LP2MP di UPH.

Tabel 4.1 Daftar *Hardware* dan *Software Minimum* Sistem Usulan dari sisi *Server*

Hardware	
Hard Disk	500 GB
RAM	4 GB
Processor	*Intel XEON 2.4 GHz*
Software	
Operating System	*Windows Server 2012*
Database	MySQL
Web Server	Apache

Tabel 4.2 Daftar *Hardware* dan *Software Minimum* dari Sisi *Client*

Hardware	
Hard Disk	20 GB
RAM	2 GB
Processor	*Intel Core i3*
Software	
Operating System	*Windows 7*
Browser	*Google Chrome*

4.1.2.3 Kelayakan Organisasi (*Organizational Feasibility*)

Dari sudut pandang organisasi, sistem ini cukup layak untuk diterapkan. Hal ini disebabkan karena pada tahap perencanaan dan perancangan hingga pembangunan, sistem ini mendapat dukungan oleh pihak-pihak terkait seperti *Head of Institutional Survey and System Improvement* dan para *Staff* LP2MP. Sistem ini dikembangkan, berdasarkan kebutuhan dan permintaan dari pihak-pihak terkait.

4.2 Tahap Analisis

Tahap analisis mencakup *user requirement*, pemodelan fungsional dan pemodelan struktural.

4.2.1 *User Requirements*

User requirements menjelaskan mengenai kebutuhan-kebutuhan dari *user* untuk sistem usulan. Pada *user requirements* terdapat meliputi *functional requirements* dan *non-functional requirements.*

4.2.1.1 *Functional Requirements*

Functional requirement mendeskripsikan tentang proses atau informasi mengenai layanan yang disediakan oleh sistem usulan. *functional requirement* dari sistem usulan akan dijelaskan berdasarkan peran dari pengguna.

1. *Login* untuk pengguna. Dalam sistem ini terdapat enam tipe *user*, yaitu *staff, part timer,* dosen, kaprodi, dekan dan *vice president. User* yang telah terdaftar dalam sistem dapat masuk ke dalam sistem untuk menggunakan fitur-fitur yang telah disediakan berdasarkan tipe *user* yang didaftarkan.
2. *User* dapat mengubah *profile* dimana *user* dapat mengubah *password* dan gelar miliknya melalui menu ubah *profile*
3. *Staff* dapat mengelola data *user,* seperti menambah dan mengubah data *user.*
4. *Staff* dapat mengelola data *campus*, seperti menambah dan mengubah data *faculty*, *program study*, dan *course.*
5. Mengelola *question set*
    a) *Staff* dapat melakukan penambahan dan pengubahan survei, seperti pertanyaan survei dan pilihan jawaban survei.
    b) *Staff* dan *part timer* dapat mengaktifkan dan menonaktifkan survei.
6. *Staff* dan *part timer* dapat melihat hasil survei tanpa mengubah hasil survei tersebut.
7. Melihat laporan penilaian
    a) *Staff* dapat melihat laporan penilaian seluruh dosen.
    b) Dosen hanya dapat melihat laporan penilaian miliknya sendiri yang berupa nilai rata-rata dan laporan berupa komentar dari mahasiswa.
    c) Kaprodi dapat melihat laporan penilaian seluruh dosen yang terdapat di jurusan yang dipimpinnya.

d) Dekan dapat melihat laporan penilaian seluruh dosen di setiap program studi yang terdapat di fakultas yang di pimpinnya.
e) *Vice president* dapat melihat laporan seluruh fakultas.

8. Mengelola jadwal
   a) *Staff* dapat melakukan penambahan dan pengubahan jadwal untuk visitasi.
   b) *Staff* dan *part timer* dapat melihat jadwal untuk visitasi ke setiap kelas.
   c) *Part timer* hanya dapat mengubah status visitasi pada jadwal visitasi.
9. Mahasiswa hanya dapat mengisi survei penilaian dosen.

4.2.1.2 *Non-Functional Requirements*

Pada sistem usulan ini memiliki *non-functional requirements* sebagai berikut:

1. *Operational*:
   a) Sistem dapat berjalan dengan menggunakan perangkat yang mendukung penggunaan *web browser*.
   b) Sistem dapat berjalan menggunakan *web browser* Google Chrome versi 66.0.3359.170, Mozilla Firefox versi 59.0.1, Internet Explorer 11.0.49, dan Microsoft Edge 41.16299.15.0.
2. *Performance:*
   a) Sistem dapat diakses selama 24 jam dalam satu hari.
   b) *Database* dapat diperbaharui secara *real-time.*
   c) Sistem dapat melakukan *update* dan penambahan data ke *database.*
3. *Security*
   a) Sistem hanya dapat diakses oleh *user* yang telah terdaftar di dalam sistem, sehingga *user* perlu melakukan *log in* dengan memasukkan *username* dan *password* .
   b) Sistem dapat membedakan level akses *Super Admin* dengan *user* lainnya. Sehingga, *user* dapat mengakses sistem sesuai dengan jabatan yang telah ditentukan pada sistem.

4. *Cultural dan Political*

Sistem usulan ini menggunakan bahasa Inggris sebagai bahasa pengantar mengikuti sistem lainnya yang ada di UPH.

4.2.2 Pemodelan Fungsional

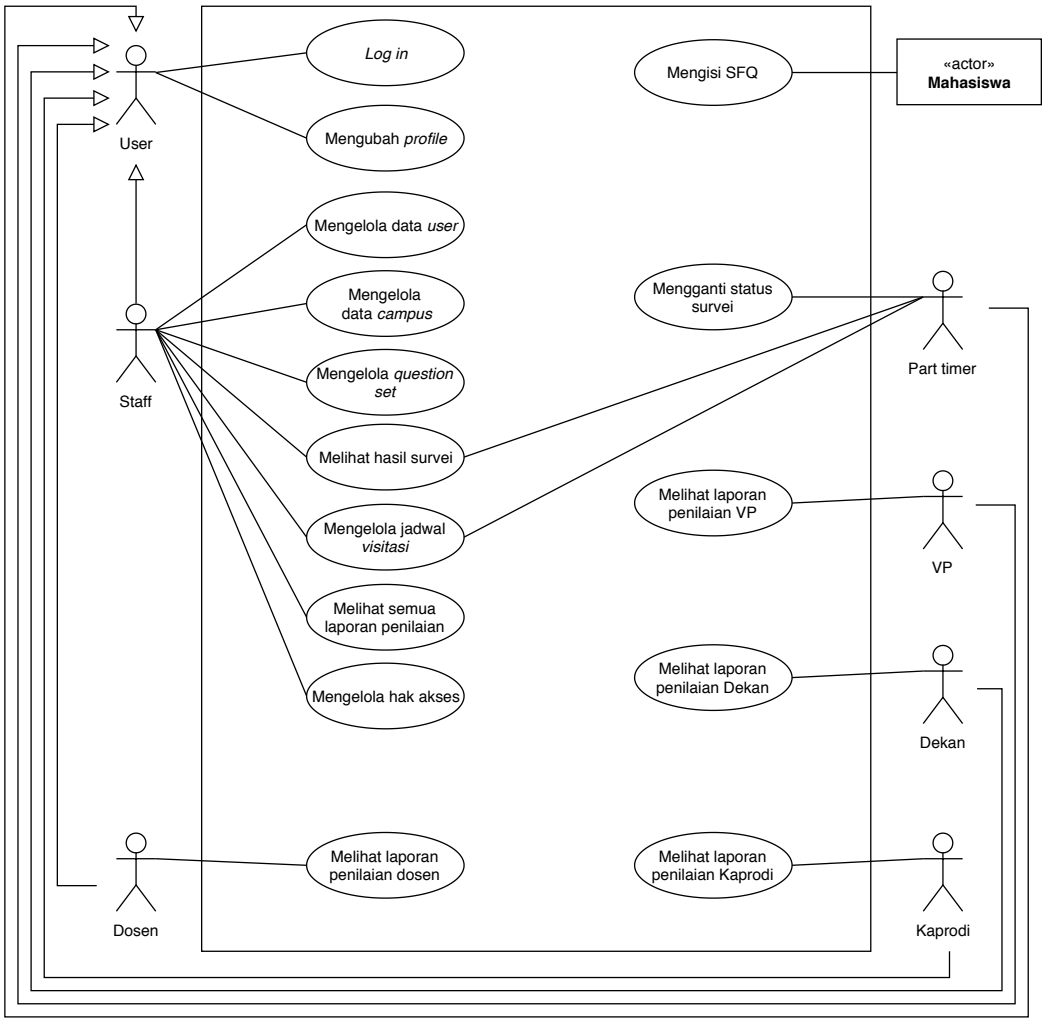
Pemodelan fungsional menjelaskan tentang interaksi *user* dengan sistem usulan ini. Dalam pemodelan fungsional dideskripsikan menjadi dua hal, yaitu *use case* dan *activity diagram.*

4.2.2.1 *Use Case Diagram*

Use case diagram menggambarkan fungsi-fungsi yang dapat dilakukan oleh *user* ketika mengakses sistem. Para pengguna akan terbagi menjadi beberapa *aktor,* yaitu:

1. *Staff* merupakan aktor yang menggambarkan pegawai di divisi *Survey Center* yang memiliki beberapa otoritas dalam pengelolaan data SFQ.
2. *Part Timer* merupakan aktor yang menggambarkan sebagai mahasiswa yang bekerja paruh waktu di divisi *Survey Center.*
3. Dosen merupakan aktor yang menggambarkan sebagai pengajar di UPH.
4. Kaprodi merupakan aktor yang menggambarkan sebagai Ketua Program Studi di UPH.
5. Dekan merupakan aktor yang menggambarkan pemimpin fakultas di UPH.
6. *Vice President* merupakan aktor yang menggambarkan sebagai *Vice President* di UPH.
7. Mahasiswa merupakan aktor yang menggambarkan seluruh mahasiswa di UPH.

Berikut adalah Gambar 4.1 yang menunjukkan *use case diagram* dari sistem usulan:



Gambar 4.1 *Use Case Diagram* Sistem Usulan

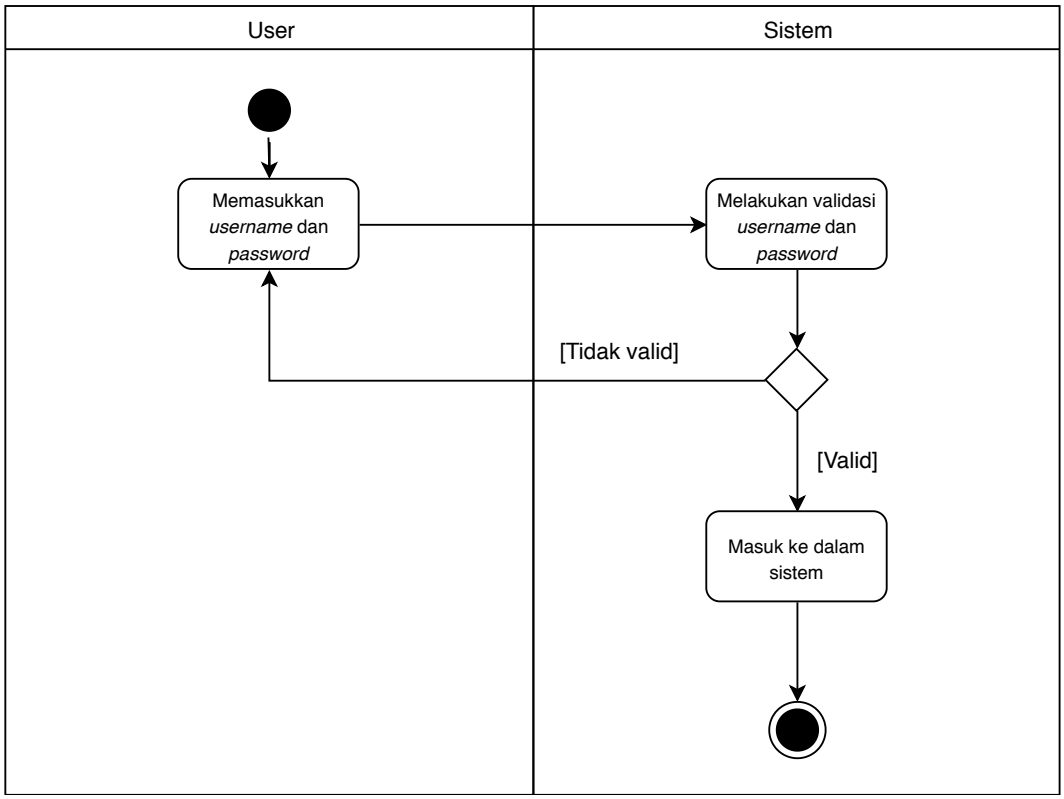
4.2.2.2 *Activity diagram*

Activity diagram merupakan alur dari satu aktivitas ke aktivitas lainnya yang dapat dilakukan oleh sistem. Berikut ini merupakan kumpulan-kumpulan gambar *Activity diagram* pada sistem usulan.

1. Proses *Log In*

   Gambar 4.2 menunjukkan *activity diagram* dari proses *log in.* Pada proses ini *user* harus memasukkan *username* dan *password* yang sesuai, kemudian sistem akan melakukan validasi. Apabila *username* dan *password* tidak sesuai atau tidak valid maka *user* harus memasukkan kembali *username* dan

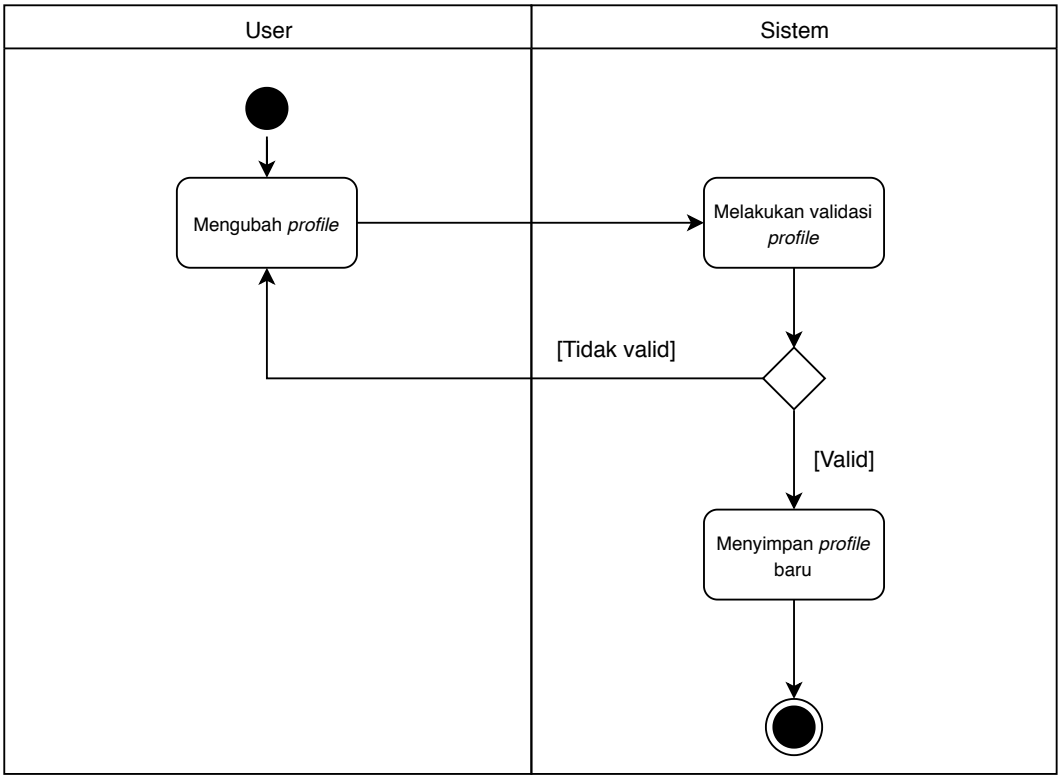
password yang benar. Namun, apabila *user* telah memasukkan *username* dan *password* yang sesuai atau valid maka *user* dapat *log in* dan mengakses sistem.



Gambar 4.2 *Activity Diagram Login Process*

2. Proses Mengubah *Profile*

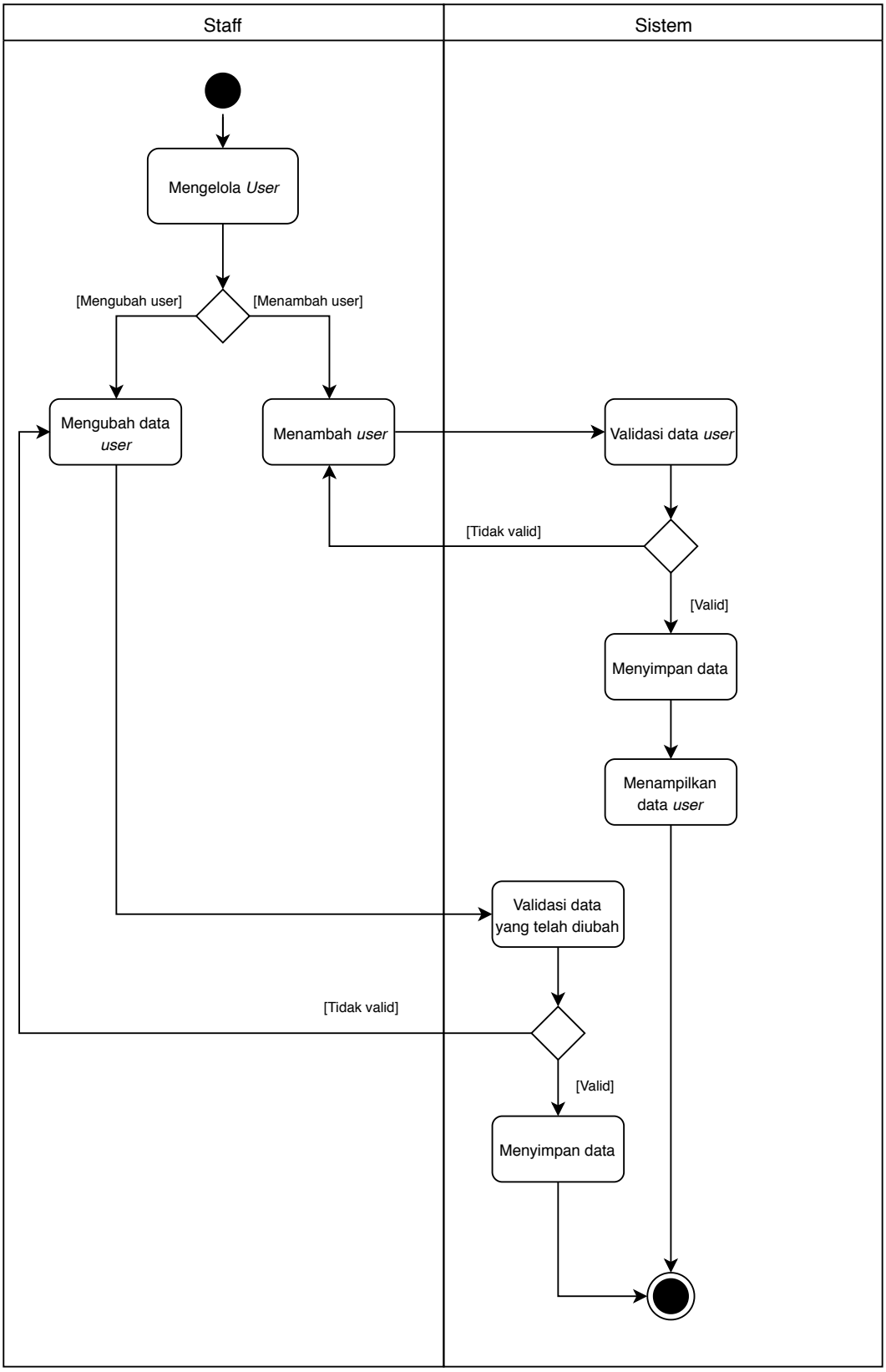
   Pada Gambar 4.3 menunjukkan *activity diagram* proses mengubah *profile*. Dalam proses ini *user* dapat mengubah *password*, gelar depan, dan gelar belakang yang dimiliki. Kemudian sistem akan melakukan validasi *password* yang telah dimasukkan. Jika tidak valid, maka *user* harus memasukkan kembali *password* yang sesuai. Dan apabila sudah valid maka sistem akan menyimpan *password*, gelar depan, dan gelar belakang yang baru.



Gambar 4.3 *Activity Diagram Proses* Mengubah *Profile*

3. Proses Mengelola Data *User*

   Pada Gambar 4.4 menunjukkan *activity diagram* mengelola data *user*. Terdapat dua proses yang dapat dilakukan dalam mengelola data *user* yaitu menambah dan mengubah *data user.* Pada proses menambah data *user, Staff* harus mengisi data *user,* kemudian sistem akan melakukan validasi data tersebut. Jika data yang dimasukkan tidak valid, maka sistem akan menampilkan notifikasi dan *user* harus mengisi kembali dengan format yang sesuai. Apabila data yang telah dimasukkan sudah valid, sistem akan menyimpan data tersebut dan menampilkan notifikasi berhasil. Selanjutnya, pada proses mengubah data *user, Staff* harus memilih data *user* dan kemudian, mengubah data yang ingin diubah. Data yang telah diubah akan divalidasi oleh sistem. Apabila data yang dimasukkan valid maka sistem akan menyimpan data. Apabila data yang dimasukkan tidak valid, maka *Staff* harus memasukkan kembali data yang sesuai.



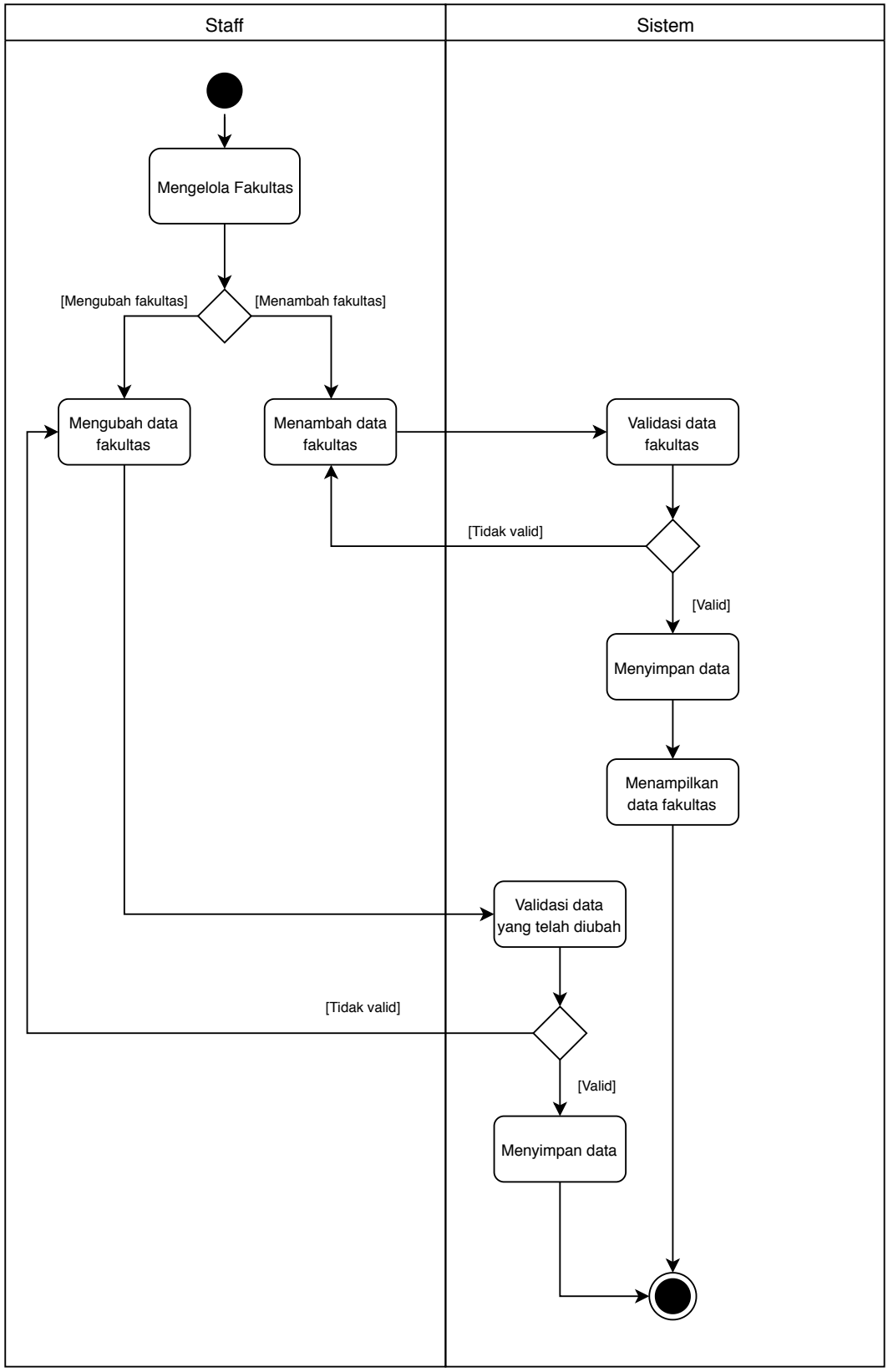
Gambar 4.4 *Activity Diagram* Mengelola *Data User*

4. Mengelola Data *Campus*

   Dalam mengelola data *campus* terbagi menjadi tiga proses yaitu, mengelola fakultas, mengelola program studi dan mengelola *course*. Berikut ini *activity diagram* dari mengelola data *campus.*

   a. Mengelola Fakultas

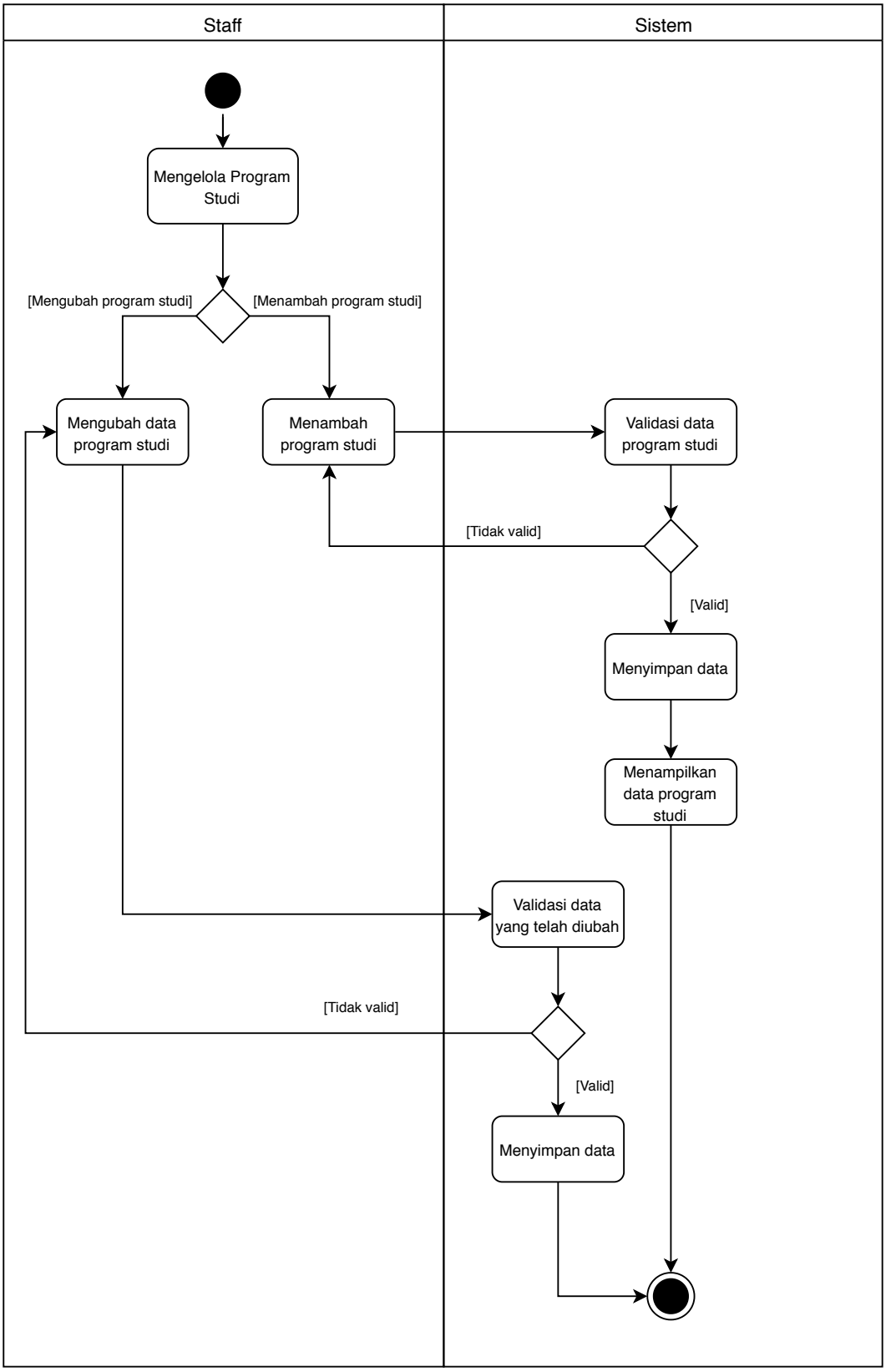
      Pada Gambar 4.5 merupakan *activity diagram* dari mengelola fakultas. Dalam mengelola fakultas, *Staff* dapat melakukan penambahan dan pengubahan data fakultas. Jika ingin menambah data, *Staff* perlu mengisi *form* terlebih dahulu, selanjutnya sistem akan melakukan validasi data yang telah dimasukkan. Jika tidak valid, *Staff* harus memasukkan data yang benar dan apabila sudah valid sistem akan menyimpan data tersebut. Kemudian untuk pengubahan data *Staff* harus memilih data yang ingin diubah. Setelah itu *Staff* dapat mengubah data yang diinginkan dan sistem akan melakukan validasi. Jika tidak valid *Staff* harus memasukkan data kembali dengan benar. Jika sudah valid, sistem akan menyimpan data.



