

**SISTEM INFORMASI PENGIKLANAN MAKANAN
BERBASIS *WEBSITE***

LAPORAN AKHIR



sebagai salah satu syarat untuk memperoleh gelar Ahli Madya (A.Md)
di Program Studi Manajemen Informatika
Jurusan Teknologi Informasi

oleh
Nurinda Febriani
E31141172

**PROGRAM STUDI MANAJEMEN INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI JEMBER
2017**

**SISTEM INFORMASI PENGIKLANAN MAKANAN
BERBASIS *WEBSITE***

LAPORAN AKHIR



sebagai salah satu syarat untuk memperoleh gelar Ahli Madya (A.Md)
di Program Studi Manajemen Informatika
Jurusan Teknologi Informasi

oleh
Nurinda Febriani
E31141172

**PROGRAM STUDI MANAJEMEN INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI JEMBER
2017**

**KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
POLITEKNIK NEGERI JEMBER**

**SISTEM INFORMASI PENGIKLANAN MAKANAN
BERBASIS WEBSITE**

Telah diuji pada tanggal 09 Agustus 2017
Telah dinyatakan Memenuhi Syarat

HALAMAN PENGESAHAN

Tim Penguji:
Ketua,



I Putu Dody Lesmana, ST, MT
NIP. 19790921200501 1 001

Sekretaris,



Syamsul Arifin, S.Kom, M.Cs
NIP. 19810615 200604 1 002

Anggota,



Taufiq Rizaldi, S.ST, MT
NIP. 19890329 201503 1 001

Mengesahkan
Ketua Jurusan Teknologi Informasi,



Wahyu Kurnia Dewanto, S.Kom, MT
NIP. 19770408 200112 1 003

**KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
POLITEKNIK NEGERI JEMBER**

**.. SISTEM INFORMASI PENGIKLANAN MAKANAN
BERBASIS WEBSITE**

**Oleh:
Nurinda Febriani
E31141172**

Di uji pada tanggal: 09 Agustus 2017

Pembimbing I,



**I Putu Dody Lesmana, S.T, M.T
NIP. 19790921200501 1 001**

Pembimbing II,



**Svamsul Arifin, S.Kom, M.Cs
NIP. 19810615 200604 1 002**

**Mengesahkan
Ketera Jurusan Teknologi Informasi,**



**Wahyu Kurnia Dewanto, S.Kom, M.T
NIP. 19710408 200112 1 003**

SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Nurinda Febriani

NIM : E31141172

Menyatakan dengan sebenar-benarnya bahwa segala pernyataan dalam laporan akhir saya yang berjudul “SISTEM INFORMASI PENGIKLANAN MAKANAN BERBASIS WEBSITE” merupakan gagasan dan hasil karya saya sendiri dengan arahan komisi pembimbing, dan belum pernah diajukan dalam bentuk apa pun pada perguruan tinggi mana pun.

Semua data dan informasi yang digunakan telah dinyatakan secara jelas dan dapat diperiksa kebenarannya. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan dari penulis lain telah disebutkan dalam naskah dan dicantumkan dalam daftar pustaka di bagian akhir Laporan Akhir ini.

Jember, 25 Agustus 2017

Nurinda Febriani

NIM E31141172

HALAMAN PERSEMBAHAN

Dengan mengucapkan syukur Alhamdulillah, karya sederhana ini teruntuk orang-orang terkasih :

- Ibu dan Ayah tersayang, orang yang telah berjuang demi keluarga. Terima kasih selalu mendoakan yang terbaik untuk anak – anaknya, sehingga saya dapat menyelesaikan tugas akhir ini..
- Bapak I Putu Dody Lesmana,S.T, M.T dan Bapak Syamsul Arifin, S.Kom,M.Cs terima kasih untuk bimbingan dan semangatnya selama penyelesaian tugas akhir ini.
- Teman – teman SEMPROSONIC yang selalu ada di saat keadaan apapun, terimakasih atas dukungan, hiburan dan semoga persahabatan kita selalu terjalin sampai nanti.
- Teman - teman seperjuangan MIF 2014, terima kasih untuk kekompakan, bantuan, dan kerja sama kalian. Terima kasih untuk waktunya selama menempuh studi di Politeknik Negeri Jember. Kalian luar biasa.

----- TerimaKasih -----

Almamaterku Tercinta

MOTTO

“ Hidup adalah seperti cermin, tersenyumlah maka kehidupan akan tersenyum kepadamu ” (Nurinda Febriani)

“Sesuatu yang belum dikerjakan seringkali tampak mustahil, kita baru yakin jika kita telah berhasil melakukannya” (Evelyn Underhill)

“Banyak kegagalan dalam hidup ini dikarenakan orang-orang tidak menyadari betapa dekatnya mereka dengan keberhasilan saat mereka menyerah”
(Thomas Alva Edison)

ABSTRAK

Perkembangan bisnis kuliner di Indonesia semakin pesat dan ramai dengan banyaknya *outlet-outlet* makanan baru. Akan tetapi masyarakat masih kesulitan mencari referensi makanan yang diinginkan karena media pemasaran yang digunakan pengusaha makanan masih menggunakan brosur dan pamflet dan media sosial yang bersifat umum dan kurang spesifik. Bahkan terjadi ketidaksesuaian informasi yang diterima oleh masyarakat dengan kenyataan di lapangan. Sehingga kami membangun sebuah sistem untuk menyelesaikan masalah ini. Sebuah sistem yang menyediakan media untuk mengiklankan produk-produk tersebut dalam satu wadah yang berbasis *website*. Keuntungan dari sistem ini adalah masyarakat semakin mudah untuk mencari referensi kuliner yang ada dan pihak produsen pun dimudahkan dengan proses pengiklanan yang mudah dan hemat waktu.

Kata Kunci : Website, Pengiklanan, Makanan

ABSTRACT

The development of culinary business in Indonesia increasingly rapid and crowded with the number of new food outlets. However, people are still difficult to find the desired food reference because the marketing media used by food entrepreneurs still use brochures, pamphlets and social media that are general and less specific. Even the inconsistency of information received by the community with the reality in the field. So we built a system to solve this problem. A system that provides media to advertise these products in one website-based container. The advantage of this system is that it is easier for people to search for existing culinary references and the producers are facilitated with an easy and time-saving advertising process.

Keywords: Website, Advertising, Food

RINGKASAN

SISTEM INFORMASI PENGIKLANAN MAKANAN BERBASIS *WEBSITE*, Nurinda Febriani, NIM E31141172, Tahun 2017, Teknologi Informasi, Politeknik Negeri Jember, I Putu Dody Lesmana, S.T, M.T, (Pembimbing I), dan Syamsul Arifin, S.Kom, M.Cs (Pembimbing II).

Perkembangan bisnis kuliner di Indonesia semakin pesat dan ramai dengan banyaknya *outlet-outlet* makanan baru. Pengusaha kuliner pun juga dituntut untuk bisa lebih kreatif dan inovatif dalam memasarkan produk kuliner yang dimilikinya. Animo masyarakat akan perkembangan dunia kuliner juga turut mempengaruhi apresiasi terhadap munculnya inovasi masakan-masakan baru, akan tetapi masih saja ada yang mengalami kesulitan dalam mencari kuliner yang diinginkan seperti terjadinya ketidaksesuaian antara informasi yang mereka dapatkan.

Media pengiklanan yang digunakan oleh produsen makanan tersebut masih berupa brosur, pamflet dan melalui media sosial yang penggunaannya kurang maksimal karena informasi yang disediakan kurang lengkap dan jelas. Sehingga kami membangun sebuah sistem untuk menyelesaikan masalah ini. Sebuah sistem yang menyediakan media untuk mengiklankan produk-produk tersebut dalam satu wadah yang berbasis *website*. Keuntungan dari sistem ini adalah masyarakat semakin mudah untuk mencari referensi kuliner yang ada dan pihak produsen pun dimudahkan dengan proses pengiklanan yang mudah dan hemat waktu.

PRAKATA

Puji syukur penulis panjatkan ke hadirat Allah SWT atas berkat rahmat dan karunia-Nya, maka penulisan karya tulis ilmiah berjudul Sistem Informasi Pengiklanan Makanan Berbasis *Website* dapat diselesaikan dengan baik.

Tulisan ini adalah laporan hasil penelitian yang dilaksanakan mulai bulan Juli 2016 sampai dengan Agustus 2017 bertempat di Politeknik Negeri Jember, yang dilakukan sebagai salah satu syarat untuk memperoleh gelar Ahli Madya (A.Md) di Program Studi Manajemen Informatika Jurusan Teknologi Informasi Politeknik Negeri Jember.

Penulis menyampaikan penghargaan dan ucapan terima kasih yang sebesar-besarnya kepada:

1. Direktur Politeknik Negeri Jember,
2. Ketua Jurusan Teknologi Informasi,
3. Ketua Program Studi Manajemen Informatika,
4. Bapak I Putu Dody Lesmana, S.T, M.T, selaku Pembimbing I,
5. Bapak Syamsul Arifin, S.Kom, M.Cs, selaku Pembimbing II,
6. Rekan-rekanku dan semua pihak yang telah ikut membantu dalam pelaksanaan penelitian dan penulisan laporan ini

Laporan Karya Tulis Ilmiah ini masih kurang sempurna, mengharapkan kritik dan saran yang sifatnya membangun guna perbaikan di masa mendatang. Semoga tulisan ini bermanfaat.

Jember, 25 Agustus 2017

Penulis

DAFTAR ISI

	Halaman
JUDUL	ii
HALAMAN PENGESAHAN	iii
SURAT PERNYATAAN	v
HALAMAN PERSEMBAHAN	vi
HALAMAN MOTTO	vii
ABSTRAK	viii
RINGKASAN	x
PRAKATA	xi
DAFTAR ISI.....	xii
DAFTAR TABEL	xv
DAFTAR GAMBAR.....	xvii
DAFTAR KODE PROGRAM	xix
PERSETUJUAN PUBLIKASI.....	xxi
BAB 1. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah.....	3
1.4 Tujuan	3
1.5 Manfaat	3
BAB 2. TINJAUAN PUSTAKA.....	4
2.1 Sistem Informasi	4
2.2 Pengiklanan.....	5
2.2.1 Pengertian Periklanan.....	5
2.2.2 Tujuan dan Sasaran Periklanan.....	6

2.3	Karya Ilmiah Yang Mendahului.....	7
2.3.1	Media Informasi dan Periklanan Pasar Tradisional Berbasis WEB Study Kasus Pasar Tradisional Kabupaten Bandung (Rizky Rahmatullah, Universitas Telkom).....	7
2.3.2	Analisis Dan Perancangan Sistem Informasi Berbasis Web Sebagai Media Promosi Dan Informasi Pada Budi Watch Collections (Adi Dwi Setiawan, Amikom Yogyakarta).....	8
2.4	The State of The Art.....	8
BAB 3. METODE KEGIATAN		10
3.1	Waktu dan Tempat	10
3.2	Alat dan Bahan	10
3.2.1	Alat	10
3.2.2	Bahan	11
3.3	Metodologi.....	11
3.4	Pelaksanaan Kegiatan.....	12
BAB 4. HASIL DAN PEMBAHASAN		14
4.1	<i>Communication</i>	14
4.2	<i>Quick Plan</i>	14
4.3	<i>Modeling Quick Design</i>	14
4.3.1	Definisi Aktor	16
4.3.2	Definisi Use Case	16
4.3.3	Skenario <i>Use Case</i>	18
4.4	Pembuatan Aplikasi	31
4.4.1	Iterasi 1	31
4.4.2	Iterasi 2	41
4.4.3	Iterasi 3	56
4.4.4	Iterasi 4	63
4.5	Hasil Kuesioner.....	72

BAB 5. KESIMPULAN DAN SARAN.....	74
5.1 Kesimpulan	74
5.2 Saran.....	74
DAFTAR PUSTAKA	75

DAFTAR TABEL

Tabel 2.1 Tabel State of The Art.....	9
Tabel 4.1 Definisi Aktor.....	16
Tabel 4.2 Definisi Usecase.....	16
Tabel 4.3 Skenario Use Case Menambah Kategori.....	18
Tabel 4.4 Skenario Use Case Mengubah Kategori.....	19
Tabel 4.5 Skenario Use Case Menghapus Kategori.....	20
Tabel 4.6 Skenario Use Case Melihat Kategori.....	20
Tabel 4.7 Skenario Use Case Mengubah Status Iklan.....	21
Tabel 4.8 Skenario Daftar dan Verifikasi Daftar.....	22
Tabel 4.9 Skenario Use Case Login.....	23
Tabel 4.10 Skenario Use Case Logout.....	24
Tabel 4.11 Skenario Use Case Verifikasi Login User.....	24
Tabel 4.12 Skenario Melihat Akun.....	25
Tabel 4.13 Skenario Use Case Mengubah Akun.....	25
Tabel 4.14 Skenario Use Case Menambah Brand.....	26
Tabel 4.15 Skenario Menghapus Brand.....	26
Tabel 4.16 Skenario Mengubah Brand.....	27
Tabel 4.17 Skenario Melihat Brand.....	27
Tabel 4.18 Skenario Use Case Menambah Iklan/Produk.....	28

