



TUGAS AKHIR - KS 141501

**RANCANG BANGUN APLIKASI BERBASIS *WEB* UNTUK
KOMUNITAS BUDIDAYA PERIKANAN DENGAN
MENGUNAKAN MODEL *PROTOTYPE***

***THE DEVELOPMENT OF WEB-BASED APPLICATION
FOR AQUACULTURE COMMUNITY USING
PROTOTYPE MODEL***

DIEGO AFIFUDIN SYAUQI
NRP 5209 100 165

Dosen Pembimbing
Feby Artwodini Muqtadiroh, S.Kom, M.T

DEPARTEMEN SISTEM INFORMASI
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember
Surabaya 2018



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - KS 141501

**RANCANG BANGUN APLIKASI BERBASIS
WEB UNTUK KOMUNITAS BUDIDAYA
PERIKANAN DENGAN MENGGUNAKAN
MODEL *PROTOTYPE***

DIEGO AFIFUDIN SYAUQI
NRP 5209 100 165

Dosen Pembimbing
Feby Artwodini Muqtadiroh, S.Kom, M.T

DEPARTEMEN SISTEM INFORMASI
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember
Surabaya 2018



ITS
Institut
Teknologi
Sepuluh Nopember

FINAL PROJECT - KS 141501

***THE DEVELOPMENT OF WEB-BASED
APPLICATION FOR AQUACULTURE
COMMUNITY USING PROTOTYPE MODEL***

**DIEGO AFIFUDIN SYAUQI
NRP 5209 100 165**

**Academic Supervisor
Feby Artwodini Muqtadiroh, S.Kom, M.T**

**DEPARTEMENT OF INFORMATION SYSTEMS
Faculty of Information Technology and
Communication
Sepuluh Nopember Institute of Technology
Surabaya 2018**

LEMBAR PENGESAHAN

RANCANG BANGUN APLIKASI BERBASIS *WEB* UNTUK KOMUNITAS BUDIDAYA PERIKANAN DENGAN MENGGUNAKAN MODEL *PROTOTYPE*

TUGAS AKHIR

Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Departemen Sistem Informasi
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember

Oleh :

DIEGO AFIFUDIN SYAUQI
NRP. 5209100165

Surabaya, Januari 2018

Pih. KEPALA
DEPARTEMEN SISTEM INFORMASI



Edwin Riksakomara, S.Kom, M.T
NIP. 196907252003121001

LEMBAR PERSETUJUAN

RANCANG BANGUN APLIKASI BERBASIS *WEB* UNTUK KOMUNITAS BUDIDAYA PERIKANAN DENGAN MENGGUNAKAN MODEL *PROTOTYPE*

TUGAS AKHIR

Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Departemen Sistem Informasi
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember

Oleh :

DIEGO AFIFUDIN SYAUQI
NRP. 5209100165

Disetujui Tim Penguji : Tanggal Ujian : 3 Juli 2017
Periode Wisuda : 117

Feby Artwodini Muqtadiroh, S.Kom, M.T (Pembimbing I)

Sholiq, S.T, M.Kom, M.SA (Penguji I)

Dr. Apol Pribadi Subriadi, S.T, M.T (Penguji II)

RANCANG BANGUN APLIKASI BERBASIS WEB UNTUK KOMUNITAS BUDIDAYA PERIKANAN DENGAN MENGGUNAKAN MODEL *PROTOTYPE*

Nama Mahasiswa : Diego Afifudin Syauqi
NRP : 5209100165
Jurusan : Sistem Informasi FTIF – ITS
Dosen Pembimbing I : Feby Artwodini Muqtadiroh,
S.Kom, M.T

ABSTRAK

Salah satu peranan strategis dalam mendukung pertumbuhan perikanan nasional adalah melalui pengembangan sumberdaya manusia di bidang kelautan dan perikanan. Sumber daya manusia yang baik dalam bidang budidaya perikanan termasuk sebagai salah satu indikator keberhasilan dalam usaha budidaya perikanan. Adanya komunitas-komunitas para pembudidaya perikanan merupakan salah satu wadah untuk pengembangan SDM bidang budidaya perikanan. Pertukaran informasi dan berbagi pengalaman dalam komunitas akan sangat membantu para pembudidaya dalam mengembangkan sumber daya manusianya.

Melihat hal tersebut, penelitian ini bertujuan untuk melakukan rancang bangun sebuah aplikasi web yang diharapkan dapat memberikan informasi secara tepat guna kepada para pelaku budidaya perikanan. Aplikasi web yang dirancang akan memiliki empat fitur utama. Fitur berita yang bertujuan untuk memberikan informasi khusus bidang budidaya perikanan. Fitur jual beli, fitur tanya jawab, fitur info pasar yang berisi informasi harga terkini dan trend pasar yang sedang berkembang.

Kata Kunci : aplikasi web, prototype model, SDLC, budaya perikanan

THE DEVELOPMENT OF WEB-BASED APPLICATION FOR AQUACULTURE COMMUNITY USING PROTOTYPE MODEL

Student Name : Diego Afifudin Syauqi
NRP : 5209100165
Department : Sistem Informasi FTIF – ITS
Supervisor I : Feby Artwodini Muqtadiroh, S.Kom, M.T

ABSTRACT

One of the strategic role in supporting the growth of national fishery is through the development of human resources in the field of marine and fisheries. Good human resources in the field of aquaculture is included as one indicator of success in the business of aquaculture. The existence of communities of fish farmers is one of the containers for the development of human resources in the field of aquaculture. The exchange of information and sharing of experiences within the community will greatly assist the farmers in developing their human resources.

Seeing this, this study aims to design a web application that is expected to provide information appropriately to the perpetrators of aquaculture. The designed web app will have four main features. News features that aims to provide specific information in the field of aquaculture. Buying and selling features, forum features, market info features that contain current pricing information and emerging market trends.

Keywords : *web app, prototype model, SDLC, aquaculture*

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Puji syukur yang sebesar-besarnya saya panjatkan kehadirat Allah SWT atas segala limpahan rahmat dan hidayah-Nya sehingga Tugas Akhir ini dapat terselesaikan.

Tak lupa saya juga mengucapkan terima kasih sebanyak-banyaknya kepada berbagai pihak yang telah banyak berkenan memberikan bantuan selama pengerjaan Tugas Akhir ini. Oleh karena itu, saya ingin mengucapkan terima kasih kepada :

- Allah SWT yang senantiasa memberikan limpahan rahmat, karunia dan hidayah-Nya sehingga Tugas Akhir ini dapat terselesaikan.
- Rasullullah Muhammad SAW atas segala syafaatnya dan bimbingannya yang tak pernah terputus. Dan semoga shalawat serta salam tetap terlimpahkan kepada beliau SAW.
- Ayahanda Sumiran dan Ibunda Lailatul Badriyah, yang telah sabar mendidik, mendukung, dan menunggu anaknya menjadi sarjana. Terima kasih yang tak terkira untuk keduanya.
- Bapak Aris Tjahjanto, Bapak Febriliyan Samopa, dan Bapak Ahmad Holil Noor Ali selaku Ketua Jurusan Sistem Informasi ITS selama saya menjalani kuliah di jurusan Sistem Informasi ITS. Terima kasih atas dukungan fasilitas selama ini.
- Ibu Feby Artwodini selaku pembimbing I yang selalu bersedia memberikan waktunya untuk membimbing dan memberikan arahan terhadap masalah yang dihadapi saat pengerjaan Tugas Akhir ini. Terima kasih atas kesabaran dan kepercayaannya selama waktu bimbingan.
- Bapak Sholiq selaku penguji I dan Bapak Apol Pribadi selaku penguji II yang bersedia menguji dan memberikan saran terhadap perbaikan Tugas Akhir ini.

- Bapak Apol Pribadi selaku dosen wali selama saya menjalani kuliah di jurusan Sistem Informasi ITS.
- Teman-teman di jurusan Sistem Informasi ITS dan teman-teman angkatan AE9IS yang menemani saya selama menjalani masa studi di jurusan Sistem Informasi.
- Seluruh penghuni warung GK51 dan semua entitas di dalamnya yang telah menghibur dan menemani saya dalam suka dan duka selama pengerjaan Tugas Akhir ini.
- Sahabat-sahabat seperjalanan yang dengan “diam”nya pun memberikan inspirasi dan solusi dalam setiap permasalahan.
- Dan kepada saudara, teman, dan orang-orang yang tidak dapat saya sebutkan satu-persatu. Terima kasih...

Saya menyadari bahwa Tugas Akhir ini masih banyak memiliki kekurangan dan masih jauh dari kata sempurna. Untuk itu saya mengharapkan saran terhadap Tugas Akhir ini yang bersifat membangun guna perbaikan di masa mendatang. Akhir kata, saya berharap tugas akhir ini dapat bermanfaat bagi kita semua.

Wassalam...

Surabaya, 07 Juli 2017

Penulis

DAFTAR ISI

ABSTRAK	v
<i>ABSTRACT</i>	vii
KATA PENGANTAR	ix
DAFTAR ISI	xi
DAFTAR GAMBAR	xiv
DAFTAR TABEL	xvi
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan Tugas Akhir	3
1.5 Manfaat Tugas Akhir	3
1.6 Target Luaran	3
1.7 Sistematika Penulisan	3
BAB II TINJAUAN PUSTAKA	7
2.1 Studi Sebelumnya	7
2.2 Dasar Teori	8
2.2.1 Budidaya Perikanan	8
2.2.2 Aplikasi <i>Web</i>	8
2.2.3 Portal <i>Web</i>	9
2.2.4 <i>Prototype Model</i>	9
2.2.5 <i>Black-Box Testing</i>	11
BAB III METODOLOGI PENELITIAN	13
3.1 Tahapan Pelaksanaan Tugas Akhir	13
3.2 Uraian Metodologi	14
3.2.1 Studi Literatur	14
3.2.2 Identifikasi Kebutuhan Perangkat Lunak	14
3.2.3 Analisis <i>Use Case</i>	14
3.2.4 Implementasi dan Pengkodean	15
3.2.5 Pengujian dan Evaluasi	15
3.2.6 Penyusunan Laporan Tugas Akhir	15
3.3 Rangkuman Metodologi	15
BAB IV ANALISIS DAN PERANCANGAN	17
4.1 Analisa Kebutuhan Sistem	17

4.1.1	Kebutuhan Fungsional Sistem	19
4.1.2	Kebutuhan non-Fungsional.....	20
4.1.3	Penentuan Aktor.....	21
4.1.4	Daftar <i>Use Case</i>	21
4.2	Perancangan Aplikasi	22
4.2.1	<i>Use Case Diagram</i>	22
4.2.2	Diagram Aktifitas.....	24
4.3	Analisis Kebutuhan Iterasi ke-2.....	30
4.3.1	Kebutuhan Fungsional Sistem Iterasi ke-2	30
4.3.2	Penentuan Aktor Iterasi ke-2	31
4.3.3	Daftar <i>Use Case</i> Iterasi ke-2.....	32
4.4	Perancangan Aplikasi Iterasi ke-2	32
4.4.1	<i>Use Case Diagram</i> Iterasi ke-2.....	33
4.4.2	Diagram Aktivitas Iterasi ke-2.....	34
BAB V	IMPLEMENTASI	37
5.1	Lingkungan Implementasi	37
5.1.1	Implementasi <i>Hardware</i>	37
5.1.2	Implementasi <i>Software</i>	37
5.2	Pembuatan <i>Database</i>	38
5.3	Struktur Direktori.....	38
5.4	Implementasi Fungsi dan Pengkodean	39
5.4.1	Implementasi <i>Login</i>	39
5.4.2	Implementasi Mengelola Berita.....	41
5.4.3	Implementasi Menghapus Topik Forum.....	45
5.4.4	Implementasi Menghapus Data Member	46
5.4.5	Implementasi Mengelola Item Jual Beli	47
5.4.6	Implementasi Mengelola Harga Trend Pasar.....	50
5.4.7	Implementasi Mendaftar Menjadi Member	51
5.4.8	Implementasi Menambahkan Topik Baru.....	53
5.4.9	Implementasi Menambahkan Komentar	55
BAB VI	HASIL PENGUJIAN DAN PEMBAHASAN.....	59
6.1	Pengujian	59
6.2	Hasil Uji Coba	60
6.3	Pembahasan Uji Coba.....	60
BAB VII	KESIMPULAN DAN SARAN	61
7.1	Kesimpulan.....	61
7.2	Saran	62

DAFTAR PUSTAKA	63
LAMPIRAN A KODE PEMBUATAN APLIKASI.....	A-1
LAMPIRAN B SKENARIO KEBUTUHAN FUNGSI.....	B-1
BIODATA PENULIS	bio

DAFTAR GAMBAR

Gambar 2.1 Alur <i>Prototype Model</i>	10
Gambar 2.2 Proses <i>Blackbox Testing</i>	12
Gambar 3.1 Tahapan Pelaksanaan Tugas Akhir	13
Gambar 4.1 <i>Use Case Diagram</i> Administrator	23
Gambar 4.2 <i>Use Case Diagram</i> Member	24
Gambar 4.3 Diagram Aktivitas Mengelola Berita	26
Gambar 4.4 Diagram Aktivitas Menghapus Topik	27
Gambar 4.5 Diagram Aktivitas Menghapus Data Member	28
Gambar 4.6 Diagram Aktivitas Mengelola Item Jual Beli	29
Gambar 4.7 Diagram Aktivitas Mengelola Harga dan Trend	30
Gambar 4.8 <i>Use Case Diagram</i> Iterasi ke-2	33
Gambar 4.9 Diagram Aktivitas Mendaftar Menjadi Member	35
Gambar 4.10 Diagram Aktivitas Menambahkan Topik	36
Gambar 5.1 Desain <i>Database</i>	38
Gambar 5.2 Struktur Direktori	39
Gambar 5.3 Halaman <i>Login Server</i>	40
Gambar 5.4 Halaman Awal Aplikasi <i>Server</i>	40
Gambar 5.5 Kode Proses <i>Login Server</i>	41
Gambar 5.6 Form Tambah Berita Baru	42
Gambar 5.7 Kode Proses Menambahkan Berita	44
Gambar 5.8 Halaman Kelola Berita	44
Gambar 5.9 Kode Proses Menghapus Berita	45
Gambar 5.10 Halaman Kelola Forum	46
Gambar 5.11 Kode Proses Menghapus Topik	46
Gambar 5.12 Halaman Data Member	47
Gambar 5.13 Kode Proses Menghapus Data Member	47
Gambar 5.14 Form Menambahkan Item Baru	48
Gambar 5.15 Kode Proses Menambahkan Item Baru	48
Gambar 5.16 Halaman Kelola Item Jual Beli	49
Gambar 5.17 Kode Proses Mengedit Item Jual Beli	49
Gambar 5.18 Kode Proses Menghapus Item	50
Gambar 5.19 Form Tambah Trend Pasar	50
Gambar 5.20 Kode Proses Menambahkan Trend Pasar	51
Gambar 5.21 Form Daftar Menjadi Member	51

Gambar 5.22 Kode Proses Mendaftar Menjadi Member	53
Gambar 5.23 Halaman Tambah Topik Tanya Jawab	54
Gambar 5.24 Kode Proses Menambahkan Topik Baru	55
Gambar 5.25 Halaman Komentar Tanya Jawab.....	55
Gambar 5.26 Kode Proses Menambahkan Komentar	57

DAFTAR TABEL

Tabel 3.1 Rangkuman Metodologi	15
Tabel 4.1 Analisis <i>User Story</i>	17
Tabel 4.2 Analisis Kebutuhan Pengguna	18
Tabel 4.3 Analisa Kebutuhan Sistem	19
Tabel 4.4 Kebutuhan Fungsional	20
Tabel 4.5 Kebutuhan non-Fungsional	20
Tabel 4.6 Aktor dan Kepentingan	21
Tabel 4.7 Daftar <i>Use Case</i>	21
Tabel 4.8 Diagram Aktivitas	25
Tabel 4.9 Kebutuhan Fungsional Iterasi ke-2	31
Tabel 4.10 Penentuan Aktor Iterasi ke-2	31
Tabel 4.11 Daftar Use Case Iterasi ke-2	32
Tabel 4.12 Diagram Aktivitas Iterasi ke-2	34
Tabel 6.1 Daftar <i>Use Case Test</i>	59
Tabel 6.2 Hasil Uji Coba	60

BAB I

PENDAHULUAN

Pada bab ini latar belakang, rumusan masalah, dan batasan permasalahan yang menjadi pokok bahasan yang akan dikerjakan dalam penelitian. Pada bab ini juga dijelaskan tujuan dan manfaat dilakukannya penelitian terhadap perkembangan dan solusi dari permasalahan yang diangkat, serta metodologi dan sistematika penulisan yang digunakan dalam penulisan Tugas Akhir ini.

1.1 Latar Belakang

Teknologi informasi berkembang dengan cepat dan telah menjangkau banyak bidang. Baik itu bidang pendidikan, pertanian, manajemen, dan lain sebagainya. Dengan didukung infrastruktur dan perangkat keras yang semakin berkembang dan dapat dijangkau oleh masyarakat luas, saat ini hampir setiap individu dapat menikmati layanan internet melalui berbagai perangkat *desktop* maupun *mobile*. Para pengembang aplikasi pun seakan berlomba-lomba untuk menciptakan terobosan baru yang memudahkan para pengguna aplikasi khususnya aplikasi *web*.

Berhubungan dengan hal tersebut diatas, penelitian ini melihat upaya pemerintah dalam memajukan usaha bidang perikanan khususnya perikanan budidaya. Melalui peraturan-peraturan pemerintah yang di keluarkan oleh Kementerian bidang Kelautan dan Perikanan, diketahui bahwa sektor perikanan budidaya menjadi salah satu sektor yang berperan penting dalam menyokong pertumbuhan perikanan nasional. Salah satu peranan strategis dalam mendukung pertumbuhan perikanan nasional tersebut adalah melalui pengembangan sumber daya manusia (SDM) di bidang kelautan dan perikanan [1].