Gambar 4.5 *Activity Diagram* Mengelola Fakultas

b) Mengelola Program Studi

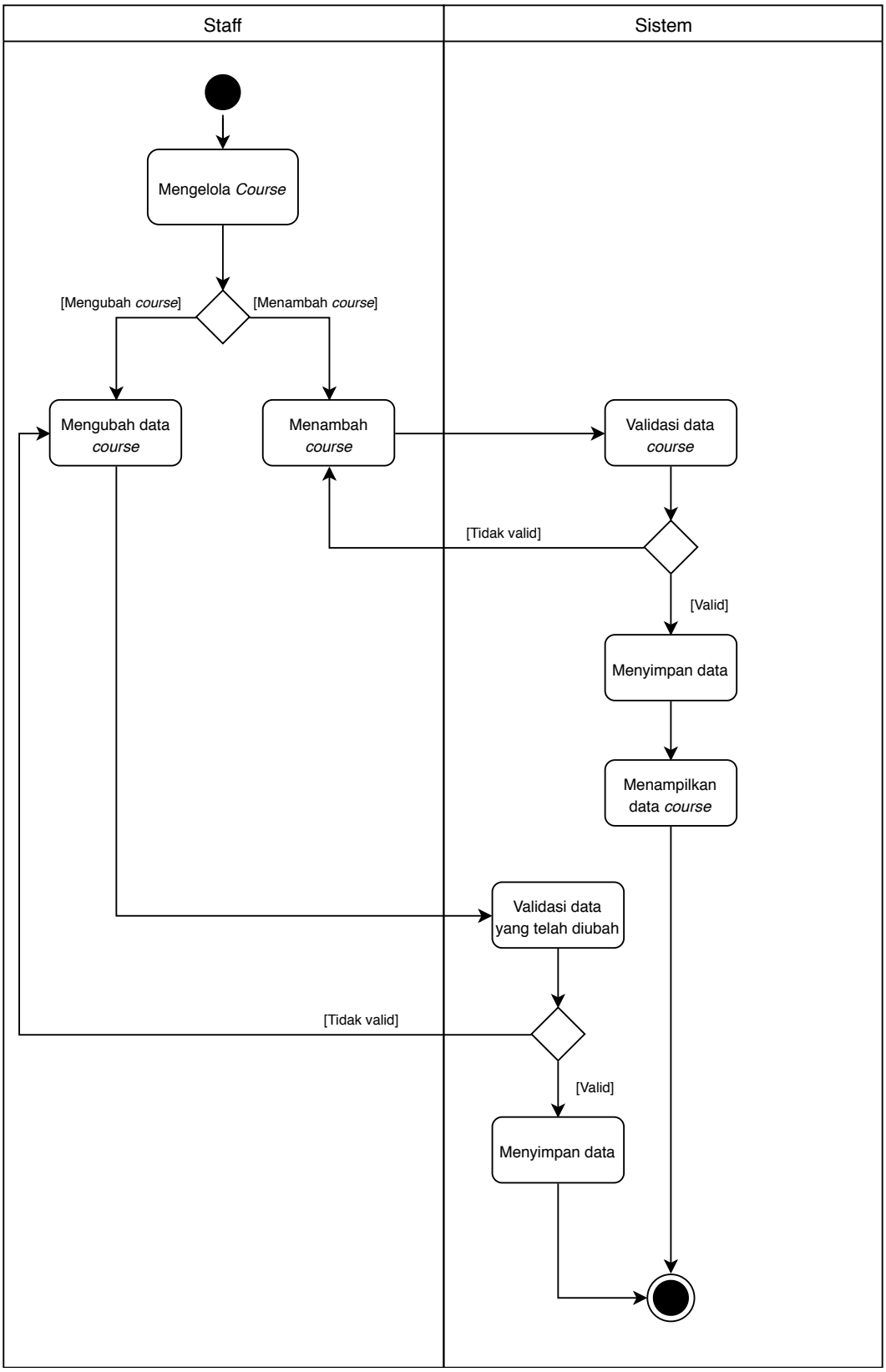
Pada Gambar 4.6 merupakan *activity diagram* dari mengelola program studi. Dalam mengelola program studi, *Staff* dapat melakukan penambahan dan pengubahan data program studi. Jika ingin menambah data, *Staff* perlu mengisi *form* terlebih dahulu, selanjutnya sistem akan melakukan validasi data yang telah dimasukkan. Jika tidak valid, *Staff* harus memasukkan data yang benar dan apabila sudah valid sistem akan menyimpan data tersebut. Kemudian untuk pengubahan data *Staff* harus memilih data yang ingin diubah. Setelah itu *Staff* dapat mengubah data yang diinginkan dan sistem akan melakukan validasi. Jika tidak valid *Staff* harus memasukkan data kembali dengan benar. Jika sudah valid, sistem akan menyimpan data.



Gambar 4.6 *Activity Diagram* Mengelola *Program Studi*

c) Mengelola *Course*

Gambar 4.7 menunjukkan *activity diagram* dari mengelola *course*. Dalam mengelola *course*, terdapat dua proses yang dapat dilakukan, yaitu penambahan dan pengubahan data *course*. Jika ingin menambah data, *Staff* perlu mengisi *form* terlebih dahulu, selanjutnya sistem akan melakukan validasi data. Jika tidak valid, *Staff* harus memasukkan data yang benar dan apabila sudah valid sistem akan menyimpan data tersebut. Kemudian untuk pengubahan data *Staff* harus memilih data yang ingin diubah. Setelah itu *Staff* dapat mengubah data yang diinginkan dan sistem akan melakukan validasi. Jika tidak valid *Staff* harus memasukkan data dengan benar. Jika sudah valid, sistem akan menyimpan data.



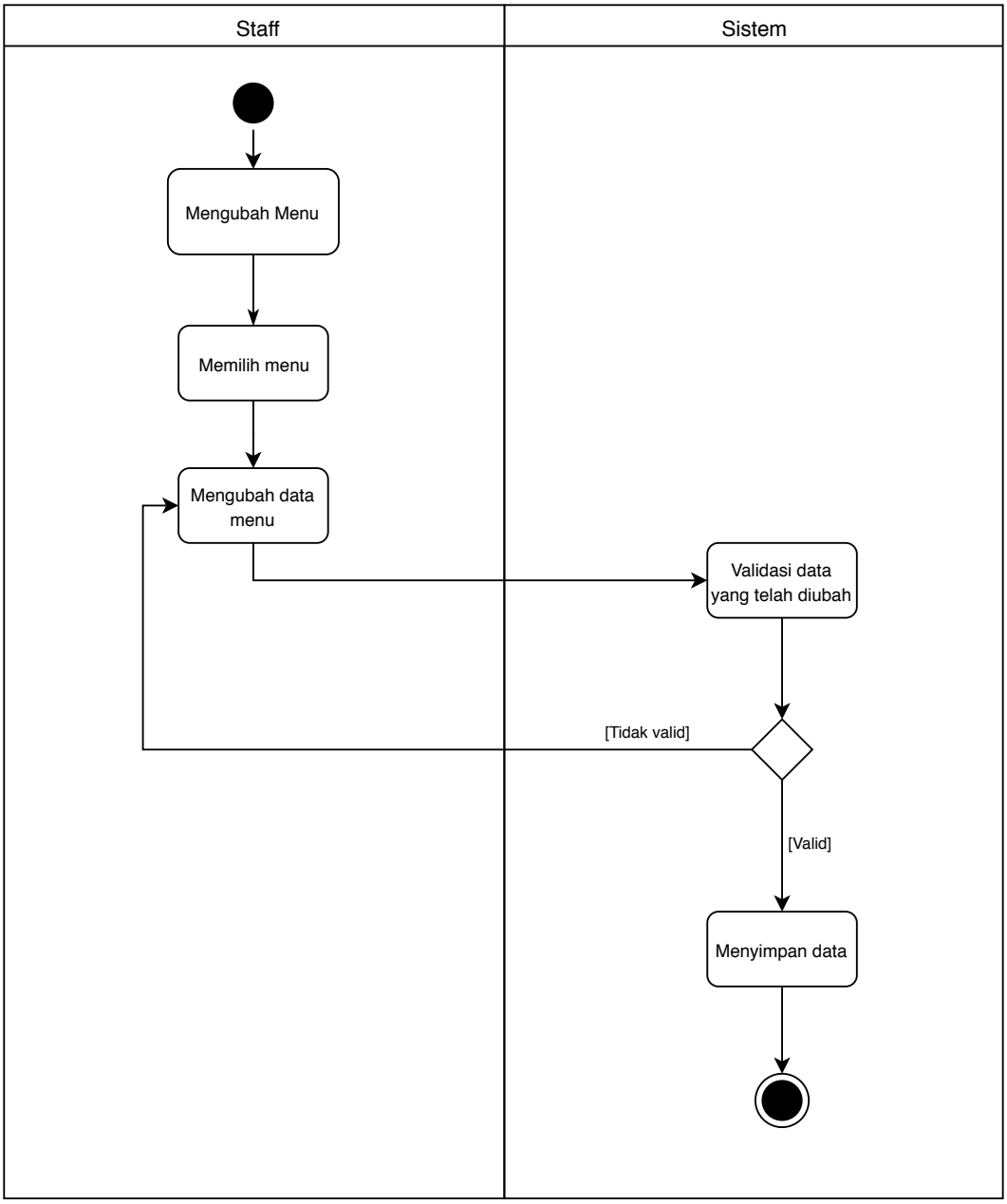
Gambar 4.7 *Activity Diagram* Mengelola *Course*

5. Mengelola Hak Akses

   Dalam mengelola data hak akses terbagi menjadi tiga proses yaitu, mengelola hak akses, mengubah menu dan mengubah sub menu. Berikut ini *activity diagram* dari mengelola data hak akses.

   a) Mengubah Menu

      Pada Gambar 4.8 merupakan activity diagram dari mengelola menu. Dalam mengelola menu, terdapat satu proses yang dapat dilakukan, yaitu pengubahan menu. Jika ingin mengubah menu *Staff* harus memilih data yang ingin diubah. Setelah itu *Staff* dapat mengubah data yang diinginkan dan sistem akan melakukan validasi. Jika tidak valid *Staff* harus memasukkan data dengan benar. Jika sudah valid, sistem akan menyimpan data.

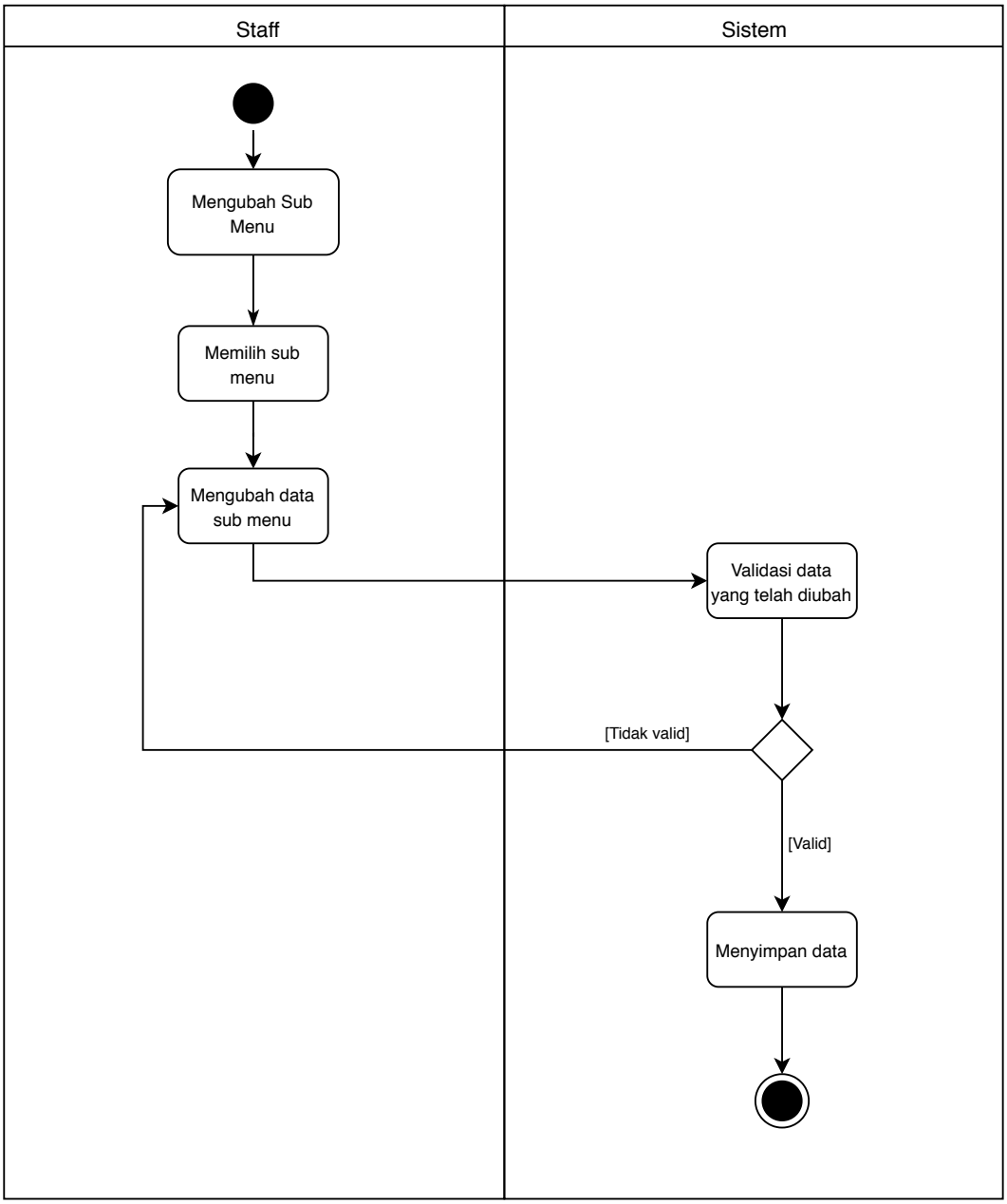


Gambar 4.8 *Activity Diagram* Mengubah Menu

b) Mengubah Sub Menu

Pada Gambar 4.9 merupakan activity diagram dari mengelola sub menu. Dalam mengelola sub menu, terdapat satu proses yang dapat dilakukan, yaitu pengubahan sub menu. Untuk mengubah sub menu *Staff* harus memilih data yang ingin diubah. Setelah itu *Staff* dapat mengubah data yang diinginkan dan sistem akan melakukan validasi.

Jika tidak valid *Staff* harus memasukkan data dengan benar. Jika sudah valid, sistem akan menyimpan data.

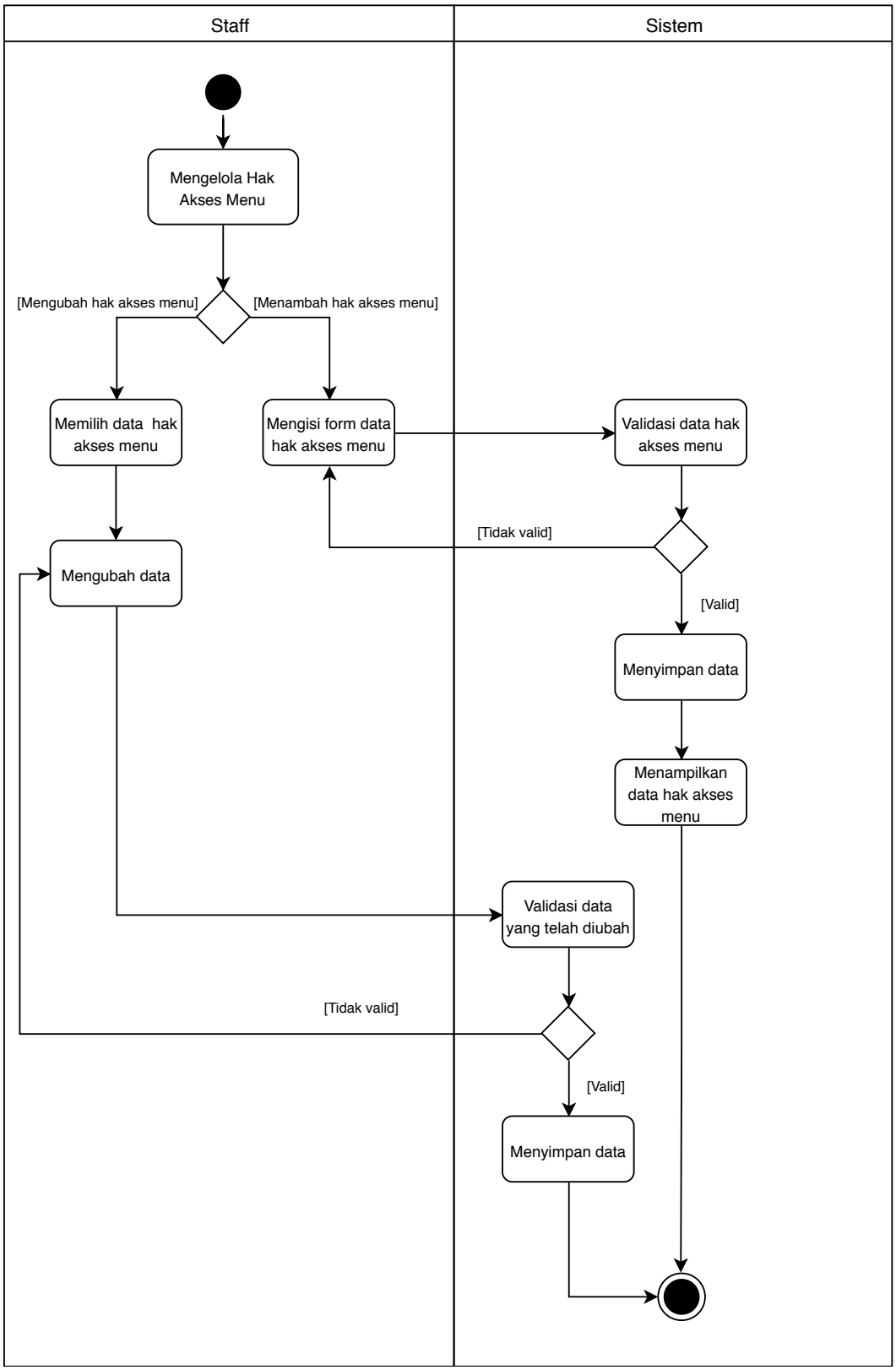


Gambar 4.9 *Activity Diagram* Mengubah *Sub Menu*

c) Mengelola Hak Akses Menu

Pada Gambar 4.10 merupakan activity diagram dari mengelola hak akses menu. Dalam mengelola hak akses menu, terdapat dua proses yang dapat dilakukan, yaitu penambahan dan pengubahan hak akses

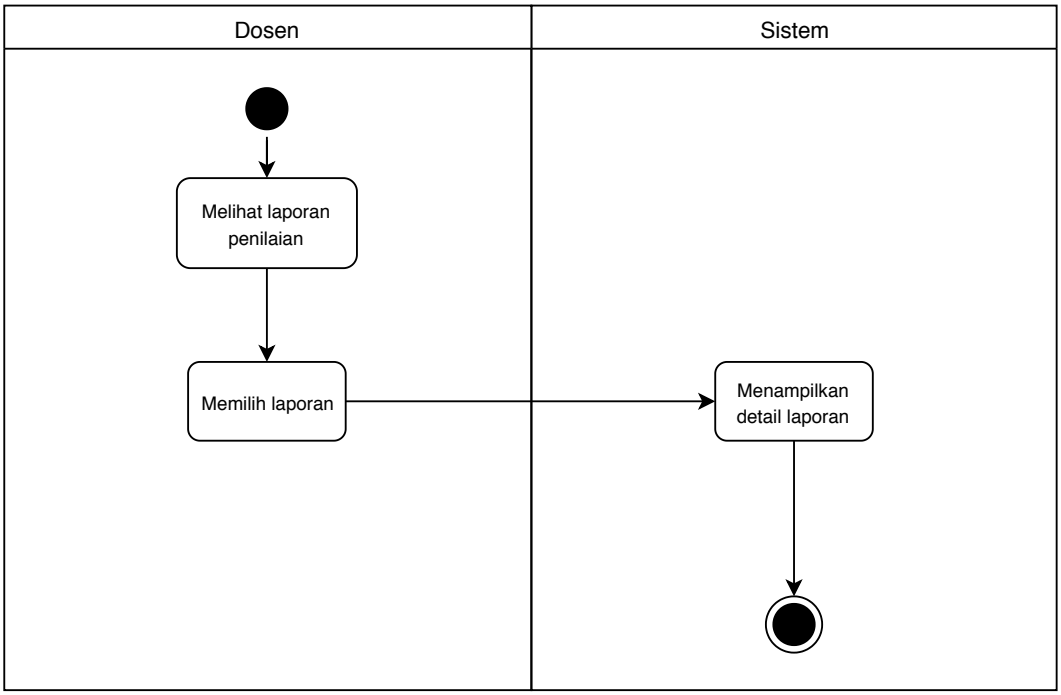
menu. Jika ingin menambah hak akses menu, *Staff* perlu mengisi *form* terlebih dahulu, selanjutnya sistem akan melakukan validasi data. Jika tidak valid, *Staff* harus memasukkan data yang benar dan apabila sudah valid sistem akan menyimpan data tersebut. Kemudian untuk pengubahan data *Staff* harus memilih data yang ingin diubah. Setelah itu *Staff* dapat mengubah data yang diinginkan dan sistem akan melakukan validasi. Jika tidak valid *Staff* harus memasukkan data dengan benar. Jika sudah valid, sistem akan menyimpan data.



Gambar 4.10 *Activity Diagram* Mengelola Hak *Akses Menu*

6. Proses Melihat Laporan Penilaian Dosen

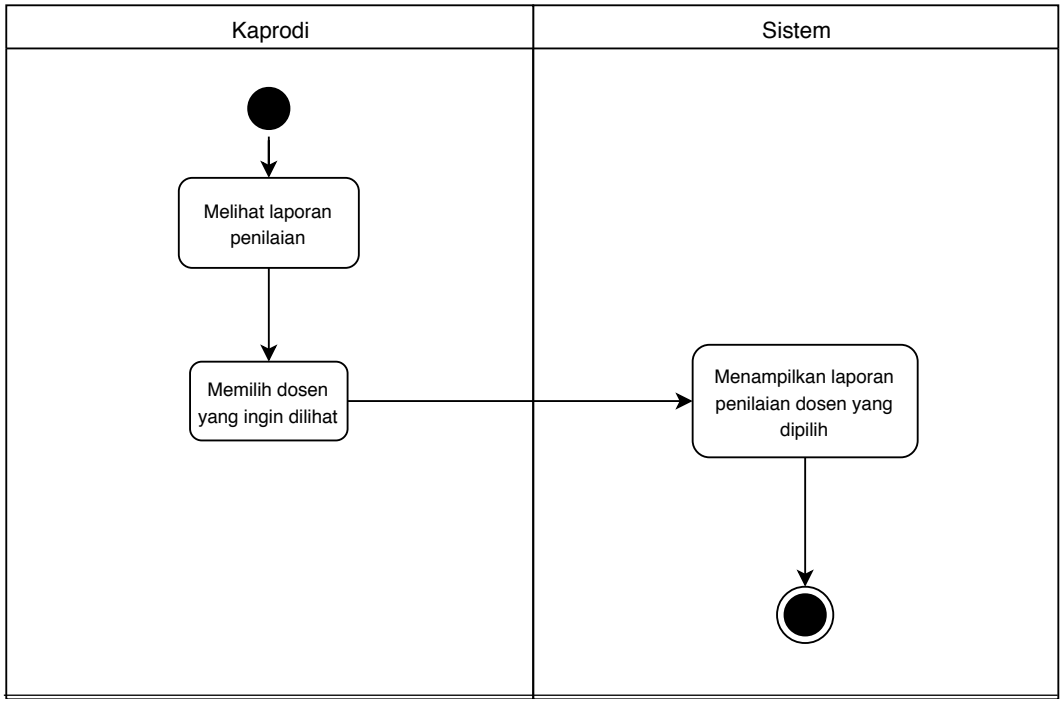
   Activity diagram melihat laporan dosen ditunjukkan pada Gambar 4.11. Dosen perlu memilih laporan yang ingin dilihat terlebih dahulu. Selanjutnya, sistem akan menampilkan detail laporan yang telah dipilih.



Gambar 4.11 *Activity Diagram* Melihat Laporan Dosen

7. Proses melihat laporan Kaprodi

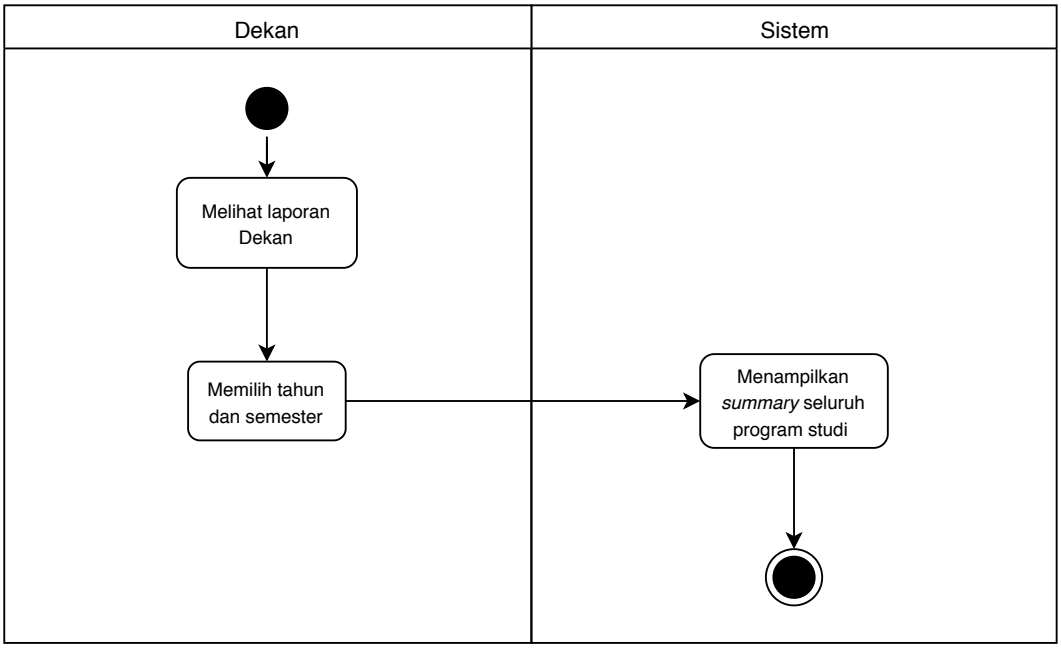
   Gambar 4.12 menunjukkan proses melihat laporan kaprodi Pada proses ini, kaprodi dapat melihat laporan penilaian hasil evaluasi dosen secara detail. Kaprodi perlu memilih dosen yang ingin dilihat kemudian sistem akan menampilkan laporan penilaian dosen secara detail.