Tabel 4.19 Skenario Menghapus Iklan.....	28
Tabel 4.20 Skenario Mengubah Iklan	29
Tabel 4.21 Skenario Melihat Iklan	30
Tabel 4.22 Skenario Melihat dan Menggunakan Filter Iklan	30
Tabel 4.23 Tabel Kategori.....	33
Tabel 4.24 Tabel <i>Feedback</i> Kategori.....	39
Tabel 4.25 <i>Feedback</i> Status Iklan.....	40
Tabel 4.26 Tabel Brand.....	44
Tabel 4.27 <i>Feedback</i> Produk	54
Tabel 4.28 <i>Feedback</i> Brand	55
Tabel 4.29 Tabel User	58
Tabel 4.30 <i>Feedback</i> Login.....	61
Tabel 4.31 <i>Feedback</i> Register.....	62

DAFTAR GAMBAR

Gambar 3.1 Model <i>Prototyping</i> menurut Pressman.....	11
Gambar 4.1 <i>Usecase</i>	15
Gambar 4.2 class diagram mengelola kategori.....	32
Gambar 4.3 Mock Up Halaman Kategori.....	32
Gambar 4.4 Mock Up Halaman Produk.....	33
Gambar 4.5 Halaman Kategori.....	38
Gambar 4.6 Halaman Produk Administrator.....	40
Gambar 4.7 <i>Class Diagram</i> Produk.....	41
Gambar 4.8 <i>Class Diagram</i> Brand.....	42
Gambar 4.9 <i>Mockup</i> Halaman Produk Produsen.....	43
Gambar 4.10 <i>mockup</i> Brand.....	44
Gambar 4.11 Halaman Produk untuk Produsen.....	53
Gambar 4.12 Halaman Brand untuk Produsen.....	55
Gambar 4.13 <i>Class Diagram</i> Login.....	56
Gambar 4.14 <i>Mockup</i> Halaman Login.....	57
Gambar 4.15 <i>Mockup</i> Halaman Register.....	57
Gambar 4.16 Halaman Login	61
Gambar 4.17 Halaman Register.....	62
Gambar 4.18 <i>Class Diagram</i> Melihat Iklan.....	63
Gambar 4.19 <i>Mockup</i> Halaman Utama.....	64
Gambar 4.20 <i>Mockup</i> Halaman Detail Produk.....	64
Gambar 4.21 <i>Mockup</i> Halaman Brand.....	65
Gambar 4.22 Halaman Utama.....	69

Gambar 4.23 Tampilan Diskon.....	70
Gambar 4.24 Tampilan Filter Kategori.....	70
Gambar 4.25 Tampilan Detail Produk.....	71
Gambar 4.26 Halaman Brand untuk Konsumen.....	71
Gambar 4.27 Tampilan Lokasi.....	72
Gambar 4.28 Grafik Hasil Pengujian Sistem Informasi.....	72

DAFTAR KODE PROGRAM

	Halaman
Kode Program 4.1 Fungsi Index Kategori	34
Kode Program 4.2 Fungsi getCateroies	34
Kode Program 4.3 Fungsi Tambah Kategori	34
Kode Program 4.4 Fungsi Tambah Kategori pada Model	35
Kode Program 4.5 Fungsi Edit Kategori.....	35
Kode Program 4.6 Fungsi Prosesedit Kategori	35
Kode Program 4.7 Fungsi Edit Kategori Pada Model	36
Kode Program 4.8 Fungsi Hapus Kategori	36
Kode Program 4.9 Fungsi Hapus Kategori Pada Model.....	36
Kode Program 4.10 Fungsi Edit Status	37
Kode Program 4.11 Fungsi Edit Status Pasa Model	37
Kode Program 4.12 Fungsi Index Halaman Produk	45
Kode Program 4.13 Fungsi GetProduct	46
Kode Program 4.14 Fungsi Tambah Produk.....	47
Kode Program 4.15 Fungsi Tambah Produk pada Model.....	47
Kode Program 4.16 Fungsi Ubah Produk	48
Kode Program 4.17 Fungsi Ubah Produk pada Model	48
Kode Program 4.18 Fungsi Hapus Produk.....	49
Kode Program 4.19 Fungsi Hapus Produk pada Model.....	49
Kode Program 4.20 Fungsi Index Halaman Brand	49
Kode Program 4.21 Fungsi Tambah Brand	50
Kode Program 4.22 Fungsi Tambah Brand pada Model.....	51
Kode Program 4.23 Fungsi Hapus Brand	51
Kode Program 4.24 Fungsi Hapus Brand pada Model	52
Kode Program 4.25 Fungsi Hapus Brand	52
Kode Program 4.26 Fungsi Hapus Brand pada Model	52
Kode Program 4.27 Fungsi Check Login.....	58

Kode Program 4.28 Fungsi <i>Login</i>	59
Kode Program 4.29 Fungsi Register	60
Kode Program 4.30 Fungsi Tambah User.....	60
Kode Program 4.31 Fungsi Index Halaman Utama	66
Kode Program 4.32 Fungsi <i>getProductById</i>	67
Kode Program 4.33 Fungsi <i>getProductByPermalink</i>	68
Kode Program 4.34 Fungsi Detail Brand	68
Kode Program 4.35 Fungsi <i>getProductByBrand</i>	68



**PERNYATAAN
PERSETUJUAN PUBLIKASI
KARYA ILMIAH UNTUK KEPENTINGAN
AKADEMIS**

Yang bertanda tangan di bawah ini, saya:

Nama : Nurinda Febriani
NIM : E31141172
Program Studi : Manajemen Informatika
Jurusan : Teknologi Informasi

Demi pengembangan Ilmu Pengetahuan, saya menyetujui untuk memberikan kepada UPT. Perpustakaan Politeknik Negeri Jember, Hak Bebas Royalti Non-Eksklusif (*Non-Exclusive Royalty Free Right*) atas Karya Ilmiah berupa **Laporan Tugas Akhir yang berjudul:**

**SISTEM INFORMASI PENGIKLANAN MAKANAN
BERBASIS WEBSITE**

Dengan Hak Bebas Royalti Non-Eksklusif ini UPT. Perpustakaan Politeknik Negeri Jember berhak menyimpan, mengalihkan media atau format, mengelola dalam bentuk Pangkalan Data (Database), mendistribusikan karya dan menampilkan atau mempublikasikannya di internet atau media lain untuk kepentingan akademis tanpa perlu meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis atau pencipta.

Saya bersedia untuk menanggung secara pribadi tanpa melibatkan pihak Politeknik Negeri Jember, segala bentuk tuntutan hukum yang timbul atas Pelanggaran Hak Cipta dalam Karya Ilmiah ini.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Jember
Pada Tanggal : 25 Agustus 2017

Yang menyatakan,

Nama : Nurinda Febriani
NIM : E31141172

BAB 1. PENDAHULUAN

1.1 Latar Belakang

Perkembangan bisnis kuliner di Indonesia semakin pesat dan ramai. Hal ini ditandai dengan banyaknya *outlet-outlet* makanan baru yang menyediakan menu-menu unik dan berbeda. Pengusaha kuliner pun juga dituntut untuk bisa lebih kreatif dan inovatif dalam memasarkan produk kuliner yang dimilikinya karena persaingan yang semakin ketat. Animo masyarakat akan perkembangan dunia kuliner juga turut mempengaruhi apresiasi terhadap munculnya inovasi masakan-masakan baru.

Akan tetapi dibalik tingginya animo masyarakat, masih saja ada yang mengalami kesulitan dalam mencari kuliner yang diinginkan. Bahkan sering terjadi ketidaksesuaian antara informasi yang mereka dapatkan dengan kenyataan di lapangan. Hal terjadi karena pengusaha kuliner tidak memasarkan produk andalannya dengan baik. Cara yang masih digunakan oleh makanan untuk pemasaran dinilai kurang efektif karena masih menggunakan pamflet ataupun brosur. Penggunaan pamflet dan brosur merugikan pengusaha makanan karena biaya pembuatannya yang mahal dan penyebarannya yang tidak dapat menjangkau segmen konsumen. Sehingga produk makanannya kurang dikenal dan berdampak pada kerugian hingga penutupan *outlet-outlet* makanan tersebut.

Untuk saat ini pengusaha makanan dimudahkan dengan canggihnya teknologi internet. Pengusaha makanan dapat memanfaatkan kemajuan teknologi yang telah ada seperti media sosial *Instagram* yang menjadi salah satu media andalan untuk melakukan komunikasi dan bisnis di dalam pemasarannya. Akan tetapi, meskipun terdapat beberapa produsen yang sudah mulai memanfaatkan media sosial sebagai media pemasarannya, namun penggunaannya kurang maksimal karena tidak menyediakan informasi yang jelas dan lengkap. Seperti hanya menyediakan informasi lokasi dan saja tanpa mengikutsertakan harga. Selain itu, dengan media sosial yang sifatnya umum, pihak konsumen pun kesulitan untuk mencari referensi *restaurant* ataupun menu-menu yang diinginkan

karena akun-akun yang menjadi media pengiklanan makanan tersebut tidak mengelompokkan menu-menu makanan yang berdasarkan brand atau mereknya.

Penelitian dan pembangunan sebuah sistem periklanan online telah banyak dilakukan sebelumnya seperti Rizky Rahmatullah yang membangun Media Informasi dan Periklanan Pasar Tradisional Berbasis WEB Study Kasus Pasar Tradisional Kabupaten Bandung (2016) dan Adi Dwi Setiawan pada Analisis Dan Perancangan Sistem Informasi Berbasis Web Sebagai Media Promosi Dan Informasi Pada Budi Watch Collections (2010).

Dari berbagai penelitian tersebut maka solusi untuk mengatasi masalah yang disebutkan diatas, dibangunlah sebuah sistem informasi yang menyediakan media untuk pengiklanan khusus makanan berbasis *website* sebagai salah satu fasilitas yang bisa kita gunakan di jaringan internet dimana penggunaannya bisa dilihat atau dibaca oleh siapapun dan kapanpun dan dari tempat manapun. Sebuah sistem yang dapat membantu masyarakat untuk mencari referensi makanan dengan mudah dan membantu pengusaha makanan untuk memasarkan produk-produknya sehingga bisa dengan mudah dijangkau oleh masyarakat.

1.2 Rumusan Masalah

Berdasarkan latar belakang permasalahan yang telah di sebutkan pada subbab sebelumnya, maka rumusan permasalahannya yaitu :

- a. Bagaimana membuat sebuah media pengiklanan makanan berbasis *website* yang dapat memudahkan pengguna mencari referensi makanan yang diinginkan ?
- b. Bagaimana membuat sebuah media pengiklanan berbasis *website* makanan yang bisa digunakan oleh beberapa produsen makanan yang menyediakan informasi dengan jelas dan lengkap ?

1.3 Batasan Masalah

Batasan masalah dalam pengembangan sistem informasi ini adalah :

- a. Sistem informasi ini untuk usaha di bidang makanan
- b. Sistem informasi ini dilengkapi dengan fitur peta lokasi
- c. Terdapat informasi tentang harga makanan, brand/merek dari makanan, alamat dan juga informasi diskon

1.4 Tujuan

Adapun tujuan yang ingin dicapai adalah membangun Sistem Informasi Pengiklanan Makanan Berbasis Website yang menyediakan media untuk pengiklanan yang dilengkapi dengan fitur untuk mengetahui tempat makan yang dicari berdasarkan filter yang digunakan dengan informasi yang lengkap dan jelas dan bisa digunakan oleh beberapa pengusaha makanan.

1.5 Manfaat

Adapun manfaat yang diharapkan yaitu :

- a. Memudahkan masyarakat dalam mencari dan menentukan makanan dan lokasi tempat makan yang akan dituju
- b. Memudahkan pengusaha makanan untuk mengiklankan produknya

BAB 2. TINJAUAN PUSTAKA

2.1 Sistem Informasi

Pengertian sistem menurut Benner (2010) bahwa sistem merupakan suatu kumpulan yang kompleks dan juga saling berinteraksi satu sama lainnya apabila kumpulan tersebut digabungkan menjadi satu kesatuan yang utuh. Sistem ini menggambarkan suatu kejadian – kejadian dan kesatuan yang nyata adalah suatu objek nyata, seperti tempat, benda, dan orang – orang yang betul – betul ada dan terjadi. Dan informasi menurut Laudon (2010) adalah sebuah data yang telah dibuat dan juga diolah dengan metode tertentu ke dalam bentuk yang nantinya akan memiliki arti tertentu bagi manusia sebagai penggunaanya. Jadi Laudon (2010) mengatakan bahwa yang dimaksud dengan sistem informasi adalah suatu komponen yang saling bekerja satu sama lain untuk mengumpulkan, mengolah, menyimpan dan juga menyebarkan informasi untuk mendukung kegiatan suatu organisasi, seperti pengambilan keputusan, koordinasi, pengendalian, analisis masalah, dan juga visualisasi dari organisasi.