Salah satu cara mengembangkan sumber daya manusia di bidang budidaya perikanan adalah melalui komunitas. Adanya komunitas-komunitas para pembudidaya perikanan merupakan salah satu wadah untuk pengembangan sumber daya manusia bidang budidaya perikanan yang efektif. Pertukaran informasi dan berbagi pengalaman dalam komunitas akan sangat membantu para pembudidaya dalam mengembangkan sumber daya manusianya [2].

Melihat hal tersebut, penelitian ini bertujuan untuk melakukan rancang bangun sebuah aplikasi *web* yang diharapkan dapat memberikan informasi secara tepat guna kepada para pelaku budidaya perikanan sebagai media pendukung komunitas pembudidaya perikanan. Aplikasi *web* yang dirancang akan memiliki empat fitur utama. Fitur berita yang bertujuan untuk memberikan informasi khusus bidang budidaya perikanan. Fitur jual beli, fitur tanya jawab, dan fitur info pasar yang berisi informasi harga perikanan terkini dan trend pasar yang sedang berkembang.

Hasil dari penelitian ini adalah sebuah *prototype* aplikasi berbasis *web* yang dapat diakses melalui *desktop* dan *mobile*.

1.2 Rumusan Masalah

Berdasarkan uraian latar belakang, maka rumusan permasalahan yang menjadi fokus dan akan diselesaikan dalam Tugas Akhir ini adalah bagaimana hasil rancang bangun aplikasi berbasis *web* untuk para pembudidaya perikanan dengan menggunakan metode pengembangan perangkat lunak *Prototype Model*.

1.3 Batasan Masalah

Pengerjaan Tugas Akhir ini memiliki beberapa batasan masalah sebagai berikut:

1. Rancang bangun aplikasi berdasarkan penggalian kebutuhan dari pelaku usaha budidaya perikanan.
2. Aplikasi berbasis *web* hanya diakses menggunakan jaringan internet.
3. Aplikasi dibangun menggunakan bahasa pemrograman *PHP* dan *database MySQL*.

1.4 Tujuan Tugas Akhir

Tujuan pembuatan Tugas Akhir ini adalah membuat aplikasi berbasis *web* yang membantu para pembudidaya perikanan dalam mendapatkan informasi.

1.5 Manfaat Tugas Akhir

Melalui Tugas Akhir ini diharapkan dapat memberi manfaat yaitu:

1. Membantu para pembudidaya perikanan mendapatkan informasi dan berita seputar budidaya perikanan.
2. Membantu memberikan wadah bagi para pembudidaya perikanan dalam berbagi pengalaman.
3. Mengetahui harga dan trend pasar komoditas perikanan budidaya terkini.

1.6 Target Luaran

Target luaran dalam pengerjaan Tugas Akhir ini adalah sebagai berikut:

1. Aplikasi berbasis *web* bagi pelaku budidaya perikanan.
2. Dokumen Laporan Tugas Akhir.

1.7 Sistematika Penulisan

Sistematika penulisan buku Tugas Akhir dibagi menjadi tujuh bab sebagai berikut:

BAB I PENDAHULUAN

Bab pendahuluan ini akan menjelaskan Latar Belakang Masalah, Perumusan Masalah, Tujuan Tugas Akhir, Batasan Masalah, Manfaat Tugas Akhir dan Relevansi.

BAB II TINJAUAN PUSTAKA

Bab tinjauan pustaka ini akan menjelaskan studi sebelumnya dari penelitian ini dan dasar teori dari Tugas Akhir ini.

BAB III METODOLOGI PENELITIAN

Bab Metodologi ini akan menjelaskan mengenai tahapan pelaksanaan dari Tugas Akhir ini dan kebutuhan fungsional serta jadwal kegiatan dari Tugas Akhir. Rangkaian pengerjaan Tugas Akhir ini mengacu pada model pengembangan perangkat lunak *Prototype Model*.

BAB IV ANALISIS DAN PERANCANGAN

Pada bab ini akan dijelaskan mengenai rancangan pengembangan aplikasi. Pembuatan desain aplikasi *web* berpedoman pada tahap-tahap pengembangan perangkat lunak dengan metode *Prototype Model*.

BAB V IMPLEMENTASI

Bab implementasi ini menjelaskan bagaimana tahap-tahap penelitian diimplementasikan, termasuk hambatan dan rintangan yang dihadapi selama proses penelitian berjalan. Bab ini juga menjelaskan tentang cara melakukan penelitian secara teknis agar dapat dilakukan kembali dengan mudah.

BAB VI HASIL PENGUJIAN DAN PEMBAHASAN

Bagian ini berisi kesimpulan dari seluruh proses pengerjaan Tugas Akhir beserta saran yang diajukan untuk proses pengembangan selanjutnya.

BAB VII KESIMPULAN DAN SARAN

Bagian ini berisi kesimpulan dari seluruh proses pengerjaan Tugas Akhir beserta saran yang diajukan untuk proses pengembangan selanjutnya.

Halaman ini sengaja dikosongkan

BAB II

TINJAUAN PUSTAKA

Bab ini akan menjelaskan mengenai referensi yang berkaitan dengan penelitian pada Tugas Akhir. Terdiri dari penjelasan mengenai studi sebelumnya dan dasar teori pendukung.

2.1 Studi Sebelumnya

Pada penelitian sebelumnya terdapat beberapa penelitian yang melakukan rancang bangun aplikasi berbasis *web* dengan metode pengembangan perangkat lunak menggunakan *Prototype Model*. Penelitian dari Tommy Prasetyo yang berjudul “*Rancang Bangun Aplikasi Website Peneliti Indonesia*” merupakan penelitian yang menjadikan aplikasi *web* sebagai salah satu solusi untuk mengatasi bagaimana cara memudahkan pencarian rekanan sesama peneliti di Indonesia. Penelitian ini menggunakan metode pengembangan perangkat lunak *Prototype Model*. Model *Prototype* digunakan dalam penelitian ini karena model ini baik digunakan pada saat pengembang aplikasi merasa tidak yakin dengan tingkat efisiensi dari sebuah algoritma, tingkat adaptasi dari sistem operasi, atau tampilan *GUI (Graphical User Interface)* yang akan dilihat oleh pengguna [3]. Hasil yang diperoleh oleh penelitian yang dilakukan oleh peneliti ini adalah dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) dan diagram-diagram *UML* sebagai pelengkap untuk mempermudah dalam memodelkan sistem.

Penelitian yang dijadikan tinjauan pustaka selanjutnya adalah penelitian dari Faisal Setia Putra dengan judul “*Layanan Berbasis Lokasi untuk Menemukan Koki Rumahan yang Terdekat dan Tersedia*”. Penelitian tersebut bertujuan untuk mempertemukan antara individu yang tidak bisa memasak dengan koki rumahan. Individu akan menyediakan bahan baku dan tempat bagi koki rumahan. Sebagai gantinya, koki

rumahan akan menyediakan jasa mengolah bahan mentah menjadi makanan jadi. Aplikasi pencari koki rumahan adalah sebuah platform berbasis *web* yang terintegrasi dengan *Google Maps* yang dapat membantu dalam mencari koki rumahan di daerah sekitar yang dapat membuatkan masakan sesuai dengan keinginan dan juga menyehatkan [4]. Aplikasi tersebut menggunakan metode pengembangan perangkat lunak *Prototype Model*.

Dari kedua penelitian diatas, akan dijadikan refrensi sekaligus sebagai tinjauan pustaka dalam penelitian kali ini. Karena dari kedua penelitian tersebut diatas menjelaskan setiap langkah-langkah dalam melakukan rancang bangun dengan rinci dan terstruktur sesuai dengan motode pengembangan perangkat lunak *Prototype Model* yang juga akan digunakan dalam penelitian ini.

2.2 Dasar Teori

Pada bagian ini, akan dijelaskan mengenai teori-teori yang digunakan untuk mendukung pengerjaan Tugas Akhir.

2.2.1 Budidaya Perikanan

Budidaya perikanan adalah salah satu bentuk budidaya perairan yang khusus membudidayakan ikan di tangki atau ruang tertutup, biasanya untuk menghasilkan bahan pangan, ikan hias, dan rekreasi (pemancingan). Ikan yang paling banyak dibudidayakan adalah ikan mas, salmon, lele, dan tilapia (sejenis ikan nila). Masih banyaknya isu strategis dan permasalahan jelas menunjukkan bahwa sektor perikanan budidaya di Indonesia masih mempunyai banyak kendala dalam pengelolaan dan pengembangannya [5].

2.2.2 Aplikasi Web

Dalam rekayasa perangkat lunak, suatu aplikasi *web* (*web application* atau sering disingkat *webapp*) adalah suatu aplikasi yang diakses menggunakan penjelajah *web* melalui

suatu jaringan seperti Internet atau intranet. Ia juga merupakan suatu aplikasi perangkat lunak komputer yang dikodekan dalam bahasa yang didukung penjelajah *web* (seperti *ASP*, *HTML*, *Java*, *Javascript*, *PHP*, *Python*, *Ruby*, dll) dan bergantung pada penjelajah tersebut untuk menampilkan aplikasi. Aplikasi *web* menjadi populer karena kemudahan tersedianya aplikasi klien untuk mengaksesnya, penjelajah *web*, yang kadang disebut sebagai suatu *thin client* (klien tipis). Kemampuan untuk memperbarui dan memelihara aplikasi *web* tanpa harus mendistribusikan dan menginstalasi perangkat lunak pada kemungkinan ribuan komputer klien merupakan alasan kunci popularitasnya. Aplikasi *web* yang umum misalnya *webmail*, toko ritel, lelang online, *wiki*, papan diskusi, *weblog* [6].

2.2.3 Portal *Web*

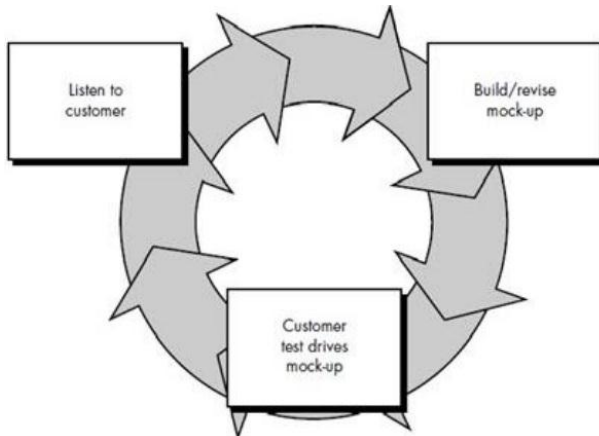
Portal *web* adalah situs *web* yang menyediakan kemampuan tertentu yang dibuat sedemikian rupa dan mencoba menuruti selera para pengunjungnya. Kemampuan portal yang lebih spesifik adalah penyediaan kandungan informasi yang dapat diakses menggunakan beragam perangkat, misalnya komputer pribadi (*desktop*), komputer jinjing (*notebook*), dan *smartphone*.

Ketika disebut sebagai Portal Komunitas, sebuah situs-*web* biasanya sengaja didesain dan dibangun berdasarkan pertimbangan pada selera (kolektif) suatu komunitas. Oleh sebab itu fasilitas-fasilitas yang disediakan pada Portal Komunitas cenderung berupaya memenuhi kebutuhan komunitas tersebut [7].

2.2.4 *Prototype Model*

Prototype Model adalah salah satu metode pengembangan perangkat lunak yang banyak digunakan. Dengan menggunakan metode *prototyping* ini, pengembang aplikasi dan pelanggan dapat saling berinteraksi selama proses pembuatan sistem. Seringkali seorang pelanggan hanya

mendefinisikan secara umum apa saja yang dibutuhkan tanpa menyertakan detail *input*, proses, maupun *output*. Pada sisi yang lain, tim pengembang (*developer*) tidak yakin terhadap efisiensi dari algoritma yang digunakan, tingkat adaptasi terhadap sistem operasi ataupun rancangan tampilan antar muka [8]. Ketika terjadi situasi seperti ini, model *prototyping* sangat membantu dalam proses pembangunan sistem.



Gambar 2.1 Alur *Prototype Model*

Adapun tahapan-tahapan dalam pengembangan dengan menggunakan model *Prototype* yaitu sebagai berikut:

1. Mendengarkan Pelanggan
 Pada tahap ini dilakukan pengumpulan kebutuhan dari sistem dengan cara melakukan wawancara dengan pelanggan. Dalam membuat suatu sistem yang sesuai dengan kebutuhan, maka harus diketahui terlebih dahulu bagaimana kondisi sistem yang sedang berjalan saat ini untuk kemudian mengetahui masalah yang terjadi [8].
2. Merancang dan Membuat *Prototype*
 Pada tahapan ini, dilakukan perancangan dan pembuatan *Prototype* dari sistem. *Prototype* yang dibuat disesuaikan dengan kebutuhan sistem yang

telah didefinisikan sebelumnya dari wawancara dengan pengguna [8].

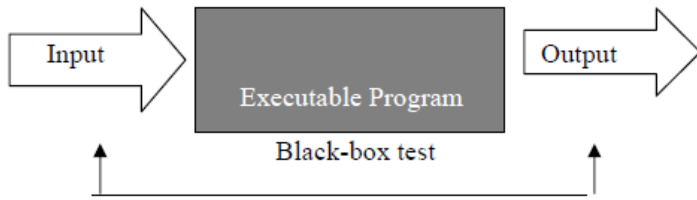
3. Uji Coba

Setelah *Prototype* jadi, maka dilakukan uji coba oleh pelanggan maupun pengguna. Kemudian dilakukan evaluasi kekurangan-kekurangan maupun masukan dari kebutuhan pelanggan. Lalu dilakukan pengembangan kembali dengan melalui tahapan dari awal yaitu mengecek kebutuhan sistem dari kebutuhan pengguna untuk memperbaiki *Prototype* yang telah ada [8].

2.2.5 *Black-Box Testing*

Black-box testing adalah metode pengujian perangkat lunak yang digunakan untuk tes fungsionalitas dari aplikasi. Pengetahuan khusus dari kode aplikasi / struktur internal dan pengetahuan pemrograman pada umumnya tidak diperlukan. *Test case* dibangun di sekitar spesifikasi dan persyaratan, yakni, apa yang seharusnya dilakukan aplikasi. Menggunakan deskripsi eksternal perangkat lunak, termasuk spesifikasi, persyaratan, dan desain untuk menurunkan *test case*. Tes ini dapat menjadi fungsional atau non-fungsional, meskipun biasanya fungsional. Perancang uji memilih input yang valid dan tidak valid dan menentukan *output* yang benar.

Metode tes ini dapat diterapkan pada semua tingkat pengujian perangkat lunak: unit, integrasi, fungsional, sistem dan penerimaan. Ini biasanya tidak hanya terjadi pada semua pengujian pada tingkat yang lebih tinggi, tetapi juga bisa mendominasi unit testing juga [9].



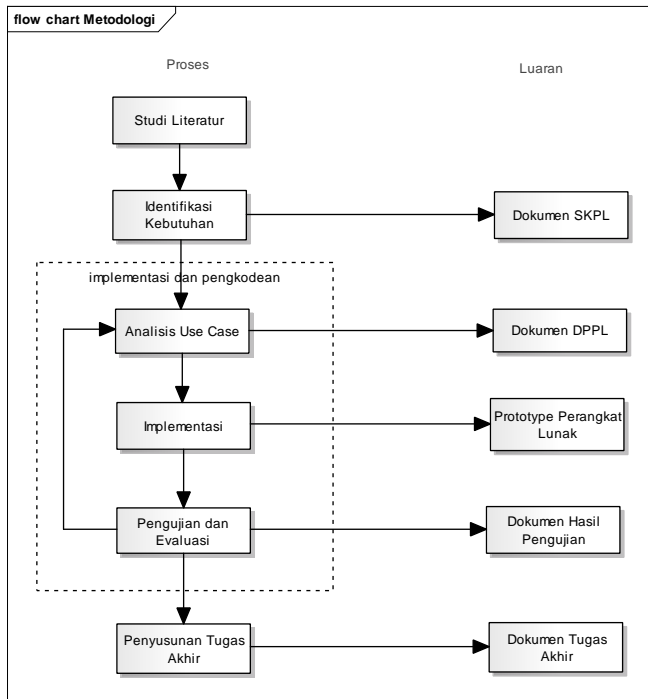
Gambar 2.2 Proses *Blackbox Testing*

BAB III METODOLOGI PENELITIAN

Bab ini berisi penjelasan mengenai tahapan-tahapan yang dilakukan dalam penelitian ini. Setiap luaran dari proses yang dilakukan akan menjadi masukan untuk proses berikutnya.

3.1 Tahapan Pelaksanaan Tugas Akhir

Terdapat beberapa tahapan-tahapan pada pelaksanaan Tugas Akhir ini. Metodologi dari penelitian ini digambarkan seperti pada Gambar 3.1.



Gambar 3.1 Tahapan Pelaksanaan Tugas Akhir

3.2 Uraian Metodologi

Berikut ini merupakan uraian dari beberapa tahapan-tahapan pada metodologi penelitian diantaranya yaitu:

3.2.1 Studi Literatur

Studi literatur merupakan upaya yang dilakukan untuk mendapatkan referensi mengenai informasi dan data yang diperlukan dalam penelitian ini. Tujuan dari tahap ini adalah agar penelitian ini memiliki dasar teori dan teknik yang jelas untuk digunakan dalam pemecahan masalah yang dibahas. Adapun literatur yang digunakan yaitu buku, dokumentasi, jurnal ilmiah, e-book, informasi dari pencarian di internet, dan sumber lain yang terkait. Luaran dari proses ini adalah metode pemodelan aplikasi yang akan menjadi masukan dalam analisis selanjutnya.

3.2.2 Identifikasi Kebutuhan Perangkat Lunak

Tahap kedua adalah tahap identifikasi kebutuhan perangkat lunak. Kebutuhan perangkat lunak didapatkan dari studi literatur, wawancara, dan perbandingan dengan aplikasi yang sejenis. Pada tahapan ini seluruh kebutuhan perangkat lunak yang didapatkan akan dipetakan menjadi kebutuhan fungsional dan non-fungsional ke dalam perangkat lunak berbasis *web*.