Gambar 4.12 *Activity Diagram* Melihat Laporan Kaprodi

8. Proses melihat laporan Dekan

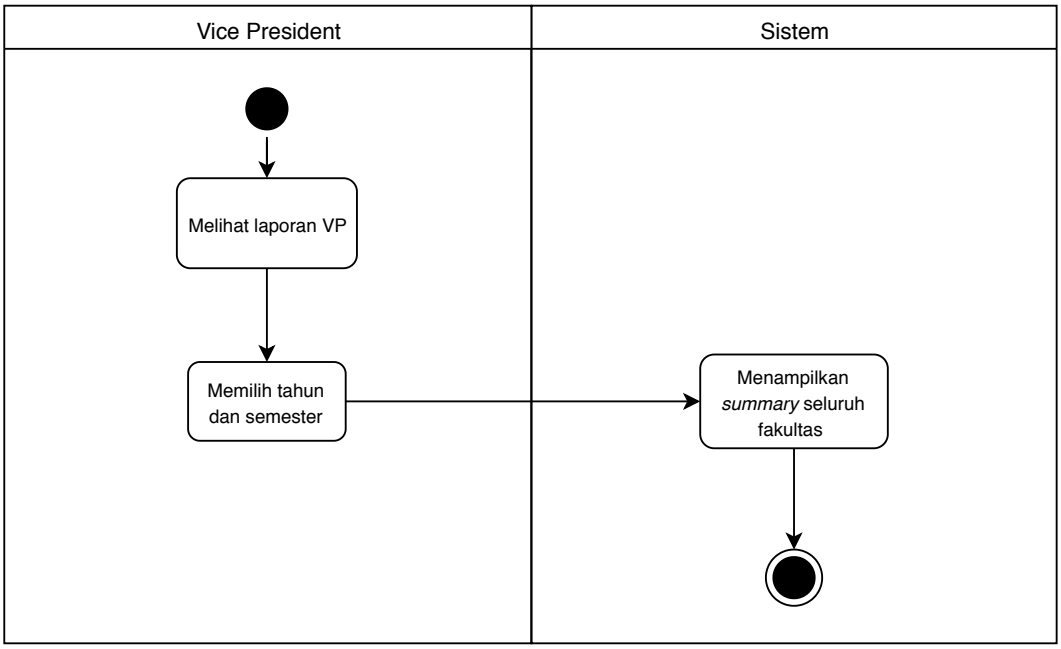
   Pada Gambar 4.13 menunjukkan *activity diagram* untuk melihat laporan Dekan. Pada proses ini Dekan dapat memilih semester yang diinginkan dan sistem akan menampilkan laporan seluruh program studi yang dipimpin.



Gambar 4.13 *Activity Diagram* Melihat Laporan Dekan

9. Proses melihat laporan *Vice President*

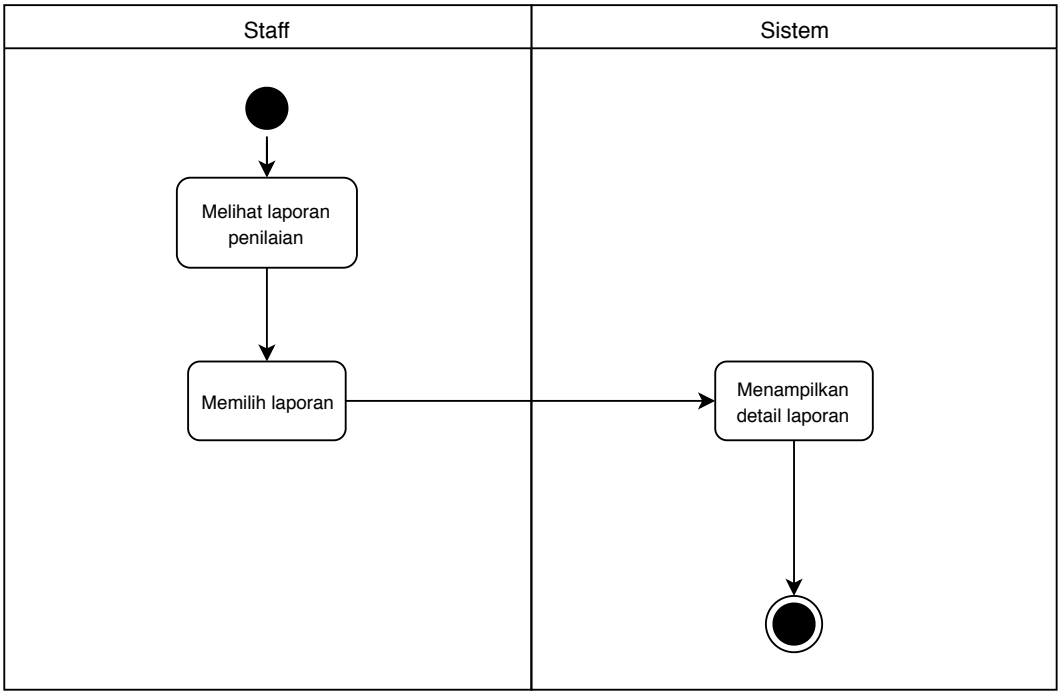
   Activity diagram untuk melihat laporan Dekan ditunjukkan pada Gambar 4.14. Pada proses ini VP dapat memilih semester yang diinginkan dan sistem akan menampilkan laporan seluruh fakultas.



Gambar 4.14 *Activity Diagram* Melihat Laporan *VP*

10. Proses Melihat Laporan Penilaian *Staff*

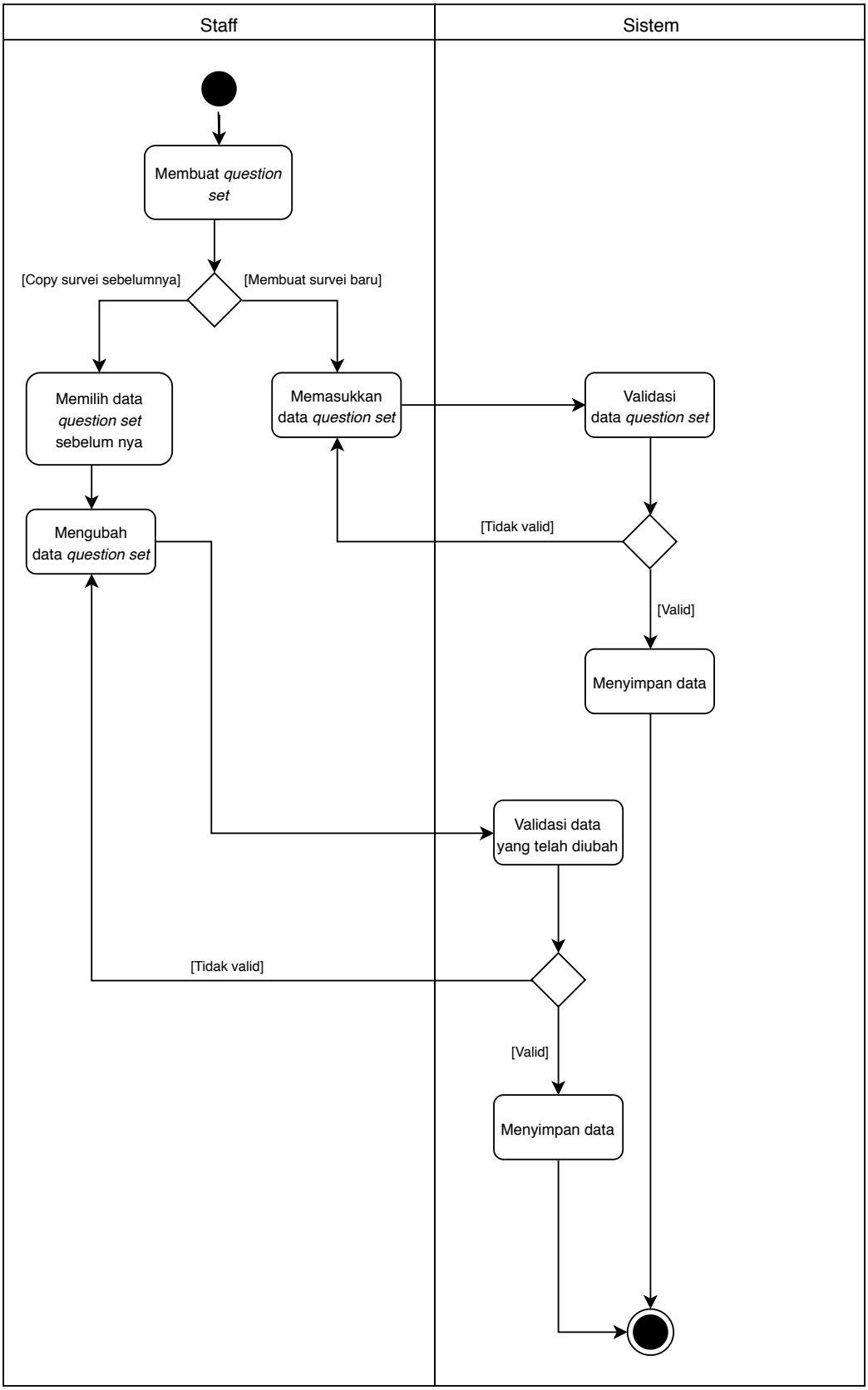
Activity diagram melihat laporan *staff* ditunjukkan pada Gambar 4.15. *Staff* perlu memilih laporan yang ingin dilihat terlebih dahulu. Selanjutnya, sistem akan menampilkan detail laporan yang telah dipilih.



Gambar 4.15 *Activity Diagram* Melihat Laporan *Staff*

11. Mengelola *Question Set*

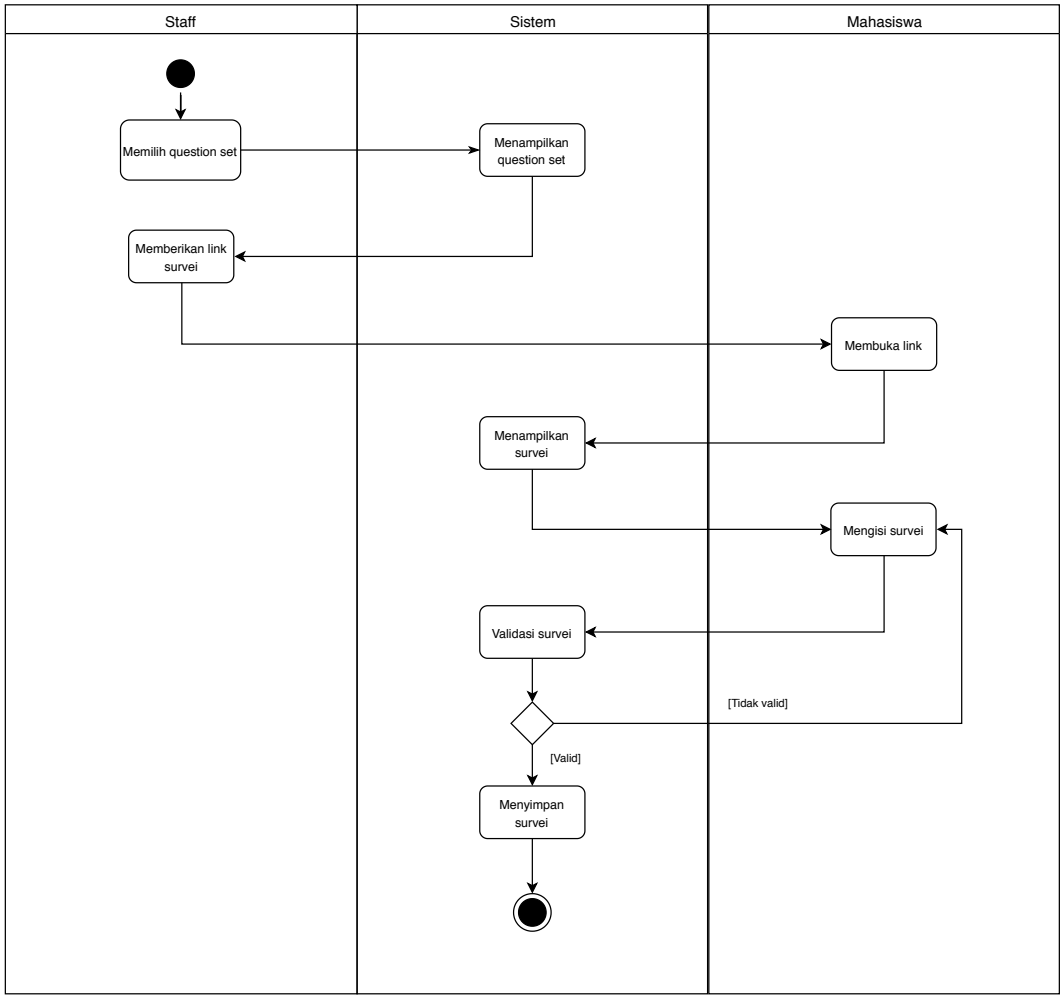
Proses mengelola *question set* ditunjukkan pada Gambar 4.16. Dalam proses ini *Staff* memilih untuk membuat survei baru atau mengambil data survei yang sebelumnya baru setelah itu diubah data yang seperlunya saja. Setelah itu, sistem akan melakukan validasi terhadap data yang dimasukkan kedalam survei baru. Jika tidak valid, maka *Staff* harus mengisi kembali dengan format yang benar. Jika valid sistem akan menyimpan survei.



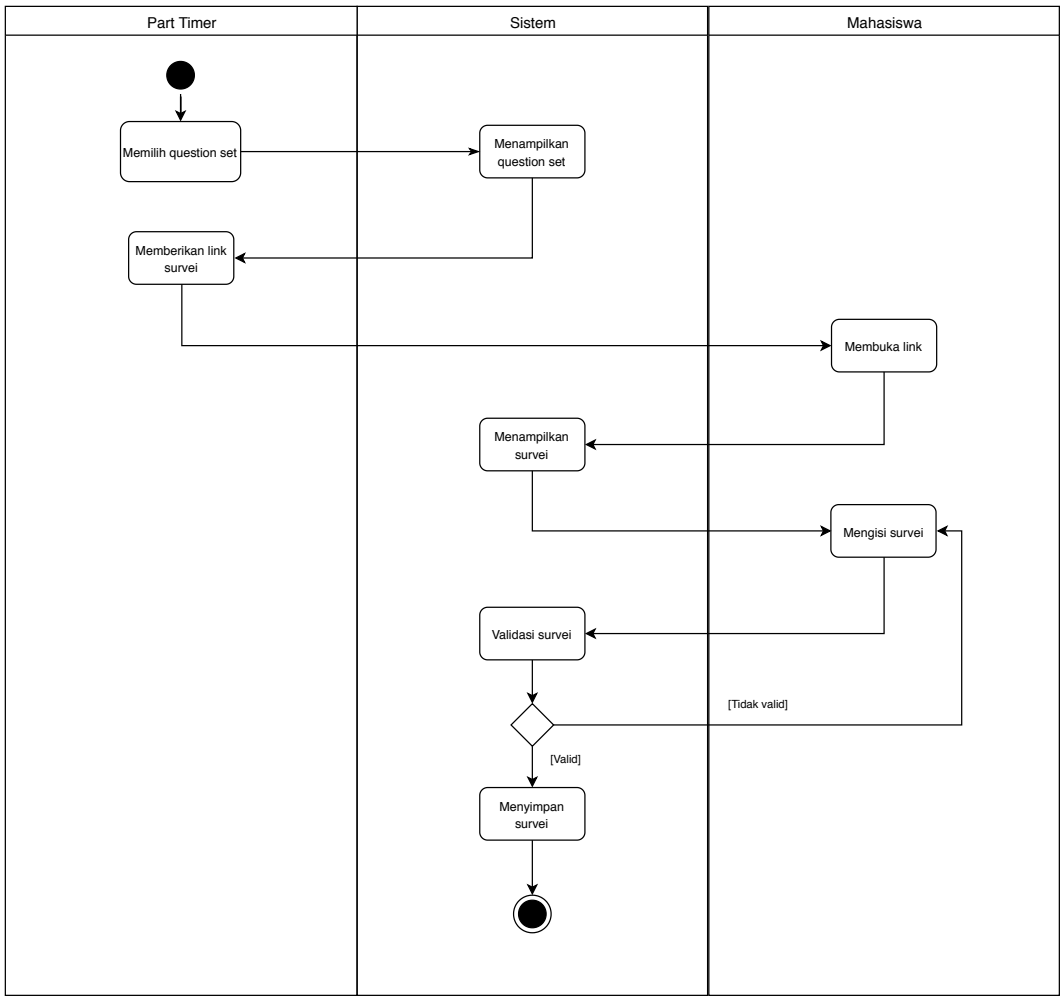
Gambar 4.16 *Activity Diagram* Mengelola *Question Set*

12. Proses Mengisi Survei

Proses mengisi survei ditunjukkan pada Gambar 4.17 dan 4.18. Dalam proses ini *Staff* atau *Part Timer* melihat jadwal terlebih dahulu dan kemudian mahasiswa akan diberikan *link* survei yang sesuai dengan jadwal kelas tersebut. Mahasiswa akan membuka *link* dan mengisi kuesioner yang telah disediakan. Setelah itu, sistem akan melakukan validasi terhadap kuesioner yang telah diisi. Jika kuesioner tidak valid, maka mahasiswa harus mengisi kuesioner dengan benar dan jika telah valid, maka sistem akan menyimpan data.



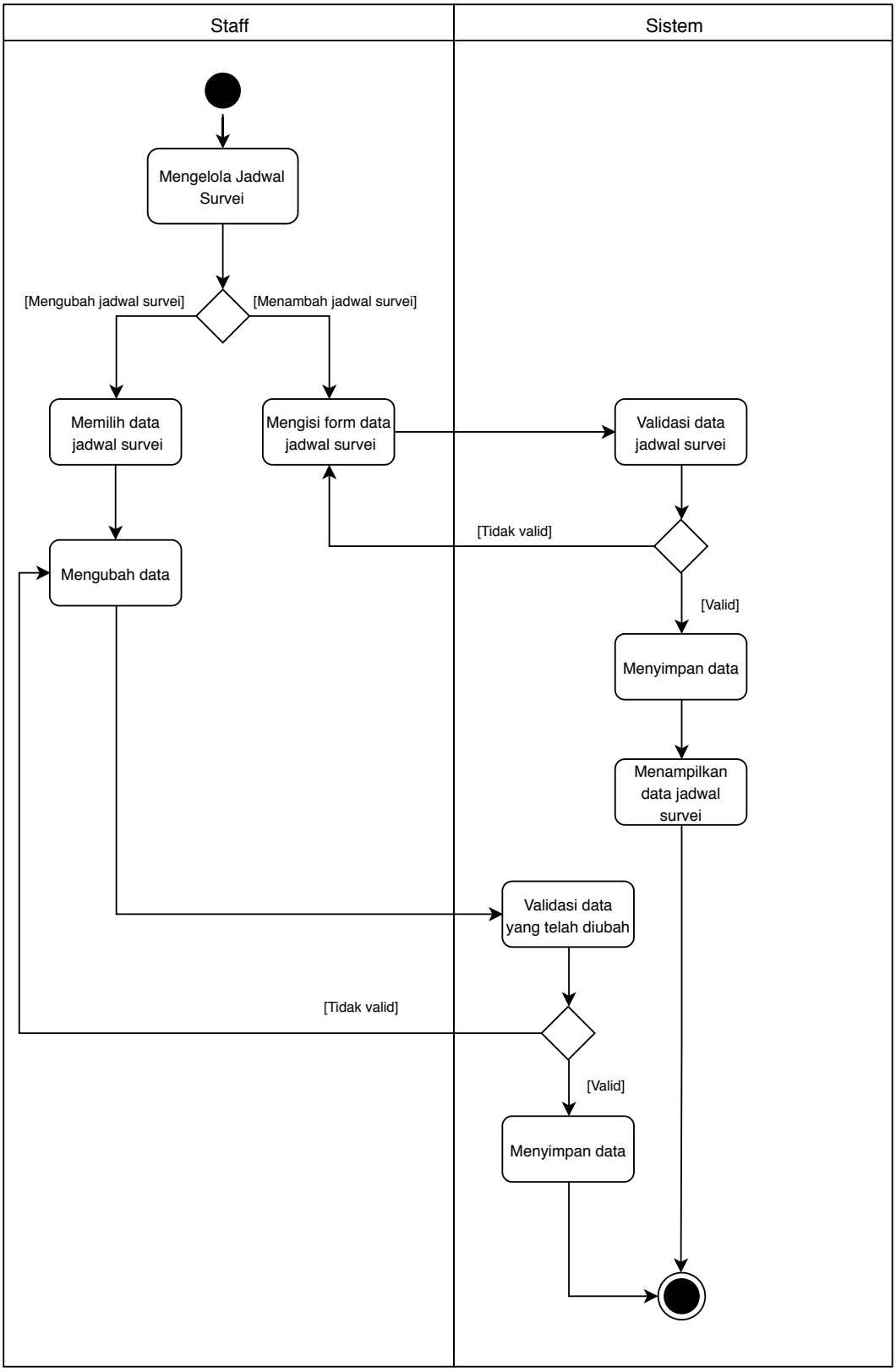
Gambar 4.17 *Activity Diagram Proses* Mengisi Survei Staff



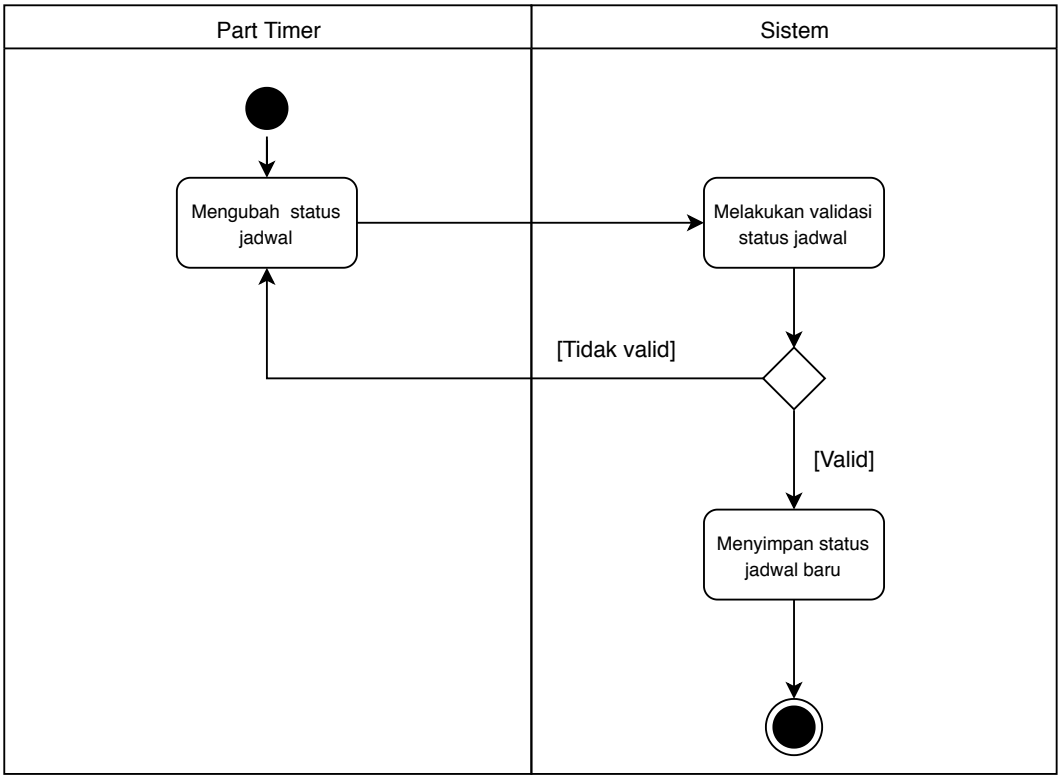
Gambar 4.18 *Activity Diagram Proses* Mengisi Survei *Part Timer*

13. Mengelola Jadwal

Proses mengelola jadwal ditunjukkan pada Gambar 4.19 dan 4.20. Dalam proses ini *staff* dapat menambah atau mengubah jadwal sedangkan *part timer* hanya dapat melihat dan mengganti status survei. Untuk menambah jadwal *staff* perlu mengisi form terlebih dahulu setelah itu akan di konfirmasi oleh sistem jika data yang dimasukkan valid maka data akan masuk ke sistem, jika tidak maka *staff* haru mengisi ulang jadwal kembali. Untuk mengganti *staff* perlu memilih jadwal yang ingin dirubah dan mengisi form ubah jadwal. *Part timer* dapat mengubah status jadwal dengan memilih jadwal tersebut dan mengubah status nya.



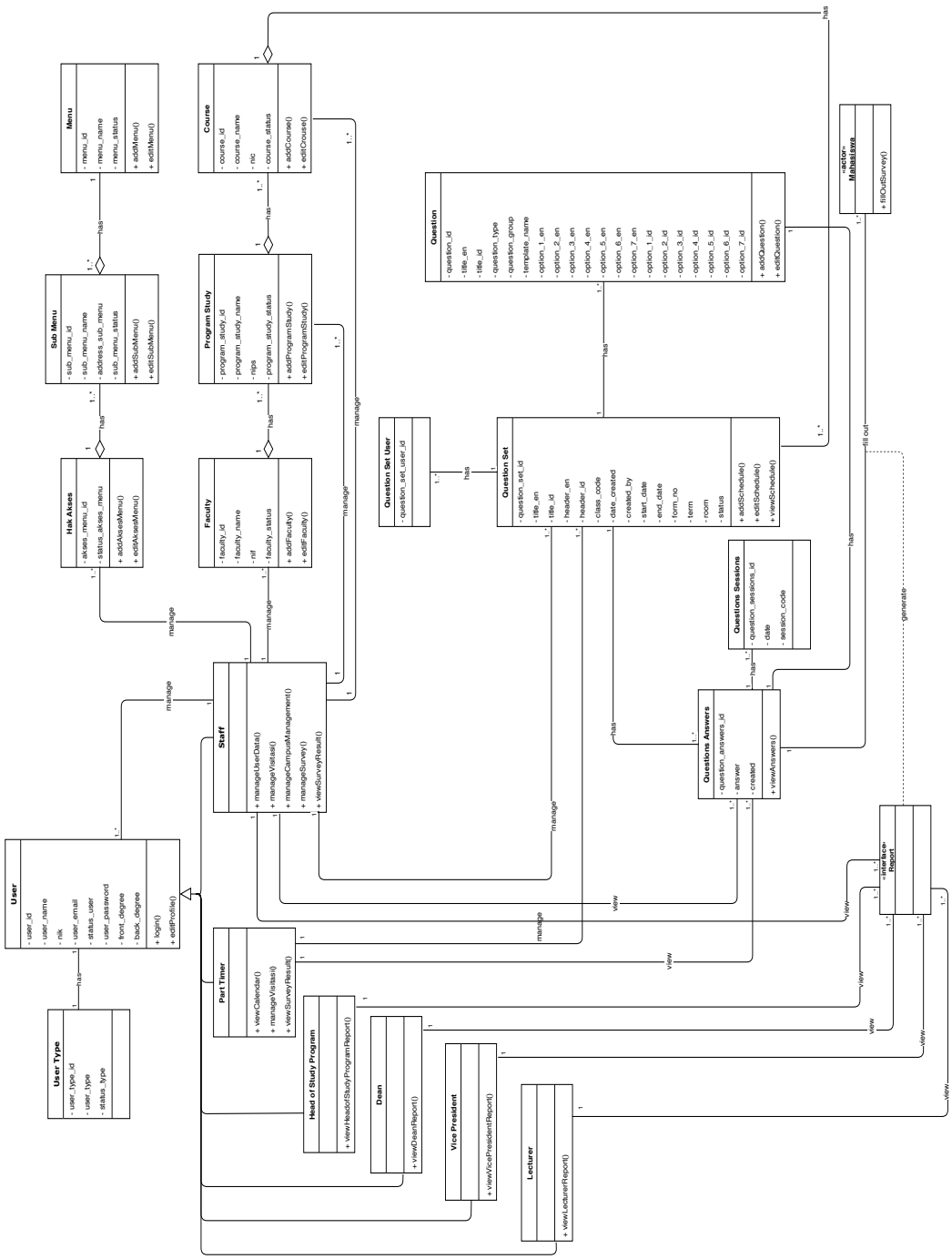
Gambar 4.19 *Activity Diagram* Mengelola Jadwal Survei



Gambar 4.20 *Activity Diagram* Mengubah Status Jadwal Survei

4.2.3 Pemodelan Struktural

Berikut ini adalah struktur *class diagram* untuk sistem usulan yang ditunjukkan pada Gambar 4.21.