Sistem informasi terdiri dari komponen – komponen yang disebut dengan istilah blok bangunan (*building block*) yaitu :

a. Blok masukan (*input block*)

Masukan mewakili data yang masuk ke dalam sistem informasi. Masukan disini termasuk metode – metode dan media yang digunakan untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen dasar.

b. Blok model (*model block*)

Blok ini terdiri dari kombinasi prosedur, logika dan metode matematik yang akan memanipulasi data masukan dan data yang tersimpan di basisdata dengan cara yang sudah tertentu untuk menghasilkan keluaran yang sudah diinginkan.

c. Blok keluaran (*output block*)

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

d. Blok teknologi (*technology block*)

Teknologi digunakan untuk menerima masukan, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian diri secara keseluruhan. Teknologi terdiri dari unsur utama:

- 1) Teknisi (*human ware* atau *brain ware*)
- 2) Perangkat lunak (*software*)
- 3) Perangkat keras (*hardware*)

e. Blok basisdata (*database block*)

Merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya.

f. Blok kendali (*control block*)

Banyak faktor yang dapat merusak sistem informasi, misalnya bencana alam, kegagalan sistem, kesalahan ketidakefisienan, sabotase dan sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal – hal yang dapat merusak sistem dapat dicegah atau bila terlanjur terjadi maka kesalahan dapat langsung diatasi.

2.2 Pengiklanan

2.2.1 Pengertian Periklanan

Menurut Basu Swastha (2011:205) iklan adalah salah satu komponen marketing mix yang umum dilakukan oleh perusahaan. Iklan adalah setiap bentuk komunikasi yang dimaksudkan untuk memotivasi seseorang pembeli potensial dan mempromosikan penjual suatu produk atau jasa, untuk mempengaruhi pendapat publik, memenangkan dukungan publik untuk berpikir atau bertindak

sesuai dengan keinginan si pemasang iklan. Selain itu, semua iklan dibuat dengan tujuan yang sama yaitu untuk memberi informasi dan membujuk para konsumen untuk mencoba atau mengikuti apa yang ada di iklan tersebut, dapat berupa aktivitas mengkonsumsi produk dan jasa yang ditawarkan

2.2.2 Tujuan dan Sasaran Periklanan

Menurut Lee dan Johnson (2007:3) periklanan adalah komunikasi komersil dan non-personal tentang sebuah organisasi dan produk - produknya yang ditransmisikan ke suatu khalayak target melalui media bersifat massal seperti televisi, radio, koran, majalah, *direct mail* (pengeposan langsung), reklame luar ruang, atau kendaraan umum. Iklan informatif bertujuan menciptakan kesadaran merek dan pengetahuan tentang produk atau fitur baru produk yang ada. Iklan persuasif bertujuan menciptakan kesukaan, preferensi, keyakinan, dan membeli produk atau jasa.

Adapun fungsi dan tujuan utama dari periklanan yaitu :

- a. *Informing* (menginformasikan)
melalui iklan konsumen memperoleh informasi mengenai keberadaan produk atau jasa, memberikan informasi mengenai manfaat dari produk atau jasa tersebut dan memberikan informasi mengenai merek-merek tertentu.
- b. *Persuading* (membujuk)
iklan yang efektif akan mampu membujuk konsumen untuk mencoba produk atau jasa yang diiklankan, terkadang persuasi dapat berbentuk mempengaruhi permintaan primer yaitu menciptakan permintaan bagi keseluruhan kategori produk. Selain itu iklan berupaya membangun permintaan sekunder yaitu permintaan bagi merek perusahaan yang spesifik.
- c. *Reminding* (mengingat)
iklan menjaga agar merek perusahaan tetap segar dalam ingatan para konsumen. Saat kebutuhan muncul, yang berhubungan dengan produk yang diiklankan, maka pengaruh dari iklan-iklan yang telah mereka saksikan sebelumnya memungkinkan para pengiklan merek-merek tersebut untuk hadir dibenak konsumen sebagai kandidat merek yang akan dibeli.

d. *Adding value* (menambah nilai)

ada tiga cara dasar agar perusahaan bisa memberikan nilai tambah bagi produk mereka, yaitu melakukan inovasi, meningkatkan kualitas ataupun mengubah persepsi konsumen. Iklan yang efektif menyebabkan merek dipandang lebih elegan, lebih bergaya, lebih bergengsi, dan lebih unggul dibandingkan dengan pesaing.

2.3 Karya Ilmiah Yang Mendahului

2.3.1 Media Informasi dan Periklanan Pasar Tradisional Berbasis WEB Study Kasus Pasar Tradisional Kabupaten Bandung (Rizky Rahmatullah, Universitas Telkom)

Sistem informasi semakin berkembang dan sangat memudahkan orang-orang dalam mencari informasi, hanya memasukan *keyword* di *search engine* maka informasi yang dibutuhkan akan tampil pada layar dan tempat yang banyak digunakan dalam mencari informasi *online* adalah *website*. Salah satu informasi yang dibutuhkan oleh masyarakat adalah informasi mengenai Pasar Tradisional baik harga jual bahan pokok maupun harga toko yang akan dijual atau disewakan.

Hasil dari perancangannya adalah masyarakat dapat mengetahui informasi dan periklanan mengenai pasar tradisional seperti harga bahan pokok setiap harinya. Harga jual beli/sewa kios atau barang dan bahan di pasar tradisional. Kemudian iklan yang ditampilkan dapat ditempatkan sesuai kategori yang sudah disediakan seperti buah-buahan atau sayur-sayuran. Dan masyarakat dapat mengikuti perkembangan zaman dimana teknologi informasi semakin berkembang.

2.3.2 Analisis Dan Perancangan Sistem Informasi Berbasis Web Sebagai Media Promosi Dan Informasi Pada Budi Watch Collections (Adi Dwi Setiawan, Amikom Yogyakarta)

Budi Watch Collections merupakan sebuah perusahaan yang bergerak dalam bidang penjualan dan servis jam. Produk yang ditawarkan bervariasi diantaranya jam dengan beberapa merek, seperti jam rolex, seiko, casio, romanson, dan lain sebagainya. Dari segi harganya pun relatif lebih murah dari perusahaan sejenisnya.

Belum adanya promosi yang dilakukan oleh Budi Watch Collections, baik dalam promosi periklanan televisi, radio, atau website membuat perusahaan ini kurang begitu dikenal dan dikunjungi oleh banyak user. Hal ini tentu akan mengakibatkan dampak yang tidak baik bagi perusahaan. Oleh karena itu, hal tersebut perlu di atasi dengan melakukan promosi dan penyediaan informasi dengan menggunakan media informasi berbasis website.

2.4 The State of The Art

State of The Art dimaksudkan untuk menganalisis penelitian sebelumnya yang pernah ada, yang sejalan dan mempunyai konsep yang hampir sama dengan penelitian saat ini. Kemudian melihat sejauh mana perbedaan masing - masing penelitian, sehingga masing - masing penelitian mempunyai tema yang original.

Berdasarkan karya tulis diatas yang membedakan dengan “Sistem Informasi Pengiklanan Makanan Berbasis Website” dapat dilihat pada tabel 2.1 berikut ini.

Tabel 2.1 Tabel *State Of The Art*

No	Materi	Rizky Rahmatullah	Adi Dwi Setiawan	Nurinda Febriani
1	Topik	Media Informasi dan Periklanan Pasar Tradisional Berbasis WEB Study Kasus Pasar Tradisional Kabupaten Bandung	Analisis dan perancangan sistem informasi berbasis web sebagai media promosi dan informasi pada budi watch collections	Sistem Informasi Pengiklanan Makanan Berbasis Website
2	Objek	Pemasaran/ pengiklanan pasar tradisional daerah bandung	Pemasaran/ pengiklanan produk jam dari budi watch collection	Pemasaran/ pengiklanan produk untuk semua produsen makanan yang terdaftar.
3	Metode	Waterfall	Waterfall	<i>Prototype</i>
4	Fasilitas	Informasi harga barang, informasi berdasarkan kategori	Informasi harga barang, profil toko, informasi berdasarkan kategori	Informasi harga, lokasi, diskon, informasi berdasarkan kategori

Pada Tabel 2.1 diatas merupakan data perbandingan yang menjadi acuan untuk membangun sistem informasi pengiklanan makanan ini sehingga sistem ini memiliki keunggulan dan pembeda dari sistem informasi lainnya. Metode yang dipilih berbeda karena metode waterfall memiliki kelemahan pada saat proses perbaikan maupun perubahan sebuah sistem. Berbeda dengan prototype yang memiliki keluwesan ketika sebuah sistem membutuhkan perbaikan maupun perubahan.

BAB 3. METODE KEGIATAN

3.1 Waktu dan Tempat

Karya ilmiah tentang Sistem Informasi Pengiklanan Makanan Berbasis Website ini dilaksanakan selama enam bulan mulai bulan Agustus 2016 sampai dengan Januari 2017 di Politeknik Negeri Jember dan pelaksanaan tempat survei dilakukan di Restoran dan Perusahaan yang bergerak di bidang makanan.

3.2 Alat dan Bahan

3.2.1 Alat

a. Perangkat keras

Perangkat keras yang digunakan dalam pembuatan program ini adalah satu unit komputer dengan spesifikasi sebagai berikut :

- 1) *Processor Amd A-8 4500M 1.90 Hz*
- 2) *RAM 4 GB*
- 3) *Hard Disk 500 GB*

b. Perangkat lunak

Perangkat lunak yang digunakan dalam pembuatan program ini adalah sebagai berikut :

- 1) *Sistem operasi Windows 7 Ultimate 64 bit.*
- 2) *Microsoft Word 2010* sebagai aplikasi pengolah kata.
- 3) *Power Designer 16.1* sebagai aplikasi pembuat rancangan sistem.
- 4) *XAMPP 5.6.21* sebagai program aplikasi pengembang yang berguna untuk pengembangan website berbasis *PHP* dan *MySQL*.
- 5) *Sublime Text* sebagai program aplikasi pengembang yang berguna untuk pengembangan aplikasi.

3.2.2 Bahan

Untuk pengambilan bahan dalam penyusunan tugas akhir diperlukan sebuah teknik pengumpulan bahan. Dalam tugas akhir ini melakukan pengambilan bahan yaitu:

a. Observasi

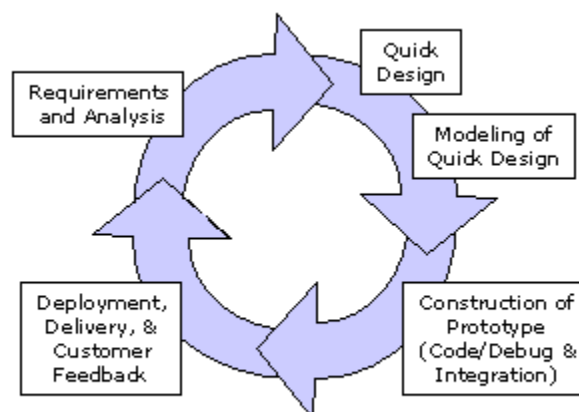
Observasi merupakan teknik pengumpulan data dengan cara mengamati secara langsung. Pada kasus ini observasi dilakukan pada sosial media Instagram yang terdapat akun-akun khusus makanan seperti banyuwangikuliner dan kulinerjember.

b. Studi Kepustakaan

Studi kepustakaan pada penyusunan tugas akhir ini bertujuan untuk membantu dalam analisis kebutuhan dari literatur lain seperti buku, karya tulis ilmiah serta situs teknologi informasi yang nantinya akan berguna dalam pembuatan rancangan yang efektif dalam penyusunan tugas akhir ini.

3.3 Metodologi

Dalam penyusunan sebuah perangkat lunak diperlukan sebuah metodologi untuk memudahkan dalam pengembangan. Pada pengembangan Sistem Informasi Pengiklanan dan Pemesanan Makanan ini menggunakan metodologi *Prototype* sebagai berikut:



Gambar 3.1 Model *Prototyping* menurut Pressman

Berdasarkan ilustrasi pada gambar 3.1 diatas metode *prototyping* yang digunakan bisa diuraikan sebagai berikut :

- a. Pengumpulan kebutuhan : *developer* dan klien bertemu dan menentukan tujuan umum, kebutuhan yang diketahui dan gambaran bagian - bagian yang akan dibutuhkan berikutnya,
- b. *Quick design* dan *modelling of quick design* : perancangan dilakukan cepat dan rancangan mewakili semua aspek *software* yang diketahui dan rancangan ini menjadi dasar pembuatan *Prototype*,
- c. *Construction of Prototype* : pembuatan prototype sesuai dengan desain yang sudah ada. Dalam tahap ini *prototyping* yang sudah disepakati diterjemahkan ke dalam bahasa pemrograman yang sesuai,
- d. Pengujian dan Evaluasi Sistem : Setelah sistem sudah menjadi suatu perangkat lunak yang siap pakai, harus dites dahulu sebelum digunakan. Pelanggan mengevaluasi apakah sistem yang sudah jadi sudah sesuai dengan yang diharapkan.