3.2.3 Analisis *Use Case*

Pada tahap ini dilakukan analisa terhadap *use case* yang didapatkan dari analisa kebutuhan perangkat lunak pada tahapan sebelumnya. Kemudian data yang didapatkan dari analisa kebutuhan diubah kedalam bentuk kebutuhan pengguna yang dibuat menjadi *use case* aplikasi. *Use case* aplikasi digunakan untuk mendapatkan desain model yang efektif dan sesuai dengan kebutuhan dan permasalahan yang akan diselesaikan. Pada tahapan ini menggunakan metode *software development life cycle (SDLC)* dengan menggunakan pendekatan *Prototype Model*.

3.2.4 Implementasi dan Pengkodean

Pada tahap ini dilakukan pengkodean yang didasarkan pada desain aplikasi dari tahap sebelumnya. Pada tahapan ini pembuatan *database* menggunakan *MySQL*, sedangkan pembuatan aplikasi *web* dilakukan dengan menggunakan bahasa pemrograman *PHP*. Untuk tampilan aplikasi menggunakan *HTML*, *CSS*, dan *javascript*. Hasil dari tahapan ini berupa *Prototype* aplikasi berbasis *web*.

3.2.5 Pengujian dan Evaluasi

Pada tahap ini dilakukan pengujian terhadap aplikasi yang telah di buat. Pengujian dilakukan untuk menguji fungsional aplikasi. Apabila terdapat fungsi aplikasi yang kurang atau tidak bekerja sesuai kebutuhan yang ada maka harus dilakukan pengkodean ulang untuk menambah atau memperbaiki fungsi aplikasi.

3.2.6 Penyusunan Laporan Tugas Akhir

Pada tahap ini dilakukan dokumentasi proses-proses yang telah dilakukan dalam penelitian ini. Dokumentasi dikemas dalam bentuk buku Tugas Akhir. Bentuk penulisan buku Tugas Akhir disesuaikan dengan format buku panduan Tugas Akhir.

3.3 Rangkuman Metodologi

Rangkuman metodologi berisi hal-hal yang dilakukan pada penelitian ini, diawali dengan rangkaian aktivitas, tujuan, masukan, luaran dan metode yang digunakan seperti yang terdapat pada Tabel 3.1 di bawah.

Tabel 3.1 Rangkuman Metodologi

Aktivitas	Tujuan	Masukkan	Luaran	Metode
Studi Literatur	Mengetahui refrensi dalam pengerjaan penelitian	Buku, jurnal, dan bahan bacaan	Dasar teori	Studi pustaka

		tentang budidaya perikanan		
Identifikasi Kebutuhan	Mengidentifikasi kebutuhan aplikasi	Hasil wawancara	Analisis kebutuhan user	wawancara
Analisis <i>Use Case</i>	Mengetahui hasil analisis dan perancangan untuk aplikasi	Hasil Analisis kebutuhan user	Dokumen spesifikasi kebutuhan perangkat lunak	Pengembangan perangkat lunak dengan metode <i>Prototype Model</i>
Implementasi dan Pengkodean	Membuat aplikasi berbasis <i>web</i> berdasarkan hasil perancangan	Hasil analisis kebutuhan perangkat lunak	Aplikasi yang sesuai dengan kebutuhan user	Pengkodean aplikasi <i>web</i>
Pengujian dan Evaluasi	Menguji kesesuaian aplikasi dengan kebutuhan	Pengujian aplikasi	Dokumen pengujian	<i>Blackbox testing</i>
Penyusunan Laporan Tugas Akhir	Menyusun keseluruhan penelitian Tugas Akhir	Seluruh data penelitian	Buku Tugas Akhir	Penyusunan data dan Aplikasi

BAB IV ANALISIS DAN PERANCANGAN

Bab ini menjelaskan mengenai tahapan-tahapan dalam melakukan analisa dan perancangan aplikasi. Tahapan-tahapan tersebut meliputi mendengarkan pelanggan, membangun dan memperbaiki *prototype* hingga pengujian *prototype*. Pengembangan aplikasi dilakukan menggunakan pendekatan *Prototyping Model*, oleh karena itu, proses pengembangan akan dibagi menjadi beberapa kali iterasi. Tiap iterasi akan menggambarkan adanya perubahan kebutuhan.

4.1 Analisa Kebutuhan Sistem

Tahap ini dimulai dengan pengumpulan kebutuhan. Pada tahap ini dilakukan analisis kebutuhan dan identifikasi terhadap segala kebutuhan sistem. Kebutuhan tersebut didapatkan dari beberapa wawancara langsung dengan dua orang pembudidaya ikan. Wawancara pertama dilakukan dengan bapak Haris Sudono yang telah lama menjadi petani tambak dan wawancara kedua dilakukan dengan bapak Zainul yang baru beberapa bulan mencoba usaha budidaya ikan lele. Dari hasil wawancara yang telah dilakukan dirangkum menjadi sebuah tabel seperti yang terdapat pada Tabel 4.1 dibawah.

Tabel 4.1 Analisis *User Story*

<i>User Story</i>			
ID US	ID Pertanyaan	Aktor	<i>Story</i>
US01	QA1	Petani Tambak	Petani tambak ingin mengetahui harga pasar
US02	QA1	Petani Tambak	Petani tambak ingin mengetahui ikan yang sedang dicari masyarakat
US03	QA2	Pembudidaya ikan lele	Pembudidaya ikan ingin mengetahui informasi yang dapat membantu pengembangan

			usahanya
US04	QA2	Pembudidaya ikan lele	Pembudidaya ikan ingin kemudahan akses bertanya kepada pembudidaya ikan yang lebih berpengalaman
US05	QA2	Pembudidaya ikan lele	Pembudidaya ikan ingin menjual hasil budidayanya dengan mudah
US06	QA2	Pembudidaya ikan lele	Pembudidaya ingin mencari harga benih dan pakan yang murah
US07	QA2	Pembudidaya ikan lele	Pembudidaya ingin mengetahui harga pasar yang berlangsung
US08	QA2	Pembudidaya ikan lele	Pembudidaya ingin mengetahui ikan apa yang stabil di pasaran

Dari hasil wawancara yang dilakukan, didapatkan analisis terhadap kebutuhan pengguna dari aplikasi yang akan dikembangkan. Kebutuhan pengguna awal dari aplikasi didapatkan berdasarkan hasil dari pengumpulan data yang telah dilakukan dan dijelaskan seperti yang terdapat pada Tabel 4.2 dibawah.

Tabel 4.2 Analisis Kebutuhan Pengguna

Kebutuhan Pengguna	
ID User Need	Deskripsi
UN01	Pengguna dapat melihat berita terkini tentang budidaya perikanan
UN02	Pengguna dapat melihat berita terpopuler dalam bidang budidaya perikanan
UN03	Pengguna dapat menjual hasil budidaya perikanan
UN04	Pengguna dapat mencari benih, pakan, dan hasil budidaya perikanan
UN05	Pengguna dapat bertukar informasi antar sesama pengguna aplikasi
UN06	Pengguna dapat memberikan informasi kepada para pembudidaya perikanan
UN07	Pengguna dapat melihat trend pasar yang sedang terjadi

Dari hasil analisis *user story* dan kebutuhan pengguna, maka dapat dirangkum berdasarkan kebutuhan seperti yang terdapat pada Tabel 4.3 dibawah.

Tabel 4.3 Analisa Kebutuhan Sistem

Kebutuhan Sistem		
ID User Story	Kebutuhan	ID User Need
US01	Sistem harus dapat menyediakan informasi harga pasar	UN07
US02	Sistem harus dapat menyediakan informasi trend pasar	UN07
US03	Sistem harus dapat menyediakan informasi tentang budidaya perikanan	UN01, UN03, UN05
US04	Sistem harus dapat menyediakan tempat untuk melakukan tanya jawab	UN05
US05	Sistem harus dapat menyediakan tempat untuk memasarkan produk	UN03
US06	Sistem harus dapat menyediakan tempat untuk memasarkan produk	UN04
US07	Sistem harus dapat menyediakan informasi harga pasar	UN07
US08	Sistem harus dapat menyediakan informasi trend pasar	UN07

4.1.1 Kebutuhan Fungsional Sistem

Sistem yang menyediakan berita dan informasi telah banyak beredar di dunia maya, dan para pengguna telah dimudahkan dalam pencarian informasi. Namun informasi tentang budidaya perikanan belum banyak yang terorganisir dalam satu wadah aplikasi. Dan belum banyak dijumpai aplikasi yang menyediakan fitur-fitur yang lengkap seperti yang diinginkan oleh pengguna khususnya para pembudidaya perikanan. Dari analisis kebutuhan pengguna yang telah dilakukan, penulis mendapatkan kebutuhan fungsional seperti yang terlampir pada Tabel 4.4 dibawah.

Tabel 4.4 Kebutuhan Fungsional

Kebutuhan Fungsional		
ID KF	Deskripsi KF	Refrensi
KF01	Sistem dapat menyediakan informasi tentang budidaya perikanan	US03
KF02	Sistem dapat menyediakan tempat untuk melakukan tanya jawab	US04
KF03	Sistem dapat menyediakan tempat untuk memasarkan produk	US05, US06
KF04	Sistem dapat menyediakan informasi harga pasar	US01, US07
KF05	Sistem dapat menyediakan informasi trend pasar	US02, US08

Kebutuhan fungsional adalah sebuah tuntutan kepada pengembang agar sistem yang dikembangkan dapat memenuhi kebutuhan dari pengguna aplikasi. Setelah mendapatkan kebutuhan fungsional, maka dibuatlah aplikasi sesuai dengan kebutuhan tersebut.

4.1.2 Kebutuhan non-Fungsional

Pada tahap ini dilakukan pengelompokan kebutuhan berdasarkan area non-fungsional yang berhubungan dengan perangkat lunak. Kebutuhan non-fungsional dari aplikasi dapat dilihat pada Tabel 4.5 dibawah.

Tabel 4.5 Kebutuhan non-Fungsional

Kebutuhan nonFungsional	
ID KnF	Deskripsi KnF
KnF01	Sistem dapat diakses 24 jam/7 hari
KnF02	Aplikasi dapat diakses di berbagai platform

4.1.3 Penentuan Aktor

Berdasarkan hasil wawancara yang telah dilakukan, dapat diketahui aktor yang akan berperan dalam aplikasi. Tiap-tiap aktor memiliki kepentingan seperti pada Tabel 4.6 dibawah.

Tabel 4.6 Aktor dan Kepentingan

Aktor	
Aktor	Kepentingan
Administrator	Mengelola seluruh kegiatan aplikasi
Member	Semua entitas pengguna aplikasi yang berperan sebagai member

4.1.4 Daftar Use Case

Berdasarkan analisis kebutuhan yang telah dilakukan, fungsional dari perangkat lunak yang dibuat dapat diterjemahkan kedalam bentuk *use case*. Dari hasil analisa *user story* yang telah dilakukan didapatkan daftar kebutuhan fungsional, dan dari analisis kebutuhan fungsional didapatkan daftar *use case*. Daftar *use case* berdasarkan kebutuhan fungsional dapat dilihat pada Tabel 4.7 dibawah.

Tabel 4.7 Daftar Use Case

Use Case Description		
ID KF	ID UC	Deskripsi Use Case
KF01	UC01	Admin dapat mengelola berita
	UC02	Pengguna dapat mengakses berita
KF02	UC03	Admin dapat menghapus topik forum
	UC04	Admin dapat menghapus komentar
	UC05	Pengguna dapat menambahkan topik/pertanyaan
	UC06	Pengguna dapat menawab pertanyaan dari pengguna lain
KF03	UC07	Admin dapat mengelola item jual budidaya perikanan
	UC08	Pengguna dapat melihat item jual budidaya perikanan

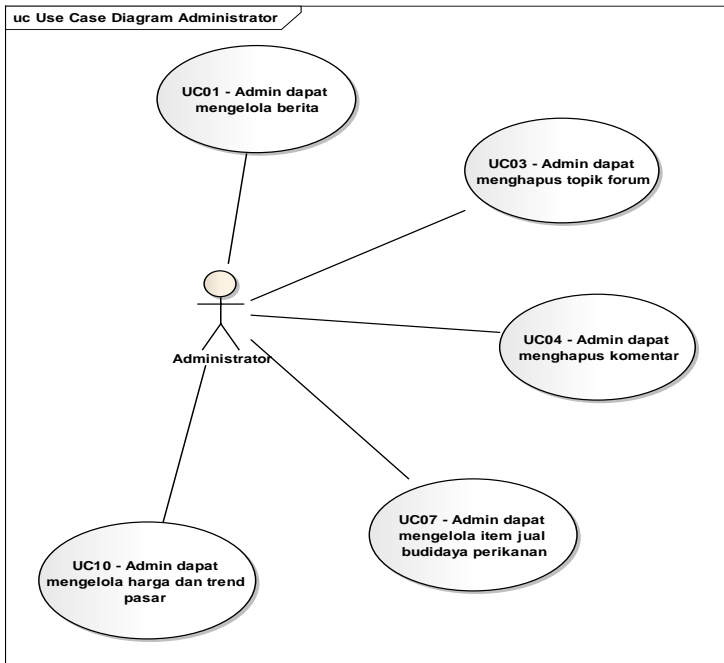
	UC09	Pengguna dapat menjual hasil budidaya perikanan
KF04	UC10	Admin dapat mengelola harga dan trend pasar
	UC11	Pengguna dapat mengakses informasi harga dan trend pasar

4.2 Perancangan Aplikasi

Tahapan ini dijelaskan mengenai pembuatan diagram *use case* dan *activity diagram*. Aplikasi ini akan dikembangkan menjadi sebuah aplikasi berbasis *web* yang dapat diakses oleh pengguna melalui *desktop* maupun *mobile*. Pengguna dapat mengakses aplikasi melalui koneksi internet dengan tampilan antar muka yang telah disesuaikan dengan *desktop* maupun *mobile*. Pengolahan data dari *database* akan dikelola oleh administrator melalui aplikasi yang bertindak sebagai *server*.

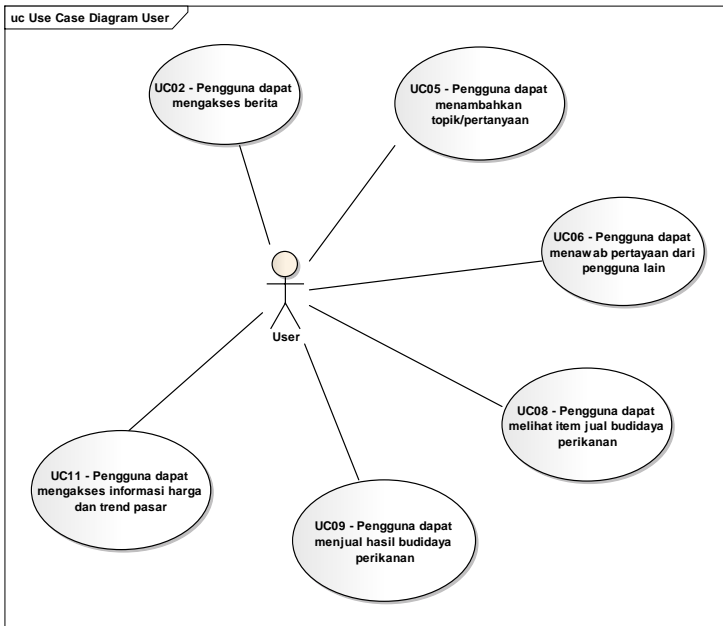
4.2.1 Use Case Diagram

Use Case Diagram adalah diagram *use case* yang menggambarkan sekelompok *use case* dan aktor yang disertai dengan hubungan diantara keduanya. Diagram *use case* menjelaskan kebutuhan yang diinginkan oleh pengguna aplikasi. Kebutuhan-kebutuhan tersebut didapatkan dari hasil wawancara dengan pelaku budidaya perikanan yang kemudian dianalisis sehingga didapatkan daftar kebutuhan fungsional aplikasi. Dari analisis kebutuhan fungsional tersebut didapatkan daftar *use case* dan aktor yang berperan dalam aplikasi. Dari daftar aktor yang didapatkan yaitu administrator dan member, dapat dipetakan menurut *use case* menjadi seperti pada Gambar 4.1 berikut.



Gambar 4.1 Use Case Diagram Administrator

Pada Gambar 4.1 diagram *use case* administrator dapat diketahui bahwa aktor administrator memiliki beberapa *use case*. Dari gambar diatas dapat diketahui bahwa administrator memiliki lima aktivitas yang diantaranya adalah admin dapat mengelola berita. Admin dapat menambahkan berita, melihat detail berita, dan menghapus berita. Kemudian admin dapat menghapus topik forum. Dari fitur aplikasi yaitu fitur tanya jawab, admin juga dapat menghapus komentar dari member. Selain itu, admin dapat mengelola item jual beli. Dan yang terakhir adalah admin dapat mengelola harga dan trend pasar dengan menambahkan harga dari aplikasi *server*.



Gambar 4.2 Use Case Diagram Member

Dari diagram *use case* pada Gambar 4.2 dapat diketahui bahwa aktor member memiliki enam *use case* yaitu member dapat mengakses berita, member dapat menambahkan topik/pertanyaan, member dapat menjawab pertanyaan dari pengguna lain, member dapat melihat item jual budidaya perikanan, member dapat menjual hasil budidaya perikanan, dan member dapat mengakses informasi harga dan trend pasar.

4.2.2 Diagram Aktifitas

Diagram aktivitas digunakan untuk menggambarkan proses bisnis (alur kerja) pada aplikasi. Sebuah diagram aktivitas digunakan untuk menunjukkan suatu alur kegiatan secara berurutan. Diagram aktivitas dibuat berdasarkan *use case* yang sebelumnya telah dibuat. Daftar diagram aktivitas dan

pemetaannya terhadap *use case* dapat dilihat pada Tabel 4.8 di bawah ini.