Gambar 4.21 *Class Diagram* Sistem Usulan

4.3 Tahap Perancangan

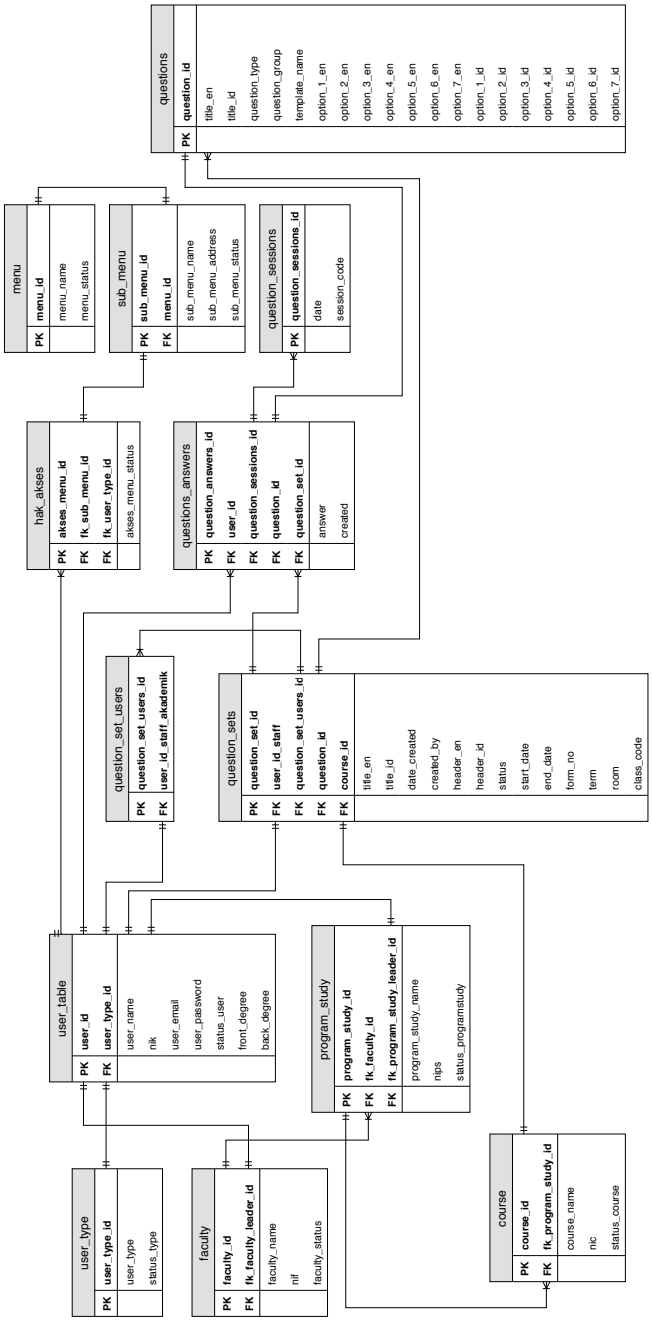
Tahap perancangan terbagi menjadi dua tahap, yaitu perancangan lapisan manajemen data dan lapisan antar muka.

4.3.1 Perancangan Lapisan Manajemen Data

Pada perancangan lapisan manajemen data terdiri atas *entity relationship diagram* dan *table description*

4.3.1.1 *Entity Relationship Diagram*

Gambar 4.22 menunjukkan *entity relationship diagram* dari sistem usulan



Gambar 4.22 *Entity Relationship Diagram* Sistem Usulan

4.3.2 *Table Description*

Table description menjelaskan tentang tabel *database* yang diperlukan dalam sistem usulan. Berikut ini adalah tabel – tabel *database* sistem usulan.

1) Tabel *user*

Tabel *user* merupakan tabel yang berisikan informasi mengenai *user* yang terdaftar dalam sistem. Tabel *user* dapat ditunjukkan pada Tabel 4.3.

Tabel 4.3 *Table Description* dari Tabel *User*

Nama *Field*	**Tipe Data**	***Length***	***Key***
user_id	*Integer*	-	*Primary key*
nik	*Varchar*	50	
user_name	*Varchar*	100	
user_email	*Text*	-	
user_password	*Varchar*	100	
user_type_id	*Integer*	-	*Foreign key*
status_user	*Varchar*	20	
front_degree	*Varchar*	100	
back_degree	*Varchar*	100	

2) Tabel *user type*

Tabel *user type* merupakan tabel yang berisikan informasi mengenai tipe – tipe user yang terdaftar dalam sistem. Tabel *user type* ditunjukkan pada Tabel 4.4.

Tabel 4.4 *Table Description* dari Tabel *User Type*

Nama *Field*	**Tipe Data**	***Length***	***Key***
user_type_id	*Integer*	-	*Primary Key*
user_type	*Varchar*	100	
status_type	*Varchar*	20	

3) Tabel menu

Tabel menu merupakan tabel yang berisikan informasi mengenai menu – menu apa saja yang terdaftar dalam sistem. Tabel menu ditunjukkan pada Tabel 4.5.

Tabel 4.5 *Table Description* dari Tabel *Menu*

Nama *Field*	**Tipe Data**	***Length***	***Key***
menu_id	*Integer*	-	*Primary key*
menu_name	*Varchar*	50	
menu_status	*Varchar*	20	

4) Tabel sub menu

Tabel sub menu merupakan tabel yang berisikan informasi mengenai sub menu apa saja yang terdaftar dalam sistem. Tabel sub menu ditunjukkan pada Tabel 4.6.

Tabel 4.6 *Table Description* dari Tabel *Sub Menu*

Nama *Field*	**Tipe Data**	***Length***	***Key***
sub_menu_id	*Integer*	-	*Primary key*
fk_menu_id	*Integer*	-	*Foreign key*
sub_menu_name	*Varchar*	50	
sub_menu_address	*Varchar*	50	
sub_menu_status	*Varchar*	20	

5) Tabel hak akses

Tabel akses menu merupakan tabel yang berisikan informasi mengenai hak akses user terhadap menu dan sub menu tertentu di dalam sistem. Tabel akses menu ditunjukkan pada Tabel 4.7.

Tabel 4.7 *Table Description* dari Tabel *Akses Menu*

Nama *Field*	**Tipe Data**	***Length***	***Key***
akses_menu_id	*Integer*	-	*Primary key*
fk_sub_menu_id	*Integer*	-	*Foreign key*
fk_user_type_id	*Integer*	-	*Foreign key*
akses_menu_status	*Varchar*	20	

6) Tabel *faculty*

Tabel *faculty* merupakan tabel yang berisikan informasi mengenai fakultas – fakultas yang terdapat di UPH. Tabel fakultas ditunjukkan pada Tabel 4.8.

Tabel 4.8 Table Description dari Tabel *Faculty*

Nama *Field*	**Tipe Data**	***Length***	***Key***
faculty_id	*Integer*	-	*Primary key*
nif	*Varchar*	50	
faculty_name	*Varchar*	250	
fk_faculty_leader_id	*Integer*	-	*Foreign key*
faculty_status	*Varchar*	20	

7) Tabel program studi

Tabel program studi merupakan tabel yang berisikan informasi mengenai program studi yang terdapat di UPH. Tabel program studi ditunjukkan pada Tabel 4.9.

Tabel 4.9 *Table Description* dari Tabel *Program Study*

Nama *Field*	**Tipe Data**	***Length***	***Key***
program_study_id	*Integer*	-	*Primary key*
nips	*Varchar*	50	
program_study_name	*Varchar*	250	
fk_faculty_id	*Integer*	-	*Foreign key*
fk_program_study_leader_id	*Integer*	-	*Foreign key*
status_programstudy	*Varchar*	20	

8) Tabel *course*

Tabel *course* merupakan tabel yang berisikan informasi mengenai mata kuliah yang terdapat di UPH. Tabel *course* ditunjukkan pada Table 4.10.

Tabel 4.10 *Table Description* dari Tabel *Course*

Nama *Field*	**Tipe Data**	***Length***	***Key***
course_id	*Integer*	-	*Primary key*
nic	*Varchar*	50	
course_name	*Varchar*	250	
fk_program_study_id	*Integer*	-	*Foreign key*
status_course	*Varchar*	20	

9) Tabel *question set*

Tabel *question set* merupakan tabel yang berisikan informasi mengenai jadwal survei yang akan dilakukan oleh pihak LP2MP. Tabel *question set* ditunjukkan pada Tabel 4.11.

Tabel 4.11 *Table Description* dari Tabel *Question Set*

Nama *Field*	**Tipe Data**	***Length***	***Key***
question_set_id	*Integer*	-	*Primary key*
user_id_staff	*Integer*	-	*Foreign key*
question_set_users_id	*Integer*	-	*Foreign key*
question_id	*Integer*	-	*Foreign key*
course_id	*Integer*	-	*Foreign key*
title_id	*Varchar*	200	
title_en	*Varchar*	200	
class_code	*Varchar*	50	
created_by	*Integer*	-	
date_created	*Datetime*	-	
header_en	*Text*	-	
header_id	*Text*	-	
status	*Varchar*	20	
start_date	*Varchar*	40	
end_date	*Varchar*	40	
form_no	*Varchar*	40	
term	*Varchar*	40	
room	*Varchar*	40	

10) Tabel *question set users*

Tabel *question set users* merupakan tabel yang berisikan informasi mengenai user yang di nilai di dalam survei. Tabel *question set users* ditunjukkan pada Tabel 4.12.

Tabel 4.12 *Table Description* dari Tabel *Question Set Users*

Nama *Field*	**Tipe Data**	***Length***	***Key***
question_set_users_id	*Integer*	-	*Primary key*
user_id_staff_akademik	*Integer*	-	*Foreign key*

11) Tabel *question sessions*

Tabel *question sessions* merupakan tabel yang berisikan informasi mengenai session_code yang digunakan untuk mahasiswa yang mengisi survei. Tabel *question sessions* ditunjukkan pada Tabel 4.13.

Tabel 4.13 *Table Description* dari Tabel *Question Sessions*

Nama *Field*	**Tipe Data**	***Length***	***Key***
question_sessions_id	*Integer*	-	*Primary key*
date	*Datetime*	-	
session_code	*Varchar*	50	*Foreign key*

12) Tabel *question answers*

Tabel *question answers* merupakan tabel yang berisikan jawaban – jawaban dari mahasiswa yang telah mengisi survei yang diberikan. Tabel *question answers* ditunjukkan pada Tabel 4.14.

Tabel 4.14 *Table Description* dari Tabel *Question Answers*

Nama *Field*	**Tipe Data**	***Length***	***Key***
question_answers_id	*Integer*	-	*Primary key*
user_id	*Integer*	-	*Foreign key*
question_set_id	*Integer*	-	*Foreign key*
question_sessions_id	*Integer*	-	*Foreign key*
question_id	*Integer*	-	*Foreign key*
answer	*Varchar*	200	
created	*Datetime*	-	

13) Tabel *questions*

Tabel *questions* merupakan tabel yang berisikan informasi mengenai pertanyaan – pertanyaan survei yang ada. Tabel *questions* ditunjukkan pada Tabel 4.15.

Tabel 4.15 *Table Description* dari Tabel *Questions*

Nama *Field*	**Tipe Data**	***Length***	***Key***
questions_id	*Integer*	-	*Primary key*
template_name	*Varchar*	200	
title_en	*Varchar*	200	
title_id	*Varchar*	200	
question_type	*Varchar*	100	
question_group	*Varchar*	100	
option_1_en	*Varchar*	100	
option_2_en	*Varchar*	100	
option_3_en	*Varchar*	100	
option_4_en	*Varchar*	100	
option_5_en	*Varchar*	100	
option_6_en	*Varchar*	100	
option_7_en	*Varchar*	100	
option_1_id	*Varchar*	100	
option_2_id	*Varchar*	100	
option_3_id	*Varchar*	100	
option_4_id	*Varchar*	100	
option_5_id	*Varchar*	100	
option_6_id	*Varchar*	100	
option_7_id	*Varchar*	100	

4.4 Tahap Implementasi

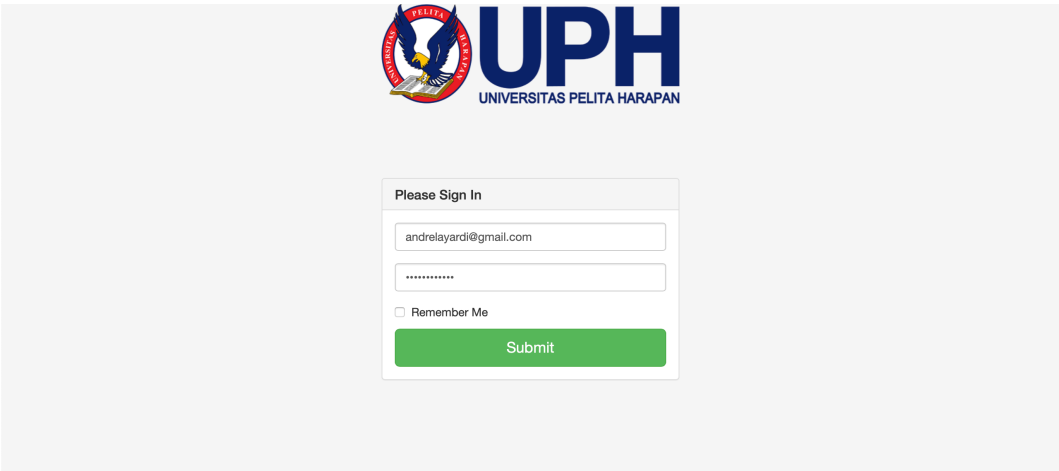
Tahap implementasi merupakan tahap di mana rancangan yang telah dikembangkan, selanjutnya diimplementasikan ke dalam sistem baru dan kemudian dilakukan uji coba.

4.4.1 Lapisan Antarmuka (*Interface*)

Pada bagian ini akan membahas lapisan antar muka (*interface*) dari sistem usulan yang telah dikembangkan.

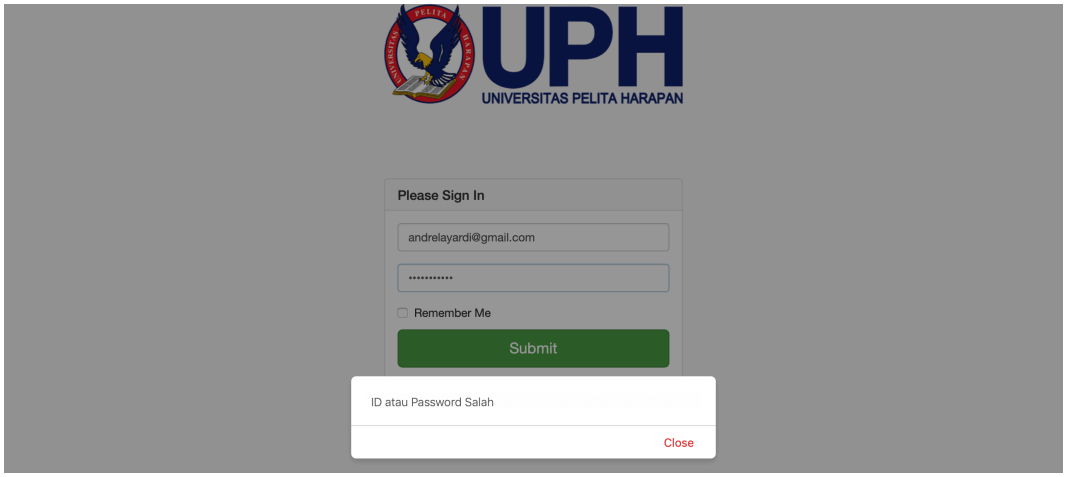
1) Tampilan halaman *login*

   Pada sistem usulan ini terdiri dari tujuh level akses, yaitu *Super Admin, Staff*, *Part Timer*, Dosen, Kaprodi, Dekan dan *VP*. Setiap level akses memiliki menu dan fungsi yang berbeda – beda. *User* harus melakukan *login* terlebih dahulu untuk mengakses sistem. *User* yang tidak memiliki akses, harus menghubungi pihak LP2MP untuk mendapatkan akses. Tampilan *login* ditunjukkan pada Gambar 4.23.



Gambar 4.23 Tampilan Halaman *Login*

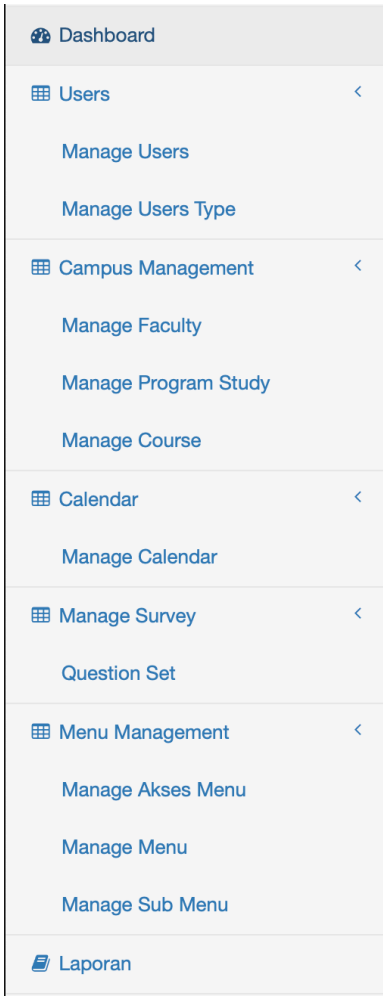
Jika *user* salah memasukkan email atau *password*, maka sistem akan menampilkan pesan eror seperti yang ditunjukkan pada Gambar 4.24.



Gambar 4.24 Tampilan Halaman *Login Error*

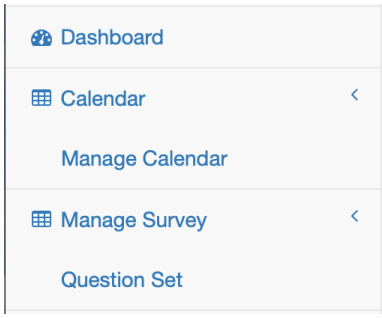
2) Tampilan menu

Setiap *user* memiliki tampilan menu yang berbeda-beda sesuai dengan peran yang telah didaftarkan dalam sistem. *Super admin* dan *staff* memiliki menu *dashboard*, *users, campus management, calendar, manage survey, menu management* dan laporan seperti yang ditunjukkan Gambar 4.25.

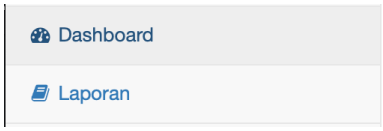


Gambar 4.25 Tampilan *Menu Super Admin* dan *Staff*

Sedangkan untuk *part timer* memiliki menu *dashoard*, *question set* dan *calendar* (Gambar 4.26). *Staff* akademik seperti dosen, kaprodi, dekan, dan *vice president* hanya memiliki menu *dashboard* dan laporan saja (Gambar 4.27)



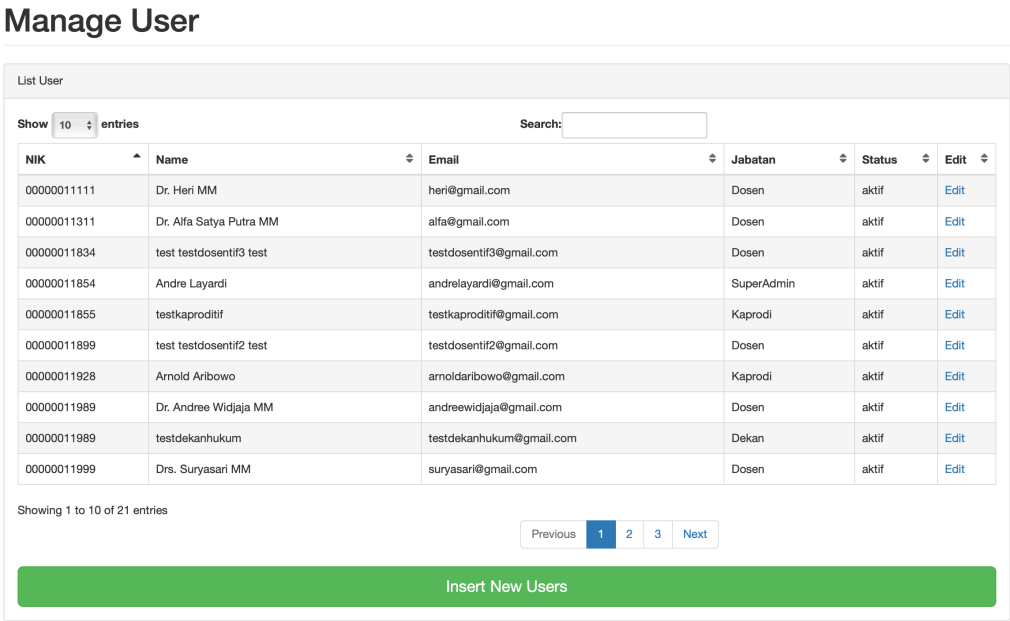
Gambar 4.26 Tampilan *Menu Part Time*



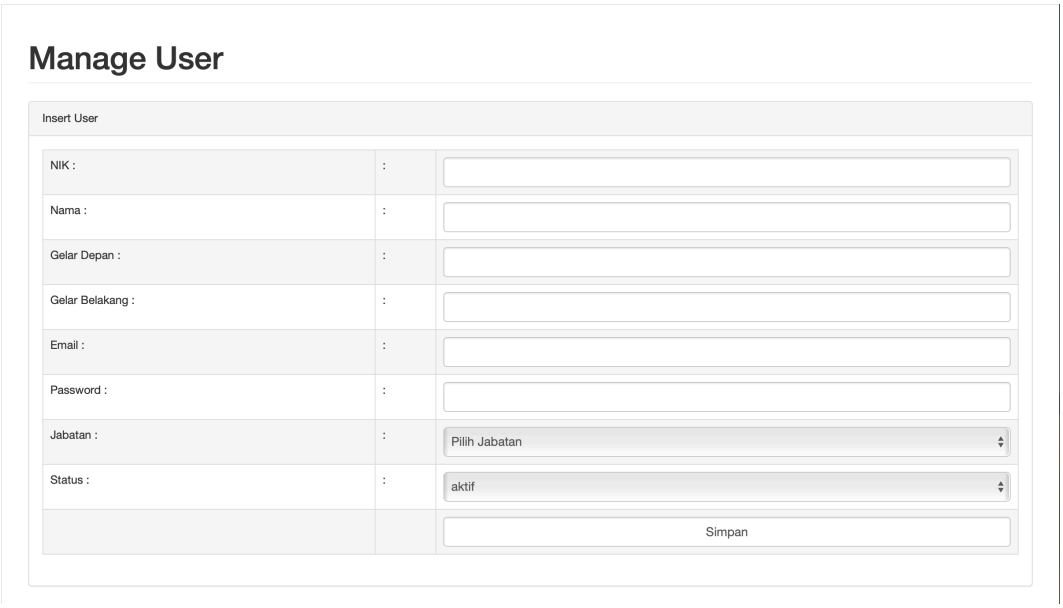
Gambar 4.27 Tampilan *Menu Staff Akademik*

3) Tampilan *user data*

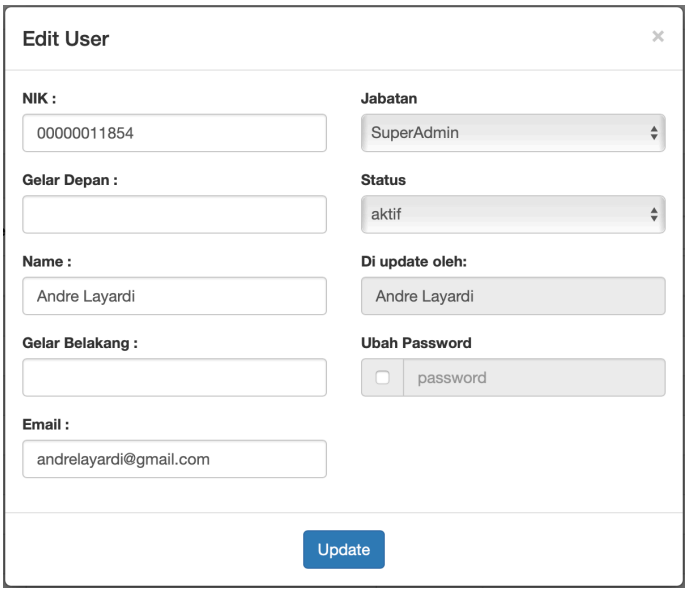
User Maintenance merupakan menu untuk mengelola *user*. Menu ini terdapat dua sub menu yaitu *user data* dan *user right*. Untuk halaman *user data* yang ditunjukkan pada Gambar 4.28. Tampilan halaman *form* menambah *user* ditunjukkan pada Gambar 4.29. Sedangkan tampilan *form* pengubahan data *user* ditunjukkan pada Gambar 4.30.



Gambar 4.28 Tampilan *Manage Users*



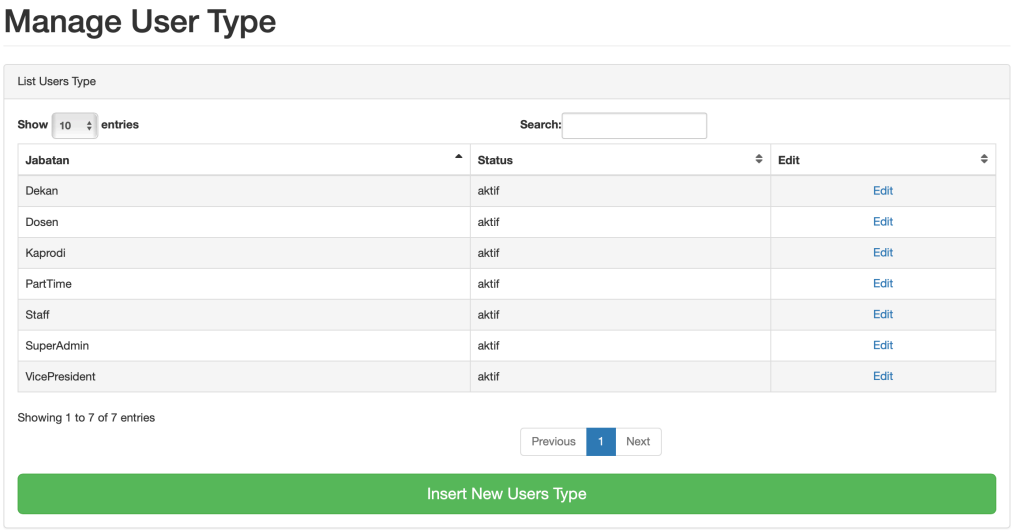
Gambar 4.29 Tampilan *Add Users*



Gambar 4.30 Tampilan *Edit User*

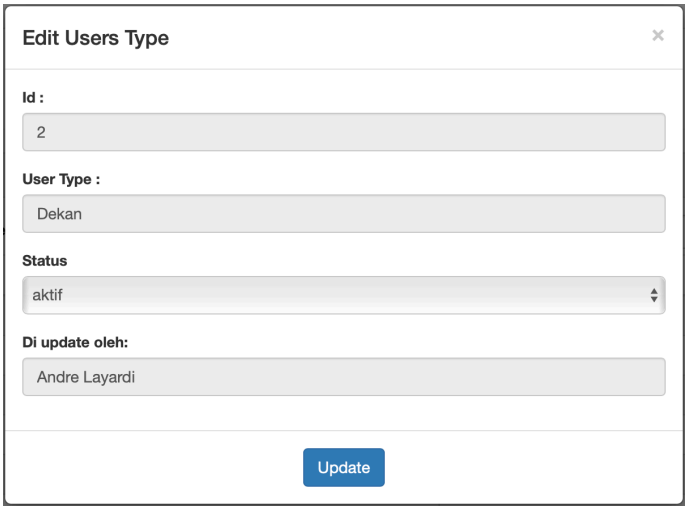
4) Tampilan *user type*

Sedangkan *user type* yang ditujukkan pada Gambar 4.31 berfungsi untuk melihat daftar tipe atau peran dari *user*. Tampilan halaman *form* menambah *user type* ditunjukkan pada Gambar 4.32. Tampilan pengubahan data *user type* ditunjukkan pada Gambar 4.33.