3.4 Pelaksanaan Kegiatan

Pelaksanaan Kegiatan pada penyusunan tugas akhir ini dilakukan dengan mengikuti metode kegiatan diatas yaitu :

- a. Tahap pertama, dilakukan pengumpulan data-data yang harus dipenuhi dengan melihat beberapa literatur seperti buku ataupun karya tulis ilmiah. Kemudian data yang diperoleh tersebut divalidasi melalui survei di Sosial Media *Instagram*. Data awal tersebut dikumpulkan dalam bentuk dokumen untuk menyusun desain perangkat lunak pada tahap berikutnya.
- b. Tahap kedua, dilakukan perancangan model sistem berbasis mobile ini diawali dengan menyusun desain dalam bentuk *use case diagram*, *activity diagram* dan *class diagram*.
- c. Tahap ketiga, desain program mulai ditulis dalam bentuk kode program dengan *database MySQL*. Kode program ditulis dengan menggunakan aplikasi android studio sedangkan *MySQL* digunakan untuk menyimpan data.

- d. Tahap keempat, di integrasikan ke dalam unit - unit program yang sudah dibangun dan dilakukan *testing* secara keseluruhan. *Testing* ini akan dilakukan oleh beberapa pengguna terutama pihak penjual (pemilik usaha makanan) dan masyarakat. Tujuannya adalah apakah program sudah sesuai dengan apa yang di inginkan dan sesuai dengan aturan atau rule yang dihasilkan dari tahapan pengumpulan kebutuhan. Tahap ini juga merupakan tahapan penentu apakah program yang dibuat sudah selesai atau belum. Jika tahap ini gagal maka dilakukan proses perbaikan ditahap mana yang terjadi kegagalan sampai tidak ada bug atau kesalahan yang terjadi pada program yang dikembangkan. Tahap ini akan terus dilakukan sampai program benar-benar sesuai dengan apa yang di inginkan. Selanjutnya, program yang sudah dibangun dioperasikan pada lingkungan sesungguhnya, pada tahap akhir ini juga dilakukan penyesuaian dan update database atau perubahan data sesuai hasil penelitian terbaru.

BAB 4. HASIL DAN PEMBAHASAN

4.1 *Communication*

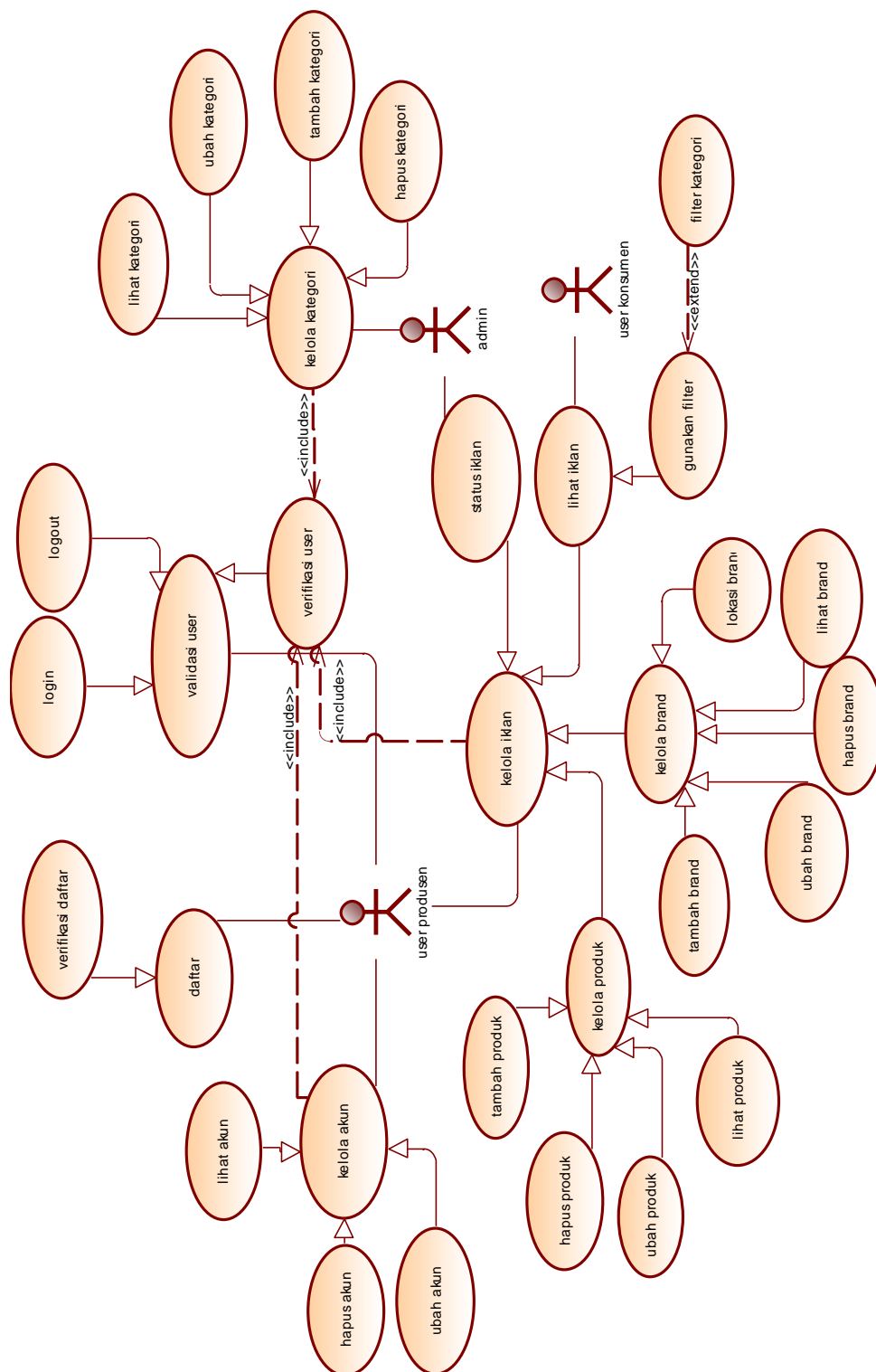
Tahap ini adalah tahap awal yang dilakukan dalam membangun Sistem Informasi Pengiklanan Makanan Berbasis *Website*. Pada tahap ini dilakukan survei dari beberapa pemilik usaha di bidang kuliner mengenai cara yang dilakukan dalam mengiklankan produk-produk mereka. Selain itu juga dilakukan pengamatan terhadap situs pengiklanan ada. Setelah mengamati beberapa sistem pengiklanan maka didapat beberapa fitur yang nantinya akan menjadi pembeda dan menjadi kelebihan dari sistem yang pengembang kembangkan.

4.2 *Quick Plan*

Tahap selanjutnya adalah tahap Quick Plan yaitu tahap dimana developer mulai memikirkan perencanaan alur proses pembangunan prototype secara cepat. Developer akan menentukan dan menjelaskan input dan output dari prototype yang akan dibangun. Pada tahap ini developer akan menyesuaikan apakah analisisnya sudah sama dengan apa yang diinginkan oleh pelanggan. Tahap ini juga merupakan tahap lanjutan dari Communication, karena di tahap ini semua data harus sudah lengkap untuk menentukan input dan outputnya secara benar agar dapat dilanjutkan ke tahap desain.

4.3 *Modeling Quick Design*

Tahap Modeling Quick Design yaitu membuat model untuk mempresentasikan perangkat lunak yang dapat dipahami oleh pengguna, sehingga pengguna mengerti bagaimana alur kerja perangkat lunak yang akan dibuat. Pengguna dapat melihat rancangan alur kerja sistem yang akan dimodelkan menggunakan permodelan UML. Untuk mengetahui jalannya proses didalam sistem informasi dapat di lihat pada Gambar 4.1 di bawah ini.



Gambar 4.1 Usecase

Pada gambar 4.1 merupakan gambaran dari use case yang digunakan pada sistem informasi pengiklanan ini. Terdapat 3 aktor yang berperan di dalam sistem ini diantaranya Administrator dan Pengguna yang dibagi menjadi 2 bagian yaitu produsen dan konsumen.

4.3.1 Definisi Aktor

Berikut adalah deskripsi pendefinisian aktor pada sistem informasi pengiklanan makanan berbasis website. Pada tabel 4.1 merupakan pendefinisian aktor kepada usecase sistem yang dikembangkan.

Tabel 4.1 Definisi Aktor

No.	Aktor	Deskripsi
1.	Administrator	Orang yang bertugas dan memiliki hak akses untuk mengolah data sertifikat, data kategori, data fasilitas, dan melihat data pengguna
2.	Pengguna	Orang yang menggunakan sistem untuk memasang iklan maupun mencari iklan

Pada tabel 4.1 merupakan deskripsi dari masing-masing aktor yang berperan dalam *use case* yaitu Administrator dan Pengguna. Pengguna pada hal ini merupakan produsen makanan dan konsumen yang dijadikan satu pada tabel diatas.

4.3.2 Definisi Use Case

Pada tabel 4.2 berikut merupakan penjelasan masing-masing *Use Case* yang terdapat pada sistem informasi pengiklanan makanan berbasis *website*.

Tabel 4.2 Definisi Usecase

No.	Use Case	Deskripsi
1.	Menambah kategori	Merupakan proses untuk menambah data kategori yang hanya dapat dilakukan oleh administrator
2.	Mengubah kategori	Merupakan proses untuk mengubah data kategori yang hanya dapat dilakukan oleh administrator

Tabel 4.2 Definisi Usecase (lanjutan)

No.	Use Case	Deskripsi
3.	Menghapus kategori	Merupakan proses untuk menghapus data kategori yang hanya dapat dilakukan oleh administrator
4.	Melihat kategori	Merupakan proses untuk melihat data kategori yang ada pada database
5.	Melihat akun	Merupakan proses untuk melihat data akun masing-masing yang dapat dilakukan oleh pengguna
6.	Hapus Akun	Merupakan proses untuk menghapus data berupa akun yang ada yang dapat dilakukan oleh pengguna
7.	Mengubah Akun	Merupakan proses untuk mengubah / mengedit data akun yang dilakukan oleh pengguna
8.	Daftar	Merupakan proses registrasi untuk pengguna yang belum memiliki akun
9.	Verifikasi Daftar	Merupakan proses untuk memastikan apakah proses pendaftaran sudah benar atau belum
10.	Login	Merupakan proses untuk pengguna agar dapat masuk kedalam sistem dan dapat memasang iklan, mengubah iklan, menghapus iklan
11.	Logout	Merupakan proses untuk pengguna agar dapat keluar dari sistem
12.	Verifikasi User	Merupakan proses untuk melakukan pengecekan kesesuaian terhadap pengguna yang masuk kedalam sistem
13.	Validasi user	Merupakan proses untuk memastikan apakah user yang masuk ke sistem benar atau tidak
14.	Mencari iklan	Merupakan proses untuk mencari semua iklan dari data yang disimpan pada database yang dapat dilakukan oleh pengguna
15.	Mencari iklan berdasarkan filter (gunakan filter)	Merupakan proses untuk mencari semua iklan dari data yang disimpan pada database sesuai dengan filter yang sudah dimasukkan oleh pengguna
16.	Menambah produk	Merupakan proses untuk input data produk yang dilakukan oleh pengguna yang sudah melakukan login

Tabel 4.2 Definisi Usecase (lanjutan)

No.	Use Case	Deskripsi
17.	Mengubah produk	Merupakan proses untuk mengubah detail dari produk yang dapat dilakukan oleh pemilik akun, dapat dilakukan jika pengguna sudah melakukan login
18.	Menghapus produk	Merupakan proses untuk menghapus produk oleh pengguna, dapat dilakukan jika pengguna sudah melakukan login
19.	Melihat produk	Merupakan proses untuk melihat detail produk dipilih atau dicari oleh pengguna
20.	Menambah brand	Merupakan proses untuk memasukkan brand yang dilakukan oleh pengguna jika pengguna sudah melakukan login
21.	Mengubah brand	Merupakan proses untuk mengubah detail dari brand yang dapat dilakukan oleh pemilik akun, dapat dilakukan jika pengguna sudah melakukan login
22.	Menghapus produk	Merupakan proses untuk menghapus produk oleh pengguna, dapat dilakukan jika pengguna sudah melakukan login
23.	Melihat produk	Merupakan proses untuk melihat detail produk dipilih atau dicari oleh pengguna
24.	Koordinat wilayah	Merupakan proses untuk melihat letak atau lokasi tempat kuliner tersebut berada

4.3.3 Skenario *Use Case*

Berikut ini adalah skenario dari masing-masing usecase yang telah di definisikan pada pembahasan sebelumnya:

a. Definisi Use Case dari aktor Administrator

Nama Use Case : Menambah kategori

Skenario : Skenario usecase terdapat pada tabel 4.3

Tabel 4.3 Skenario Use Case Menambah Kategori

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. memasukkan data kategori dengan benar	
	3. Menyimpan data kategori ke database
	4. Kembali ke form kategori

Tabel 4.3 merupakan gambaran dari urutan proses yang terjadi ketika Administrator sedang melakukan proses menambah kategori.