Tabel 4.8 Diagram Aktivitas

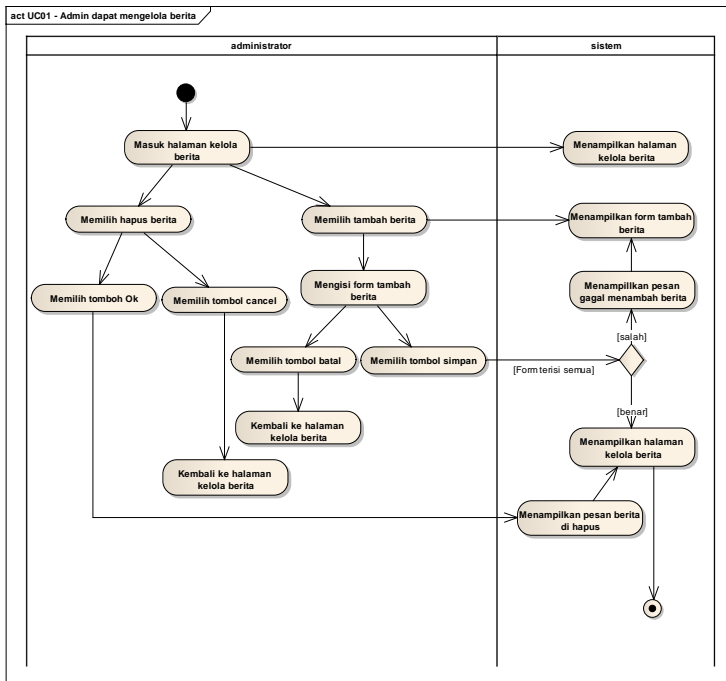
<i>Activity Diagram</i>			
ID Use Case	ID AD	Aktor	Aktivitas
UC01	AD01	Administrator	menelola berita
UC02	AD02	Member	mengakses berita
UC03	AD03	Administrator	menghapus topik forum
UC04	AD04	Administrator	menghapus member
UC05	AD05	Member	menambahkan topik/pertanyaan
UC06	AD06	Member	Menjawab pertanyaan dari pengguna lain
UC07	AD07	Administrator	menelola item jual beli
UC08	AD08	Member	melihat item jual budidaya perikanan
UC09	AD09	Member	Menjual hasil budidaya perikanan
UC10	AD10	Administrator	menelola harga dan trend pasar
UC11	AD11	Member	mengakses informasi harga dan trend pasar

Diagram aktivitas yang memiliki peran penting dalam aplikasi ini akan dijelaskan seperti di bawah.

4.2.2.1 Mengelola Berita

Pada Gambar 4.3 menjelaskan mengenai alur bagaimana administrator mengelola berita pada aplikasi *server*. Admin telah login ke dalam *backpanel* admin. Dan memilih halaman kelola berita. Kemudian admin dapat menambah berita dengan menekan tombol tambah berita. Admin mengisi judul berita, kategori berita, isi berita, dan gambar berita. Lalu admin menekan button simpan untuk menyimpan berita baru. Admin dapat melihat detail berita, dan dapat melakukan hapus berita. Jika terdapat *field* yang belum lengkap, namun admin telah menekan tombol simpan maka sistem akan menampilkan

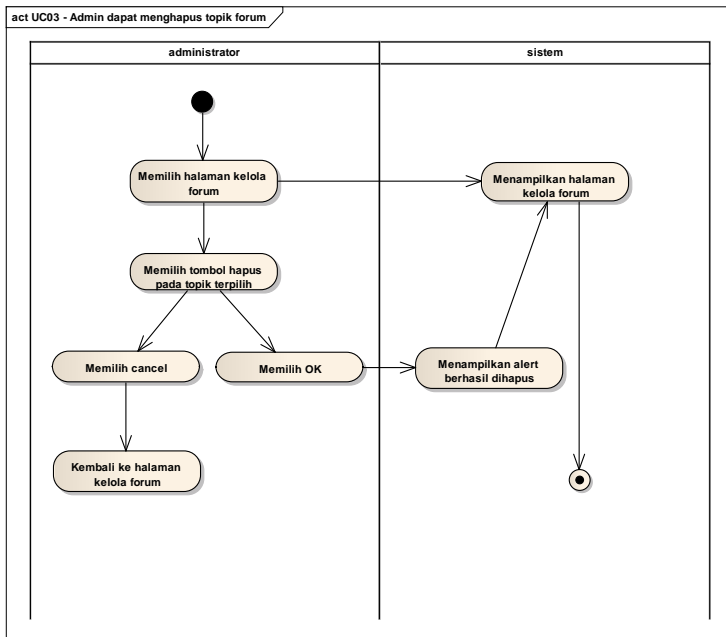
pesan peringatan “*Please fill the field*”. Jika admin menghapus berita, akan menampilkan pesan peringatan “anda ingin menghapus berita?” tekan *OK* atau *cancel*.



Gambar 4.3 Diagram Aktivitas Mengelola Berita

4.2.2.2 Menghapus Topik Forum

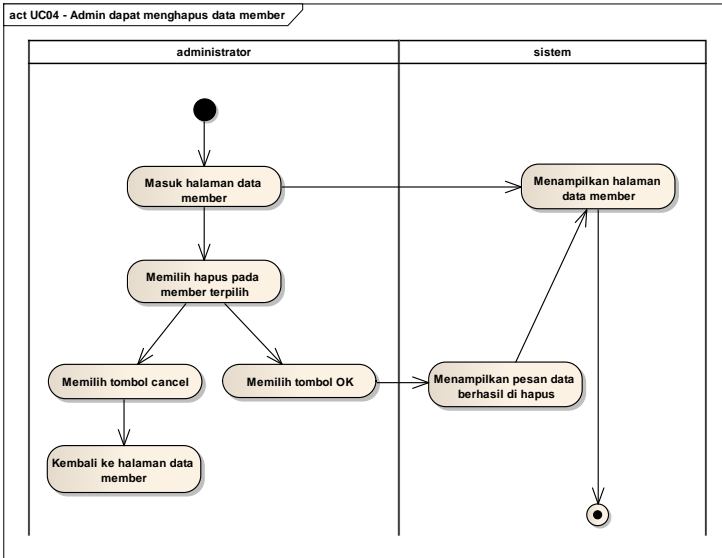
Pada Gambar 4.4 menjelaskan mengenai alur bagaimana administrator menghapus topik forum pada aplikasi *server*. Admin telah login ke dalam *backpanel* admin. Dan memilih halaman kelola forum. Admin berada di halaman kelola forum. Admin dapat melihat detail topik, dan dapat melakukan hapus topik. Jika admin menghapus topik, akan menampilkan pesan peringatan “anda ingin menghapus topik?” tekan *OK* atau *cancel*.



Gambar 4.4 Diagram Aktivitas Menghapus Topik

4.2.2.3 Menghapus Data Member

Pada Gambar 4.5 menjelaskan mengenai alur bagaimana administrator menghapus data member pada aplikasi *server*. Admin telah login ke dalam *backpanel* admin. Dan memilih halaman data member. Admin berada di halaman data member. Admin dapat melihat detail member, dan dapat melakukan hapus member. Jika admin menghapus member, akan menampilkan pesan peringatan “anda ingin menghapus member?” tekan *OK* atau *cancel*.

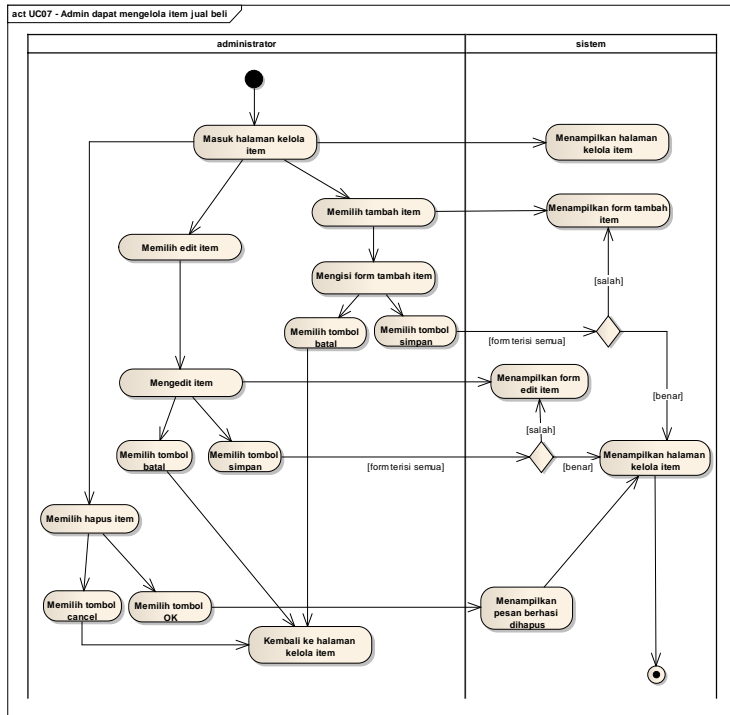


Gambar 4.5 Diagram Aktivitas Menghapus Data Member

4.2.2.4 Mengelola Item Jual Beli

Pada Gambar 4.6 menjelaskan mengenai alur bagaimana administrator mengelola item jual beli pada aplikasi *server*. Admin telah login ke dalam *backpanel* admin. Dan memilih halaman data item. Admin berada di halaman *backpanel* data item. Admin menambah item dengan menekan button tambah item. Admin mengisikan nama item, kategori, nama supplier, harga, dan jumlah item tersedia. Lalu admin menekan tombol simpan untuk meyimpan item baru baru. Admin dapat melihat detail item, dan dapat melakukan hapus item. Jika terdapat *field* yang belum lengkap, namun admin telah menekan tombol simpan maka sistem akan menampilkan pesan peringatan “*Please fill the field!*”.

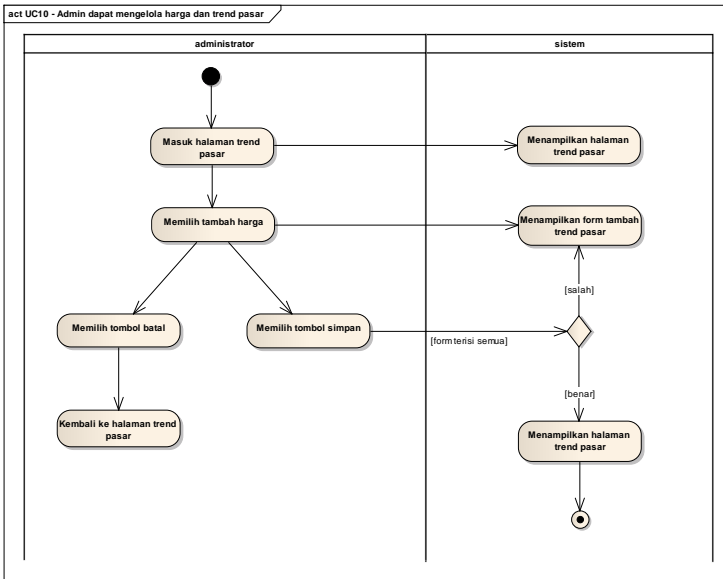
Jika admin menghapus item, akan menampilkan pesan peringatan “anda ingin menghapus item?” tekan *OK* atau *cancel*.



Gambar 4.6 Diagram Aktivitas Mengelola Item Jual Beli

4.2.2.5 Mengelola Harga dan Trend Pasar

Pada Gambar 4.7 menjelaskan mengenai alur bagaimana administrator mengelola harga dan trend pasar pada aplikasi server. Admin telah login ke dalam *backpanel* admin. Dan memilih halaman trend pasar. Admin berada di halaman *backpanel* trend pasar. Admin menambah harga dengan menekan tombol tambah. Admin mengisikan nama ikan, tanggal, dan harga. Lalu admin memilih tombol simpan untuk menyimpan harga baru. Jika terdapat *field* yang belum lengkap, namun admin telah menekan tombol simpan maka sistem akan menampilkan pesan peringatan “*Please fill the field!*”.



Gambar 4.7 Diagram Aktivitas Mengelola Harga dan Trend

4.3 Analisis Kebutuhan Iterasi ke-2

Pengembangan aplikasi dengan menggunakan metode *prototyping* membutuhkan beberapa kali perulangan. Pada saat telah dilakukan implementasi pada iterasi pertama, terdapat beberapa kekurangan pada aplikasi. Maka perlu dilakukan perbaikan terhadap aplikasi dengan melakukan perubahan pada kebutuhan sistem. Berikut ini beberapa perubahan-perubahan yang terjadi pada iterasi ke-2.

4.3.1 Kebutuhan Fungsional Sistem Iterasi ke-2

Dari hasil iterasi ke-2, dibutuhkan perubahan pada aplikasi. Perubahan yang dilakukan adalah dengan menambahkan aktor baru. Pada aplikasi sebelumnya hanya terdapat dua aktor yaitu administrator dan member. Karena pengunjung aplikasi selain member seharusnya juga bisa mengakses aplikasi tanpa melakukan *login*, maka perlu perubahan aplikasi dengan

menambahkan aktor baru yaitu pengunjung. Penambahan aktor baru juga akan merubah sebagian *use case*. Oleh karena itu pada iterasi ke-2 ini akan dijelaskan perubahan yang terjadi dimulai dari kebutuhan fungsional sistem.

Tabel 4.9 Kebutuhan Fungsional Iterasi ke-2

Kebutuhan Fungsional		
ID KF	KF Description	Refrensi
KF04	Sistem dapat menyediakan informasi harga dan trend pasar	US01, US07, US02, US08

Terjadi perubahan pada kebutuhan fungsional sistem dengan menggabungkan kebutuhan fungsional menjadi seperti pada Tabel 4.9 diatas. Perubahan pada kebutuhan fungsional KF04 dan KF05 menjadi KF04 dengan deskripsi sistem dapat menyediakan informasi harga dan trend pasar.

4.3.2 Penentuan Aktor Iterasi ke-2

Perubahan aktor dilakukan pada iterasi ke-2 dengan melakukan penambahan aktor baru, yaitu pengunjung. Tiap-tiap aktor memiliki kepentingan seperti pada Tabel 4.10 di bawah.

Tabel 4.10 Penentuan Aktor Iterasi ke-2

Aktor	
Aktor	Kepentingan
Administrator	Mengelola seluruh kegiatan aplikasi
Member	Semua entitas pengguna aplikasi yang telah mendaftarkan diri menjadi member.
Pengunjung	Semua entitas pengguna aplikasi yang dapat mengakses aplikasi tanpa menggunakan akun member.

4.3.3 Daftar *Use Case* Iterasi ke-2

Pada Tabel 4.11 berikut ini adalah tambahan dan perubahan *use case* pada iterasi ke-2. Perubahan ini didapatkan dari penambahan aktor baru dan penggabungan kebutuhan fungsional iterasi k-2 yang kemudian diturunkan menjadi *use case* iterasi ke-2 yang baru.

Tabel 4.11 Daftar Use Case Iterasi ke-2

Use Case Description		
ID KF	ID UC	Description
KF01	UC01	Admin dapat mengelola berita
	UC02	Pengguna (member dan pengunjung) dapat mengakses berita
KF02	UC03	Admin dapat menghapus topik forum
	UC04	Admin dapat menghapus member
	UC05	Member dapat menambahkan topik dan berkomentar
	UC06	Pengunjung dapat mendaftar menjadi member
KF03	UC07	Admin dapat mengelola item jual beli
	UC08	Pengguna (member dan pengunjung) dapat melihat item jual beli
	UC09	Member dapat memesan item
KF04	UC10	Admin dapat mengelola harga dan trend pasar
	UC11	Pengguna (member dan pengunjung) dapat mengakses informasi harga dan trend pasar

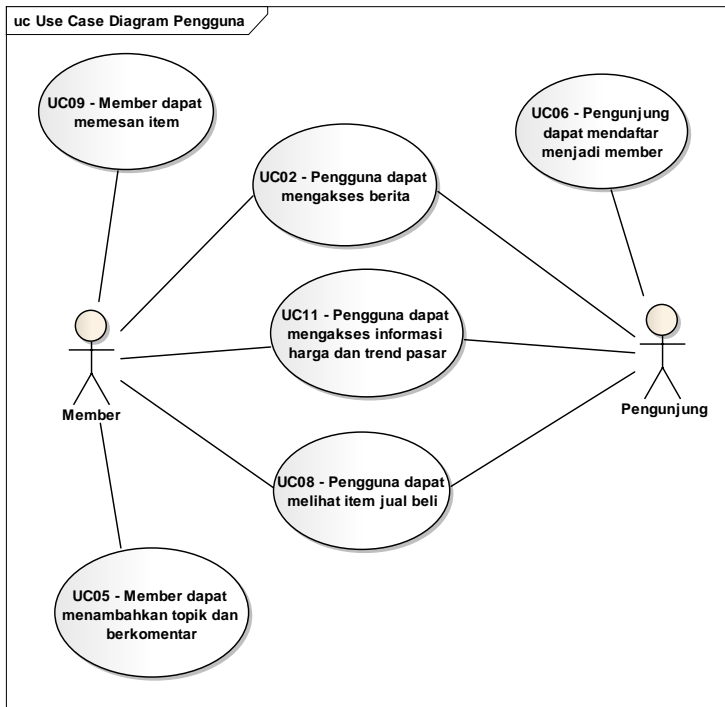
Terjadi perubahan pada *use case* deskripsi UC05, UC06, dan UC09.

4.4 Perancangan Aplikasi Iterasi ke-2

Dengan adanya perubahan kebutuhan sistem dan penambahan aktor baru, maka *use case* dan *activity diagram* juga akan berubah. Berikut ini adalah perancangan aplikasi pada iterasi kedua yang meliputi *use case* dan *activity diagram*.

4.4.1 Use Case Diagram Iterasi ke-2

Dengan adanya perubahan kebutuhan pada iterasi ke-2, maka *use case* juga mengalami perubahan khususnya pada *use case* dengan aktor baru. Berikut ini adalah daftar *use case diagram* pada perubahan perubahan pada iterasi ke-2.