Gambar 4.31 Tampilan *Users Type*

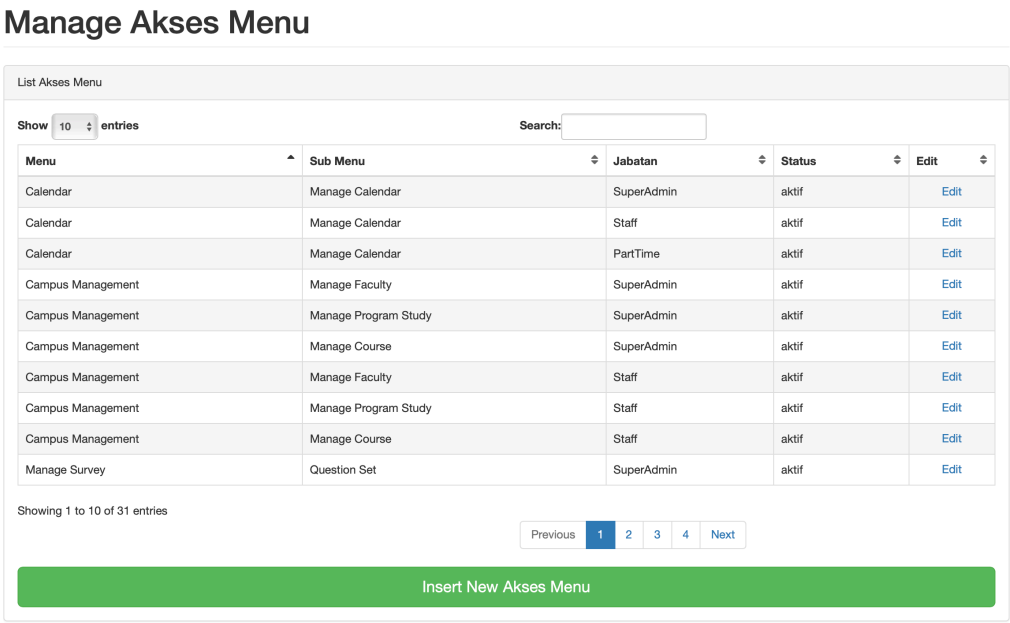
Gambar 4.32 Tampilan *Add Users Type*



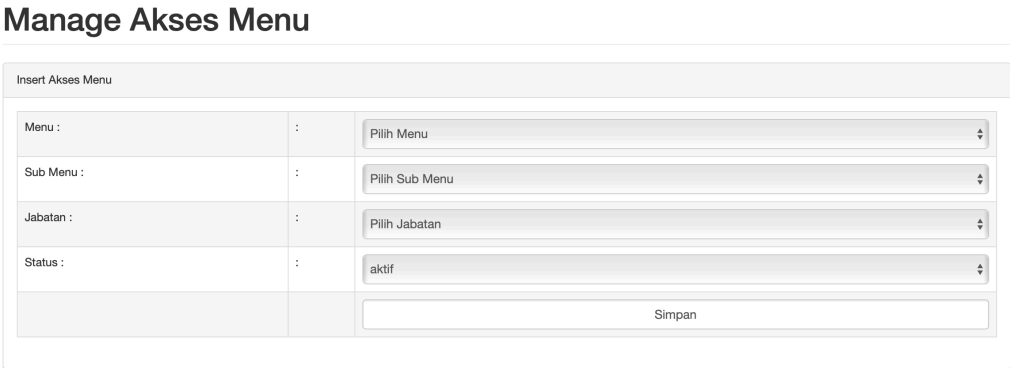
Gambar 4.33 Tampilan *Edit User Type*

5) Tampilan halaman *manage* hak *akses*

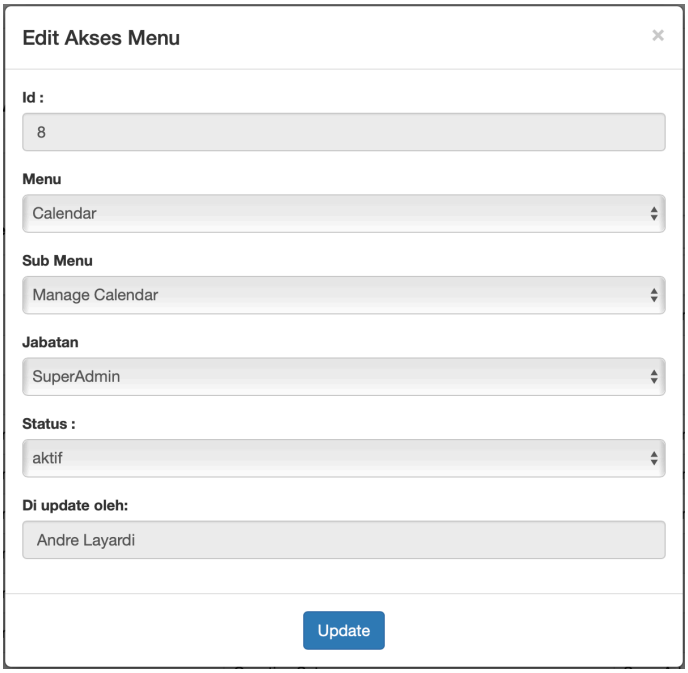
Hak *akses* merupakan *sub menu* dari *menu menu management yang* berfungsi untuk mengelola data hak *akses* yang terdapat di program ini, hak *akses* mengatur *menu* mana saja yang dapat diakses oleh *user type* tertentu. Untuk halaman hak *akses* ditunjukkan pada Gambar 4.34. Selanjutnya pada halaman hak *akses* memiliki fungsi menambah data *akses menu* yang ditunjukkan pada Gambar 4.35 dan fungsi mengubah data *akses menu* ditunjukkan pada Gambar 4.36.



Gambar 4.34 Tampilan *Manage Akses Menu*



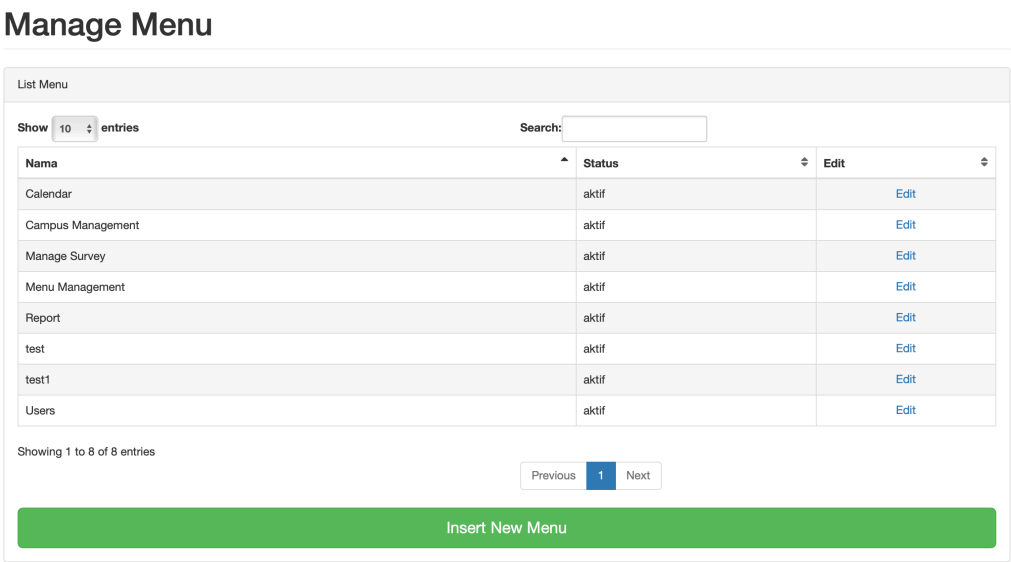
Gambar 4.35 Tampilan *Add Akses Menu*



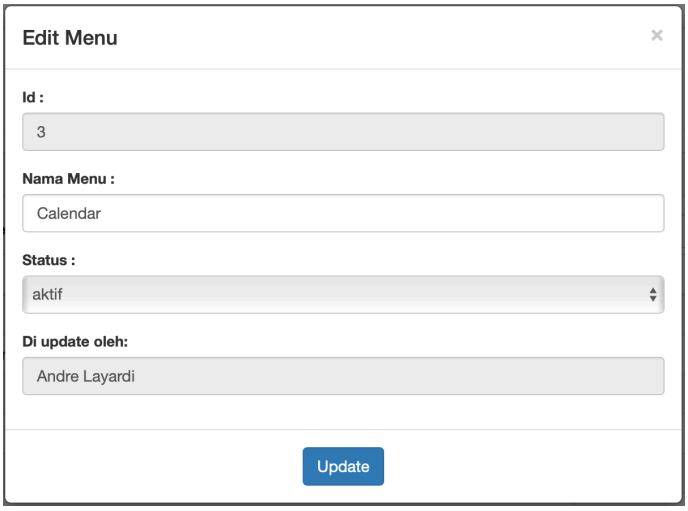
Gambar 4.36 Tampilan *Edit Akses Menu*

6) Tampilan halaman *manage menu*

Menu merupakan *sub menu* dari *menu menu management yang* berfungsi untuk mengelola data *menu* yang terdapat di program ini. Untuk halaman *menu* ditunjukkan pada Gambar 4.37. Selanjutnya pada halaman *menu* memiliki fungsi mengubah data *menu* ditunjukkan pada Gambar 4.38.



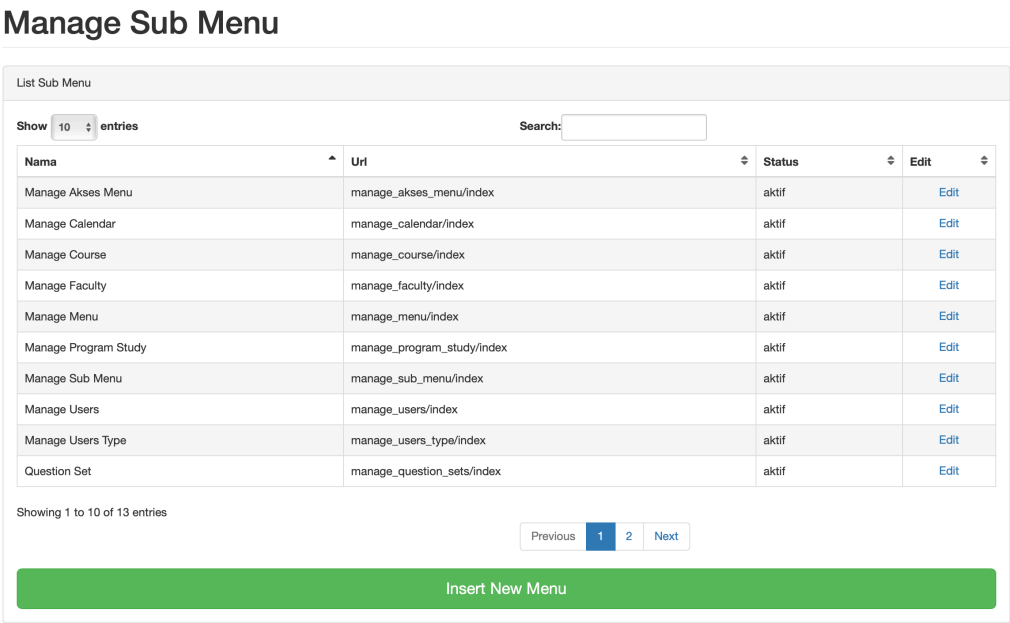
Gambar 4.37 Tampilan *Manage Menu*



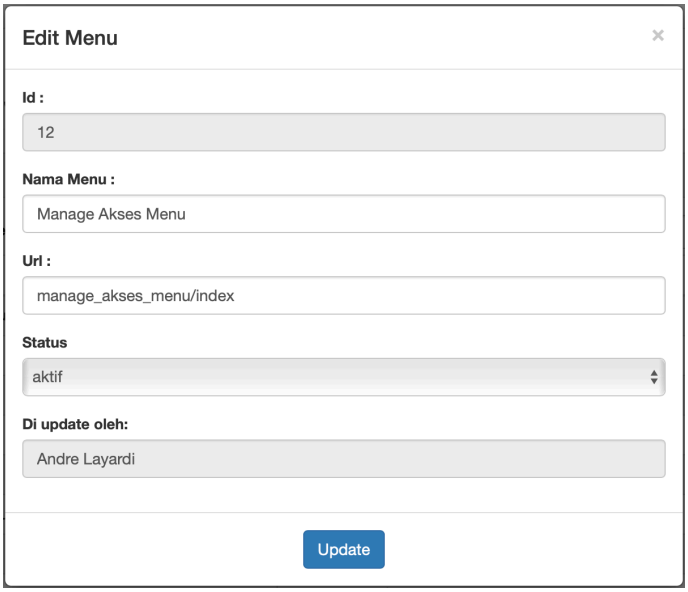
Gambar 4.38 Tampilan *Edit Menu*

7) Tampilan halaman *sub menu*

Sub menu merupakan *sub menu* dari *menu menu management yang* berfungsi untuk mengelola data *sub menu* yang terdapat di program ini. Untuk halaman *sub menu* ditunjukkan pada Gambar 4.39. Selanjutnya pada halaman *sub menu* memiliki fungsi mengubah data *sub menu* ditunjukkan pada Gambar 4.40.



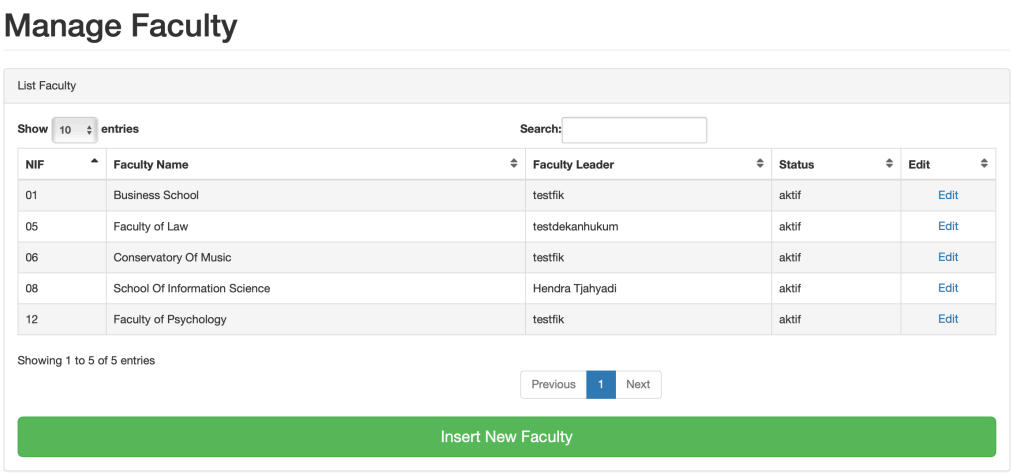
Gambar 4.39 Tampilan *Manage Sub Menu*



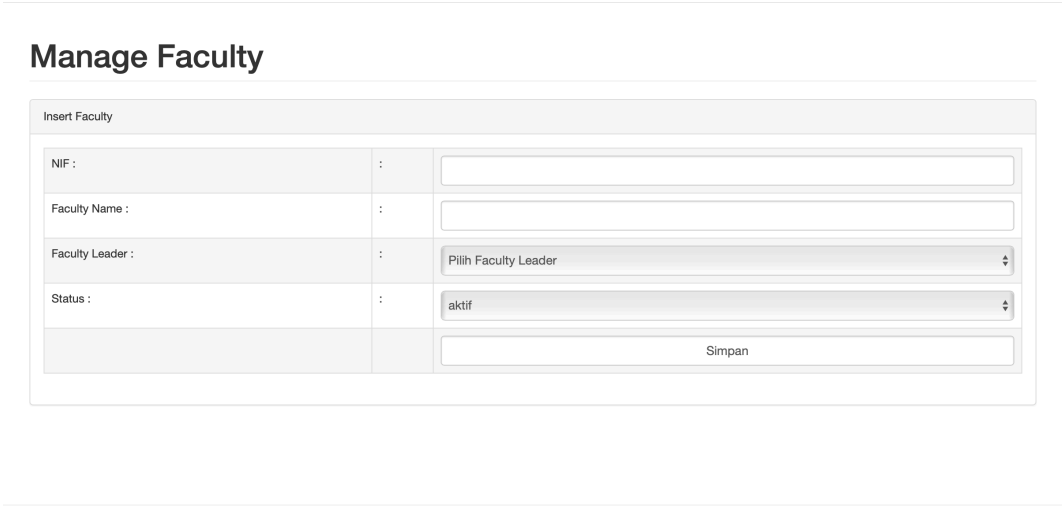
Gambar 4.40 Tampilan *Edit Sub Menu*

8) Tampilan halaman *faculty*

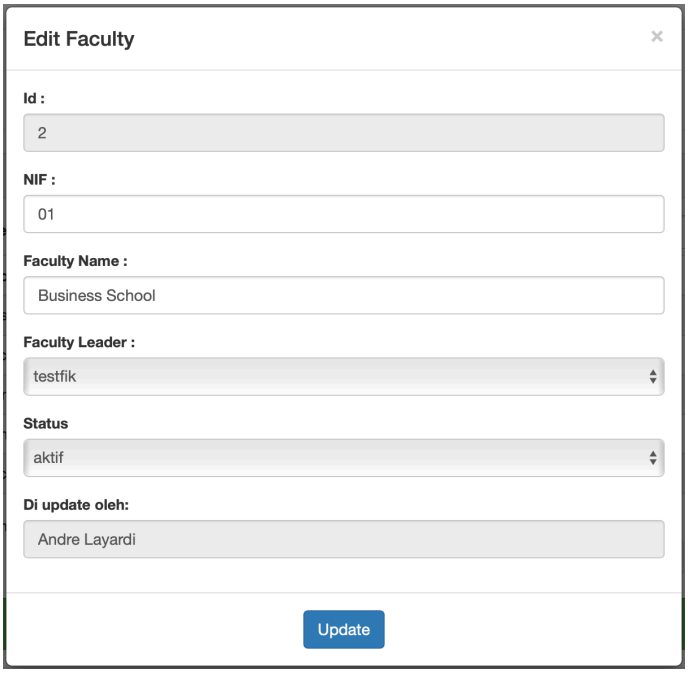
Faculty merupakan *sub menu* dari menu *campus management yang* berfungsi untuk mengelola data fakultas yang terdapat di UPH dan memberikan informasi mengenai dekan yang ada disetiap fakultas. Untuk halaman *faculty* ditunjukkan pada Gambar 4.41. Selanjutnya pada halaman *faculty* memiliki fungsi menambah data fakultas yang ditunjukkan pada Gambar 4.42 dan fungsi mengubah data fakultas ditunjukkan pada Gambar 4.43.



Gambar 4.41 Tampilan *Manage Faculty*



Gambar 4.42 Tampilan *Add Faculty*

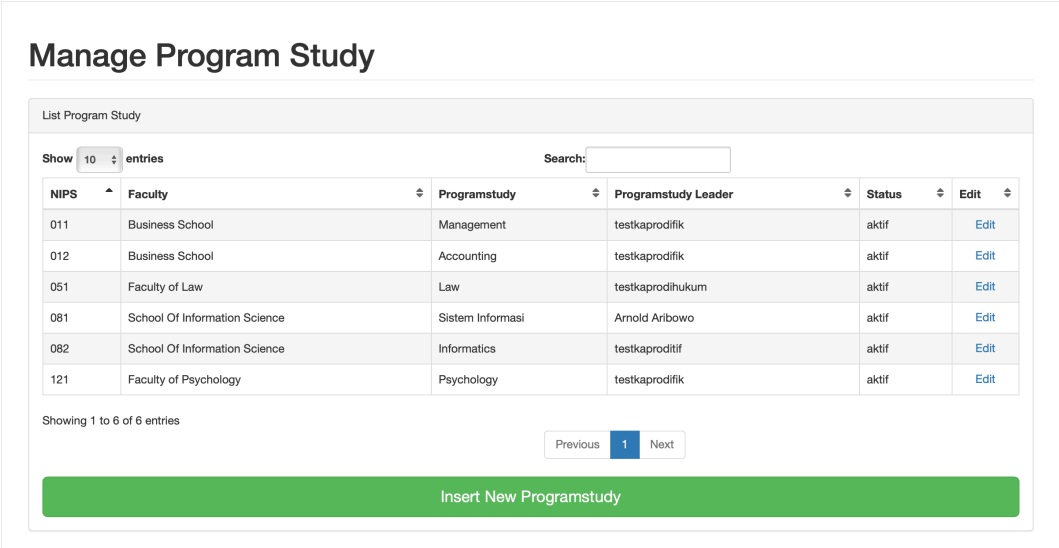


Gambar 4.43 Tampilan *Edit Faculty*

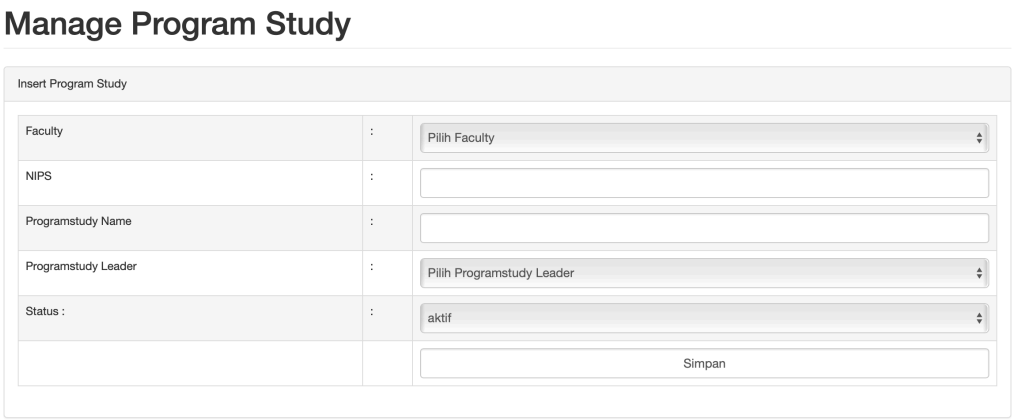
9) Tampilan halaman *program study*

Program Study merupakan *sub menu* kedua dari menu *campus management* yang berfungsi untuk mengelola data program studi di UPH dan dapat mengetahui Kaprodi disetiap program studi. Pada Gambar 4.44 menunjukkan tampilan halaman *program study* untuk melihat daftar program studi. *Super admin* dan *staff* dapat menambah data program studi dengan mengisi *form* pada Gambar 4.45. Jika ada perubahan nama atau

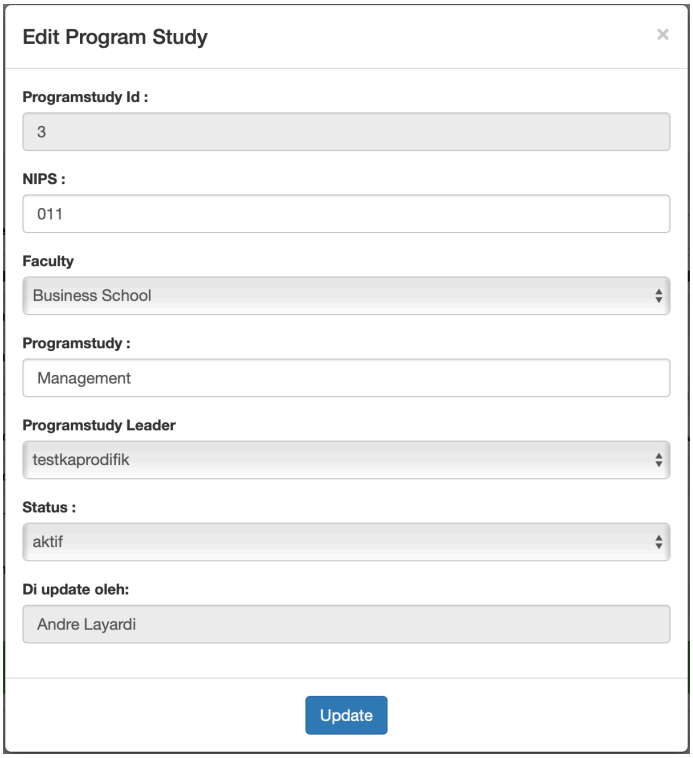
status program studi, maka *Super admin* atau *staff* dapat mengubah data program studi seperti yang ditunjukkan pada Gambar 4.46.



Gambar 4.44 Tampilan *Manage Program Study*



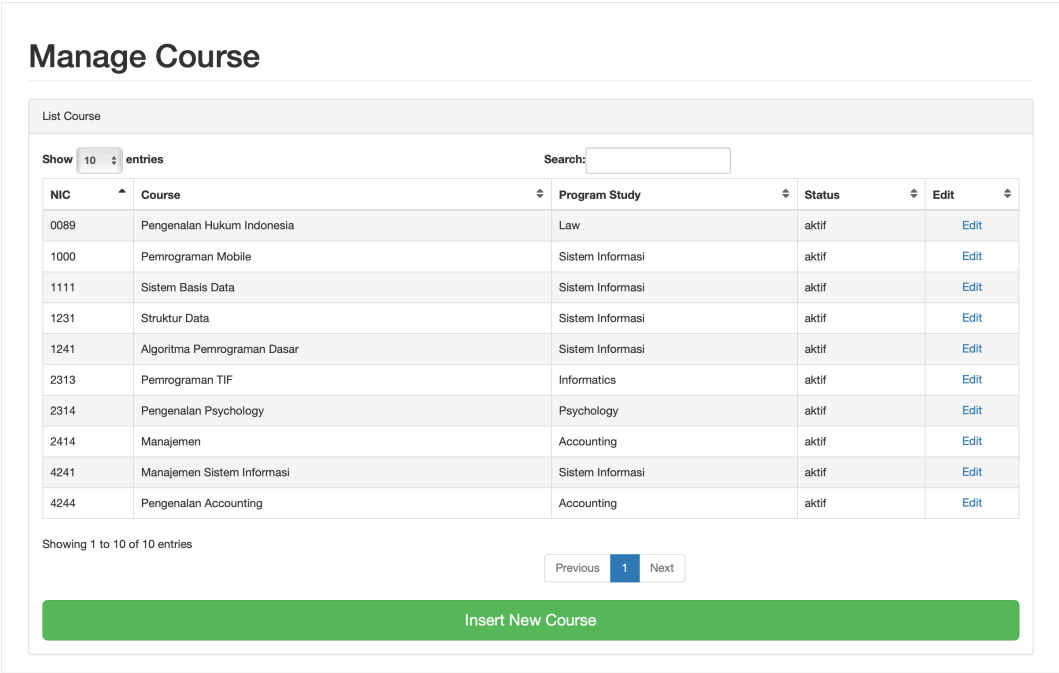
Gambar 4.45 Tampilan *Add Program Study*



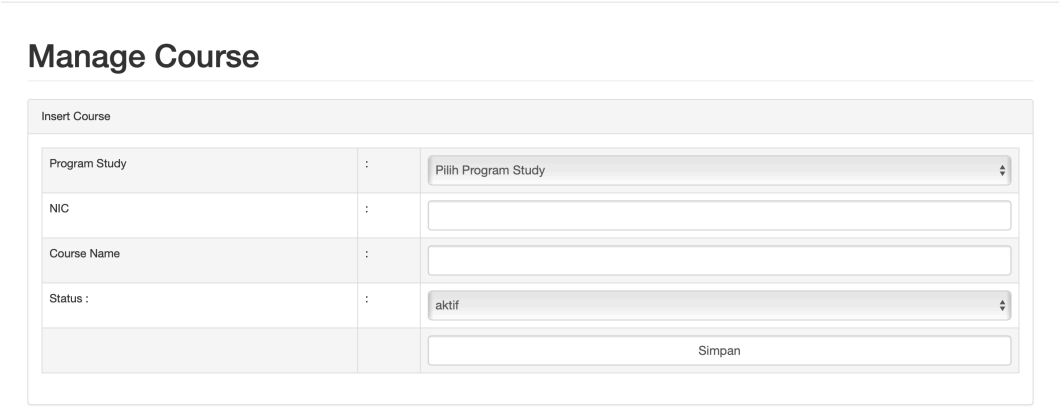
Gambar 4.46 Tampilan *Edit Program Study*

10) Tampilan halaman *course*

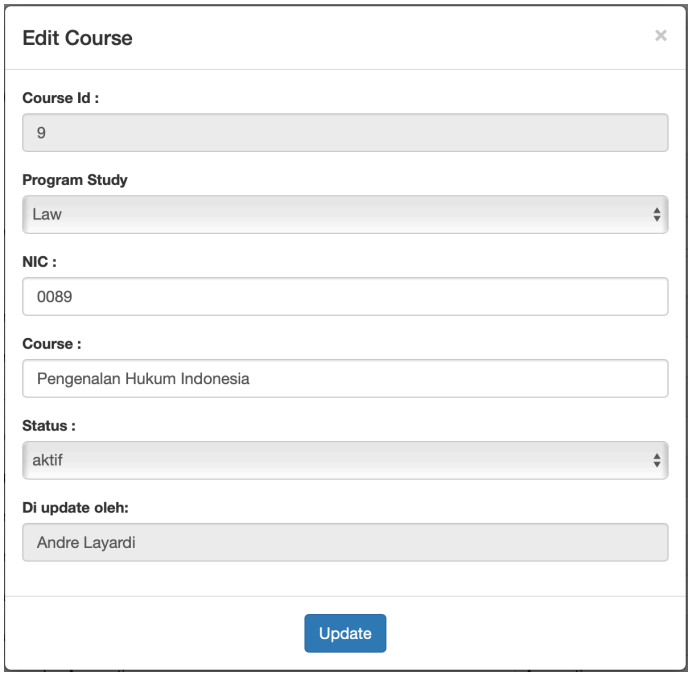
Course merupakan *sub menu* ketiga pada menu *campus management* yang berfungsi untuk mengelola data mata kuliah yang terdapat di UPH. Pada Gambar 4.47 merupakan tampilan halaman daftar mata kuliah. Kemudian *super admin* maupun *staff* dapat menambahkan data dengan mengisi *form* yang ditunjukkan pada Gambar 4.48. Jika ada perubahan data dapat di ubah seperti yang ditunjukkan pada Gambar 4.49.