Nama Use Case : Mengubah kategori

Skenario : Skenario usecase terdapat pada tabel 4.4

Tabel 4.4 Skenario Use Case Mengubah Kategori

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih data yang akan diubah	
	3. Menampilkan semua kolom isian berdasarkan data yang sudah dipilih
4. Mengubah data kategori	
	5. Menyimpan data kategori yang sudah diubah ke database
	6. Kembali ke form kategori

Tabel 4.4 merupakan gambaran dari urutan proses yang terjadi ketika Administrator sedang melakukan proses mengubah kategori.

Nama Use Case : Menghapus kategori

Skenario : Skenario usecase terdapat pada tabel 4.5

Tabel 4.5 Skenario Use Case Menghapus Kategori

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih data yang akan dihapus	
	3. Menghapus data yang sudah dipilih dari database

Pada tabel 4.5 merupakan gambaran dari urutan proses yang terjadi ketika Administrator sedang melakukan proses menghapus kategori.

Nama Use Case : Melihat kategori

Skenario : Skenario usecase terdapat pada tabel 4.6

Tabel 4.6 Skenario Use Case Melihat Kategori

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
	2. Menampilkan semua data kategori yang terdapat di database

Pada tabel 4.6 merupakan gambaran dari urutan proses yang terjadi ketika Administrator sedang melakukan proses melihat kategori.

Nama Use Case : Kelola Status Iklan

Skenario : Skenario usecase terdapat pada tabel 4.7

Tabel 4.7 Skenario Use Case Mengubah Status Iklan

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih data yang akan diubah	
	3. Menampilkan kolom isian berdasarkan data yang sudah dipilih
4. Mengubah status iklan	
	5. Menyimpan data iklan yang sudah diubah
	6. Kembali ke form iklan/produk

Pada tabel 4.7 merupakan gambaran dari urutan proses yang terjadi ketika Administrator melakukan pengelolaan status iklan yang telah di upload oleh pengguna (Produsen).

b. Definisi Use Case dari aktor pengguna bagian produsen

Nama Use Case : Daftar dan Verifikasi Daftar

Skenario : Skenario usecase terdapat pada tabel 4.8

Tabel 4.8 Skenario Daftar dan Verifikasi Daftar

Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Mengisi data pengguna sesuai isian yang ada	2. Mengecek username sudah berupa email dan password sesuai dengan verifikasi
	3. Menyimpan data pengguna yang telah diisi ke database
	4. Masuk ke halaman login
Skenario Alternatif	
1. Mengisi data pengguna sesuai isian yang ada	2. Mengecek username tidak berupa email dan password tidak sesuai dengan verifikasi
	3. Menampilkan pesan username harus berupa email dan password harus benar
4. Mengisi Username berupa email dan password dengan benar	5. Menyimpan data pengguna yang telah diisi ke database
	6. Masuk ke halaman login

Pada tabel 4.8 diatas merupakan urutan proses yang terjadi ketika pengguna melakukan proses pendaftaran. Tabel diatas berisi skenario normal ketika langkah-langkahnya benar dan skenario alternatif yaitu ketika terjadi kesalahan katika proses pengisian tidak sesuai atau salah.

Nama Use Case : Login

Skenario : Skenario usecase terdapat pada tabel 4.9

Tabel 4.9 Skenario Use Case Login

Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Memasukkan Username dan Password	2. Memeriksa benar tidaknya data yang dimasukkan dengan memeriksa ke tabel users
	3. Jika valid masuk ke halaman administrator
Skenario Alternatif	
1. Memasukkan Username dan Password	2. Memeriksa benar tidaknya data masukan dengan users
	3. Jika tidak benar menampilkan pesan bahwa username atau password salah
4. Memasukkan Username dan Password	5. Memeriksa benar tidaknya data masukan
	6. Masuk ke halaman administrator

Pada tabel 4.9 diatas merupakan urutan proses yang terjadi ketika pengguna melakukan proses *login*. Terdapat 2 skenario yang diberikan yaitu skenario normal ketika username dan password diisi dengan benar dan skenario alternatif yaitu ketika username dan password salah.

Nama Use Case : Logout

Skenario : Skenario usecase terdapat pada tabel 4.10

Tabel 4.10 Skenario Use Case Logout

Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Menekan tombol Logout	
	2. Melakukan Logout
	3. Masuk Halaman Home

Pada tabel 4.10 diatas merupakan urutan proses yang terjadi ketika pengguna melakukan proses *logout*.

Nama Use Case : Verifikasi Login User

Skenario : Skenario usecase terdapat pada tabel 4.11

Tabel 4.11 Skenario Use Case Verifikasi Login User

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa ke variabel session sebagai penanda apakah pengguna sudah melakukan login
	2. Mengembalikan status login atau belum

Pada Tabel 4.11 merupakan gambaran dari urutan proses yang terjadi ketika sistem melakukan verifikasi dari proses login yang dilakukan oleh pengguna.

Nama Use Case : Melihat akun

Skenario : Skenario usecase terdapat pada tabel 4.12

Tabel 4.12 Skenario Melihat Akun

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
	2. Menampilkan data akun yang terdapat di database berdasarkan session

Pada tabel 4.12 diatas merupakan urutan proses yang terjadi ketika pengguna melakukan kegiatan melihat akunnya.

Nama Use Case : Mengubah akun

Skenario : Skenario usecase terdapat pada tabel 4.13

Tabel 4.13 Skenario Use Case Mengubah Akun

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
	2. Memilih data yang akan diubah
	3. Menampilkan semua kolom isian berdasarkan data yang sudah dipilih dari tabel user
	4. Mengubah data akun
	5. Menyimpan data akun yang sudah diubah ke database
	6. Kembali ke form akun

Pada Tabel 4.13 diatas merupakan urutan proses yang terjadi ketika pengguna melakukan pengubahan pada akunnya.

Nama Use Case : Menambah brand

Skenario : skenario usecase terdapat pada tabel 4.14

Tabel 4.14 Skenario Use Case Menambah Brand

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Mengisi data sesuai isian yang ada	3. Menyimpan data ke dalam database
	4. Menampilkan data yang sudah disimpan

Pada Tabel 4.14 diatas merupakan urutan proses yang terjadi ketika pengguna melakukan proses menambah brand miliknya.

Nama Use Case : Menghapus brand

Skenario : skenario usecase terdapat pada tabel 4.15

Tabel 4.15 Skenario Menghapus Brand

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih data yang akan dihapus	3. Menghapus data yang sudah dipilih dari database

Pada Tabel 4.15 diatas merupakan urutan proses yang terjadi ketika pengguna melakukan penghapusan brand miliknya.

Nama Use Case : Mengubah brand

Skenario : Skenario usecase terdapat pada tabel 4.16

Tabel 4.16 Skenario Mengubah Brand

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih data yang ingin diubah	3. Mencari data yang dipilih
	4. Menampilkan detail data yang sudah dipilih
5. Mengubah isian pada data yang sudah ditampilkan	6. Menyimpan data yang sudah diubah
	7. Menampilkan data yang sudah diubah

Pada Tabel 4.16 diatas merupakan urutan proses yang terjadi ketika pengguna melakukan proses pengubahan brand miliknya.

Nama Use Case : Melihat brand

Skenario : Skenario usecase terdapat pada tabel 4.17

Tabel 4.17 Skenario Melihat Brand

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
	2. Menampilkan semua data brand yang terdapat di database yang pernah disimpan oleh pengguna

Pada Tabel 4.17 diatas merupakan urutan proses yang terjadi ketika pengguna melakukan proses melihat brand yang dimilikinya.

Nama Use Case : Menambah produk

Skenario : skenario usecase terdapat pada tabel 4.18

Tabel 4.18 Skenario Use Case Menambah Iklan/Produk

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	2. Memeriksa status login
2. Mengisi data produk sesuai isian yang ada	
	3. Menyimpan data produk ke dalam database
	4. Menampilkan data produk yang sudah disimpan

Pada Tabel 4.18 diatas merupakan urutan proses yang terjadi ketika pengguna melakukan proses penambahan produk yang akan di iklankan.

Nama Use Case : Menghapus produk

Skenario : skenario usecase terdapat pada tabel 4.19

Tabel 4.19 Skenario Menghapus Iklan

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	2. Memeriksa status login
2. Memilih data yang akan dihapus	
	3. Menghapus data yang sudah dipilih dari database

Pada Tabel 4.19 diatas merupakan urutan proses yang terjadi ketika pengguna melakukan proses penghapusan produk yang dimiliki.

Nama Use Case : Mengubah produk

Skenario : skenario usecase terdapat pada tabel 4.20

Tabel 4.20 Skenario Mengubah Iklan

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	2. Memeriksa status login
2. Memilih data yang ingin diubah	
	3. Mencari data yang dipilih
	4. Menampilkan detail data yang sudah dipilih
5. Mengubah isian pada data iklan yang sudah ditampilkan	
	6. Menyimpan data iklan yang sudah diubah
	7. Menampilkan data iklan yang sudah diubah

Pada Tabel 4.20 diatas merupakan urutan proses yang terjadi ketika pengguna melakukan pengubahan pada produk yang dimiliki.

Nama Use Case : Melihat produk

Skenario : Skenario usecase terdapat pada tabel 4.21

Tabel 4.21 Skenario Melihat Iklan

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
	2. Menampilkan semua data database yang pernah disimpan oleh pengguna

Pada Tabel 4.21 merupakan urutan proses yang terjadi ketika pengguna melakukan proses melihat semua produk yang dimilikinya.

c. Definisi Use Case dari aktor pengguna bagian konsumen

Nama Use Case : Melihat iklan dan Menggunakan Filter

Skenario : skenario usecase terdapat pada tabel 4.22

Tabel 4.22 Skenario Melihat dan Menggunakan Filter Iklan

Aksi Aktor	Reaksi Sistem
Skenario Normal	
2. Memilih kategori yang akan dicari	3. Mencari data iklan berdasarkan kategori yang sudah dipilih
	4. Menampilkan data iklan yang sudah dicari dari database
5. Memilih iklan yang dicari	6. Menampilkan detail iklan yang dipilih

Pada Tabel 4.22 diatas merupakan urutan proses yang terjadi ketika pengguna (Konsumen) melakukan proses pencarian produk berdasarkan kategori yang diinginkan.

4.4 Pembuatan Aplikasi

Pada tahap ini dilakukan pengkodean dari sistem informasi pengiklanan makanan berbasis website. Pada tahap ini perancangan yang dibuat ialah dari sisi administrator dan dari sisi pengguna baik itu konsumen maupun produsen makanannya yang akan dijelaskan pada iterasi 1 dari aktor administrator, iterasi 2 dari aktor produsen dan iterasi 3 dari aktor konsumen.

4.4.1 Iterasi 1

Pada iterasi 1 dilakukan pembuatan aplikasi dari aktor administrator yang meliputi pengelolaan kategori dan pengelolaan status iklan. Berikut adalah tahapan prototype pada pembuatan proses mengelola kategori dan mengelola status iklan.

a. *Communication*

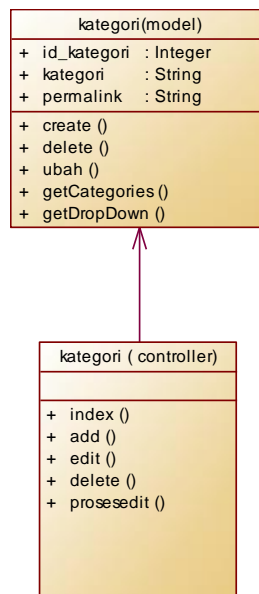
Pada tahap *Communication*, usecase yang digunakan adalah proses yang dapat dilakukan dari sisi administrator. Use Case yang dilakukan adalah menambah kategori, mengubah kategori, menghapus kategori, melihat kategori dan status iklan.

b. *Quick Plan*

Pada tahap *Quick Plan*, macam-macam usecase yang digunakan adalah:

- 1) Menambah kategori : Skenario usecase mengacu pada tabel 4.3
- 2) Mengubah kategori : Skenario usecase mengacu pada tabel 4.4
- 3) Menghapus kategori : Skenario usecase mengacu pada tabel 4.5
- 4) Melihat kategori : Skenario usecase mengacu pada tabel 4.6
- 5) Kelola Status Iklan : Skenario usecase mengacu pada tabel 4.7

Sementara *class diagram* mengacu pada gambar 4.2 berikut ini :

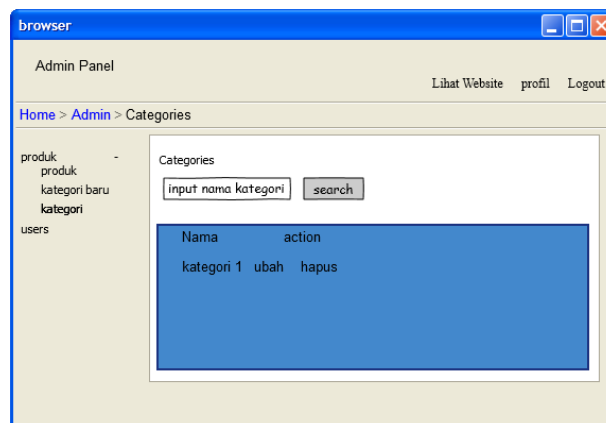


Gambar 4.2 class diagram mengelola kategori

Pada Gambar 4.2 merupakan Class Diagram pengelolaan kategori. Terdapat perintah-perintah atau aksi yang terdapat pada model diantaranya *create()* untuk proses menambah kategori, *delete()* untuk proses menghapus kategori, *ubah()* untuk proses ubah kategori, *getCategories()* untuk proses melihat semua kategori, *getDropDown()* untuk menampilkan kategori dengan bentuk *drop down list* dan *getCategoryById()* untuk melihat kategori berdasarkan Idnya.