Gambar 4.8 Use Case Diagram Iterasi ke-2

Seperti yang terlihat pada Gambar 4.8 diatas, *use case diagram* iterasi kedua terjadi penambahan pada aktor. Aktor pengunjung mendapatkan *use case* baru yaitu mendaftar menjadi member. Dengan penambahan aktor pengunjung maka pengguna aplikasi akan berubah tidak hanya member yang telah terdaftar, namun juga pengunjung yang tertarik untuk mengakses aplikasi ini. Dengan penambahan aktor baru

akan merubah *use case* pada aktor lama. Perubahan terjadi pada aktor member yang memiliki *use case* baru, yaitu member dapat memesan item dan member dapat menambahkan topik dan berkomentar. *Use case* dengan aktor pengguna merupakan *use case* yang dapat dilakukan oleh member dan pengunjung.

4.4.2 Diagram Aktivitas Iterasi ke-2

Diagram aktivitas iterasi ke-2 dibuat berdasarkan *use case* pada iterasi ke-2 yang sebelumnya telah dilakukan perubahan. Daftar diagram aktivitas dan pemetaannya terhadap *use case* iterasi ke-2 dapat dilihat pada Tabel 4.12 di bawah ini.

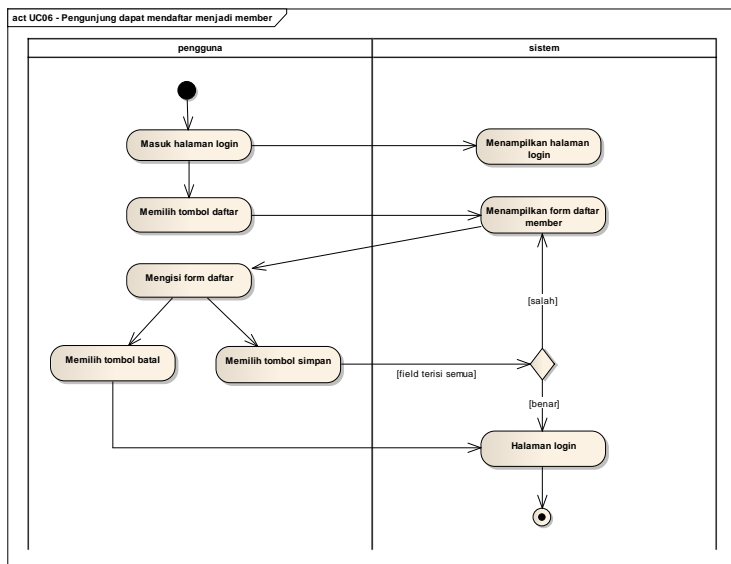
Tabel 4.12 Diagram Aktivitas Iterasi ke-2

Activity Diagram			
ID Use Case	ID AD	Aktor	Aktivitas
UC01	AD01	Administrator	menelola berita
UC02	AD02	Pengguna	mengakses berita
UC03	AD03	Administrator	menghapus topik forum
UC04	AD04	Administrator	menghapus member
UC05	AD05	Member	menambahkan topik dan berkomentar
UC06	AD06	Pengunjung	mendaftar menjadi member
UC07	AD07	Administrator	menelola item jual beli
UC08	AD08	Pengguna	melihat item jual beli
UC09	AD09	Member	memesan item
UC10	AD10	Administrator	menelola harga dan trend pasar
UC11	AD11	Pengguna	mengakses informasi harga dan trend pasar

Berikut ini adalah daftar diagram aktivitas pada perubahan-perubahan pada iterasi kedua.

4.4.2.1 Mendaftar Menjadi Member

Pada Gambar 4.9 menjelaskan mengenai alur bagaimana pengunjung mendaftar menjadi member. Pengunjung telah masuk ke halaman *login* aplikasi dan memilih tombol daftar. Sistem memunculkan form daftar member. Setelah pengunjung mengisi form dan memilih tombol simpan, maka sistem akan menyimpan data pada *database* dan mengarahkan menuju halaman login. Apabila terdapat *field* kosong, akan muncul peringatan “*please fill the field*”.

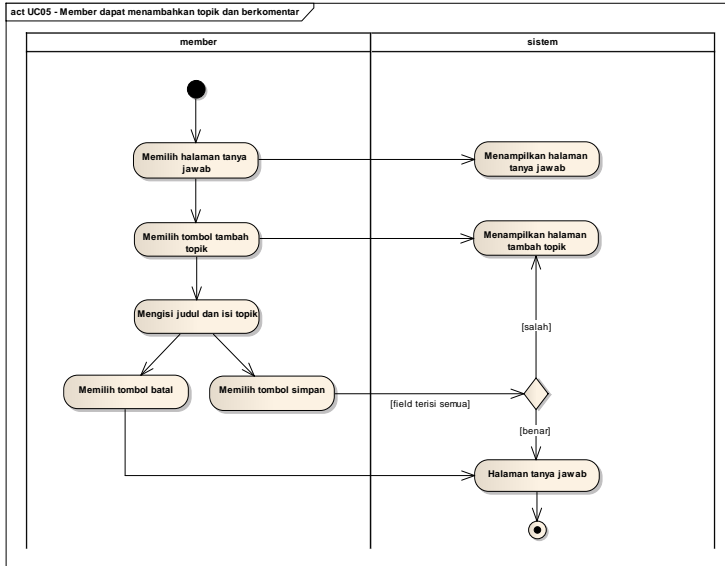


Gambar 4.9 Diagram Aktivitas Mendaftar Menjadi Member

4.4.2.2 Menambahkan Topik dan Berkomentar

Pada Gambar 4.10 menjelaskan mengenai alur bagaimana pengunjung dapat menambahkan topik dan berkomentar pada halaman tanya jawab. Pengunjung telah masuk ke halaman awal dan melakukan *login*. Member memilih tab menu tanya jawab. Member kemudian memilih tambah topik. Sistem memunculkan form tambah topik. Setelah member

mengisi form dan memilih tombol simpan, maka sistem akan menyimpan data pada *database* dan mengarahkan menuju halaman tanya jawab. Apabila terdapat *field* kosong, akan muncul peringatan “*please fill the field*”.



Gambar 4.10 Diagram Aktivitas Menambahkan Topik

BAB V IMPLEMENTASI

Bab ini menjelaskan mengenai bagaimana tahap-tahap penelitian diimplementasikan. Pada bab ini juga menjelaskan mengenai cara melakukan penelitian secara teknis agar dapat dilakukan kembali dengan mudah. Hal-hal yang akan dibahas pada bab ini adalah lingkungan implementasi, konfigurasi sistem, antar-muka, serta kode program.

5.1 Lingkungan Implementasi

Aplikasi ini dikembangkan dengan menggunakan beberapa perangkat keras diantaranya adalah *handphone*, *tablet* dan *laptop*. Pada tahapan ini terdapat dua poin dalam implementasi lingkungan, yaitu perangkat keras (*hardware*) dan perangkat lunak (*software*).

5.1.1 Implementasi *Hardware*

Lingkungan implementasi pada *hardware* yang digunakan untuk mengelola aplikasi *server* komputer. *Hardware* yang digunakan dalam komputer *server* tersebut adalah:

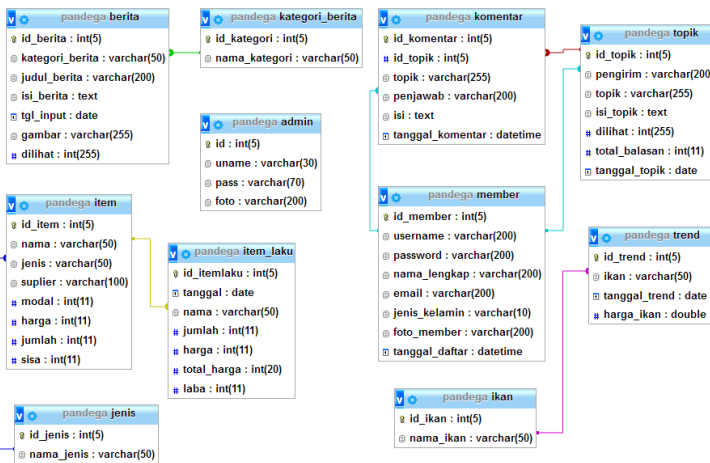
- *Processor : intel core i3 1.80GHz*
- *Memory RAM : 4 Gb*
- *Operating System : windows 7*

5.1.2 Implementasi *Software*

Spesifikasi *software* yang diimplementasikan harus mampu menjalankan *web browser*. Sistem operasi minimal yang disarankan menggunakan *windows 7* karena telah tersedia fasilitas *web browser* seperti *Google Chrome*, *Mozilla*, dan *Internet Explorer* yang mendukung *javascript*, *HTML5*, dan *CSS3*.

5.2 Pembuatan Database

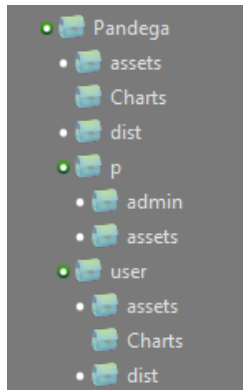
Setelah dilakukan analisis kebutuhan sistem serta mendapatkan hasil perancangan aplikasi, selanjutnya adalah membuat desain basis data pada sistem. *Database* dibuat berdasarkan desain model data fisik. Pembuatan model *database* pada aplikasi ini menggunakan *database MySQL*. Desain *database* aplikasi dapat dilihat pada Gambar 5.1.



Gambar 5.1 Desain Database

5.3 Struktur Direktori

Struktur direktori aplikasi bisa dilihat pada Gambar 5.2. Direktori yang terdapat pada aplikasi dibedakan menjadi tiga bagian, yaitu direktori */p/admin*, direktori *user*, dan direktori utama. Pada direktori *assets* berfungsi untuk menempatkan file *CSS* dan *javascript*. Direktori gambar berisi file gambar untuk gambar berita dan item. Direktori foto berisi file gambar untuk profile pengguna. Dan direktori *Chart* dan *dist* berisi file untuk menampilkan grafik. File-file lain berada pada direktori utama. Gambar direktori utama.



Gambar 5.2 Struktur Direktori

5.4 Implementasi Fungsi dan Pengkodean

Pengkodean dan implementasi fungsi dilakukan setelah *database* selesai dibuat. Pada pengkodean ini, peneliti menggunakan bahasa pemrograman *PHP* dan dengan *database MySQL*. Pada pembuatan aplikasi ini terdapat berbagai fitur yang diimplementasikan, berikut ini merupakan beberapa contoh implementasi fungsi:

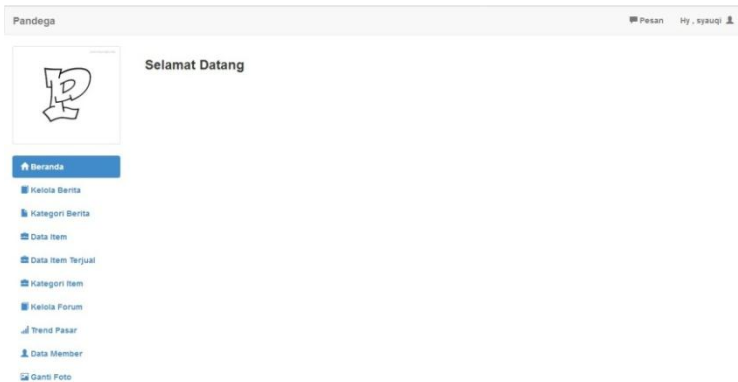
5.4.1 Implementasi *Login*

Pada awal admin membuka halaman aplikasi *server*, akan dihadapkan dengan halaman awal dan untuk dapat mengakses aplikasi admin harus melakukan login terlebih dahulu.

Silahkan Login ..

Gambar 5.3 Halaman *Login Server*

Tampilan fitur login seperti pada Gambar 5.3. Saat *login* admin diminta untuk memasukkan *username* dan *password* yang telah didaftarkan pada *database*. Setelah *login* admin akan diarahkan ke halaman awal aplikasi *server*.



Gambar 5.4 Halaman Awal Aplikasi *Server*

Pengkodean halaman *login* seperti pada Gambar 5.5.

```

<?php
session_start();
include 'admin/config.php';
$username=$_POST['uname'];
$password=$_POST['pass'];
$pass=md5($password);
$query=mysql_query("select * from admin where
uname='$username' and pass='$pass'")or die(mysql_error());
if(mysql_num_rows($query)==1){
    $_SESSION['uname']=$username;
    header("location:admin/index.php");
}else{
    header("location:index.php?pesan=gagal")or
die(mysql_error());
    // mysql_error();
}
// echo $pass;
?>

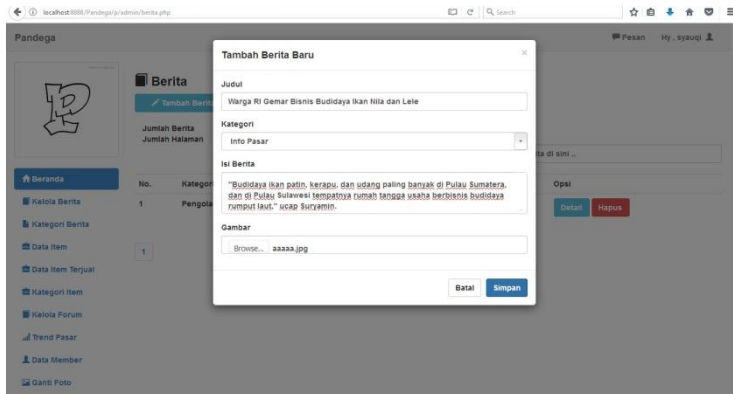
```

Gambar 5.5 Kode Proses Login Server

Untuk melakukan *login* pada aplikasi *server*, terdapat proses mengenkripsi *password* dengan menggunakan *md5*. Dan pada proses *login*, *username* akan disimpan kedalam *session*.

5.4.2 Implementasi Mengelola Berita

Pada implementasi ini admin dianggap telah melakukan *login* dan masuk halaman awal aplikasi *server*. Kemudian admin memilih halaman kelola berita pada bar navigasi dan memilih tombol tambah berita. Dan akan muncul form untuk menambahkan berita baru.



Gambar 5.6 Form Tambah Berita Baru

Pada proses menambahkan berita dapat dilihat pengkodean yang telah dilakukan seperti dibawah.


```

<?php
include 'config.php';
$kategori_berita=$_POST['kategori_berita'];
$judul_berita=$_POST['judul_berita'];
$isi_berita=$_POST['isi_berita'];
$tgl_input=date("Y/m/d");
$dilihat=(0);
#tangkap gambar
$namafolder="gambar/"; //folder tempat menyimpan file
if (!empty($_FILES["gambar"]["tmp_name"]))
{
    $jenis_gambar=$_FILES['gambar']['type'];
    if($jenis_gambar=="image/jpeg"
$jenis_gambar=="image/jpg"
$jenis_gambar=="image/gif"
$jenis_gambar=="image/png")
    {
        $gambar          =          $namafolder
basename($_FILES['gambar']['name']);
        if
(move_uploaded_file($_FILES['gambar']['tmp_name'],
$gambar)) {
            mysql_query("insert          into          berita
values(',$kategori_berita',$judul_berita,$isi_berita',$tgl_i
nput',$gambar',$dilihat)");
                ?>
                    <script
language="javascript">
                        alert('Berhasil menambahkan');
                        document.location.href="berita.php";
                    </script>
                <?php
            } else {
                ?>
                    <script
language="javascript">
                        alert('Gagal menambahkan');

```

```

        document.location.href="berita.php";
    </script>

    <?php

    }
} else {
    ?>
    <script language="javascript">
        alert("Gambar harus berformat .jpg .png .gif");
        document.location.href="berita.php";
    </script>
    <?php

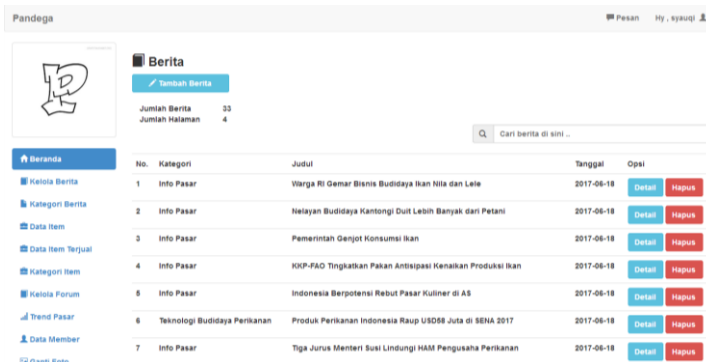
    }
} else {
    echo "Berita gagal disimpan";
}
?>

```

Gambar 5.7 Kode Proses Menambahkan Berita

Pengkodean pada Gambar 5.7 diatas akan menyimpan data pada *database* apabila gambar yang di masukkan berhasil disimpan pada direktori gambar.

Pada halaman kelola berita, terdapat dua pilihan untuk mengelola berita, yaitu detail berita dan hapus berita.



No.	Kategori	Judul	Tanggal	Opsi
1	Info Pasar	Warga RI Gemar Bisnis Budidaya Ikan Nila dan Lela	2017-06-18	Detail Hapus
2	Info Pasar	Nelayan Budidaya Kantong Duit Lebih Banyak dari Petani	2017-06-18	Detail Hapus
3	Info Pasar	Pemerintah Genjot Konsumsi Ikan	2017-06-18	Detail Hapus
4	Info Pasar	KKP-PAO Tingkatkan Pakan Antipati Kenaikan Produksi Ikan	2017-06-18	Detail Hapus
5	Info Pasar	Indonesia Berpotensi Rebut Pasar Kuliner di AS	2017-06-18	Detail Hapus
6	Teknologi Budidaya Perikanan	Produk Perikanan Indonesia Raup USD58 Juta di SENA 2017	2017-06-18	Detail Hapus
7	Info Pasar	Tiga Jurus Menteri Susi Lindungi HAM Pengusaha Perikanan	2017-06-18	Detail Hapus

Gambar 5.8 Halaman Kelola Berita

Pada pilihan detail, admin dapat melihat detail berita. Pada pilihan hapus, admin dapat menghapus berita. Pengkodean untuk menghapus berita seperti pada Gambar 5.9.