Gambar 4.47 Tampilan *Manage Course*



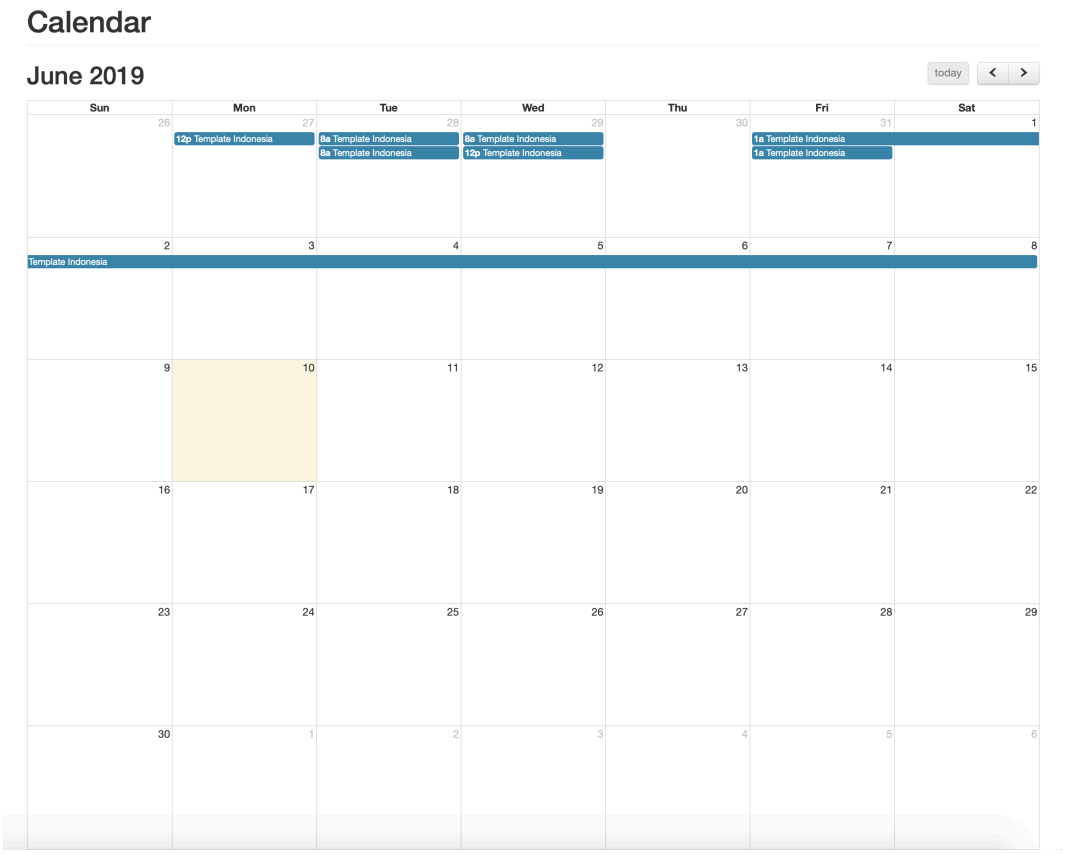
Gambar 4.48 Tampilan *Add Course*



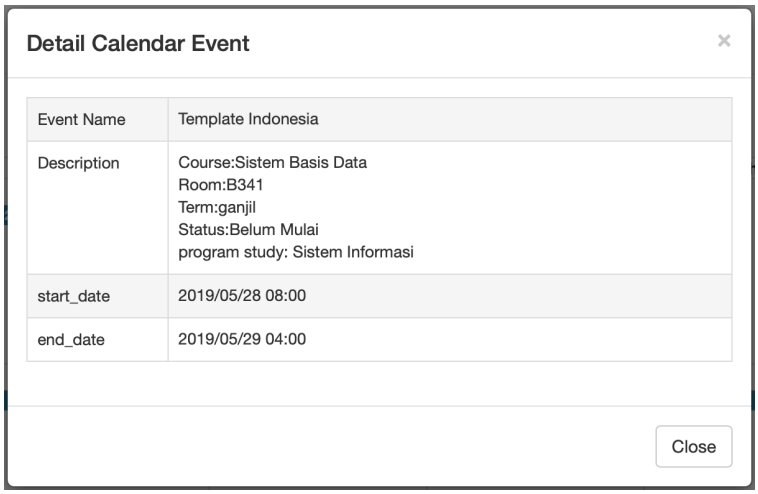
Gambar 4.49 Tampilan *Edit Course*

11) Tampilan *calendar*

Calendar merupakan sub menu dari menu *calendar* yang memiliki fungsi untuk memperlihatkan jadwal kunjungan survei dalam bentuk sebuah kalender agar lebih mudah di lihat oleh *super admin*, *staff* dan *part timer* Gambar 4.50. Kalender tersebut juga bisa di tekan agar memunculkan detail survei pada hari itu Gambar 4.51.



Gambar 4.50 Tampilan *Calendar*

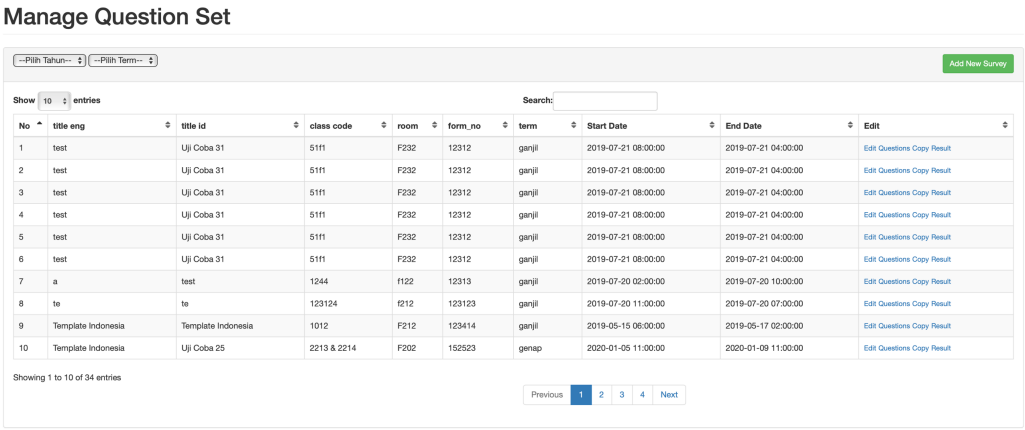


Gambar 4.51 Tampilan *Detail Calendar*

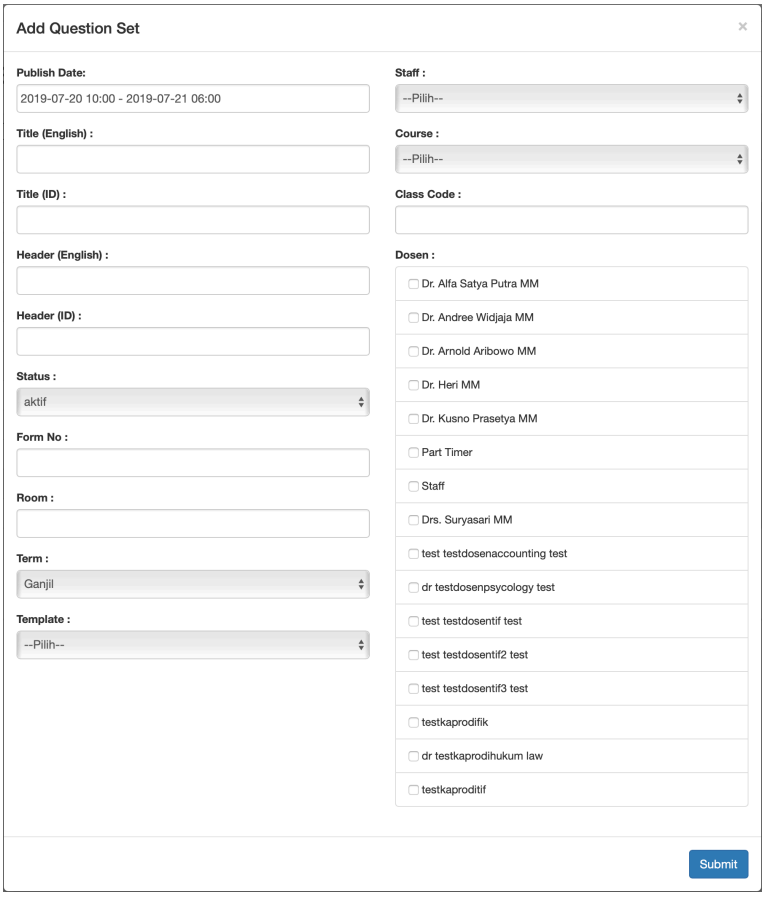
12) Tampilan halaman *question set*

Question Set merupakan *sub menu* dari menu *manage survey* yang memiliki fungsi untuk mengelola survei. Pada Gambar 4.52 merupakan tampilan

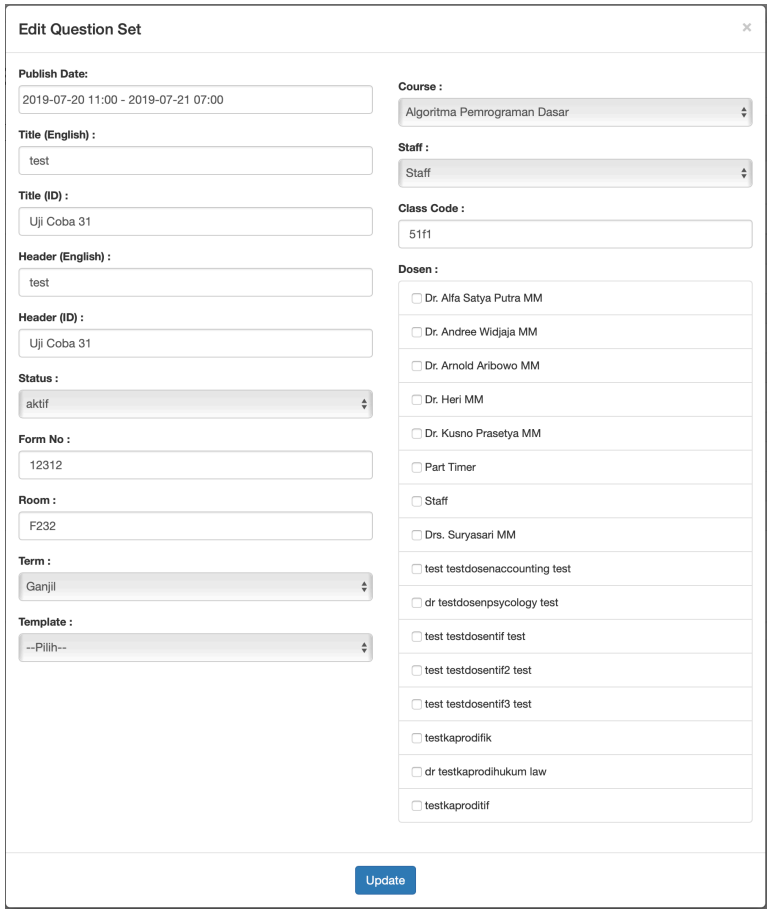
halaman *question sets* yang berisikan daftar survei untuk setiap kelas. *Super admin*, *staff* dan *part timer* dapat menambah *question set* baru yang ditunjukkan pada Gambar 4.53. *Super admin*, *staff* dan *part timer* juga dapat mengubah *question set* yang ada ditunjukkan pada Gambar 4.54.



Gambar 4.52 Tampilan *Manage Question Set*



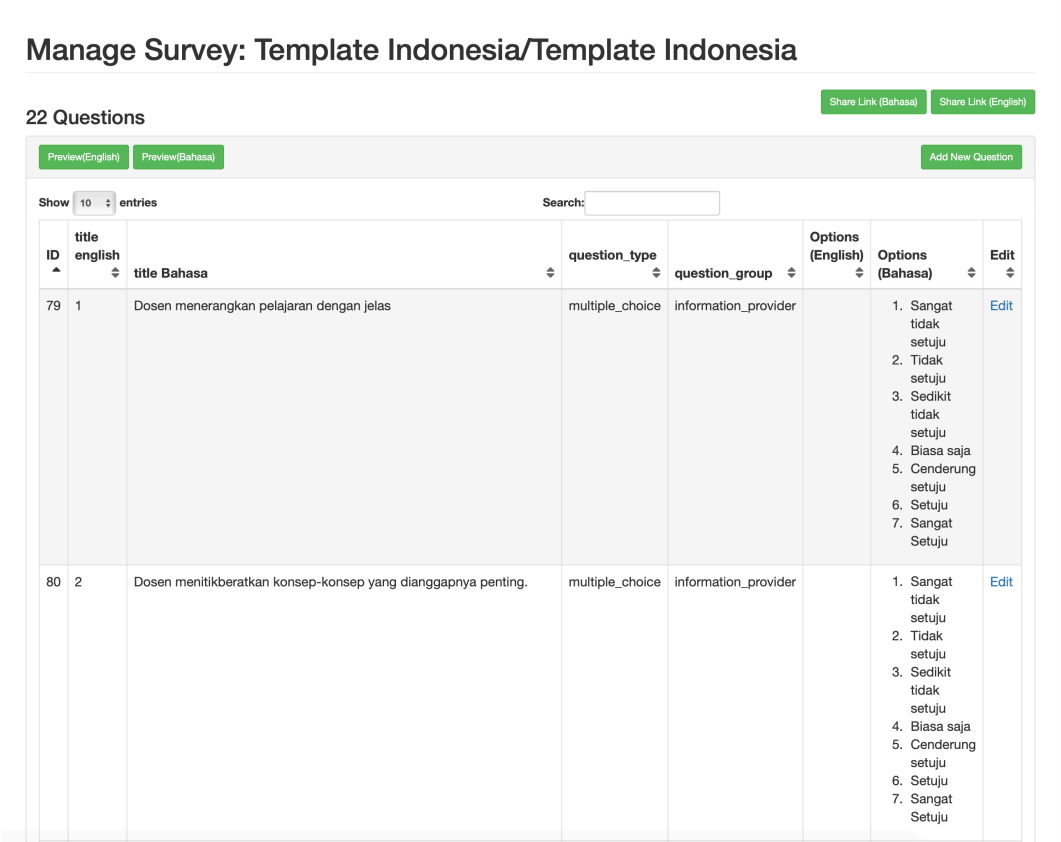
Gambar 4.53 Tampilan *Add Question Set*



Gambar 4.54 Tampilan *Edit Question Set*

13) Tampilan halaman *question*

Question merupakan *button* yang terletak pada kanan sebuah *question set*. Pada Gambar 4.55 merupakan tampilan halaman *question* oleh *question set* tertentu. *Super admin*, *staff* dan *part timer* dapat menambah *question multiple choice* baru yang ditunjukkan pada Gambar 4.56. *Super admin*, *staff* dan *part timer juga* dapat menambah *question text* baru yang ditunjukkan pada Gambar 4.57. *Super admin*, *staff* dan *part timer* juga dapat mengubah *question set* yang ada ditunjukkan pada Gambar 4.58.



Manage Survey: Template Indonesia/Template Indonesia

22 Questions

Share Link (Bahasa) Share Link (English)

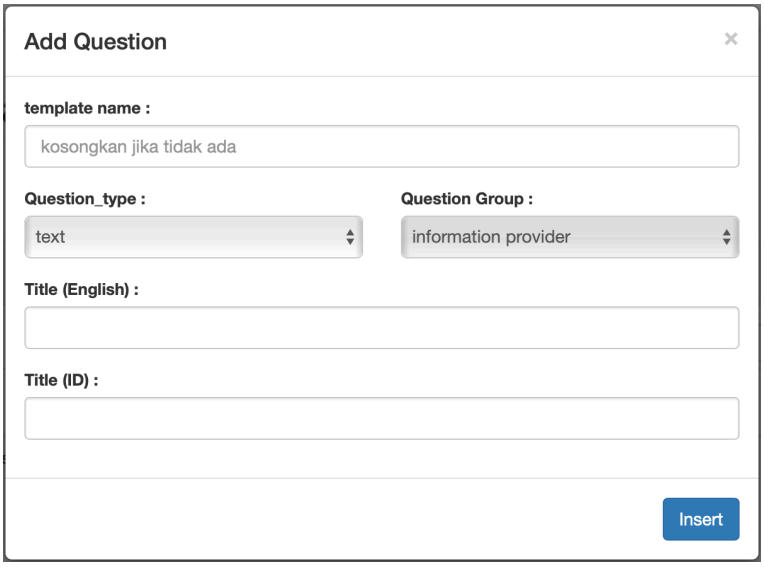
Preview(English) Preview(Bahasa) Add New Question

Show 10 entries Search:

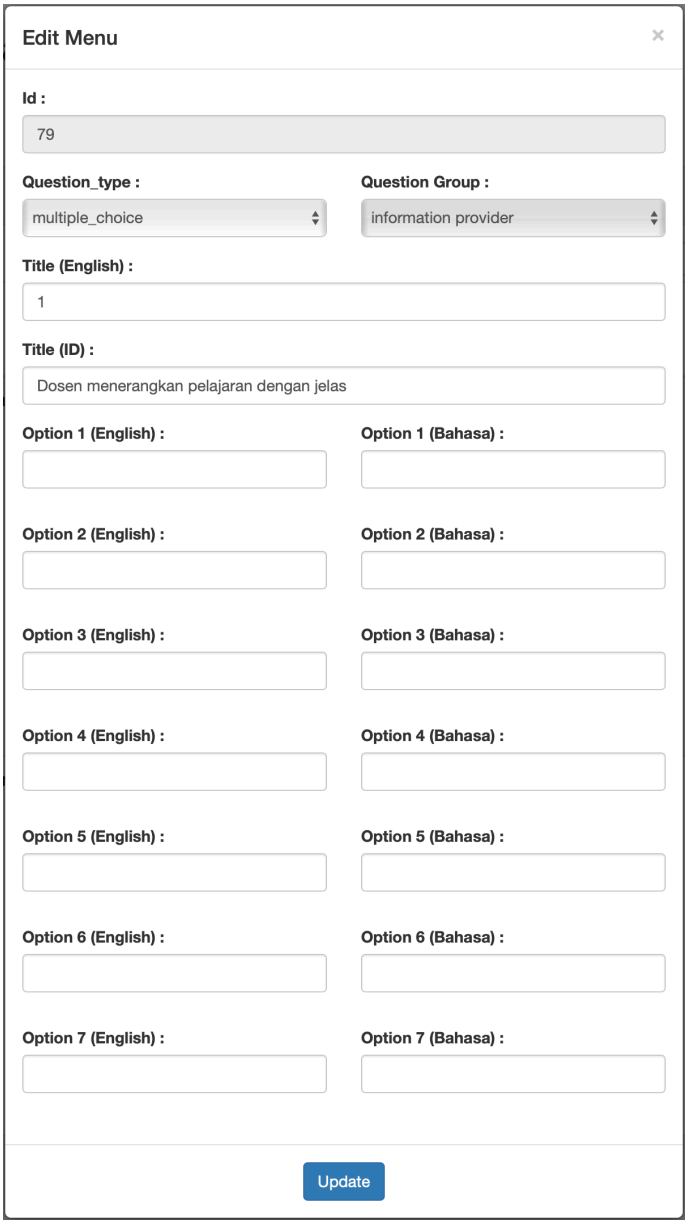
ID	title english	title Bahasa	question_type	question_group	Options (English)	Options (Bahasa)	Edit
79	1	Dosen menerangkan pelajaran dengan jelas	multiple_choice	information_provider		1. Sangat tidak setuju 2. Tidak setuju 3. Sedikit tidak setuju 4. Biasa saja 5. Cenderung setuju 6. Setuju 7. Sangat Setuju	Edit
80	2	Dosen menitikberatkan konsep-konsep yang dianggapnya penting.	multiple_choice	information_provider		1. Sangat tidak setuju 2. Tidak setuju 3. Sedikit tidak setuju 4. Biasa saja 5. Cenderung setuju 6. Setuju 7. Sangat Setuju	Edit

Gambar 4.55 Tampilan *Manage Question*

Gambar 4.56 Tampilan *Add Question (Multiple Choice)*



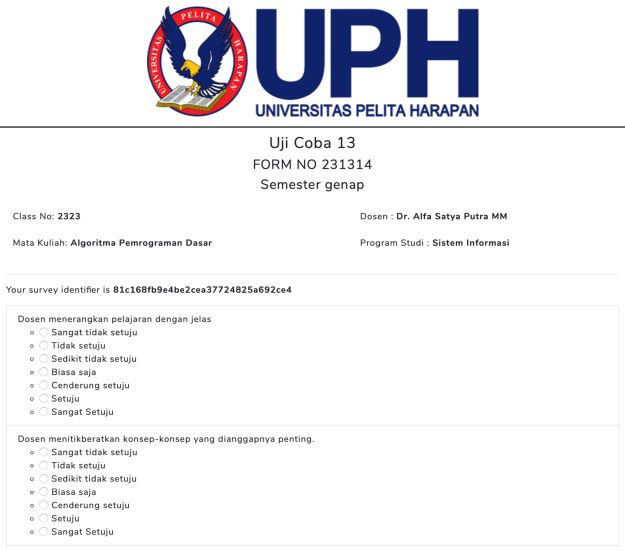
Gambar 4.57 Tampilan *Add Question (Text)*



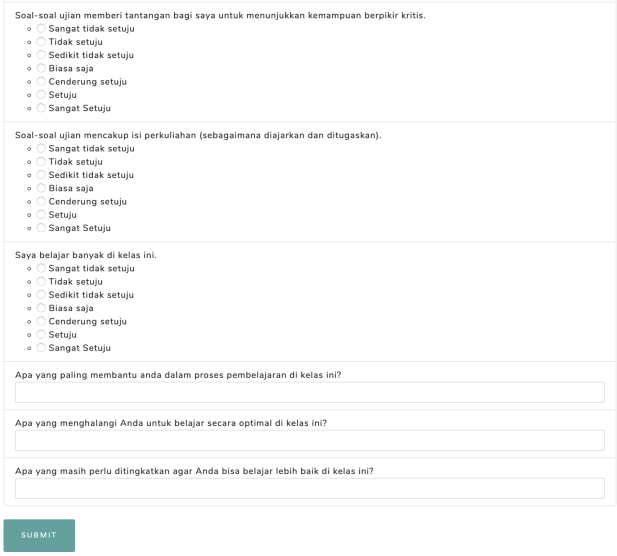
Gambar 4.58 Tampilan *Edit Question*

14) Tampilan halaman survei

Gambar 4.59. Merupakan tampilan halaman survei. Survei dapat dilihat dengan membuka *link* yang telah terdapat dalam *question set*. Setelah itu mahasiswa dapat memulai survei. Jika dalam satu kelas memilik lebih dari satu dosen yang mengajar maka mahasiswa akan mengisi survei sejumlah dosen yang mengajar. Ketika mahasiswa sudah selesai menjawab maka sistem akan menampilkan tanda selesai seperti pada Gambar 4.60.



Gambar 4.59 Tampilan Survei



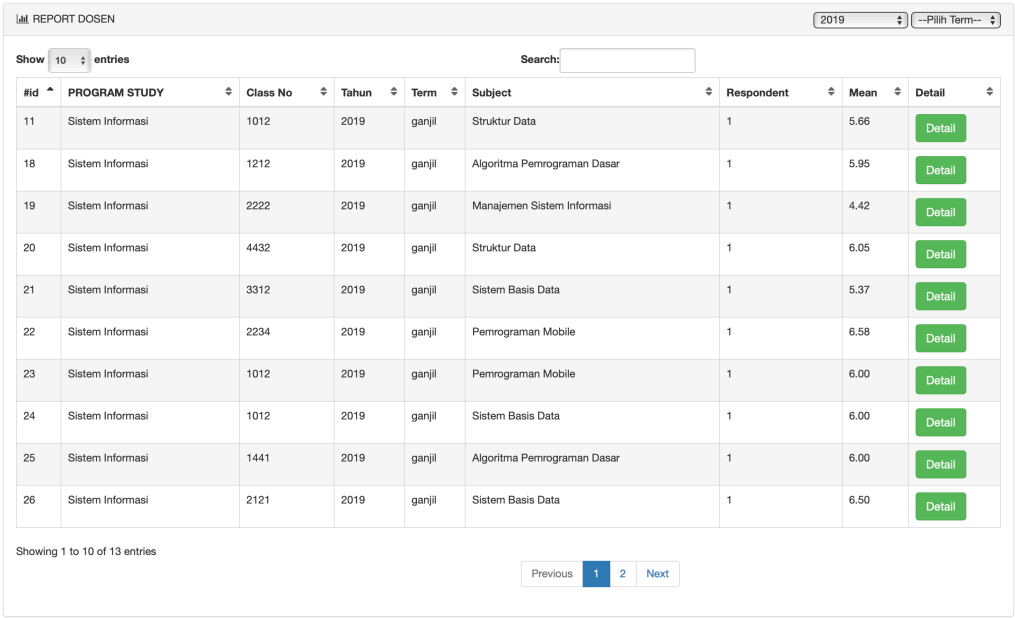
Gambar 4.59 Tampilan Survei Lanjut

Gambar 4.60 Tampilan Selesai Survei

15) Tampilan halaman *report* untuk *super admin* dan *staff*

Setelah mahasiswa telah mengisi survei, data yang masuk dalam sistem akan diolah menjadi informasi yang berbentuk laporan penilaian. Laporan

di bagi menjadi dua yaitu laporan dosen dan laporan detail dosen. Penilaian dosen ditunjukkan pada Gambar 4.61. Setiap dosen akan memiliki laporan penilaian dosen sesuai dengan mata kuliah yang diajar oleh dosen tersebut. Laporan dosen dapat dilihat pada Gambar 4.62. Sedangkan tampilan detail laporan dosen ditunjukkan pada Gambar 4.63.



#id	PROGRAM STUDY	Class No	Tahun	Term	Subject	Respondent	Mean	Detail
11	Sistem Informasi	1012	2019	ganjil	Struktur Data	1	5.66	Detail
18	Sistem Informasi	1212	2019	ganjil	Algoritma Pemrograman Dasar	1	5.95	Detail
19	Sistem Informasi	2222	2019	ganjil	Manajemen Sistem Informasi	1	4.42	Detail
20	Sistem Informasi	4432	2019	ganjil	Struktur Data	1	6.05	Detail
21	Sistem Informasi	3312	2019	ganjil	Sistem Basis Data	1	5.37	Detail
22	Sistem Informasi	2234	2019	ganjil	Pemrograman Mobile	1	6.58	Detail
23	Sistem Informasi	1012	2019	ganjil	Pemrograman Mobile	1	6.00	Detail
24	Sistem Informasi	1012	2019	ganjil	Sistem Basis Data	1	6.00	Detail
25	Sistem Informasi	1441	2019	ganjil	Algoritma Pemrograman Dasar	1	6.00	Detail
26	Sistem Informasi	2121	2019	ganjil	Sistem Basis Data	1	6.50	Detail

Gambar 4.61 Tampilan Penilaian *Super Admin* dan *Staff*

REPORT DETAIL

DETAIL SURVEY

Dosen:	course_name:	class_code:	term:	program_study_name:	Tanggal:
testkaprodifik	Struktur Data	1012	ganjil	Sistem Informasi	2019-05-15 00:00:00

Question	question group	Mean	Indicator
Dosen menerangkan pelajaran dengan jelas (id) 1 (en)	information_provider	7.00	excellent
Dosen menitikberatkan konsep-konsep yang dianggapnya penting. (id) 2 (en)	information_provider	6.00	excellent
Cara dosen menyampaikan materi kuliah membuat mahasiswa memperhatikan dengan baik. (id) 3 (en)	information_provider	5.00	Good
Dosen menguasai materi dengan baik. (id) 4 (en)	information_provider	7.00	excellent
Dosen mampu menjawab pertanyaan mahasiswa dengan baik. (id) 5 (en)	information_provider	5.00	Good
Dosen membantu saya meningkatkan kemampuan berpikir kritis dalam bidang yang diajarkan. (id) 6 (en)	information_provider	6.00	excellent
Dosen membantu saya memahami bagaimana ilmu yang diperoleh dari perkuliahan ini dapat diterapkan dalam berbagai situasi. (id) 7 (en)	information_provider	7.00	excellent
Dosen memulai kuliah tepat waktu. (id) 8 (en)	student_learning	5.00	Good
Dosen mengakhiri kuliah tepat waktu. (id) 9 (en)	student_learning	4.00	Fair
Dosen menunjukan komitmen dalam mengajar (memiliki persiapan yang baik, memperhatikan apakah mahasiswa sudah memahami penjelasannya atau belum). (id) 10 (en)	student_learning	5.00	Good
Dosen memperlakukan mahasiswa dengan pantas. (id) 11 (en)	student_learning	6.00	excellent
Dosen menunjukkan integritasnya (jujur, adil, beretika). (id) 12 (en)	student_learning	5.00	Good
Dosen menyediakan waktu untuk memberi bimbingan. (id) 13 (en)	facilitator	6.00	excellent

Gambar 4.62 Tampilan *Report Super Admin* dan *Staff*

Dosen mendorong mahasiswa agar berpartisipasi aktif di kelas. (id) 14 (en)	facilitator	7.00	excellent
Dosen memberi tugas-tugas yang relevan. (id) 15 (en)	assesor	7.00	excellent
Dosen memberi masukkan pada mahasiswa mengenai kemajuan studi mereka (a.1. memeri umpan balik tepat waktu bagi tugas-tugas mahasiswa). (id) 16 (en)	assesor	5.00	Good
Soal-soal ujian memberi tantangan bagi saya untuk menunjukkan kemampuan berpikir kritis. (id) 17 (en)	assesor	6.00	excellent
Soal-soal ujian mencakup isi perkuliahan (sebagaimana diajarkan dan ditugaskan). (id) 18 (en)	assesor	5.00	Good
Saya belajar banyak di kelas ini. (id) 19 (en)	assesor	5.00	Good

REPORT BY QUESTION GROUP

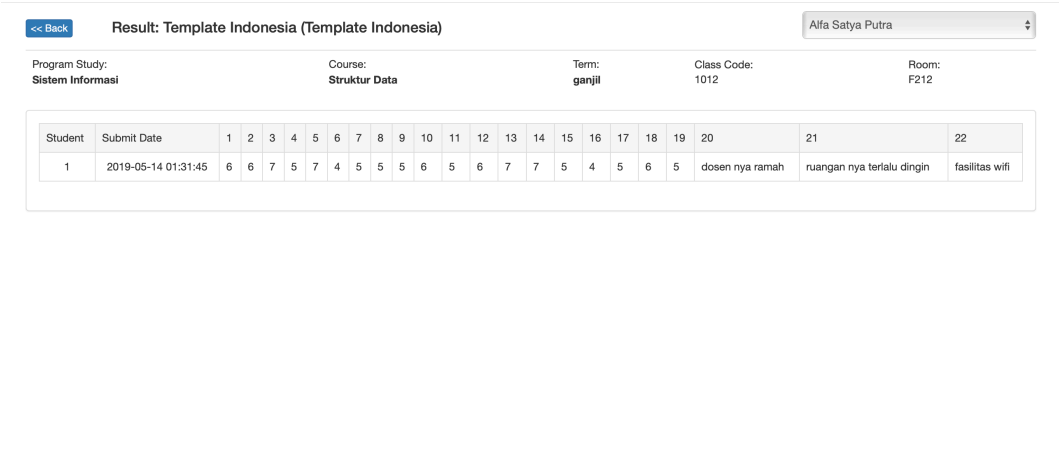
Show 10 entries Search:

program_study_name	course_name	Term	class_code	Responden	assesorr	facilitator	information_provider	student_learning	Avg Mean
Sistem Informasi	Struktur Data	ganjil	1012	1	5.60	6.50	6.14	5.00	5.66

Showing 1 to 1 of 1 entries

Previous 1 Next

Gambar 4.62 Tampilan *Report Super Admin* dan *Staff* Lanjut



Student	Submit Date	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
1	2019-05-14 01:31:45	6	6	7	5	7	4	5	5	5	6	5	6	7	7	5	4	5	6	5	dosen nya ramah	ruangan nya terlalu dingin	fasilitas wifi

Gambar 4.63 Tampilan *Detail Report Super Admin* dan *Staff*

16) Tampilan halaman *report* untuk dosen

Setelah mahasiswa telah mengisi survei, data yang masuk dalam sistem akan diolah menjadi informasi yang berbentuk laporan penilaian. Laporan di bagi menjadi dua yaitu laporan dosen dan laporan detail dosen. Penilaian dosen ditunjukkan pada Gambar 4.64. Setiap dosen akan memiliki laporan penilaian dosen sesuai dengan mata kuliah yang diajar oleh dosen tersebut. Laporan dosen dapat dilihat pada Gambar 4.65. Sedangkan tampilan detail laporan dosen ditunjukkan pada Gambar 4.66.