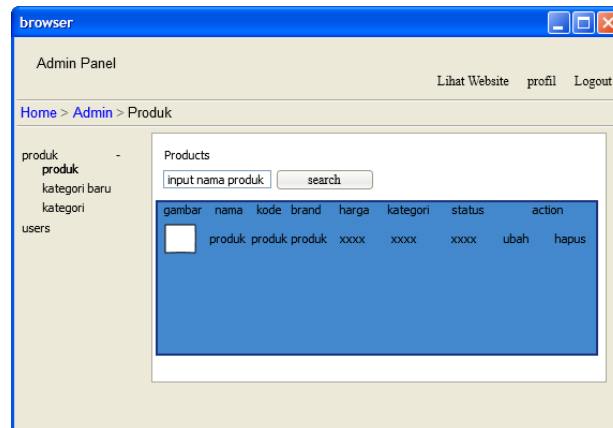
c. Quick Design

Tahap ini merupakan proses dari pembuatan mockup atau design tampilan dari halaman mengelola kategori dan status iklan. Design mockup dapat dilihat pada gambar 4.3 dan 4.4.



Gambar 4.3 Mockup Halaman Kategori

Pada gambar 4.3 adalah design dari halaman kategori untuk administrator.



Gambar 4.4 *Mockup* Halaman Produk

Pada gambar 4.4 adalah design dari halaman produk untuk administrator. Administrator dapat mengubah status dari produk iklan yang di upload oleh user. Status tersebut mempengaruhi apakah produk tersebut akan muncul pada halaman utama atau tidak. Selain itu, berikut adalah design database dari tabel kategori.

Tabel 4.23 Tabel Kategori

No	Nama Field	Jenis	Keterangan
1.	Kat_id	int(17)	<i>Primary Key</i>
2.	kategori	varchar(20)	
3.	permalink	Varchar(20)	

Tabel 4.23 merupakan Tabel Kategori yang ada pada database. Terdapat 3 *field* dengan Kat_id sebagai *Primary Key* nya.

d. *Constuction of Prototype*

Pada tahap ini yaitu mengimplementasikan code sesuai design yang telah dibangun.

```

1) function index() {
2)   $data['categories']=$this->kategori->getCategories();
3)   $data['content'] = 'admin/categories/index';
4)   $this->load->view($this->template, $data);
5) }

```

Kode Program 4.1 Fungsi Index Kategori

Pada kode program 4.1 diatas merupakan fungsi yang akan dipanggil pertama kali ketika membuka halaman kategori dengan memanggil fungsi `getCategories` pada model.

```

1) function getCategories() {
2)   $this->db->select('*');
3)   $this->db->order_by('kategori','ASC');
4)   $query = $this->db->get($this->table);
5)   if ($query->num_rows() > 0) {
6)     return $query->result_array(); }
7) }

```

Kode Program 4.2 Fungsi getCategories

Pada kode program 4.2 merupakan fungsi `getCategories` yang ada pada model. Fungsi tersebut bertugas untuk memanggil seluruh data yang ada pada tabel kategori dan diurutkan berdasarkan abjad.

```

1) function add() {
2)   $this->form_validation->set_rules('name', 'name',
   'required');
3)   $this->form_validation->set_error_delimiters(' ', '<br/>');
4)   if ($this->form_validation->run() == TRUE) {
5)     $this->kategori->create();
6)     $this->session->set_flashdata('success', 'Kategori added');
7)     redirect('admin/categories'); }
8)   $data['content'] = 'admin/categories/add';
9)   $this->load->view($this->template, $data);
10) }

```

Kode Program 4.3 Fungsi Tambah Kategori

Pada gambar 4.7 merupakan fungsi dari menambah kategori. Fungsi ini memanggil fungsi `create` yang ada pada model apabila inputan sudah benar atau terisi.

```

1) function create() {
2) $data = array(
3) 'kategori' => $this->input->post('name'),
4) 'permalink' => url_title(strtolower($this->input->
   >post('name'))) );
5) $this->db->insert($this->table, $data);
6) if ($this->db->affected_rows() == 1) {
7) return TRUE; }
8) return FALSE;
9) }

```

Kode Program 4.4 Fungsi Tambah Kategori pada Model

Pada Kode Program 4.4 merupakan fungsi create yang ada model kategori. Inputan nama kategori dijadikan array bersamaan dengan permalink yang isinya mengambil data dari inputan nama secara otomatis. Kemudian data yang berupa array tersebut disimpan pada tabel kategori.

```

1) function edit($id = null) {
2) if ($id == null) {
3) $id = $this->input->post('kat_id'); }
4) $data['category'] = $this->kategori->getCategoryById($id);
5) $data['content'] = 'admin/categories/edit';
6) $this->load->view($this->template, $data);
7) }

```

Kode Program 4.5 Fungsi Edit Kategori

Pada Kode Program 4.5 merupakan fungsi edit data kategori. Fungsi ini memanggil getCategoryById yang terdapat pada model yang bertugas memanggil data berdasarkan id.

```

1) function prosesedit(){
2) $id=$this->input->post('kat_id');
3) $nama=$this->input->post('name');
4) $permalink=$this->input->post('permalink');
5) $this->kategori->ubah($id,$nama);
6) redirect('admin/categories','refresh'); }

```

Kode Program 4.6 Fungsi Prosesedit Kategori

Pada Kode Program 4.6 ini merupakan proses dari perubahan data kategori. Data dari database akan dikirim ke textbox masing-masing. Fungsi ini memanggil fungsi ubah yang terdapat pada model kategori.


```

1) function ubah($id,$nama){
2) $this->db->set('kategori',$nama);
3) $this->db->where('kat_id', $id);
4) $this->db->update($this->table);
5) }

```

Kode Program 4.7 Fungsi Edit Kategori Pada Model

Pada Kode Program 4.7 ini merupakan proses pengiriman data yang diubah pada halaman kategori menuju database kategori.

```

1) function delete($id = null) {
2) if ($id == null) {
3) $this->session->set_flashdata('error', 'Invalid category');
4) redirect('admin/categories');
5) } else {
6) if ($this->kategori->delete($id)) {
7) $this->session->set_flashdata('success', 'Category deleted');
8) } else {
9) $this->session->set_flashdata('error','Failed, try again !');
10) }
11) }
12) redirect('admin/categories');
13) }

```

Kode Program 4.8 Fungsi Hapus Kategori

Pada Kode Program 4.8 diatas merupakan fungsi dari delete/hapus data kategori berdasarkan id nya. Fungsi ini memanggil fungsi delete yang ada pada model kategori.

```

1) function delete($id) {
2) $this->db->where('kat_id', $id);
3) $this->db->delete($this->table);
4) if ($this->db->affected_rows() == 1) {
5) return TRUE; }
6) return FALSE;
7) }

```

Kode Program 4.9 Fungsi Hapus Kategori Pada Model

Pada Kode Program 4.9 merupakan fungsi dari delete yang ada pada kategori. Pada fungsi inilah terdapat perintah untuk menghapus data yang ada pada database berdasarkan id nya.

```

1) function edit($id = null){
2)   if ($id == null) {
3)     $id = $this->input->post('id'); }
4)   $data['product'] = $this->Products->getProductsById($id);
5)   $data['status'] = $this->Products->status;
6)   $data['content'] = 'admin/products/edit';
7)   $this->load->view($this->template, $data); }

8) function editproses(){
9)   $id=$this->input->post('id');
10)  $status = $this->input->post('status');
11)  $this->products->ubah($id,$status);
12)  redirect('admin/produk','refresh');
13) }

```

Kode Program 4.10 Fungsi Edit Status

Pada Kode Program 4.10 merupakan fungsi dari edit status dari produk iklan yang ditampilkan berdasarkan id nya. Fungsi tersebut memanggil fungsi edit yang ada pada model product.

```

1) var $status = array(
2)   0 => 'banned',
3)   1 => 'published' );

4) function ubah($id,$status){
5)   $this->db->set('enabled',$status);
6)   $this->db->where('id', $id);
7)   $this->db->update($this->table);
8) }

```

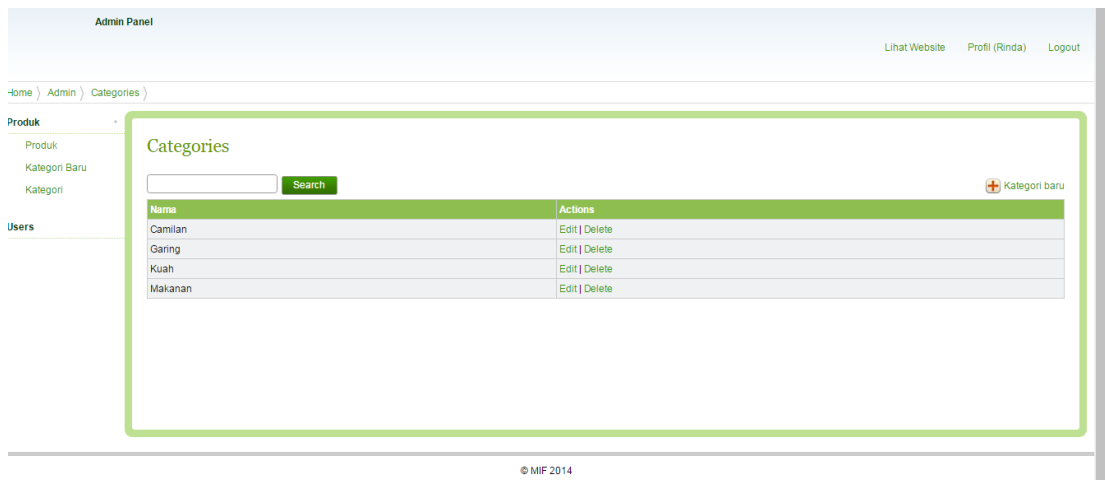
Kode Program 4.11 Fungsi Edit Status Pasa Model

Pada Kode Program 4.11 merupakan fungsi edit status yang terdapat pada model. Status berupa array yang terdiri dari draft dan published yang akan disimpan pada database.

e. *Deployment Delivery and Feedback*

Pada tahap ini merupakan tahapan pengujian pada iterasi mengelola kategori, apabila pengguna tidak sesuai dengan hasil yang di harapkan maka pengembang memperbaiki sistem tersebut sehingga aplikasi atau sistem yang di buat sesuai dengan yang di inginkan oleh pengguna pengujian ini menggunakan black box.

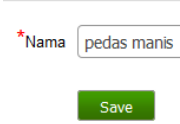
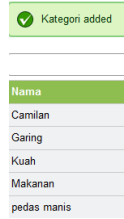
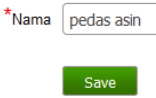

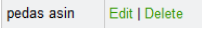
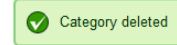
1) *Deployment delivery* kategori



Gambar 4.5 Halaman Kategori

Pada gambar 4.5 diatas merupakan tampilan dari halaman kategori yang sudah di aplikasikan pada *website*. Administrator dapat mengubah ataupun menghapus kategori, selain itu terdapat fungsi pencarian berdasarkan nama kategori yang dapat mempermudah penggunaan halaman tersebut.

2) *Feedback* kategoriTabel 4.24 Tabel *Feedback* Kategori

No.	Skenario Pengujian	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1.	Memasukkan data kategori dengan benar, klik simpan		Sistem sukses menyimpan, menampilkan data kategori		Valid
2.	Mengupdate data kategori dengan benar, klik simpan		Sistem sukses mengupdate, menampilkan data kategori		Valid
3.	Menghapus data kategori		Sistem sukses menghapus kategori		Valid

Pada Tabel 4.24 diatas merupakan data dari pengujian yang telah dilakukan pada pengelolaan kategori.

3) *Deployment delivery status*



Gambar 4.6 Halaman Produk Administrator

Pada gambar 4.6 merupakan tampilan dari halaman produk untuk administrator dimana administrator hanya bisa merubah status dari produk yang akan di iklankan.

4) *Feedback status*

Tabel 4.25 *Feedback Status Iklan*

No.	Skenario Pengujian	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1.	Mengupdate status dengan benar, klik simpan	<div> <div>Status published</div> <div>Simpan</div> </div>	Sistem sukses mengupdate status	<div> <div>Status</div> <div>banned</div> </div>	Valid

Pada Tabel 4.25 merupakan data dari hasil pengujian perubahan status iklan yang telah dilakukan.