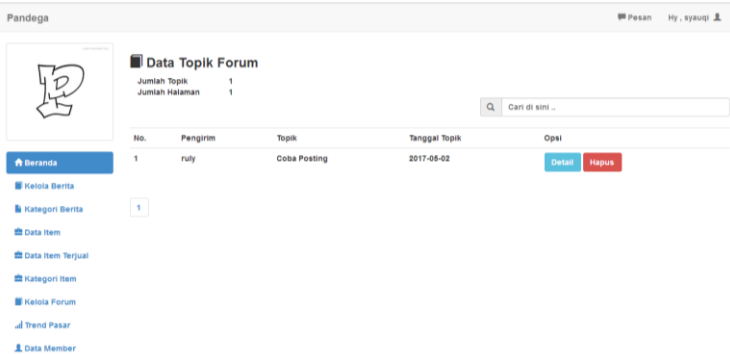
```
<?php
include 'config.php';
$Sid=$_GET['id'];
mysql_query("delete from berita where id_berita='$id'");
header("location:berita.php");
?>
```

Gambar 5.9 Kode Proses Menghapus Berita

Setelah admin memilih hapus berita dan memilih *OK*, maka berita akan di *delete* dan diarahkan menuju halaman kelola berita.

5.4.3 Implementasi Menghapus Topik Forum

Pada implementasi ini admin dianggap telah melakukan login dan masuk halaman awal aplikasi *server*. Kemudian admin memilih halaman kelola forum pada bar navigasi dan memilih tombol hapus untuk menghapus topik atau tombol detail untuk melihat detail topik.



Gambar 5.10 Halaman Kelola Forum

Apabila admin memilih tombol hapus dan memilih *OK*, maka topik akan dihapus dari *database*. Pengkodean menghapus topik adalah seperti pada Gambar 5.11 dibawah.

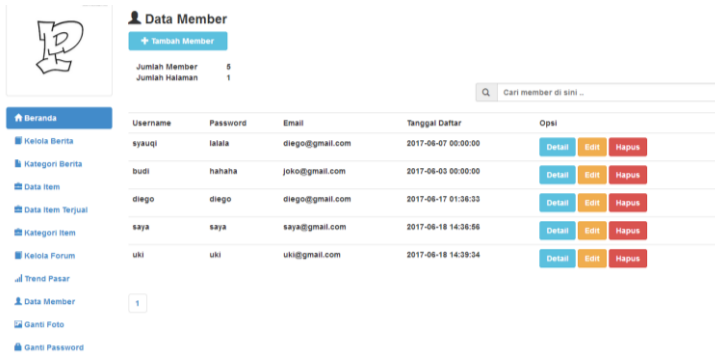
```
<?php
include 'config.php';
$id=$_GET['id'];
mysql_query("delete from topik where id_topik='$id'");
header("location:topik.php");
?>
```

Gambar 5.11 Kode Proses Menghapus Topik

Setelah admin memilih *OK* proses diatas akan berjalan dengan melakukan *query database* yaitu *delete*.

5.4.4 Implementasi Menghapus Data Member

Pada implementasi ini admin dianggap telah melakukan login dan masuk halaman awal aplikasi *server*. Kemudian admin memilih halaman data member pada bar navigasi dan memilih tombol hapus untuk menghapus member atau tombol detail untuk melihat detail member.



Username	Password	Email	Tanggal Datar	Ops
syauqi	lalata	diego@gmail.com	2017-06-07 00:00:00	Detail Edit Hapus
budi	hahaha	joko@gmail.com	2017-06-03 00:00:00	Detail Edit Hapus
diego	diego	diego@gmail.com	2017-06-17 01:38:33	Detail Edit Hapus
saya	saya	saya@gmail.com	2017-06-18 14:38:56	Detail Edit Hapus
uki	uki	uki@gmail.com	2017-06-18 14:39:34	Detail Edit Hapus

Gambar 5.12 Halaman Data Member

Apabila admin memilih tombol hapus dan memilih *OK*, maka topik akan dihapus dari *database*. Pengkodean menghapus topik adalah seperti pada Gambar 5.13 dibawah.

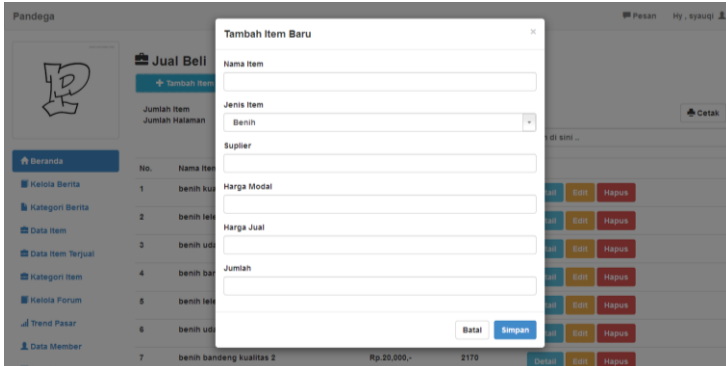
```
<?php
include 'config.php';
$id=$_GET['id'];
mysql_query("delete          from          member          where
id_member='$id'");
header("location:member.php");
?>
```

Gambar 5.13 Kode Proses Menghapus Data Member

Setelah admin memilih *OK* proses diatas akan berjalan dengan melakukan *query database* yaitu *delete*.

5.4.5 Implementasi Mengelola Item Jual Beli

Pada implementasi ini admin dianggap telah melakukan login dan masuk halaman awal aplikasi *server*. Kemudian admin memilih halaman data item pada bar navigasi dan memilih tombol tambah item. Dan akan muncul form untuk menambahkan item baru.



Gambar 5.14 Form Menambahkan Item Baru

Pada proses menambahkan item dapat dilihat pengkodean yang telah dilakukan seperti pada Gambar 5.15 dibawah.

```
<?php
include 'config.php';
$nama=$_POST['nama'];
$jenis=$_POST['jenis'];
$suplier=$_POST['suplier'];
$modal=$_POST['modal'];
$harga=$_POST['harga'];
$jumlah=$_POST['jumlah'];
$sisas=$_POST['jumlah'];
mysql_query("insert          into          item
values('$nama','$jenis','$suplier','$modal','$harga','$jumlah
h','$sisas)");
header("location:item.php");
?>
```

Gambar 5.15 Kode Proses Menambahkan Item Baru

Setelah admin memilih *OK* proses diatas akan berjalan dengan melakukan *query database* yaitu *insert*. Pada halaman data item, terdapat tiga pilihan untuk mengelola item, yaitu detail item, edit item, dan hapus item.

No.	Nama Item	Harga Jual	Jumlah	Opsi
1	benih kualitas 1	Rp.22.000,-	1000	Detail Edit Hapus
2	benih lele kualitas 1	Rp.16.000,-	1000	Detail Edit Hapus
3	benih udang kualitas 1	Rp.31.000,-	700	Detail Edit Hapus
4	benih bandeng kualitas 1	Rp.20.500,-	2000	Detail Edit Hapus
5	benih lele kualitas 2	Rp.18.000,-	3800	Detail Edit Hapus
6	benih udang kualitas 2	Rp.23.500,-	600	Detail Edit Hapus
7	benih bandeng kualitas 2	Rp.20.000,-	2170	Detail Edit Hapus

Gambar 5.16 Halaman Kelola Item Jual Beli

Pada pilihan detail, admin dapat melihat detail item. Pada pilihan edit, admin dapat merubah detail item. Dan pada pilihan hapus, admin dapat menghapus item. Apabila admin memilih edit, akan muncul form untuk merubah detail item. Pengkodean untuk mengedit item seperti pada Gambar 5.17.

```
<?php
include 'config.php';
$id=$_POST['id'];
$nama=$_POST['nama'];
$jenis=$_POST['jenis'];
$supplier=$_POST['supplier'];
$modal=$_POST['modal'];
$harga=$_POST['harga'];
$jumlah=$_POST['jumlah'];
mysql_query("update item set nama='$nama',
jenis='$jenis', supplier='$supplier', modal='$modal',
harga='$harga', jumlah='$jumlah' where id_item='$id'");
header("location:item.php");
?>
```

Gambar 5.17 Kode Proses Mengedit Item Jual Beli

Setelah admin memilih edit dan memilih *OK*, maka item akan di *update* dan diarahkan menuju halaman data item. Kemudian apabila admin memilih hapus, maka proses pengkodean seperti Gambar 5.18 di bawah.

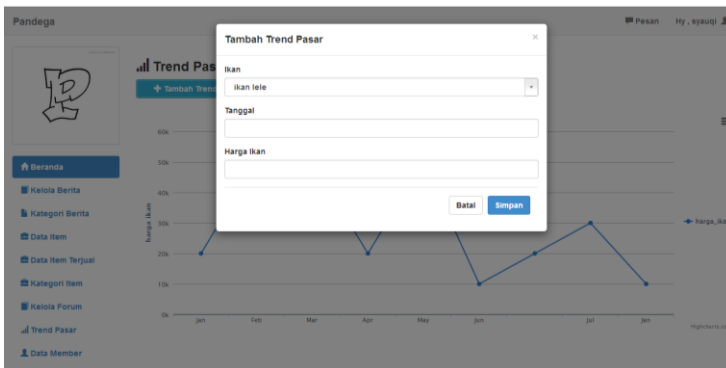
```
<?php
include 'config.php';
$id=$_GET['id'];
mysql_query("delete from item where id_item='$id'");
header("location:item.php");
?>
```

Gambar 5.18 Kode Proses Menghapus Item

Setelah admin memilih hapus item dan memilih *OK*, maka item akan di *delete* dan diarahkan menuju halaman data item.

5.4.6 Implementasi Mengelola Harga dan Trend Pasar

Pada implementasi ini admin dianggap telah melakukan login dan masuk halaman awal aplikasi server. Kemudian admin memilih halaman trend pasar pada bar navigasi dan memilih tombol tambah untuk menambahkan harga pasar.



Gambar 5.19 Form Tambah Trend Pasar

Setelah form terisi penuh, tombol simpan akan melakukan perintah *insert* pada *database*. Dan pengkodean untuk menambahkan trend pasar seperti pada Gambar 5.20 di bawah.

```
<?php
include 'config.php';
$ikan=$_POST['ikan'];
$tanggal_trend=$_POST['tanggal_trend'];
$harga_ikan=$_POST['harga_ikan'];
mysql_query("insert                into                trend
values('','$ikan','$tanggal_trend','$harga_ikan')");
header("location:trend.php");
?>
```

Gambar 5.20 Kode Proses Menambahkan Trend Pasar

Proses akan melakukan *query* yaitu *insert* dan mengarahkan pada halaman awal trend pasar.

5.4.7 Implementasi Mendaftar Menjadi Member

Pada implementasi ini pengguna dianggap telah masuk halaman awal aplikasi. Kemudian pengguna memilih tombol login pada *header*. Setelah masuk halaman *login*, pengguna memilih tombol daftar.

Daftar ..

Nama Lengkap

Username

Password

Email

Jenis Kelamin

Foto
Browse... No file selected.

Daftar Daftar

Daftar

Gambar 5.21 Form Daftar Menjadi Member

Setelah form terisi penuh, tombol simpan akan melakukan perintah *insert* pada *database*. Dan pengkodean untuk menambahkan data member seperti pada Gambar 5.22 di bawah.

```

<?php
include 'user/config.php';
$username=$_POST['username'];
$password=$_POST['password'];
$nama_lengkap=$_POST['nama_lengkap'];
$email=$_POST['email'];
$jenis_kelamin=$_POST['jenis_kelamin'];
$stanggal_daftar=date("Y-m-d H:i:s");

#tangkap gambar
$namafolder="../pandega/p/admin/foto/"; //folder tempat
menyimpan file
if (empty($_FILES["foto_member"]["tmp_name"]))
{
    $jenis_gambar=$_FILES['foto_member']['type'];
    if($jenis_gambar=="image/jpeg" ||
    $jenis_gambar=="image/jpg" || $jenis_gambar=="image/gif" ||
    $jenis_gambar=="image/png")
    {
        $foto_member = $namafolder .
        basename($_FILES['foto_member']['name']);
        if
        (move_uploaded_file($_FILES['foto_member']['tmp_name'],
        $foto_member)) {
            mysql_query("insert into member
            values(',$username','$password','$nama_lengkap','$email','$je
            nis_kelamin','$foto_member','$stanggal_daftar)");
            ?>
            <script
            language="javascript">
                alert('Berhasil menambahkan');
                document.location.href="login.php";

```

```

        </script>
        <?php
    } else {
        ?>
            <script
language="javascript">
                alert('Gagal menambahkan');
                document.location.href="daftar.php";
            </script>
            <?php
        }
    } else {
        ?>
            <script language="javascript">
                alert('Gambar harus berformat .jpg .png .gif');
                document.location.href="daftar.php";
            </script>
            <?php
        }
    } else {
        echo "Gagal";
    }
    ?>

```

Gambar 5.22 Kode Proses Mendaftar Menjadi Member

Proses akan melakukan *query* yaitu *insert* dan mengarahkan pada halaman *login*.

5.4.8 Implementasi Menambahkan Topik Baru

Pada implementasi ini pengguna dianggap telah masuk halaman awal aplikasi dan telah melakukan login. Kemudian pengguna memilih menu tanya jawab pada bagian menu. Setelah masuk halaman tanya jawab, pengguna memilih tombol tambah topik.

Gambar 5.23 Halaman Tambah Topik Tanya Jawab

Setelah form tambah topik telah terisi, tombol simpan akan melakukan perintah *insert* pada *database*. Dan pengkodean untuk menambahkan topik seperti pada Gambar 5.24 di bawah.

```

<?php
include 'config.php';
$topik      = $_POST['topik'];
$pengirim   = $_POST['pengirim'];
$isi_topik  = $_POST['isi_topik'];
$dilihat    = 0;
$total_balasan = 0;
$tanggal_topik = date("Y-m-d H:i:s");

$sql = "INSERT INTO topik VALUES ('', '$pengirim',
'$topik', '$isi_topik', '$dilihat', '$total_balasan',
'$tanggal_topik')";
$hasil = mysql_query($sql);

if($hasil){
                                ?>
                                <script
language="javascript">
    alert('Berhasil menambahkan');
    document.location.href="forum.php?id=$id";

```

```

        </script>
        <?php
    }
else {
    echo "topik gagal disimpan <br>";
}
?>

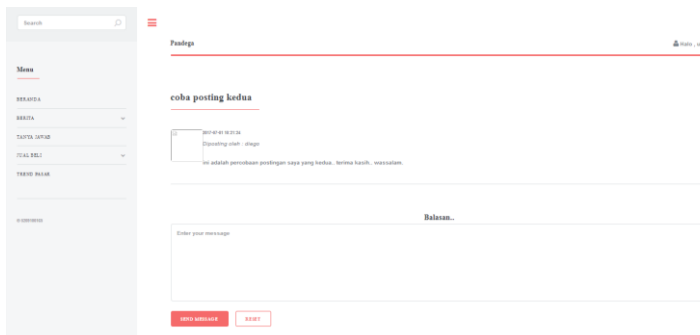
```

Gambar 5.24 Kode Proses Menambahkan Topik Baru

Proses akan melakukan *query* yaitu *insert* dan mengarahkan pada halaman tanya jawab.

5.4.9 Implementasi Menambahkan Komentar

Pada implementasi ini pengguna dianggap telah masuk halaman awal aplikasi dan telah melakukan *login*. Kemudian pengguna memilih menu tanya jawab pada bagian menu. Setelah masuk halaman tanya jawab, pengguna memilih tombol lihat komentar pada postingan yang dipilih.



Gambar 5.25 Halaman Komentar Tanya Jawab

Setelah form balasan telah terisi, tombol simpan akan melakukan perintah *insert* pada *database*. Dan pengkodean untuk menambahkan topik seperti pada Gambar 5.26 di bawah.

```

<?php
include 'config.php';
$idd          = $_POST['id_topikk'];
$stopik       = $_POST['topik'];
$penjawab     = $_POST['penjawab'];
$sisi         = $_POST['isi'];
$stanggal     = date("Y-m-d H:i:s");

$sql = "INSERT INTO komentar VALUES ('', '$idd',
'$stopik', '$penjawab', '$isi', '$tanggal')";
$hasil = mysql_query($sql);

if($hasil){
    ?>
    <script
language="javascript">
    alert('Berhasil menambahkan');

document.location.href="moretopik.php?id=$id";
    </script>
    <?php
    }
else {
    echo "komentar gagal disimpan <br>";
    }

$query_balasan = mysql_query("SELECT id_topik FROM
komentar WHERE id_topik='$id'");
$total_balas = mysql_num_rows($query_balasan);
$total_balasan = $total_balas;

//memasukan total balasan ke database
$sql2          =          "UPDATE          topik          SET
total_balasan='$total_balasan' WHERE id_topik='$id'";
$hasil2 = mysql_query($sql2);

if($hasil2){

```

```
        echo "ok";
    }
else {
    echo "komentar gagal disimpan <br>";
}
?>
```

Gambar 5.26 Kode Proses Menambahkan Komentar

Proses akan melakukan *query* yaitu *insert* dan mengarahkan pada halaman tanya jawab.

Halaman ini sengaja dikosongkan

BAB VI HASIL PENGUJIAN DAN PEMBAHASAN

Pada bab ini menjelaskan mengenai hasil dan pembahasan mengenai aplikasi setelah dilakukan implementasi.

6.1 Pengujian

Pengujian dilakukan setelah pengkodean/implementasi selesai. Pengujian dilakukan berdasarkan kebutuhan fungsional dan kebutuhan non-fungsional. Pengujian kebutuhan fungsional dilakukan dengan menggunakan metode *blackbox testing*. Pengujian dengan menggunakan metode ini dilakukan dengan cara menguji skenario utama dan skenario alternatif pada masing-masing *use case*. Teknis untuk melakukan uji coba fungsional ini menggunakan *test case* seperti pada Tabel 6.1 dibawah. Hasil uji coba berupa *Requirement Traceability Matrix (RTM)*.