REPORT DOSEN

2019 | ganjil

Show 10 entries Search:

#id	PROGRAM STUDY	Class No	Tahun	Term	Subject	Respondent	Mean	Detail
11	Sistem Informasi	1012	2019	ganjil	Struktur Data	1	5.66	Detail
18	Sistem Informasi	1212	2019	ganjil	Algoritma Pemrograman Dasar	1	5.95	Detail
19	Sistem Informasi	2222	2019	ganjil	Manajemen Sistem Informasi	1	4.42	Detail
20	Sistem Informasi	4432	2019	ganjil	Struktur Data	1	6.05	Detail
21	Sistem Informasi	3312	2019	ganjil	Sistem Basis Data	1	5.37	Detail
22	Sistem Informasi	2234	2019	ganjil	Pemrograman Mobile	1	6.58	Detail
23	Sistem Informasi	1012	2019	ganjil	Pemrograman Mobile	1	6.00	Detail
24	Sistem Informasi	1012	2019	ganjil	Sistem Basis Data	1	6.00	Detail
25	Sistem Informasi	1441	2019	ganjil	Algoritma Pemrograman Dasar	1	6.00	Detail
26	Sistem Informasi	2121	2019	ganjil	Sistem Basis Data	1	6.50	Detail

Showing 1 to 10 of 13 entries

Previous 1 2 Next

Gambar 4.64 Tampilan Penilaian Dosen

REPORT DETAIL

DETAIL SURVEY

Dosen:	course_name:	class_code:	term:	program_study_name:	Tanggal:
Alfa Satya Putra	Sistem Basis Data	2121	ganjil	Sistem Informasi	2019-05-28 00:00:00

Question	question group	Mean	Indicator
Dosen menerangkan pelajaran dengan jelas (id) 1 (en)	information_provider	7.00	excellent
Dosen menitikberatkan konsep-konsep yang dianggapnya penting. (id) 2 (en)	information_provider	7.00	excellent
Cara dosen menyampaikan materi kuliah membuat mahasiswa memperhatikan dengan baik. (id) 3 (en)	information_provider	7.00	excellent
Dosen menguasai materi dengan baik. (id) 4 (en)	information_provider	7.00	excellent
Dosen mampu menjawab pertanyaan mahasiswa dengan baik. (id) 5 (en)	information_provider	7.00	excellent
Dosen membantu saya meningkatkan kemampuan berpikir kritis dalam bidang yang diajarkan. (id) 6 (en)	information_provider	7.00	excellent
Dosen membantu saya memahami bagaimana ilmu yang diperoleh dari perkuliahan ini dapat diterapkan dalam berbagai situasi. (id) 7 (en)	information_provider	7.00	excellent
Dosen memulai kuliah tepat waktu. (id) 8 (en)	student_learning	7.00	excellent
Dosen mengakhiri kuliah tepat waktu. (id) 9 (en)	student_learning	7.00	excellent
Dosen menunjukan komitmen dalam mengajar (memiliki persiapan yang baik, memperhatikan apakah mahasiswa sudah memahami penjelasannya atau belum). (id) 10 (en)	student_learning	7.00	excellent
Dosen memperlakukan mahasiswa dengan pantas. (id) 11 (en)	student_learning	7.00	excellent
Dosen menunjukkan integritasnya (jujur, adil, beretika). (id) 12 (en)	student_learning	7.00	excellent
Dosen menyediakan waktu untuk memberi bimbingan. (id) 13 (en)	facilitator	7.00	excellent

Gambar 4.65 Tampilan *Report* Dosen

Dosen mendorong mahasiswa agar berpartisipasi aktif di kelas. (id) 14 (en)	facilitator	7.00	excellent
Dosen memberi tugas-tugas yang relevan. (id) 15 (en)	assesor	7.00	excellent
Dosen memberi masukkan pada mahasiswa mengenai kemajuan studi mereka (a.1. memeri umpan balik tepat waktu bagi tugas-tugas mahasiswa). (id) 16 (en)	assesor	7.00	excellent
Soal-soal ujian memberi tantangan bagi saya untuk menunjukkan kemampuan berpikir kritis. (id) 17 (en)	assesor	7.00	excellent
Soal-soal ujian mencakup isi perkuliahan (sebagaimana diajarkan dan ditugaskan). (id) 18 (en)	assesor	7.00	excellent
Saya belajar banyak di kelas ini. (id) 19 (en)	assesor	7.00	excellent

REPORT BY QUESTION GROUP

Show 10 entries Search:

program_study_name	course_name	Term	class_code	Responden	assesorr	facilitator	information_provider	student_learning	Avg Mean
Sistem Informasi	Sistem Basis Data	ganjil	2121	1	7.00	7.00	7.00	7.00	6.50

Showing 1 to 1 of 1 entries

Previous 1 Next

Gambar 4.65 Tampilan *Report* Dosen Lanjut

<< Back Result: Template Indonesia (Uji Coba 10)

Program Study: **Sistem Informasi** Course: **Sistem Basis Data** Term: **ganjil** Class Code: 2121 Room: B341

Student	Submit Date	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
1	2019-05-27 22:13:21	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	test	test	test

Gambar 4.66 Tampilan *Detail Report* Dosen

17) Tampilan halaman *report* untuk kaprodi

Setelah mahasiswa telah mengisi survei, data yang masuk dalam sistem akan diolah menjadi informasi yang berbentuk laporan penilaian. Laporan kaprodi dapat dilihat pada Gambar 4.67.

REPORT KAPRODI

Semester ganjil-2019 Submit

Lecturer	course_name	Term	class_code	Responden	assesorr	facilitator	information_provider	student_learning	Avg Mean
testkaprodifik	Struktur Data	ganjil	1012	1	5.00	7.00	5.71	5.40	5.78
Alfa Satya Putra	Struktur Data	ganjil	1012	6	6.51	6.64	6.59	6.43	6.54
Suryasari	Manajemen Sistem Informasi	ganjil	2222	3	6.13	6.00	5.57	5.73	5.86
Heri	Algoritma Pemrograman Dasar	ganjil	1212	2	5.93	6.00	6.14	6.20	6.07
Andree Widjaja	Pemrograman Mobile	ganjil	1012	2	5.67	5.67	5.67	5.67	5.67
Arnold Aribowo	Sistem Basis Data	ganjil	3312	2	4.70	4.75	6.21	6.30	5.49
Kusno Prasetya	Sistem Basis Data	ganjil	1012	1	6.00	0.00	0.00	0.00	1.50

Gambar 4.67 Tampilan *Report* Kaprodi

18) Tampilan halaman *report* untuk dekan

Setelah mahasiswa telah mengisi survei, data yang masuk dalam sistem akan diolah menjadi informasi yang berbentuk laporan penilaian. Laporan dekan dapat dilihat pada Gambar 4.68.

REPORT DEKAN

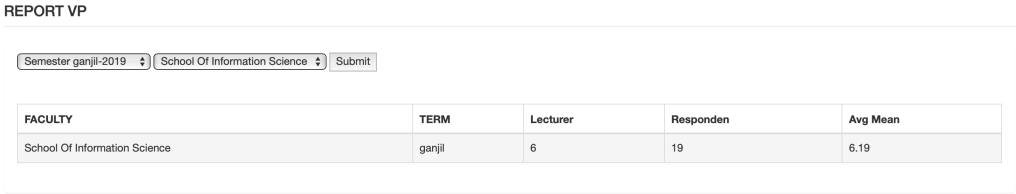
Semester ganjil-2019 Sistem Informasi Submit

PROGRAM STUDY	TERM	Lecturer	Responden	Avg Mean
School Of Information Science	ganjil	6	19	6.19

Gambar 4.68 Tampilan *Report* Dekan

19) Tampilan halaman *report* untuk *vice president*

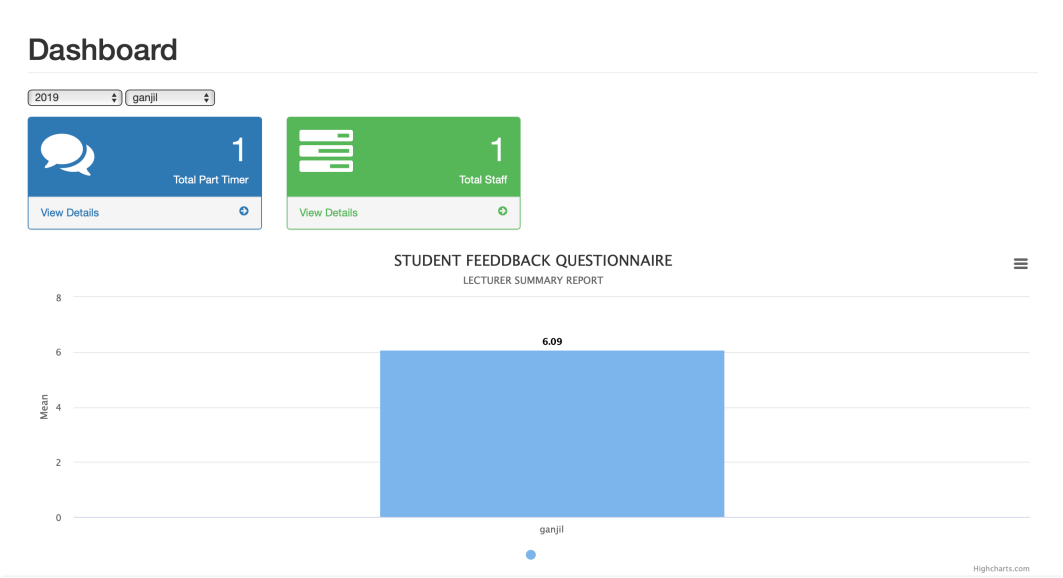
Setelah mahasiswa telah mengisi survei, data yang masuk dalam sistem akan diolah menjadi informasi yang berbentuk laporan penilaian. Laporan *vice president* dapat dilihat pada Gambar 4.69.



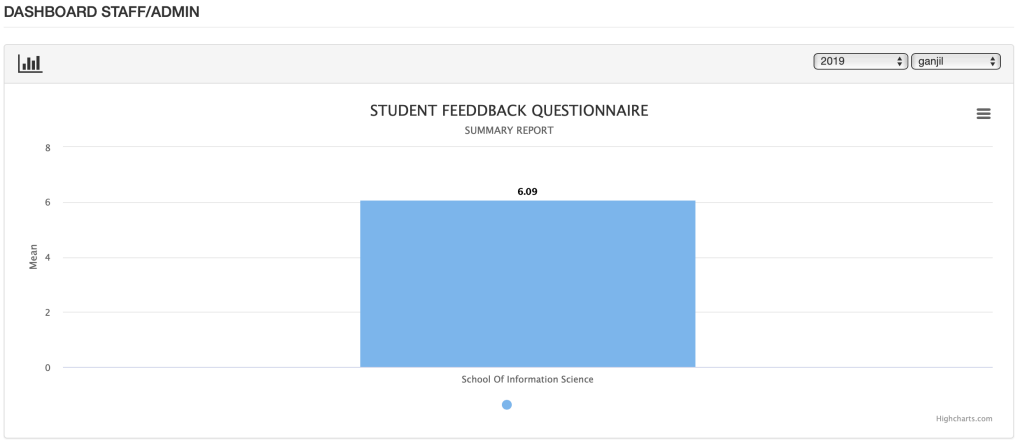
Gambar 4.69 Tampilan *Report VP*

20) Tampilan dashboard *super admin* dan *staff*

Setelah mahasiswa telah mengisi survei, data yang masuk dalam sistem akan diolah menjadi informasi yang berbentuk grafik. Grafik ini akan ditampilkan pada dashboard *super admin* dan *staff* pada Gambar 4.70. Dan jika grafik nya di tekan maka akan memunculkan grafik-grafik fakultas pada Gambar 4.71.



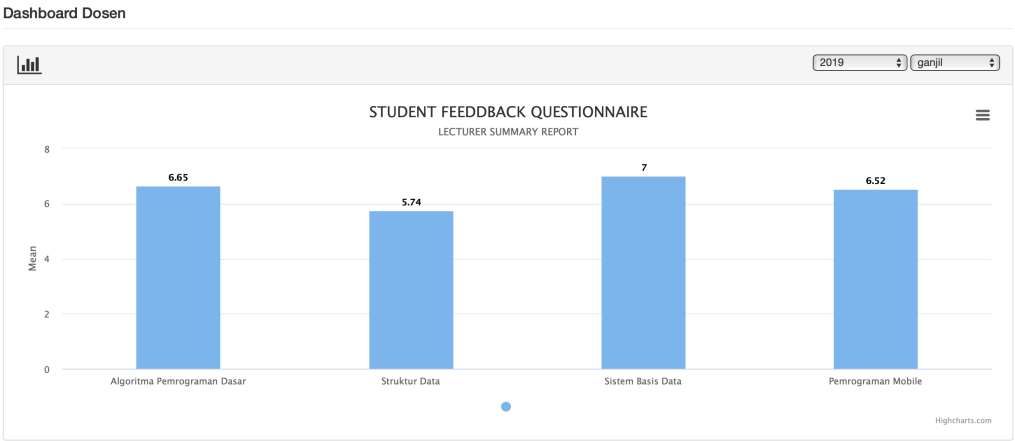
Gambar 4.70 Tampilan *Dashboard Super Admin* dan *Staff*



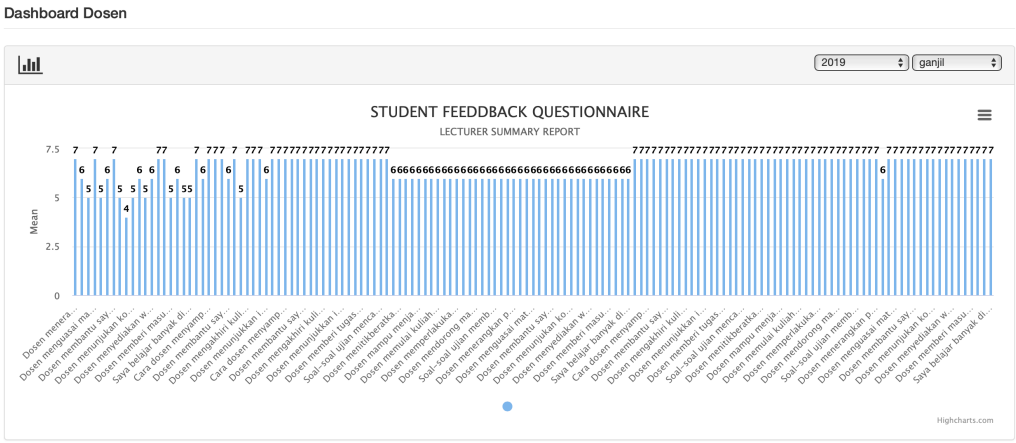
Gambar 4.71 Tampilan *Dashboard Detail Super Admin* dan *Staff*

21) Tampilan dashboard dosen

Setelah mahasiswa telah mengisi survei, data yang masuk dalam sistem akan diolah menjadi informasi yang berbentuk grafik. Grafik ini akan ditampilkan pada dashboard dosen pada Gambar 4.72. Dan jika grafik nya di tekan maka akan memunculkan grafik-grafik pertanyaan survei pada Gambar 4.73.



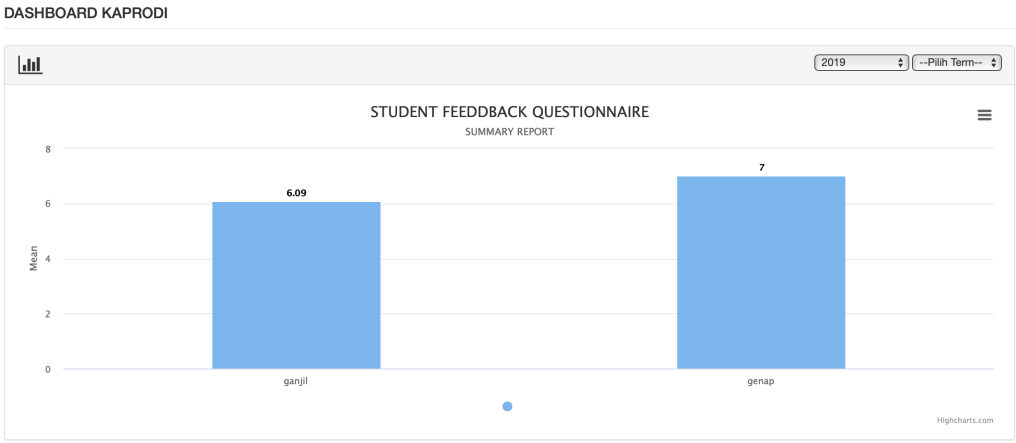
Gambar 4.72 Tampilan *Dashboard* Dosen



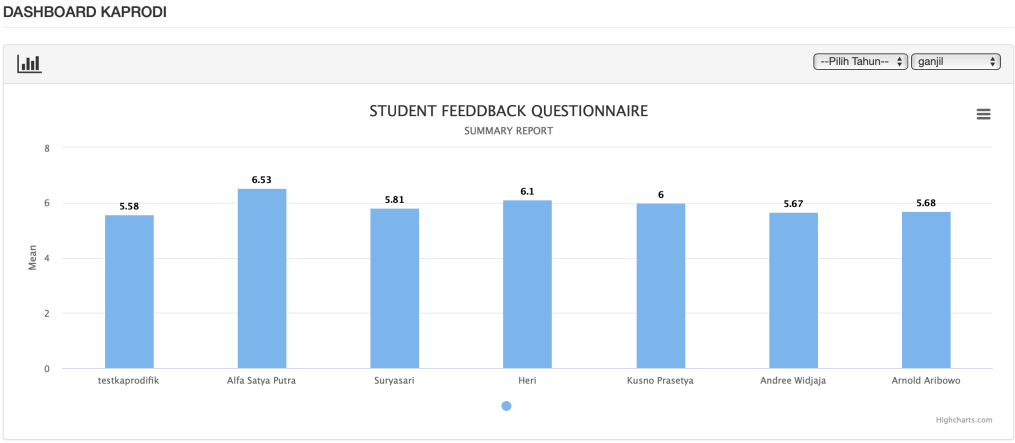
Gambar 4.73 Tampilan *Dashboard Detail* Dosen

22) Tampilan dashboard kaprodi

Setelah mahasiswa telah mengisi survei, data yang masuk dalam sistem akan diolah menjadi informasi yang berbentuk grafik. Grafik ini akan ditampilkan pada dashboard kaprodi pada Gambar 4.74. Jika grafik nya di tekan maka akan memunculkan grafik-grafik dosen pada Gambar 4.75.



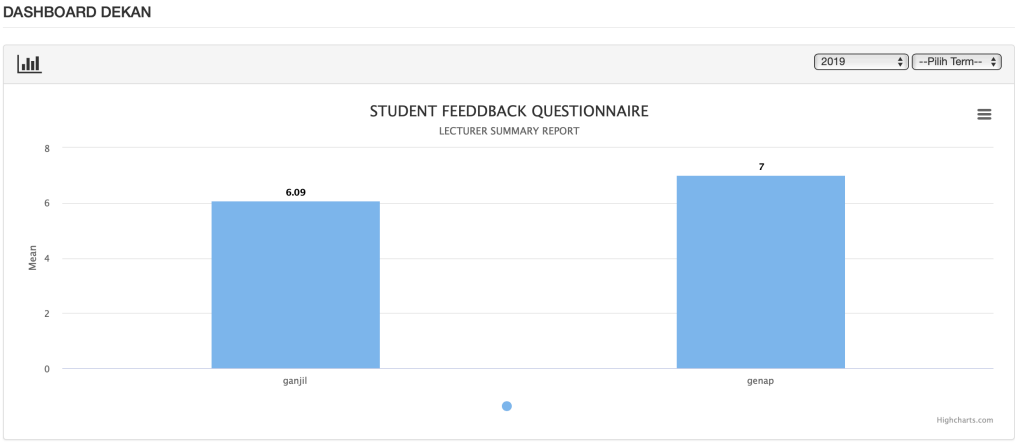
Gambar 4.74 Tampilan *Dashboard* Kaprodi



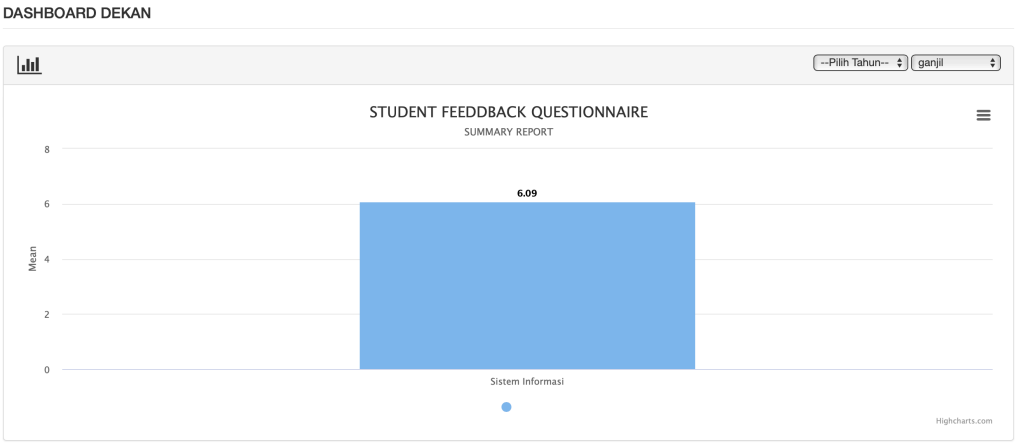
Gambar 4.75 Tampilan *Dashboard Detail* Kaprodi

23) Tampilan dashboard dekan

Setelah mahasiswa telah mengisi survei, data yang masuk dalam sistem akan diolah menjadi informasi yang berbentuk grafik. Grafik ini akan ditampilkan pada dashboard dekan pada Gambar 4.76. Jika grafik nya di tekan maka akan memunculkan grafik-grafik program studi pada Gambar 4.77.



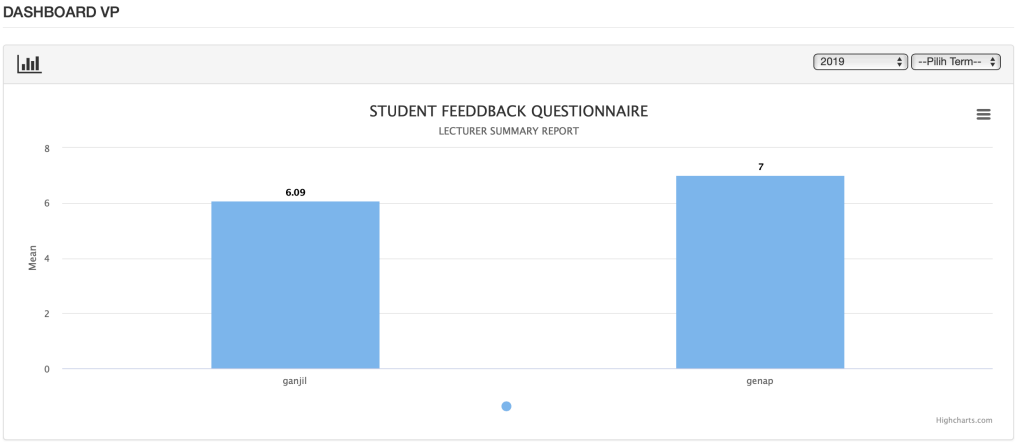
Gambar 4.76 Tampilan *Dashboard* Dekan



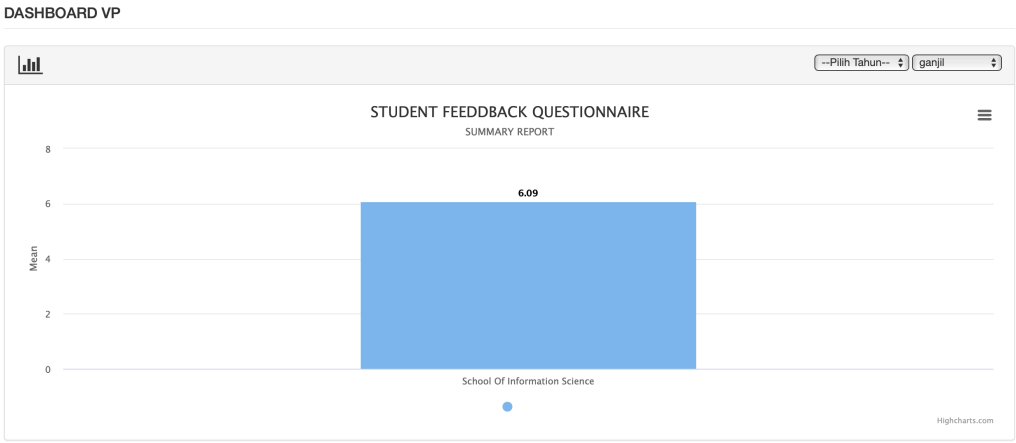
Gambar 4.77 Tampilan *Dashboard Detail* Dekan

24) Tampilan dashboard *vice president*

Setelah mahasiswa telah mengisi survei, data yang masuk dalam sistem akan diolah menjadi informasi yang berbentuk grafik. Grafik ini akan ditampilkan pada dashboard *vp* pada Gambar 4.78 Jika grafik nya di tekan maka akan memunculkan grafik-grafik fakultas pada Gambar 4.79.



Gambar 4.78 Tampilan *Dashboard Vice President*



Gambar 4.79 Tampilan *Dashboard Detail Vice President*

4.4.2 Pengujian Sistem

Proses pengujian ini dilakukan untuk memastikan setiap fungsi dapat berjalan dengan baik dan *output* yang dihasilkan dapat sesuai dengan yang diharapkan. Pengujian ini akan dilakukan dengan metode *black box testing.* Dalam proses *testing,* terdapat beberapa istilah yang akan digunakan, yaitu *pass (p)* dan *failed* (*f*). *Pass* menandakan bahwa fungsi dan hasil yang dikeluarkan oleh sistem telah berjalan dengan baik dan sesuai dengan yang diharapkan. Sedangkan *failed* menandakan fungsi tidak berjalan dengan baik dan hasil yang dikeluarkan tidak sesuai dengan apa yang diharapkan. Berikut ini beberapa skenario pengujian yang telah dilakukan. Skenario pengujian selanjutnya dapat dilihat pada lampiran.