4.4.2 Iterasi 2

Pada iterasi 2 dilakukan pembuatan aplikasi dari aktor user produsen yang meliputi pengelolaan produk dan pengelolaan brand atau merk . Berikut adalah tahapan prototype pada pembuatan proses mengelola produk dan mengelola brand.

a. *Communication*

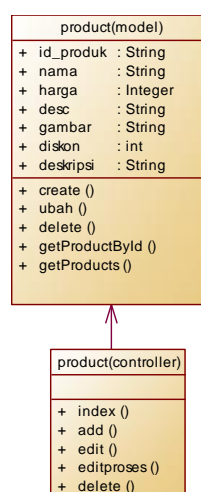
Pada tahap *Communication*, usecase yang digunakan adalah proses yang dapat dilakukan dari sisi user produsen. Use Case yang dilakukan adalah menambah produk, mengubah produk, menghapus produk, melihat produk, menambah brand, menghapus brand, mengubah brand dan melihat brand.

b. *Quick Plan*

Pada tahap *Quick Plan*, macam-macam usecase yang digunakan adalah:

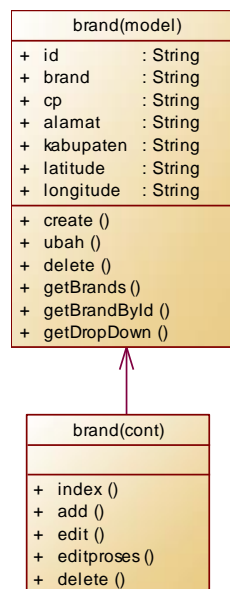
- 1) Menambah produk : Skenario usecase mengacu pada tabel 4.18
- 2) Mengubah produk : Skenario usecase mengacu pada tabel 4.20
- 3) Menghapus produk : Skenario usecase mengacu pada tabel 4.19
- 4) Melihat produk : Skenario usecase mengacu pada tabel 4.21
- 5) Menambah brand : Skenario usecase mengacu pada tabel 4.14
- 6) Mengubah brand : Skenario usecase mengacu pada tabel 4.16
- 7) Menghapus brand : Skenario usecase mengacu pada tabel 4.15
- 8) Melihat brand : Skenario usecase mengacu pada tabel 4.17

Sementara *class diagram* mengacu pada gambar 4.18 dan 4.19 berikut ini :



Gambar 4.7 *Class Diagram* Produk

Pada Gambar 4.7 diatas merupakan *Class Diagram* dari pengelolaan produk. Terdapat perintah *create()* untuk menambah produk, *delete()* untuk menghapus produk, *ubah()* untuk mengubah produk dan *getProductById()* untuk melihat produk berdasarkan Idnya.

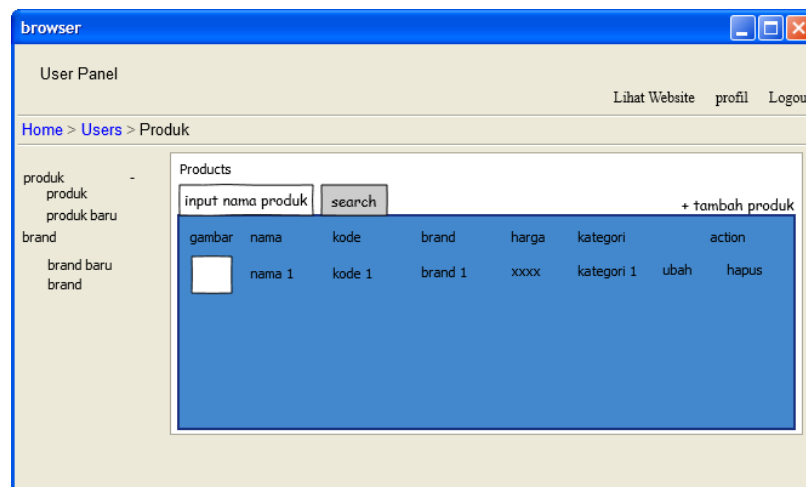


Gambar 4.8 *Class Diagram* Brand

Pada Gambar 4.8 diatas merupakan *Class Diagram* dari pengelolaan Brand. Terdapat perintah *create()* untuk menambah brand, *delete()* untuk menghapus brand, *ubah()* untuk mengubah brand, *getBrands()* untuk melihat semua brand yang ada, *getBrandById* untuk melihat brand berdasarkan Idnya dan *getDropDown()* untuk melihat brand dalam bentuk *dropdownlist*.

c. *Quick Design*

Tahap ini merupakan proses dari pembuatan *mockup* atau design tampilan dari halaman mengelola produk dan mengelola brand. Design mockup dapat dilihat pada gambar 4.9 dan 4.10.



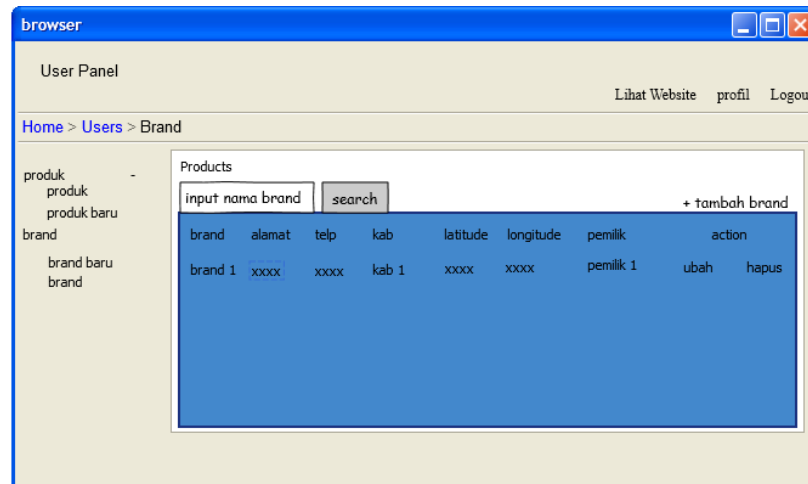
Gambar 4.9 Mockup Halaman Produk Produsen

Pada gambar 4.9 merupakan tampilan design mockup dari halaman produk untuk user produsen dan berikut merupakan design dari database produk.

Tabel 4.25 Tabel Produk

No	Nama Field	Jenis	Keterangan
1.	Id	int(17)	<i>Primary Key</i>
2.	kode	varchar(20)	
3.	nama	Varchar(20)	
4.	Permalink	Varchar(20)	
5.	deskripsi	text	
6.	Harga	double	
7.	Gambar	text	
8.	kategori	Int(11)	
9.	Brand	Int(11)	
10.	Enable	Tyniint(1)	

Pada Tabel 4.25 merupakan tabel produk yang ada pada database. Terdapat 10 field dengan ID sebagai *Primary Key* nya.



Gambar 4.10 mockup Brand

Pada gambar 4.10 merupakan tampilan design mockup dari halaman brand untuk user produsen berikut merupakan design dari database brand.

Tabel 4.26 Tabel Brand

No	Nama Field	Jenis	Keterangan
1.	Id	int(17)	<i>Primary Key</i>
2.	brand	varchar(20)	
3.	Permalink	Varchar(20)	
4.	cp	Varchar(20)	
6.	Alamat	Varchar(20)	
7.	kabupaten	Varchar(20)	
8.	Id_user	Int(11)	
9.	latitude	Text	
10.	longitude	text	

Pada Tabel 4.25 merupakan Tabel Brand yang terdapat pada database. Terdiri dari 10 *field* dengan ID sebagai *Primary Key* nya.

d. *Constuction of Prototype*

Pada tahap ini yaitu mengimplementasikan code sesuai design yang telah dibangun.

```

1) function index() {
2)   $products = $this->Userproduct->getProducts();
3)   $config['uri_segment'] = 4;
4)   $config['total_rows'] = count($products);
5)   $config['per_page']=$this->general-
   >getSetting('paginationLimit');
6)   $config['base_url'] = base_url() .
   'c_user/c_userproduk/index/';
7)   $this->pagination->initialize($config);
8)   if ($this->input->get('q')):
9)     $data['products'] = $this->Userproduct->getProducts
   ($config['per_page'], $this->uri->segment(4), $this->input->
   get('q'));
10)  else:
11)    $data['products'] = $this->Userproduct->
   getProducts($config['per_page'], $this->uri->segment(4));
12)  endif;
13)  $data['pagination'] = $this->pagination->create_links();
14)  $data['status'] = $this->Userproduct->status;
15)  $data['content'] = 'user/produk/index';
16)  $this->load->view($this->template, $data);
17) }
```

Kode Program 4.12 Fungsi Index Halaman Produk

Pada Kode Program 4.12 merupakan fungsi index dari halaman produk untuk user, dimana ketika user membuka halaman tersebut maka fungsi ini yang akan dipanggil. Pada fungsi ini juga terdapat code untuk pencarian data berdasarkan inputan nama produk. Fungsi ini memanggil fungsi getProduct yang terdapat pada model.

```

1) function getProducts($perPage = null, $offset = null, $key =
   null) {
2) $this->db->select('p.id, p.sku,p.nama,p.permalink,
   p.deskripsi,P.harga,p.diskon,p.harga_net,p.gambar, p.enabled,
   c.kategori as categoryName, b.brand as brand');
3) $this->db->join('kategori c', 'c.kat_id = p.kategori');
4) $this->db->join('brand b', 'b.id=p.brand');
5) $this->db->join('users u', 'u.id=b.id_user');
6) $this->db->where('b.id_user', $this->session-
   >userdata('id'));
7) $this->db->limit($perPage, $offset);
8) if ($key != null) {
9) $this->db->like('p.nama', $key);
10) $this->db->or_like('p.deskripsi', $key); }
11) $query = $this->db->get('produk p');
12) if ($query->num_rows() > 0) {
13) return $query->result_array();
14) }
15) }

```

Kode Program 4.13 Fungsi GetProduct

Pada Kode Program 4.13 ini merupakan fungsi getProduct() yang terdapat pada model. Fungsi ini memanggil data yang ada pada database dan terdiri dari beberapa tabel diantaranya tabel produk, brand, kategori dan user menggunakan perintah join.

```

1) function add() {
2) $this->form_validation->set_rules('name', 'Nama',
   'required');
3) $this->form_validation->set_rules('description', 'Deskripsi',
   '');
4) $this->form_validation-
   >set_rules('price', 'harga', 'required');
5) $this->form_validation->set_rules('kategori', 'kategori',
   'required');
6) $this->form_validation->set_rules('brand', 'brand',
   'required');
7) $this->form_validation->set_error_delimiters('', '<br/>');
8) if ($this->form_validation->run() == TRUE) {
9) if ($_FILES['image']['error'] != 4) {
10) $config['upload_path'] = $this->path;
11) $config['allowed_types']=$this->general->
   getSetting('imageAllowed');
12) $config['max_size'] = $this->general-
   >getSetting('maxImageSize');
13) $this->load->library('upload', $config);
14) if (!$this->upload->do_upload("image")) {
15) $this->session->set_flashdata('error', $this->upload->
   display_errors());
16) redirect('user/produk/index');
17) } else {
18) $data = $this->upload->data();
19) $filename = $data['file_name'];

```

```

20) if ($this->Userproduct->create($filename)) {
21) $this->session->set_flashdata('success', 'Produk
    Ditambahkan'); redirect('c_user/c_userproduk/index');
22) } else {
23) $this->session->set_flashdata('error', 'Gagal, Coba Lagi!');
24) redirect('c_user/c_userproduk/add');
25) }
26) }
27) }
28) }
29) $data['categories_brands'] = $this->brand-
    >getDropDown_brands();
30) $data['categories_gender'] = $this->kategori->
    getDropDown_gender();
31) $data['content'] = 'user/produk/add';
32) $this->load->view($this->template, $data);
33) }

```

Kode Program 4.14 Fungsi Tambah Produk

Pada Kode Program 4.14 merupakan fungsi untuk menambah data produk atau fungsi add. Pada fungsi ini inputan divalidasi terlebih dahulu sebelum akhirnya disimpan pada database. Pada fungsi ini juga terdapat inputan berupa *image*.

```

1) function create($filename) {
2) $data = array(
3) 'nama' => $this->input->post('name'),
4) 'permalink' => url_title(strtolower($this->input-
    >post('name'))),
5) 'deskripsi' => $this->input->post('description'),
6) 'harga' => $this->input->post('price'),
7) 'diskon' => $this->input->post('discount_percent'),
8) 'harga_net' => $this->input->post('price') - ($this->input->
    post('discount_percent') / 100 * $this->input-
    >post('price')),
9) 'gambar' => $filename,
10) 'kategori' => $this->input->post('kategori'),
11) 'brand' => $this->input->post('brand'),
12) 'enabled' => 1
13) );
14) $this->db->insert($this->table, $data);
15) if ($this->db->affected_rows() == 1) {
16) return TRUE;
17) }
18) return FALSE;
19) }

```

Kode Program 4.15 Fungsi Tambah Produk pada Model

Pada Kode Program 4.15 diatas merupakan fungsi create data yang ada pada model. Data yang diisi di inisialisasi dan selanjutnya dikirim pada model untuk kemudian di simpan pada database produk.

```

1) function editproses(){
2)   $id = $this->input->post('id');
3)   $nama = $this->input->post('name');
4)   $harga = $this->input->post('price');
5)   $diskon = $this->input->post('discount_percent');
6)   $harga_net = $this->input->post('price') - ($this->input->
    post('discount_percent') / 100 * $this->input->
    post('price'));
7)   $deskripsi = $this->input->post('description');
8)   $kategori = $this->input->post('kategori');
9)   $brand = $this->input->post('brand');
10)  $status = 1;
11)  $this->Userproduct->
    ubah($id,$nama,$harga,$diskon,$harga_net,$deskripsi,$kategori
    ,$brand,$status);
12)  redirect('c_user/c_userproduk','refresh');
13)  }