Tabel 6.1 Daftar Use Case Test

Deskripsi			
ID KF	ID UC	ID UTC	Deskripsi
KF01	UC02	UTC02	Pengguna (member dan pengunjung) dapat mengakses berita
KF02	UC05	UTC05-1 UTC05-2	Member dapat menambahkan topik dan berkomentar
	UC06	UTC06	Pengunjung dapat mendaftar menjadi member
KF03	UC08	UTC08	Pengguna (member dan pengunjung) dapat melihat item jual beli
	UC09	UTC09	Member dapat memesan item
KF04	UC11	UTC011	Pengguna (member dan pengunjung) dapat mengakses informasi harga dan trend pasar

6.2 Hasil Uji Coba

Pada bagian ini akan dijelaskan mengenai hasil uji coba yang telah dilakukan terhadap aplikasi. Hasil uji coba yang dilakukan terhadap aplikasi meliputi hasil uji coba kebutuhan fungsional dari aplikasi dengan menggunakan *Requirement Traceability Matrix* (RTM). Teknis untuk melakukan uji coba fungsional ini menggunakan *test case* seperti pada Tabel 6.2.

Tabel 6.2 Hasil Uji Coba

ID KF	ID UC	ID AD	ID TC	Hasil Tes
KF01	UC02	AD01	TC02	Sukses
KF02	UC05	AD05	TC05-1	Sukses
			TC05-2	Sukses
	UC06	AD06	TC06	Sukses
KF03	UC08	AD08	TC08	Sukses
	UC09	AD09	TC09	Gagal
KF04	UC11	AD11	TC11	Sukses

6.3 Pembahasan Uji Coba

Berdasarkan hasil uji coba kebutuhan fungsional yang dilakukan menggunakan *Requirement Traceability Matrix* (RTM), dapat dilihat bahwa hampir semua fitur pada aplikasi terpenuhi sesuai dengan tabel *tes case*. Semua kebutuhan fungsional telah melewati skenario uji coba dengan hasil yang telah sesuai dengan *output* ekspektasi yang dapat dilihat pada Lampiran B.

BAB VII

KESIMPULAN DAN SARAN

Bab ini menjelaskan mengenai kesimpulan dan saran dari Tugas Akhir sesuai dengan yang terdapat pada bab sebelumnya.

7.1 Kesimpulan

Tugas Akhir ini membuat rancang bangun aplikasi portal berita berbasis *web* dengan menggunakan model *Prototype* untuk budidaya perikanan. Dari hasil uji coba yang dilakukan pada penelitian ini didapatkan kesimpulan sebagai berikut:

1. Aplikasi ini telah dibuat berdasarkan kebutuhan pengguna yang meliputi:
 - a. Sistem dapat menyediakan informasi tentang budidaya perikanan.
 - b. Sistem dapat menyediakan tempat untuk melakukan tanya jawab.
 - c. Sistem dapat menyediakan tempat untuk memasarkan produk.
 - d. Sistem dapat menyediakan informasi harga pasar.
 - e. Sistem dapat menyediakan informasi trend pasar.
2. Metode pengembangan yang digunakan untuk mengembangkan perangkat lunak adalah *Prototype Model* dengan tahapan sebagai berikut:
 - a. Pengumpulan kebutuhan dari sistem dengan cara mendengar keluhan dari pelanggan.
 - b. Merancang dan membuat *Prototype* dari sistem. *Prototype* yang dibuat disesuaikan dengan kebutuhan sistem yang telah didefinisikan sebelumnya dari wawancara dengan pelaku budidaya perikanan.
 - c. Melakukan uji coba dengan bertindak sebagai pengguna. Kemudian dilakukan evaluasi kekurangan-kekurangan maupun masukan dari

kebutuhan pengguna. Lalu dilakukan pengembangan kembali dengan melalui tahapan dari awal untuk memperbaiki *prototype* yang telah ada.

3. Aplikasi ini telah melewati skenario uji coba untuk semua kebutuhan fungsional dengan hasil yang ditampilkan sistem telah sesuai dengan kebutuhan.

7.2 Saran

Dari penelitian Tugas Akhir ini dapat diberikan saran untuk pengembangan selanjutnya, antara lain:

1. Untuk penelitian selanjutnya dapat ditambahkan fitur grafik trend/*dasboard* untuk tiap jenis ikan.
2. Aplikasi dapat dikembangkan lagi dengan berbasis *android* atau *mobile* agar lebih mudah digunakan.
3. Aplikasi dapat dikembangkan dengan menambahkan fitur jual beli secara *online*.
4. Aplikasi dapat dikembangkan dengan menambahkan setelan pada akun member.

DAFTAR PUSTAKA

- [1] Kementerian Kelautan dan Perikanan, “*Rencana Strategis Kementerian Kelautan Dan Perikanan Tahun 2015-2019*”, Jakarta: KKP RI, 2015.
- [2] Zulkarimen. Nasution, “*Komunikasi Pembangunan: Pengenalan Teori dan Penerapannya. Edisi Revisi*”, Jakarta: Divisi Buku Perguruan Tinggi PT Raja Grafindo Persada, 2004.
- [3] Prasetyo. Tomy, “*Rancang Bangun Aplikasi Website Peneliti Indonesia*”, Surabaya: ITS, 2013.
- [4] Putra. Faisal Setia, “*Layanan Berbasis Lokasi untuk Menemukan Koki Rumahan yang Terdekat dan Tersedia*”, Surabaya: ITS, 2017.
- [5] Kementrian PPN/Bappenas, “*Isu Strategis dan Permasalahannya. Bab 6. hal 6-1 sampai 6.15*”, Jakarta: Bappenas RI, 2016.
- [6] Kadir. Abdul, “*From Zero To A Pro - Membuat Aplikasi Web Dengan PHP dan Database MySQL*”, Yogyakarta: Penerbit Andi, 2009.
- [7] Jenkins. Henry, “*Convergence Culture: Where Old and New Media Collide*”, New York: New York University Press, 2006.
- [8] R. S. Pressman, “*Software Engineering*”, The McGraw Hill, 2005.
- [9] R. D, a. M. Sthephens, “*Use Case Driven Object Modelling with UML-Theory and Practice*”, p. 470, Apress, 2007.

Halaman ini sengaja dikosongkan

LAMPIRAN A

KODE PEMBUATAN APLIKASI

```

<?php
include 'config.php';
$kategori_berita=$_POST['kategori_berita'];
$judul_berita=$_POST['judul_berita'];
$isi_berita=$_POST['isi_berita'];
$stgl_input=date("Y/m/d");
$dilihat=(0);

#tangkap gambar
$namafolder="gambar/"; //folder tempat menyimpan file
if (!empty($_FILES["gambar"]["tmp_name"]))
{
    $jenis_gambar=$_FILES['gambar']['type'];
    if($jenis_gambar=="image/jpeg" ||
    $jenis_gambar=="image/jpg" || $jenis_gambar=="image/gif" ||
    $jenis_gambar=="image/png")
    {
        $gambar = $namafolder .
basename($_FILES['gambar']['name']);
        if (move_uploaded_file($_FILES['gambar']['tmp_name'],
$gambar)) {
            mysql_query("insert into berita
values('$kategori_berita','$judul_berita','$isi_berita','$stgl_inp
ut','$gambar','$dilihat')");
            ?>
                <script
language="javascript">
                    alert('Berhasil menambahkan');
                    document.location.href="berita.php";
                </script>
            <?php
        } else {
            ?>

```

```
                                <script
language="javascript">
    alert('Gagal menambahkan');
    document.location.href="berita.php";
</script>
                                <?php
    }
} else {
    ?>
                                <script language="javascript">
    alert('Gambar harus berformat .jpg .png .gif');
    document.location.href="berita.php";
</script>
                                <?php
    }
} else {
    echo "Berita gagal disimpan";
}
?>
```

Gambar A- 1 Kode Tambah Berita


```
<?php

include 'config.php';
$nama=$_POST['nama'];
$jenis=$_POST['jenis'];
$suplier=$_POST['suplier'];
$modal=$_POST['modal'];
$harga=$_POST['harga'];
$jumlah=$_POST['jumlah'];
$sisas=$_POST['jumlah'];

mysql_query("insert                into                item
values('$nama','$jenis','$suplier','$modal','$harga','$jumlah','$
sisas)");
header("location:item.php");

?>
```

Gambar A- 2 Kode Tambah Item

```
<?php

include 'config.php';
$nama_kategori=$_POST['nama_jenis'];

mysql_query("insert into jenis values('$nama_jenis')");
header("location:jenis.php");

?>
```

Gambar A- 3 Kode Tambah Jenis Item

A-4

```
<?php
include 'config.php';
$nama_kategori=$_POST['nama_kategori'];

mysql_query("insert          into          kategori_berita
values('$nama_kategori')");
header("location:kategori_berita.php");

?>
```

Gambar A- 4 Kode Tambah Kategori Berita

```

<?php
include 'config.php';
$username=$_POST['username'];
$password=$_POST['password'];
$name_lengkap=$_POST['nama_lengkap'];
$email=$_POST['email'];
$jenis_kelamin=$_POST['jenis_kelamin'];
$tanggal_daftar=date("Y-m-d H:i:s");

#tangkap gambar
$namafolder="foto"; //folder tempat menyimpan file
if (!empty($_FILES["foto_member"]["tmp_name"]))
{
    $jenis_gambar=$_FILES['foto_member']['type'];
    if($jenis_gambar=="image/jpeg" //
    $jenis_gambar=="image/jpg" || $jenis_gambar=="image/gif" //
    $jenis_gambar=="image/png")
    {
        $foto_member = $namafolder .
        basename($_FILES['foto_member']['name']);
        if
        (move_uploaded_file($_FILES['foto_member']['tmp_name'],
        $foto_member)) {
            mysql_query("insert into member
            values('$username','$password','$nama_lengkap','$email','$je
            nis_kelamin','$foto_member','$tanggal_daftar')");
            ?>
            <script
            language="javascript">
                alert('Berhasil menambahkan');
                document.location.href="member.php";
            </script>
            <?php
        } else {
            ?>
            <script

```

A-6

```
language="javascript">
    alert('Gagal menambahkan');
    document.location.href="member.php";
</script>
        <?php
    }
} else {
    ?>
        <script language="javascript">
            alert('Gambar harus berformat .jpg .png .gif');
            document.location.href="member.php";
        </script>
        <?php
    }
} else {
    echo "Gagal";
}
?>
```

Gambar A- 5 Kode Tambah Member

```
<?php

include 'config.php';
$ikan=$_POST['ikan'];
$tanggal_trend=$_POST['tanggal_trend'];
$harga_ikan=$_POST['harga_ikan'];

mysql_query("insert          into          trend
values(',$ikan','$tanggal_trend','$harga_ikan)");
header("location:trend.php");

?>
```

Gambar A- 6 Kode Tambah Trend

```
<?php

include 'config.php';
$id=$_POST['id'];
$name=$_POST['nama'];
$type=$_POST['jenis'];
$supplier=$_POST['supplier'];
$modal=$_POST['modal'];
$harga=$_POST['harga'];
$total=$_POST['jumlah'];

mysql_query("update item set nama='$name', jenis='$type',
supplier='$supplier', modal='$modal', harga='$harga',
jumlah='$total' where id_item='$id'");
header("location:item.php");

?>
```

Gambar A- 7 Kode Edit Item

```
<?php

include 'config.php';
$id=$_POST['id'];
$name_type=$_POST['nama_jenis'];

mysql_query("update jenis set nama_jenis='$name_type'
where id_jenis='$id'");
header("location:jenis.php");

?>
```

Gambar A- 8 Kode Edit Jenis Item

```
<?php
include 'config.php';
$id=$_POST['id'];
$nama_kategori=$_POST['nama_kategori'];

mysql_query("update          kategori_berita          set
nama_kategori='$nama_kategori' where id_kategori='$id'");
header("location:kategori_berita.php");

?>
```

Gambar A- 9 Kode Edit Kategori Berita

```
<?php
include 'config.php';
$id=$_GET['id'];
mysql_query("delete from berita where id_berita='$id'");
header("location:berita.php");

?>
```

Gambar A- 10 Kode Hapus Berita

```
<?php
include 'config.php';
$id=$_GET['id'];
mysql_query("delete from item where id_item='$id'");
header("location:item.php");

?>
```

Gambar A- 11 Kode Hapus Item

```
<?php
include 'config.php';
$Sid=$_GET['id'];
mysql_query("delete from member where id_member='$Sid'");
header("location:member.php");

?>
```

Gambar A- 12 Kode Hapus Member

```
<?php
include 'config.php';
$Sid=$_GET['id'];
mysql_query("delete from topik where id_topik='$Sid'");
header("location:topik.php");

?>
```

Gambar A- 13 Kode Hapus Topik

```
<?php
session_start();
include 'user/config.php';
$username=$_POST['username'];
$password=$_POST['password'];

if (empty($username) || empty($password))
    {
        header("location:login.php?pesan=gagal");
    }
else{
$sql = "SELECT * FROM member WHERE username =
'$username' AND password = '$password'";
$hasil = mysql_query($sql);
$record = mysql_fetch_array($hasil);

if($record['username'] == ""){

    header("location:login.php?pesan=gagal");
    exit();
}

if($record['username']){
    $_SESSION['username'] = $username;
    header ("location:user/index.php");

}

}
?>
```

Gambar A- 14 Kode *Login Member*


```
<?php
session_start();
include 'admin/config.php';
$username=$_POST['uname'];
$password=$_POST['pass'];
$pass=md5($password);
$query=mysql_query("select * from admin where
uname='$username' and pass='$pass'")or die(mysql_error());
if(mysql_num_rows($query)==1){
    $_SESSION['uname']=$username;
    header("location:admin/index.php");
}else{
    header("location:index.php?pesan=gagal")or
die(mysql_error());
    // mysql_error();
}
// echo $pass;
?>
```

Gambar A- 15 Kode *Login Admin*

```

<?php
include 'user/config.php';
$username=$_POST['username'];
$password=$_POST['password'];
$nama_lengkap=$_POST['nama_lengkap'];
$email=$_POST['email'];
$jenis_kelamin=$_POST['jenis_kelamin'];
$tanggal_daftar=date("Y-m-d H:i:s");

#tangkap gambar
$namafolder="../pandega/p/admin/foto/"; //folder tempat
menyimpan file
if (!empty($_FILES["foto_member"]["tmp_name"]))
{
    $jenis_gambar=$_FILES['foto_member']['type'];
    if($jenis_gambar=="image/jpeg" ||
$jenis_gambar=="image/jpg" || $jenis_gambar=="image/gif" ||
$jenis_gambar=="image/png")
    {
        $foto_member = $namafolder
        basename($_FILES['foto_member']['name']);
        if
        (move_uploaded_file($_FILES['foto_member']['tmp_name'],
        $foto_member)) {
            mysql_query("insert into member
            values('$username','$password','$nama_lengkap','$email','$je
            nis_kelamin','$foto_member','$tanggal_daftar')");
            ?>
            <script
            language="javascript">
                alert('Berhasil menambahkan');
                document.location.href="login.php";
            </script>
            <?php
        } else {
            ?>

```

```
                                <script
language="javascript">
    alert('Gagal menambahkan');
    document.location.href="daftar.php";
</script>
                                <?php
    }
} else {
    ?>
                                <script language="javascript">
    alert('Gambar harus berformat .jpg .png .gif');
    document.location.href="daftar.php";
</script>
                                <?php
    }
} else {
    echo "Gagal";
}
?>
```

Gambar A- 16 Kode Proses Daftar

```

<?php
include 'config.php';
$id = $_POST['id_topikk'];
$topik = $_POST['topik'];
$penjawab = $_POST['penjawab'];
$isi = $_POST['isi'];
$stanggal = date("Y-m-d H:i:s");

$sql = "INSERT INTO komentar VALUES ('", $id, "', '$topik',
'$penjawab', '$isi', '$stanggal')";
$hasil = mysql_query($sql);

if($hasil){
    <?>
    <script
language="javascript">
    alert('Berhasil menambahkan');
    document.location.href="moretopik.php?id=$id";
    </script>
    <?php
    }
else {
    echo "komentar gagal disimpan <br>";
    }

$query_balasan = mysql_query("SELECT id_topik FROM
komentar WHERE id_topik='$id'");
$total_balas = mysql_num_rows($query_balasan);
$total_balasan = $total_balas;

//memasukan total balasan ke database
$sql2 = "UPDATE topik SET total_balasan='$total_balasan'
WHERE id_topik='$id'";
$hasil2 = mysql_query($sql2);

if($hasil2){

```

```
        echo "ok";
    }
else {
    echo "komentar gagal disimpan <br>";
}
?>
```

Gambar A- 17 Kode Proses Tambah Komentar

```
<?php
session_start();
session_destroy();
header("location:../index.php");
?>
```

Gambar A- 18 Kode *Logout* Admin

```
<?php

$_SESSION['username'] = "";
unset($_SESSION['username']);

print "<script>alert('Logout berhasil'); location.href =
'/Pandega/user/..'; </script>";

?>
```