1) ***Scenario* : *Login***

   ***Scenario Description* :** *Scenario* ini menjelaskan *proses* masuk ke dalam sistem.

   ***Test Scripts* :**

   1) *Login*

   ***Use Case* :**

   1) *Login*

   ***Use Groups* :**

   1) *User*

Script* 1.1 : *Login

***Script Description* :** *Test script* ini menjelaskan proses masuk kedalam sistem.

***Setup* :**

1) *User* sedang mengakses halaman *website*

***Script Steps* :**

Tabel 4.16 *Test Case* Melakukan *Login*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*User* memasukan username dan kata sandi (akun *valid)*	Sistem menampilkan halaman home	P
2	*User* memasukan *username* dan kata sandi (akun tidak *valid*)	Menampilkan pesan eror	P
3	*User* melakukan *log out* setelah *log in*	Sistem menampilkan halaman *log in*	P

2) ***Scenario* : Mengelola data *user***

***Scenario Description* :** *Scenario* ini menjelaskan proses mengelola data *user*

***Test Scripts* :**

1) Menambah data *user*
2) Mengubah data *user*
3) Melihat data *user*

***Use Case* :**

1) Mengelola data *user*

***Use Groups* :**

1) *Staff*

***Script* 2.1 :** Menambah data *user*

***Script Description* :** *Test script* ini menjelaskan proses menambah data *user*

***Setup* :**

1) *Staff* telah melakukan *login*

2) *Staff* sedang mengakses halaman *user*

***Script Steps* :**

Tabel 4.17 *Test Case* Menambah *Data User*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Staff* memilih *add user*	Sistem menampilkan form penambahan data *user*	P
2	*Staff* menambah data *user*	Sistem menyimpan data *user* dan kembali ke halaman *user*	P

***Script* 2.2 :** Mengubah dan melihat data *user*

***Script Description* :** *Test script* ini menjelaskan proses mengubah data *user*

***Setup* :**

1) *Staff* telah melakukan *login*
2) *Staff* sedang mengakses halaman *user*

***Script Steps* :**

Tabel 4.18 *Test Case* Mengubah *Data User*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Staff* memilih *edit user*	Sistem menampilkan form pengubahan data *user*	P
2	*Staff* mengubah data *user*	Sistem menyimpan data *user* yang telah diubah dan kembali ke halaman *user*	P

3) ***Scenario* : Mengelola *user type***

***Scenario Description* :** *Scenario* ini menjelaskan proses mengelola data *user type*

***Test Scripts* :**

1) Mengubah data *user type*

***Use Case* :**

1) Mengelola data *user type*

***Use Groups* :**

1) *Staff*

***Script* 3.1 :** Menambah data *user type*

***Script Description* :** *Test script* ini menjelaskan proses menambah data *user type*

***Setup* :**

1) *Staff* telah melakukan *login*
2) *Staff* sedang mengakses halaman *user*

***Script Steps* :**

Tabel 4.19 *Test Case* Mengelola *User Type*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Staff* memilih *add user type*	Sistem menampilkan form penambahan data *user type*	P
2	*Staff* menambah data *user type*	Sistem menyimpan data *user type* dan kembali ke halaman *user type*	P

***Script* 3.2 :** Mengubah dan melihat data *user type*

***Script Description* :** *Test script* ini menjelaskan proses mengubah dan melihat data *user type*

***Setup* :**

1) *Staff* telah melakukan *login*
2) *Staff* sedang mengakses halaman *user*

***Script Steps* :**

Tabel 4.20 *Test Case* Mengubah *User Type*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Staff* memilih *edit user type*	Sistem menampilkan form pengubahan data *user type*	P
2	*Staff* mengubah data *user type*	Sistem menyimpan data *user type* yang telah diubah dan kembali ke halaman *user type*	P

4) ***Scenario* : Mengelola *campus management***

***Scenario Description* :** *Scenario* ini menjelaskan proses mengelola data fakultas, program studi, dan mata kuliah

***Test Scripts* :**

1) Menambah data *faculty*
2) Mengubah data *faculty*
3) Menambah data *program study*
4) Mengubah data *program study*
5) Menambah data *course*
6) Menambah data *course*

***Use Case* :**

1) Mengelola *campus management*

***Use Groups* :**

1) *Staff*

***Script* 4.1 :** Menambah data *faculty*

***Script Description* :** *Test script* ini menjelaskan proses menambah data *faculty*

***Setup* :**

1) *Staff* telah melakukan *login*
2) *Staff* sedang mengakses halaman *faculty*

***Script Steps* :**

Tabel 4.21 *Test Case* Menambah *Faculty*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Staff* memilih *add faculty*	Sistem menampilkan form penambahan data *faculty*	P
2	*Staff* menambah data *faculty*	Sistem menyimpan data *faculty* dan kembali ke halaman *faculty*	P

***Script* 4.2 :** Mengubah data *faculty*

***Script Description* :** *Test script* ini menjelaskan proses mengubah data *faculty*

***Setup* :**

1) *Staff* telah melakukan *login*
2) *Staff* sedang mengakses halaman *faculty*

***Script Steps* :**

Tabel 4.22 *Test Case* Mengubah *Faculty*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Staff* memilih *edit faculty*	Sistem menampilkan form pengubahan data *faculty*	P
2	*Staff* mengubah data *faculty*	Sistem menyimpan data *faculty* yang telah diubah dan kembali ke halaman *faculty*	P

***Script* 4.3 :** Menambah data *program study*

***Script Description* :** *Test script* ini menjelaskan proses menambah data *program study*

***Setup* :**

1) *Staff* telah melakukan *login*

2) *Staff* sedang mengakses halaman *program study*

***Script Steps* :**

Tabel 4.23 *Test Case* Menambah *Program Study*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Staff* memilih *add program study*	Sistem menampilkan form penambahan data *program study*	P
2	*Staff* menambah data *program study*	Sistem menyimpan data *program study* dan kembali ke halaman *program study*	P

***Script* 4.4 :** Mengubah data *program study*

***Script Description* :** *Test script* ini menjelaskan proses mengubah data *program study*

***Setup* :**

1) *Staff* telah melakukan *login*

2) *Staff* sedang mengakses halaman *program study*

***Script Steps* :**

Tabel 4.24 *Test Case* Mengubah *Program Study*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Staff* memilih *edit program study*	Sistem menampilkan form pengubahan data *program study*	P
2	*Staff* mengubah data *program study*	Sistem menyimpan data *program study* yang telah diubah dan kembali ke halaman *program study*	P

***Script* 4.5 :** Menambah data *course*

***Script Description* :** *Test script* ini menjelaskan proses menambah data *course*

***Setup* :**

1) *Staff* telah melakukan *login*
2) *Staff* sedang mengakses halaman *course*

***Script Steps* :**

Tabel 4.25 *Test Case* Menambah *Course*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Staff* memilih *add course*	Sistem menampilkan form penambahan data *course*	P
2	*Staff* menambah data *course*	Sistem menyimpan data *course* dan kembali ke halaman *course*	P

***Script* 4.6 :** Mengubah data *course*

***Script Description* :** *Test script* ini menjelaskan proses mengubah data *course*

***Setup* :**

1) *Staff* telah melakukan *login*
2) *Staff* sedang mengakses halaman *course*

***Script Steps* :**

Tabel 4.26 *Test Case* Mengubah *Course*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Staff* memilih *edit course*	Sistem menampilkan form pengubahan data *course*	P
2	*Staff* mengubah data *course*	Sistem menyimpan data *course* yang telah diubah dan kembali ke halaman *course*	P

5) ***Scenario* : Mengelola *menu management***

***Scenario Description* :** *Scenario* ini menjelaskan proses mengelola data akses menu, menu, dan *sub* menu

***Test Scripts* :**

1) Menambah akses menu
2) Mengubah akses menu
3) Mengubah menu
4) Mengubah *sub* menu

***Use Case* :**

1) Mengelola menu management

***Use Groups* :**

1) *Staff*

***Script* 5.1 :** Menambah data akses menu

***Script Description* :** *Test script* ini menjelaskan proses menambah data akses menu

***Setup* :**

1) *Staff* telah melakukan *login*
2) *Staff* sedang mengakses halaman akses menu

***Script Steps* :**

Tabel 4.27 *Test Case* Menambah *Akses Menu*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Staff* memilih *add* akses menu	Sistem menampilkan form penambahan data akses menu	P
2	*Staff* menambah data akses menu	Sistem menyimpan data akses menu dan kembali ke halaman akses menu	P

***Script* 5.2 :** Mengubah data akses menu

***Script Description* :** *Test script* ini menjelaskan proses mengubah data akses menu

***Setup* :**

1) *Staff* telah melakukan *login*
2) *Staff* sedang mengakses halaman akses menu

***Script Steps* :**

Tabel 4.28 *Test Case* Mengubah *Akses Menu*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Staff* memilih *edit* akses menu	Sistem menampilkan form pengubahan data akses menu	P
2	*Staff* mengubah data akses menu	Sistem menyimpan data akses menu yang telah diubah dan kembali ke halaman akses menu	P

***Script* 5.3 :** Mengubah data menu

***Script Description* :** *Test script* ini menjelaskan proses Mengubah data menu

***Setup* :**

1) *Staff* telah melakukan *login*

2) *Staff* sedang mengakses halaman menu

***Script Steps* :**

Tabel 4.29 *Test Case* Mengubah *Menu*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Staff* memilih *edit* menu	Sistem menampilkan form pengubahan data menu	P
2	*Staff* mengubah data menu	Sistem menyimpan data menu yang telah diubah dan kembali ke halaman menu	P

***Script* 5.4 :** Mengubah data *sub* menu

***Script Description* :** *Test script* ini menjelaskan proses mengubah data *sub* menu

***Setup* :**

1) *Staff* telah melakukan *login*

2) *Staff* sedang mengakses halaman *sub* menu

***Script Steps* :**

Tabel 4.30 *Test Case* Mengubah *Sub Menu*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Staff* memilih *edit sub* menu	Sistem menampilkan form pengubahan data *sub* menu	P
2	*Staff* mengubah data *sub* menu	Sistem menyimpan data *sub* menu yang telah diubah dan kembali ke halaman *sub* menu	P

6) ***Scenario* : Mengelola *survey***

***Scenario Description* :** *Scenario* ini menjelaskan proses mengelola data survei

***Test Scripts* :**

1) Menambah *question set*
2) Mengubah *question set*
3) Menambah *question*
4) Mengubah *question*
5) Melihat question
6) Menpublish *question set*

***Use Case* :**

1) Mengelola *question set*

***Use Groups* :**

1) *Staff*
2) *Part timer*

***Script* 6.1 :** Menambah *question set*

***Script Description* :** *Test script* ini menjelaskan proses menambah data *question set*

***Setup* :**

1) *Staff* atau *part timer* telah melakukan *login*
2) *Staff* atau *part timer* sedang mengakses halaman *question set*

***Script Steps* :**

Tabel 4.31 *Test Case* Menambah *Question Set*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Staff* atau *part timer* memilih *add question set*	Sistem menampilkan form penambahan data *question set*	P
2	*Staff* atau *part timer* menambah data *question set*	Sistem menyimpan data *question set* dan kembali ke halaman *question set*	P

Tabel 4.31 *Test Case* Menambah *Question Set*

3	*Staff* atau *part timer* memilih *copy question set*	Sistem akan men *copy* data *question set* yang dipilih	P

***Script* 6.2 :** Mengubah *question set*

***Script Description* :** *Test script* ini menjelaskan proses mengubah data *question set*

***Setup* :**

1) *Staff* atau *part timer* telah melakukan *login*
2) *Staff* atau *part timer* sedang mengakses halaman *question set*

***Script Steps* :**

Tabel 4.32 *Test Case* Mengubah *Question Set*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Staff* atau *part timer* memilih *edit question set*	Sistem menampilkan form pengubahan data *question set*	P
2	*Staff* atau *part timer* mengubah data *question set*	Sistem menyimpan data *question set* yang telah diubah dan kembali ke halaman *question set*	P

***Script* 6.3 :** Menambah *question*

***Script Description* :** *Test script* ini menjelaskan proses menambah data *question*

***Setup* :**

1) *Staff* atau *part timer* telah melakukan *login*
2) *Staff* atau *part timer* sedang mengakses halaman *question*

***Script Steps* :**

Tabel 4.33 *Test Case* Menambah *Question*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Staff* atau *part timer* memilih *add question*	Sistem menampilkan form penambahan data *question*	P
2	*Staff* atau *part timer* menambah data *question*	Sistem menyimpan data *question* dan kembali ke halaman *question*	P

***Script* 6.4 :** Mengubah *question*

***Script Description* :** *Test script* ini menjelaskan proses mengubah data *question*

***Setup* :**

1) *Staff* atau *part timer* telah melakukan *login*
2) *Staff* atau *part timer* sedang mengakses halaman *question*

***Script Steps* :**

Tabel 4.34 *Test Case* Mengubah *Question*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Staff* atau *part timer* memilih *edit question*	Sistem menampilkan form pengubahan data *question*	P
2	*Staff* atau *part timer* mengubah data *question*	Sistem menyimpan data *question* yang telah diubah dan kembali ke halaman *question*	P

***Script* 6.5 :** Melihat *question*

***Script Description* :** *Test script* ini menjelaskan proses melihat data *question*

***Setup* :**

1) *Staff* atau *part timer* telah melakukan *login*

2) *Staff* atau *part timer* sedang mengakses halaman *question set*

***Script Steps* :**

Tabel 4.35 *Test Case* Melihat *Question*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Staff* atau *part timer* memilih *question set* yang ingin di preview dengan menekan tombol *question*	Sistem menampilkan data *question* dari *question set* tersebut	P
2	*Staff* atau *part timer* memilih tombol *preview* sesuai dengan bahasa yang diinginkan	Sistem menampilkan data *question* dalam bentuk survei	P

***Script* 6.5 :** Menpublish *question*

***Script Description* :** *Test script* ini menjelaskan proses menpublish data *question set*

***Setup* :**

1) *Staff* atau *part timer* telah melakukan *login*

2) *Staff* atau *part timer* sedang mengakses halaman *question set*

***Script Steps* :**

Tabel 4.36 *Test Case* Menpublish *Question*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Staff* atau *part timer* memilih *question set* yang ingin di share dan menekan tombol *question*	Sistem menampilkan data *question* dari *question set* tersebut	P

Tabel 4.36 *Test Case* Menpublish *Question*

2	*Staff* atau *part timer* memilih *share link* sesuai bahasa yang di inginkan	Sistem menampilkan data *question* dan *link question*	P

7) ***Scenario* : Melihat hasil *survey***

***Scenario Description* :** *Scenario* ini menjelaskan proses melihat hasil survei

***Test Scripts* :**

1) Melihat hasil survei

***Use Case* :**

1) Melihat hasil survei

***Use Groups* :**

1) *Staff*
2) *Part timer*

***Script* 7.1 :** Melihat *question set*

***Script Description* :** *Test script* ini menjelaskan proses melihat hasil survei yang sudah di publish

***Setup* :**

1) *Staff* atau *part timer* telah melakukan *login*
2) *Staff* atau *part timer* sedang mengakses halaman *question set*

***Script Steps* :**

Tabel 4.37 *Test Case* Melihat Hasil Survei

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Staff* atau *part timer* memilih *question set* yang ingin di lihat dengan menekan tombol *result*	Sistem menampilkan data *result* dari *question set* tersebut	P

8) ***Scenario* : Melihat laporan hasil evaluasi**

***Scenario Description* :** *Scenario* ini menjelaskan proses melihat laporan hasil penilaian untuk dosen, kaprodi, dekan dan *vice president*.

***Test Scripts* :**

1) Melihat hasil evaluasi SFQ untuk dosen.
2) Melihat hasil evaluasi SFQ untuk kaprodi.
3) Melihat hasil evaluasi SFQ untuk dekan.
4) Melihat hasil evaluasi SFQ untuk *vice president*.

***Use Case* :**

1) Melihat hasil survei

***Use Groups* :**

1) Dosen
2) Kaprodi
3) Dekan
4) *Vice president*

***Script* 8.1 :** Melihat hasil evaluasi SFQ untuk dosen.

***Script Description* :** *Test script* ini menjelaskan proses melihat hasil survei untuk dosen

***Setup* :**

1) Dosen telah melakukan *login*
2) Dosen sedang mengakses halaman dashboard

***Script Steps* :**

Tabel 4.38 *Test Case* Melihat Laporan Dosen

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	Dosen memilih grafik mata pelajaran yang ingin di lihat	Sistem menampilkan data detail laporan survei dosen	P

***Script* 8.2 :** Melihat hasil evaluasi SFQ untuk kaprodi.

***Script Description* :** *Test script* ini menjelaskan proses melihat hasil survei untuk kaprodi

***Setup* :**

1) Kaprodi telah melakukan *login*
2) Kaprodi sedang mengakses halaman dashboard

***Script Steps* :**

Tabel 4.39 *Test Case* Melihat Laporan Kaprodi

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	Kaprodi memilih grafik mata pelajaran yang ingin di lihat	Sistem menampilkan data detail laporan survei dosen	P

***Script* 8.3 :** Melihat hasil evaluasi SFQ untuk dekan.

***Script Description* :** *Test script* ini menjelaskan proses melihat hasil survei untuk dekan

***Setup* :**

1) Dekan telah melakukan *login*
2) Dekan sedang mengakses halaman dashboard

***Script Steps* :**

Tabel 4.40 *Test Case* Melihat Laporan Dekan

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	Dekan memilih grafik mata pelajaran yang ingin di lihat	Sistem menampilkan data detail laporan survei dosen	P

***Script* 8.4 :** Melihat hasil evaluasi SFQ untuk *vice president*.

***Script Description* :** *Test script* ini menjelaskan proses melihat hasil survei untuk *vice president*

***Setup* :**

1) *Vice president* telah melakukan *login*
2) *Vice president* sedang mengakses halaman dashboard

***Script Steps* :**

Tabel 4.41 *Test Case* Melihat Laporan *VP*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Vice president* memilih grafik mata pelajaran yang ingin di lihat	Sistem menampilkan data detail laporan survei dosen	P

9) ***Scenario* : Mengubah *profile***

***Scenario Description* :** *Scenario* ini menjelaskan proses mengubah *profile*

***Test Scripts* :**

1) Mengubah *profile*

***Use Case* :**

1) Mengubah *password*

***Use Groups* :**

1) *User*

***Script* 9.1 :** Mengubah *profile*.

***Script Description* :** *Test script* ini menjelaskan proses mengubah *profile*

***Setup* :**

1) *User* telah melakukan *login*
2) *User* sedang mengakses halaman ubah *profile*

***Script Steps* :**

Tabel 4.42 *Test Case* Mengubah *Profile*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*User* mengisi form yang telah disediakan dengan benar	Sistem menyimpan gelar dan *password* yang baru dimasukkan	P

10) ***Scenario* : Melihat *dashboard***

***Scenario Description* :** *Scenario* ini menjelaskan proses melihat *dashboard*

***Test Scripts* :**

1) Melihat hasil *dashboard* untuk *super admin*.
2) Melihat hasil *dashboard* untuk *staff*.

3) Melihat hasil *dashboard* untuk *part timer*.

4) Melihat hasil *dashboard* untuk dosen.

5) Melihat hasil *dashboard* untuk kaprodi.

6) Melihat hasil *dashboard* untuk dekan.

7) Melihat hasil *dashboard* untuk *vice president*.

***Use Case* :**

1) Melihat *dashboard*

***Use Groups* :**

1) *Super admin*

2) *Staff*

3) *Part timer*

4) Dosen

5) Kaprodi

6) Dekan

7) *Vice president*

***Script* 10.1 :** Melihat *dashboard* untuk *super admin*.

***Script Description* :** *Test script* ini menjelaskan proses melihat *dashboard*

***Setup* :**

1) *Super* admin telah melakukan *login*

2) *Super* admin sedang mengakses halaman *dashboard*

***Script Steps* :**

Tabel 4.43 *Test Case* Melihat *Dashboard Super Admin*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Super Admin* memilih grafik term yang telah disurvei	Sistem menampilkan data grafik term	P
2	*Super Admin* melihat grafik fakultas	Sistem menampilkan data grafik fakultas	P

***Script* 10.2 :** Melihat *dashboard* untuk *staff*.

***Script Description* :** *Test script* ini menjelaskan proses melihat *dashboard*

***Setup* :**

1) *Staff* telah melakukan *login*
2) *Staff* sedang mengakses halaman *dashboard*

***Script Steps* :**

Tabel 4.44 *Test Case* Melihat *Dashboard Staff*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Staff* memilih grafik term yang telah disurvei	Sistem menampilkan data grafik term	P
2	*Staff* melihat grafik fakultas	Sistem menampilkan data grafik fakultas	P

***Script* 10.3 :** Melihat *dashboard* untuk *part timer*.

***Script Description* :** *Test script* ini menjelaskan proses melihat *dashboard*

***Setup* :**

1) *Part timer* telah melakukan *login*
2) *Part timer* sedang mengakses halaman *dashboard*

***Script Steps* :**

Tabel 4.45 *Test Case* Melihat *Dashboard Part Timer*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Part timer* masuk ke menu *dashboard*	Sistem menampilkan jumlah *staff* dan *part timer*	P

***Script* 10.4 :** Melihat *dashboard* untuk dosen.

***Script Description* :** *Test script* ini menjelaskan proses melihat *dashboard*

***Setup* :**

1) Dosen telah melakukan *login*
2) Dosen sedang mengakses halaman dashboard

***Script Steps* :**

Tabel 4.46 *Test Case* Melihat *Dashboard* Dosen

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	Dosen memilih grafik mata kuliah yang ingin di lihat	Sistem menampilkan data detail grafik mata kuliah tersebut	P

***Script* 10.5 :** Melihat *dashboard* untuk kaprodi.

***Script Description* :** *Test script* ini menjelaskan proses melihat *dashboard*

***Setup* :**

1) Kaprodi telah melakukan *login*
2) Kaprodi sedang mengakses halaman *dashboard*

***Script Steps* :**

Tabel 4.47 *Test Case* Melihat *Dashboard* Kaprodi

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	Kaprodi memilih grafik term yang telah disurvei	Sistem menampilkan data grafik term	P
2	Kaprodi melihat grafik dosen	Sistem menampilkan data grafik dosen	P

***Script* 10.6 :** Melihat dashboard untuk dekan.

***Script Description* :** *Test script* ini menjelaskan proses melihat *dashboard*

***Setup* :**

1) Dekan telah melakukan *login*
2) Dekan sedang mengakses halaman *dashboard*

***Script Steps* :**

Tabel 4.48 *Test Case* Melihat *Dashboard* Dekan

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	Dekan memilih grafik program studi yang ingin di lihat	Sistem menampilkan data detail grafik program studi tersebut	P

***Script* 10.7 :** Melihat *dashboard* untuk *vice president*.

***Script Description* :** *Test script* ini menjelaskan proses melihat *dashboard*

***Setup* :**

1) *Vice president* telah melakukan *login*
2) *Vice president* sedang mengakses halaman *dashboard*

***Script Steps* :**

Tabel 4.49 *Test Case* Melihat *Dashboard Vice President*

Step	***Test Action***	***Expected Result***	***Result (P/F)***
1	*Vice president* memilih grafik term yang telah disurvei	Sistem menampilkan data grafik term	P
2	*Vice president* melihat grafik fakultas	Sistem menampilkan data grafik fakultas	P

4.4.3 Analisis Pengujian dan Perbandingan Sistem

Berdasarkan hasil pengujian dari sistem usulan ini, dapat dilihat perbandingan antara sistem usulan dengan aplikasi yang telah digunakan LP2MP pada Tabel 4.50. Hasil pengujian dan perbandingan sistem didapat dari hasil wawancara dan pengukuran waktu dengan pihak LP2MP. Setelah dilakukan pengujian, sistem usulan ini dapat menghemat waktu sebesar 50% dari sistem lama. Hasil ini didapat dengan membandingkan proses pembuatan survei dalam satu jadwal visitasi. Waktu yang dibutuhkan dalam pembuatan survei sistem lama, yaitu sebesar 1-2 menit. Sedangkan waktu yang dibutuhkan untuk pembuatan survei pada sistem usulan, yaitu sebesar 20-25 detik. Untuk waktu yang dibutuhkan untuk mengolah laporan pada sistem lama dibutuhkan waktu 2-3 menit untuk satu laporan sedangkan pada sistem usulan laporan akan diolah otomatis oleh sitem sehingga hasil laporan dalam langsung dilihat oleh *user*

Tabel 4.50 Tabel Perbandingan Sistem Usulan Dengan Sistem Saat Ini

No	Kriteria	Limesurvey	UPH Survey
1	Mengelola jadwal	x	v
2	Membuat survei	v	v
3	Melihat data survei	v	v
4	Mengelola data survei	x	v
5	Mendistribusikan laporan hasil survei	x	v
6	Melihat dashboard survei	x	v
7	Melihat dashboard laporan penilaian	x	v
8	Melihat laporan penilaian	x	v

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari pengerjaan tugas akhir ini dihasilkan sistem informasi pengelolaan data SFQ yang dapat berfungsi untuk membantu para *staff* LP2MP dalam mengelola dan melakukan SFQ ke setiap kelas serta dapat memberikan laporan hasil evaluasi berdasarkan dari data survei. dapat disimpulkan beberapa hal mengenai sistem usulan yang telah dibangun, yaitu:

1. Menyediakan sarana untuk memberikan informasi mengenai jadwal kelas visitasi, sehingga divisi LP2MP tidak perlu mencetak dan melihat jadwal dalam bentuk kertas.
2. Memberikan sarana untuk mengelola survei *SFQ* sehingga dapat membantu divisi LP2MP dalam membuat survei tanpa harus menggunakan aplikasi *open source*.
3. Menghemat waktu kerja pihak LP2MP dalam pembuatan survei dan pengelolaan data survei.
4. Dapat memfasilitasi kegiatan survei *online* dan mendapatkan hasil survei secara otomatis.
5. Membantu divisi LP2MP dalam mengelola data, karena hasil survei yang didapatkan diolah secara otomatis oleh sistem, sehingga divisi LP2MP tidak perlu mengolah menggunakan Microsoft Excel, Microsoft Word dan PDF.
6. Membantu divisi LP2MP dalam mendistribusikan hasil survei kepada pihak-pihak yang berkepentingan.

5.2 Saran

Saran yang dapat diusulkan dalam pengembangan sistem informasi pengelolaan data *SFQ* selanjutnya adalah:

1. Mengembangkan sistem dalam bentuk aplikasi yang berbasis *mobile* sehingga lebih mudah diakses dalam menggunakan perangkat *mobile*.

2. Mengembangkan sistem terkait dengan penggabungan seluruh UPH yang ada, di mana sistem dapat diakses di berbagai lokasi.

DAFTAR PUSTAKA

[1] R. K. Rainer and C. G. Cegielski, Introduction to Information Systems, 4th edition., London: John Wiley & Sons, Inc, 2013.

[2] A. Dennis, B. Haley Wixom, and D. Tegarden Systems analysis and design: An object-oriented approach with UML, 5th edition, John Wiley & Sons, Inc, 2015.

[3] B. K. Hamilton and R. Miles, Learning UML 2.0, O'Reilly, 2006.

[4] S. S. Alhir, Learning UML, Sebastopol: O'Reilly Media, Inc, 2003.

[5] "Welcome to CodeIgniter - CodeIgniter 3.1.5 documentation," [Online]. Available: https://www.codeigniter.com/user_guide/general/welcome.html. [Diakses 2018 November 18].

[6] G. Booch, J. Rumbaugh, and I. Jacobson, *The Unified Modeling Language User Guide*, 2nd ed, Addison Wesley Professional, 2005.

[7] "PHP: What is PHP? - Manual," [Online]. Available: http://php.net/manual/en/intro-whatis.php. [Diakses 2018 November 18].

[8] C. Pitt, Pro PHP MVC, Apress, 2012.

[9] "What is RDBMS? - Database Directory," [Online]. Available: http://www.databasedir.com/what-is-rdbms/. [Diakses 25 November 2018].

[10] A. G. Taylor, SQL For Dummies, 5th edition, Wiley Publishing, Inc, 2003.

[11] G. D. Everett and R. McLeod Jr., *Software Testing*: *Testing Across the Entire Software Development Life Cycle*, John Wiley & Sons, Inc, 2007.

[12] "About grocery CRUD - not just another codeigniter grid.," [Online]. Available: https://www.grocerycrud.com/documentation/about. [Diakses 25 November 2018].

[13] Victoria Kis, "Quality Assurance in Tertiary Education: Current Practices in OECD Countries and a Literature Review on Potential Effects," *Organ. Econ. Co-operation Dev. Dir. Educ. Educ. Train. Policy Div,* p. 47, 2005.