Gambar A- 19 Kode *Logout* Member

Halaman ini sengaja dikosongkan

LAMPIRAN B

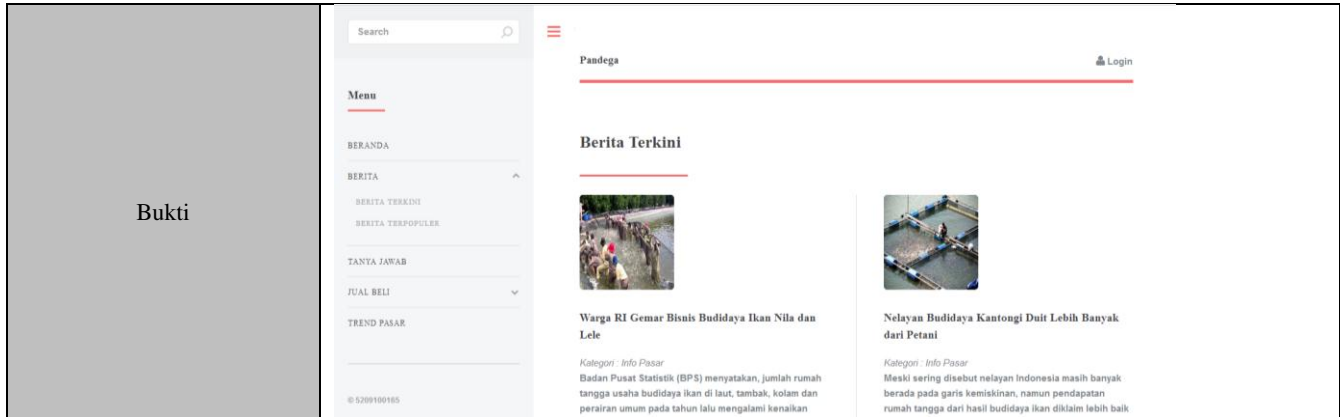
SKENARIO KEBUTUHAN FUNGSIONAL

Tabel B- 1 Requirement Traceability Matrix

ID KF	Kebutuhan Fungsional	ID UC	Use Case	ID AD	ID TC
KF01	Sistem dapat menyediakan informasi tentang budidaya perikanan	UC02	Pengguna (member dan pengunjung) dapat mengakses berita	AD01	TC02
KF02	Sistem dapat menyediakan tempat untuk melakukan tanya jawab	UC05	Member dapat menambahkan topik dan berkomentar	AD05	TC05-1 TC05-2
		UC06	Pengunjung dapat mendaftar menjadi member	AD06	TC06
KF03	Sistem dapat menyediakan tempat untuk memasarkan produk	UC08	Pengguna (member dan pengunjung) dapat melihat item jual beli	AD08	TC08
		UC09	Member dapat memesan item	AD09	TC09

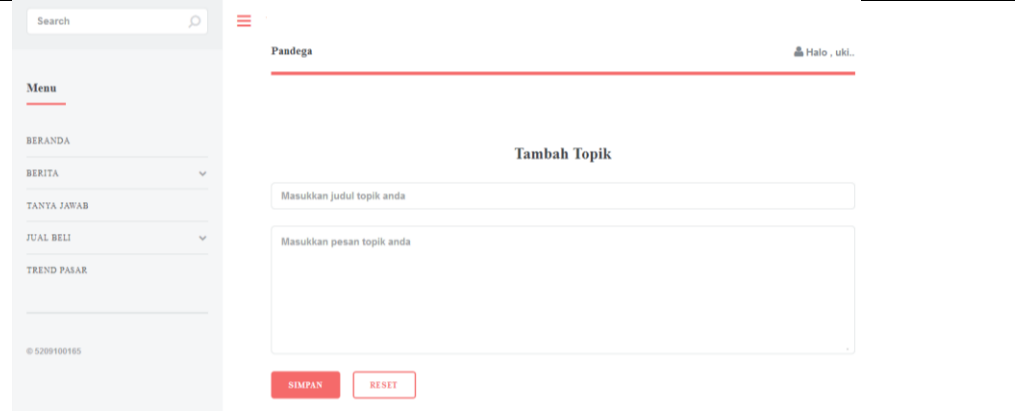
ID KF	Kebutuhan Fungsional	ID UC	Use Case	ID AD	ID TC
KF04	Sistem dapat menyediakan informasi harga pasar	UC11	Pengguna (member dan pengunjung) dapat mengakses informasi harga dan trend pasar	AD11	TC11

Tes Case Design Info		Tes Case Execution Info		
Tes Case	Usecase	Sukses/Gagal	Tanggal/Waktu	Tester
02	Pengguna (member dan pengunjung) dapat mengakses berita	sukses	21 Juni 2017 10.00 WIB	Diego Afifudin Syauqi
Objective	Pengguna mengakses berita dan informasi			
Step	<ol style="list-style-type: none"> Masuk ke url Memilih tab berita 			
Input				
Hasil Ekspektasi	<ol style="list-style-type: none"> Sistem menampilkan berita terpopuler Sistem menampilkan berita terkini 			
Hasil Testing	<ol style="list-style-type: none"> Sistem menampilkan berita terpopuler Sistem menampilkan berita terkini 			

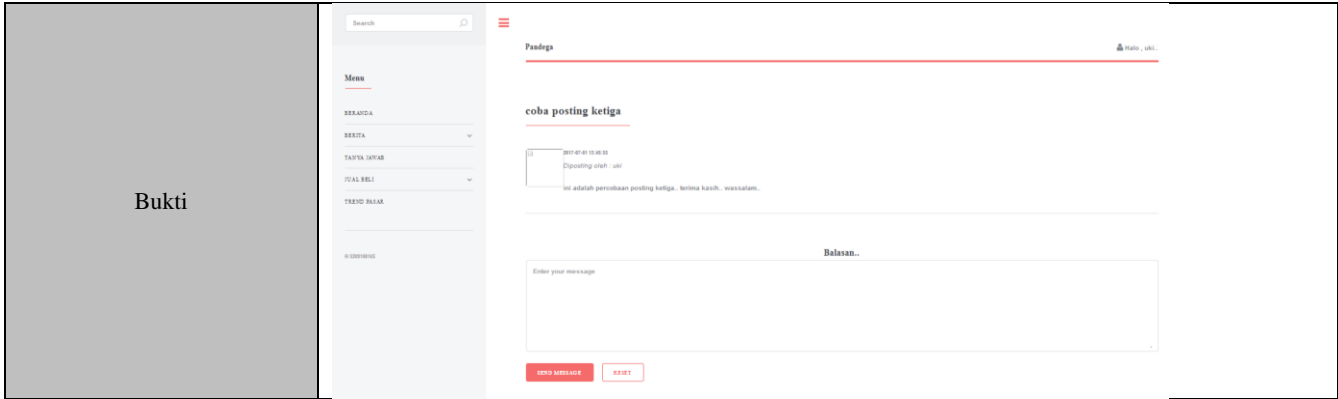


Bukti

Tes Case Design Info		Tes Case Execution Info		
Tes Case	Usecase	Sukses/Gagal	Tanggal/Waktu	Tester
05-1	Member dapat menambahkan topik	sukses	21 Juni 2017 10.00 WIB	Diego Afifudin Syauqi
Objective	Member menambahkan topik dan berkomentar			
Step	<ol style="list-style-type: none"> Login Memilih tab tanya jawab 			

	<ol style="list-style-type: none">3. Memilih tab tambah topik4. Isi form topik5. Simpan
Input	Text topik tanya jawab
Hasil Ekspektasi	<ol style="list-style-type: none">1. Sistem menngecek field kosong2. Sistem menyimpan text tanya jawab
Hasil Testing	<ol style="list-style-type: none">1. Sistem menngecek field kosong2. Sistem menyimpan text tanya jawab
Bukti	

Tes Case Design Info		Tes Case Execution Info		
Tes Case	Usecase	Sukses/Gagal	Tanggal/Waktu	Tester
05-2	Member dapat menambahkan komentar	sukses	21 Juni 2017 10.00 WIB	Diego Afifudin Syauqi
Objective	Member menambahkan topik dan berkomentar			
Step	<ol style="list-style-type: none"> 1. Login 2. Memilih tab tanya jawab 3. Memilih tab lihat komentar 4. Isi komentar 5. Simpan 			
Input	Text komentar			
Hasil Ekspektasi	<ol style="list-style-type: none"> 1. Sistem mengecek field kosong 2. Sistem menyimpan text komentar 			
Hasil Testing	<ol style="list-style-type: none"> 1. Sistem mengecek field kosong 2. Sistem menyimpan text komentar 			

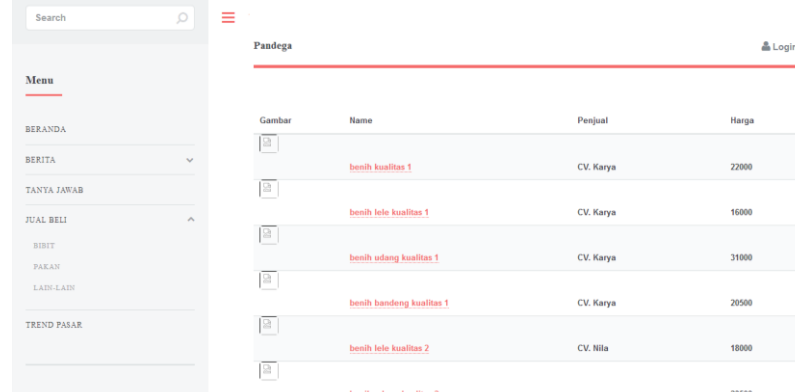


Tes Case Design Info		Tes Case Execution Info		
Tes Case	Usecase	Sukses/Gagal	Tanggal/Waktu	Tester
06	Pengunjung dapat mendaftar menjadi member	sukses	21 Juni 2017 10.00 WIB	Diego Afifudin Syauqi
Objective	Pengunjung melakukan daftar			
Step	<ol style="list-style-type: none"> Masuk ke <i>url</i> Pilih login 			

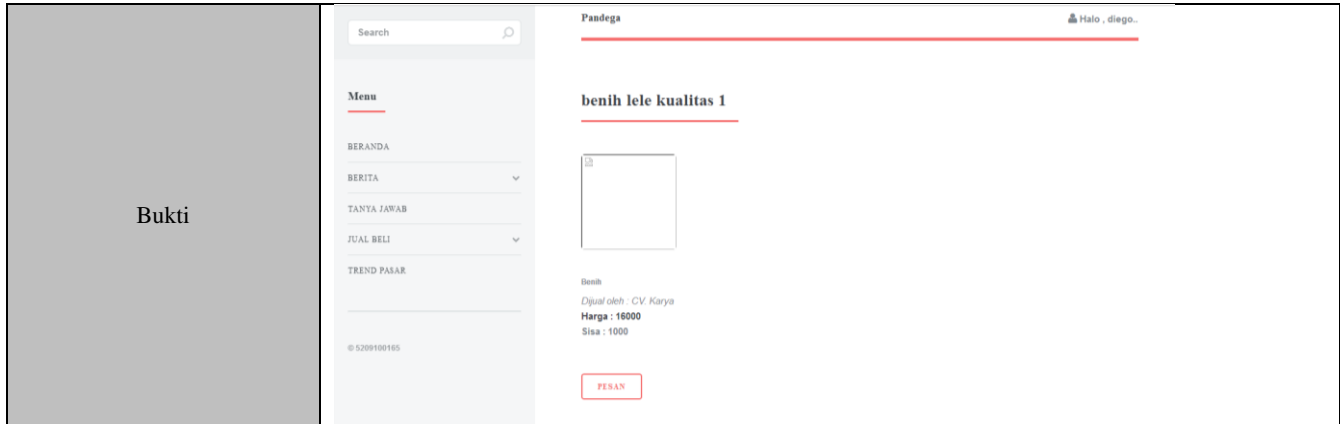
	<ol style="list-style-type: none">3. Pilih daftar4. Isi form5. Simpan
Input	
Hasil Ekspektasi	<ol style="list-style-type: none">1. Sistem mengecek field kosong2. Sistem menyimpan data pada <i>database</i>
Hasil Testing	<ol style="list-style-type: none">1. Sistem mengecek field kosong2. Sistem menyimpan data pada <i>database</i>

Bukti	<p>Daftar ..</p> <p>Nama Lengkap <input type="text"/></p> <p>Username <input type="text"/></p> <p>Password <input type="text"/></p> <p>Email <input type="text"/></p> <p>Jenis Kelamin <input type="text"/></p> <p>Foto <input type="text" value="Browse..."/> No file selected.</p> <p><input type="button" value="Reset"/> <input type="button" value="Simpan"/></p> <p><input type="button" value="⬆ Beranda"/></p>
-------	---

Tes Case Design Info		Tes Case Execution Info		
Tes Case	Usecase	Sukses/Gagal	Tanggal/Waktu	Tester
08	Pengguna (member dan pengunjung) dapat melihat item	sukses	21 Juni 2017 10.00 WIB	Diego Afifudin Syauqi

	jual beli																															
Objective	Pengguna melihat list item																															
Step	<ol style="list-style-type: none"> Masuk ke <i>url</i> Pilih tab jual beli Pilih nama item 																															
Input																																
Hasil Ekspektasi	1. Sistem menampilkan item jual beli																															
Hasil Testing	1. Sistem menampilkan item jual beli																															
Bukti	 <p>The screenshot shows a web application interface. On the left is a navigation menu with options: BERANDA, BERITA, TANYA JAWAB, JUAL BELI (highlighted), BIBIT, PAKAN, LADI LADI, and TREND PASAR. The main content area is titled 'Pandega' and features a 'Login' button. Below the title is a table of items for sale:</p> <table border="1"> <thead> <tr> <th>Gambar</th> <th>Name</th> <th>Penjual</th> <th>Harga</th> </tr> </thead> <tbody> <tr> <td></td> <td>benih kualitas 1</td> <td>CV. Karya</td> <td>22000</td> </tr> <tr> <td></td> <td>benih lele kualitas 1</td> <td>CV. Karya</td> <td>16000</td> </tr> <tr> <td></td> <td>benih udang kualitas 1</td> <td>CV. Karya</td> <td>31000</td> </tr> <tr> <td></td> <td>benih bandeng kualitas 1</td> <td>CV. Karya</td> <td>20500</td> </tr> <tr> <td></td> <td>benih lele kualitas 2</td> <td>CV. Nila</td> <td>18000</td> </tr> <tr> <td></td> <td>benih udang kualitas 2</td> <td>c</td> <td>23500</td> </tr> </tbody> </table>				Gambar	Name	Penjual	Harga		benih kualitas 1	CV. Karya	22000		benih lele kualitas 1	CV. Karya	16000		benih udang kualitas 1	CV. Karya	31000		benih bandeng kualitas 1	CV. Karya	20500		benih lele kualitas 2	CV. Nila	18000		benih udang kualitas 2	c	23500
Gambar	Name	Penjual	Harga																													
	benih kualitas 1	CV. Karya	22000																													
	benih lele kualitas 1	CV. Karya	16000																													
	benih udang kualitas 1	CV. Karya	31000																													
	benih bandeng kualitas 1	CV. Karya	20500																													
	benih lele kualitas 2	CV. Nila	18000																													
	benih udang kualitas 2	c	23500																													

Tes Case Design Info		Tes Case Execution Info		
Tes Case	Usecase	Sukses/Gagal	Tanggal/Waktu	Tester
09	Member dapat memesan item	gagal	21 Juni 2017 10.00 WIB	Diego Afifudin Syauqi
Objective	Member memesan item			
Step	<ol style="list-style-type: none"> 1. Login 2. Pilih item jual beli 3. Pilih nama item 4. Pilih pesan 			
Input				
Hasil Ekspektasi	<ol style="list-style-type: none"> 1. Sistem mencocokkan dengan data yang ada 2. Sistem menampilkan item 3. Sistem menampilkan form pemesanan 4. Sistem menyimpan form untuk di proses 			
Hasil Testing	<ol style="list-style-type: none"> 1. Sistem menampilkan item 			



Tes Case Design Info		Tes Case Execution Info		
Tes Case	Usecase	Sukses/Gagal	Tanggal/Waktu	Tester
11	Pengguna (member dan pengunjung) dapat mengakses informasi harga dan	sukses	21 Juni 2017	Diego Afifudin Syauqi
			10.00 WIB	

	trend pasar			
Objective	Pengguna melihat trend pasar			
Step	<ol style="list-style-type: none"> 1. Masuk ke <i>url</i> 2. Pilih tab trend pasar 			
Input				
Hasil Ekspektasi	<ol style="list-style-type: none"> 1. Sistem menampilkan grafik trend pasar 2. Sistem menampilkan grafik trend tiap komoditas 			
Hasil Testing	<ol style="list-style-type: none"> 1. Sistem menampilkan grafik trend pasar 			

Bukti

Search



Menu

BERANDA

BERITA

TANYA JAWAB

JUAL BELI

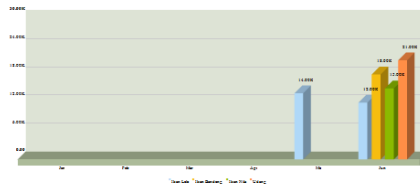
TREND PASAR

© 2020100165

Pandega

Login

Trend Harga Ikan



Harga Rata-Rata per Bulan	Jan	Feb	Mar	Apr	Mei	Jun
Ikan Lele					14000	12000
Ikan Bandeng						18000
Ikan Nila						15000
Udang						21000

Halaman ini sengaja dikosongkan

BIODATA PENULIS



Penulis lahir di Gresik pada tanggal 08 Juni 1990. Penulis merupakan anak terakhir dari tiga bersaudara. Penulis menempuh pendidikan formal di SDN Sidokumpul 1 Gresik, SMPN 3 Gresik, dan SMAN 1 Gresik. Pada tahun 2009 penulis mengikuti SNMPTN tulis dan diterima di Jurusan Sistem Informasi Institut Teknologi Sepuluh Nopember (ITS) Surabaya. Sebagai mahasiswa, penulis juga mengikuti kegiatan non akademik maupun organisasi. Penulis pernah aktif menjadi anggota tim futsal jurusan, fakultas, dan ukm futsal ITS. Selain organisasi, penulis pernah aktif dalam kepanitiaan. Penulis juga pernah menjalani kerja praktik di PT. PLN APJ Gresik selama 1,5 bulan pada tahun 2012. Pada akhir semester di Jurusan Sistem Informasi, penulis mengambil bidang minat Manajemen Sistem Informasi (MSI) untuk Tugas Akhir. Untuk kepentingan penelitian, penulis juga dapat dihubungi melalui email afifudin.syauqi@gmail.com.