```

Kode Program 4.16 Fungsi Ubah Produk

Pada Kode Program 4.16 merupakan fungsi dari edit dan editproses. Pada fungsi edit hanya untuk menyeleksi data berdasarkan id nya sedangkan pada editproses merupakan proses perubahan sebenarnya.

```

1) Function ubah
   ($id,$nama,$harga,$diskon,$harga_net,$deskripsi,$kategori,$br
   and,$status){
2)   $this->db->set('nama',$nama);
3)   $this->db->set('harga',$harga);
4)   $this->db->set('diskon',$diskon);
5)   $this->db->set('harga_net',$harga_net);
6)   $this->db->set('deskripsi',$deskripsi);
7)   $this->db->set('kategori',$kategori);
8)   $this->db->set('brand',$brand);
9)   $this->db->set('enabled',$status);
10)  $this->db->where('id', $id);
11)  $this->db->update($this->table);
12)  }

```

Kode Program 4.17 Fungsi Ubah Produk pada Model

Pada Kode Program 4.17 merupakan model dari ubah data produk. Data-data yang telah dirubah akan disimpan pada database berdasarkan id nya.

```

1) function delete($id = null) {
2)   if ($id == null) {
3)     $this->session->set_flashdata('error', 'Produk Kosong');
4)     redirect('c_user/c_userproduk/index');
5)   } else {
6)     if ($this->Userproduct->delete($id)) {
7)       $this->session->set_flashdata('success', 'Produk Berhasil
        Dihapus');
8)     } else {
9)       $this->session->set_flashdata('error', 'Gagal , Coba Lagi!');
10)    }
11)    redirect('c_user/c_userproduk/index');
12)  }
13) }

```

Kode Program 4.18 Fungsi Hapus Produk

Pada Kode Program 4.18 merupakan fungsi dari hapus data. Data tersebut akan dihapus berdasarkan id nya. Ketika data berhasil dihapus maka akan muncul pemberitahuan data berhasil dihapus, begitupun sebaliknya ketika data gagal dihapus. Maka akan muncul pemberitahuan bahwa data gagal dihapus.

```

1) function delete($id) {
2)   $this->db->where('id', $id);
3)   $this->db->delete($this->table);
4)   if ($this->db->affected_rows() == 1) {
5)     return TRUE;
6)   }
7)   return FALSE;
8) }

```

Kode Program 4.19 Fungsi Hapus Produk pada Model

Pada Kode Program 4.19 diatas merupakan fungsi delete yang ada pada model produk.

```

1) function index() {
2)   $brand = $this->brand->getBrands();
3)   $config['uri_segment'] = 4;
4)   $config['total_rows'] = count($brand);
5)   $config['per_page'] = $this->general->
        getSetting('paginationLimit');
6)   $config['base_url'] = base_url() . 'c_user/brands/index/';
7)   $this->pagination->initialize($config);
8)   if ($this->input->get('q')):
9)     $data['brand'] = $this->brand->getBrands($config['per_page'],
        $this->uri->segment(4), $this->input->get('q'));
10)  else:
11)    $data['brand'] = $this->brand->
        getBrands($config['per_page'], $this->uri->segment(4));
12)  endif;
13)  $data['pagination'] = $this->pagination->create_links();
14)  $data['content'] = 'user/brand/index';
15)  $this->load->view($this->template, $data); }

```

Kode Program 4.20 Fungsi Index Halaman Brand

Pada Kode Program 4.20 merupakan fungsi index dari halaman brand dimana ketika user membuka halaman tersebut maka fungsi ini yang akan dipanggil. Pada fungsi ini juga terdapat code untuk pencarian data berdasarkan inputan nama produk.

```

1) function add() {
2) $this->form_validation->set_rules('id_user', 'id_user',
   'required');
3) $this->form_validation->set_rules('brand', 'brand',
   'required');
4) $this->form_validation->set_rules('alamat', 'alamat',
   'required');
5) $this->form_validation->set_rules('kabupaten', 'kabupaten',
   'required');
6) $this->form_validation->set_rules('cp', 'cp', 'required');
7) $this->form_validation->set_rules('latitude', 'latitude',
   'required');
8) $this->form_validation->set_rules('longitude', 'longitude',
   'required');
9) $this->form_validation->set_error_delimiters('', '<br/>');
10) if ($this->form_validation->run() == TRUE) {
11) $this->brand->create();
12) $this->session->set_flashdata('success', 'Brand Ditambahkan
    !');
13) redirect('c_user/brands/index');
14) }
15) $data['content'] = 'user/brand/add';
16) $this->load->view($this->template, $data);
17) }
```

Kode Program 4.21 Fungsi Tambah Brand

Pada Kode Program 4.21 merupakan fungsi dari tambah data brand. Data-data yang disebutkan pada fungsi tersebut adalah wajib untuk diisi (*require*). Apabila data tidak diisi maka data tidak dapat disimpan atau penyimpanan data gagal.

```

1) function create() {
2) $data = array(
3) 'brand' => $this->input->post('brand'),
4) 'permalink' => url_title(strtolower($this->input->
   post('brand'))),
5) 'alamat'=>$this->input->post('alamat'),
6) 'kabupaten' =>$this->input->post('kabupaten'),
7) 'cp' =>$this->input->post('cp'),
8) 'latitude' =>$this->input->post('latitude'),
9) 'longitude' =>$this->input->post('longitude'),
10) 'id_user' => $this->input->post('id_user') );
11) $this->db->insert($this->table, $data);
12) if ($this->db->affected_rows() == 1) {
13) return TRUE;
14) }
15) return FALSE;
16) }

```

Kode Program 4.22 Fungsi Tambah Brand pada Model

Pada Kode Program 4.22 merupakan fungsi create yang ada pada model. Data diambil dari data yang sudah diisi pada form tambah brand kemudian dikirim ke database untuk disimpan.

```

1) function editproses(){
2) $id=$this->input->post('id');
3) $brand=$this->input->post('brand');
4) $alamat = $this->input->post('alamat');
5) $kabupaten = $this->input->post('kabupaten');
6) $id_user = $this->input->post('id_user');
7) $cp = $this->input->post('cp');
8) $latitude = $this->input->post('latitude');
9) $longitude = $this->input->post('longitude');
10) $this->brand->
   ubah($id,$brand,$alamat,$kabupaten,$id_user,$cp,$latitude,$lo
   ngitude);
11) redirect('c_user/brands','refresh');
12) }

```

Kode Program 4.23 Fungsi Hapus Brand

Pada Kode Program 4.23 diatas merupakan fungsi dari edit dan editproses yang ada pada controller brand.


```

1) function
   ubah($id,$brand,$alamat,$kabupaten,$id_user,$cp,$latitude,$
   longitude){
2) $this->db->set('brand',$brand);
3) $this->db->set('alamat',$alamat);
4) $this->db->set('cp',$cp);
5) $this->db->set('latitude',$latitude);
6) $this->db->set('longitude',$longitude);
7) //$this->db->set('diskon',$diskon);
8) $this->db->set('kabupaten',$kabupaten);
9) $this->db->where('id', $id);
10)      $this->db->update($this->table);
11)      }

```

Kode Program 4.24 Fungsi Hapus Brand pada Model

Pada Kode Program 4.24 diatas merupakan fungsi dari ubah data yang berada pada model brand. Pada model inilah terdapat perintah untuk menyimpan data yang telah diubah pada form ubah data brand.

```

1) function delete($id = null) {
2) if ($id == null) {
3) $this->session->set_flashdata('error', 'Brand Kosong');
4) redirect('c_user/brands/index');
5) } else {
6) if ($this->brand->delete($id)) {
7) $this->session->set_flashdata('success', 'Produk Berhasil
   Dihapus');
8) } else {
9) $this->session->set_flashdata('error', 'Gagal , Coba Lagi!');
10) }
11) redirect('c_user/brands/index');
12) }
13) }

```

Kode Program 4.25 Fungsi Hapus Brand

Pada Kode Program 4.25 diatas merupakan fungsi delete data. Data akan dihapus berdasarkan id dari data yang dipilih.

```

1) function delete($id) {
2) $this->db->where('id', $id);
3) $this->db->delete($this->table);
4) if ($this->db->affected_rows() == 1) {
5) return TRUE;
6) }
7) return FALSE;
8) }

```

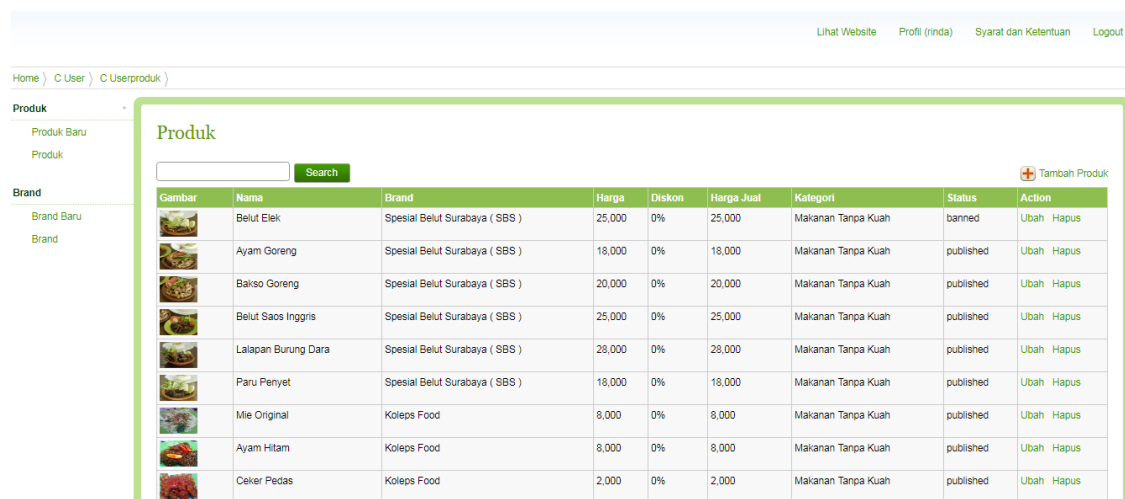
Kode Program 4.26 Fungsi Hapus Brand pada Model










Pada Kode Program 4.26 diatas merupakan model dari fungsi delete. Pada fungsi inilah terdapat perintah untuk menghapus data pada database berdasarkan id dari data yang dipilih.

e. *Deployment Delivery and Feedback*

Pada tahap ini merupakan tahapan pengujian pada iterasi mengelola produk dan mengelola brand, apabila pengguna tidak sesuai dengan hasil yang di harapkan maka pengembang memperbaiki sistem tersebut sehingga aplikasi atau sistem yang di buat sesuai dengan yang di inginkan oleh pengguna pengujian ini menggunakan black box.

1) *Deployment Delivery* Produk

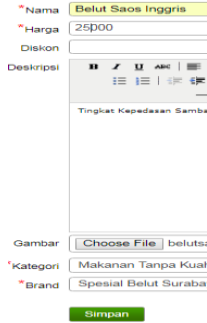

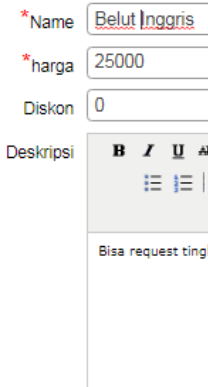
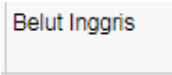




Gambar	Nama	Brand	Harga	Diskon	Harga Jual	Kategori	Status	Action
	Belut Elek	Spesial Belut Surabaya (SBS)	25,000	0%	25,000	Makanan Tanpa Kuah	banned	Ubah Hapus
	Ayam Goreng	Spesial Belut Surabaya (SBS)	18,000	0%	18,000	Makanan Tanpa Kuah	published	Ubah Hapus
	Bakso Goreng	Spesial Belut Surabaya (SBS)	20,000	0%	20,000	Makanan Tanpa Kuah	published	Ubah Hapus
	Belut Saos Inggris	Spesial Belut Surabaya (SBS)	25,000	0%	25,000	Makanan Tanpa Kuah	published	Ubah Hapus
	Lalapan Burung Dara	Spesial Belut Surabaya (SBS)	28,000	0%	28,000	Makanan Tanpa Kuah	published	Ubah Hapus
	Paru Penyet	Spesial Belut Surabaya (SBS)	18,000	0%	18,000	Makanan Tanpa Kuah	published	Ubah Hapus
	Mie Original	Koleps Food	8,000	0%	8,000	Makanan Tanpa Kuah	published	Ubah Hapus
	Ayam Hitam	Koleps Food	8,000	0%	8,000	Makanan Tanpa Kuah	published	Ubah Hapus
	Ceker Pedas	Koleps Food	2,000	0%	2,000	Makanan Tanpa Kuah	published	Ubah Hapus

Gambar 4.11 Halaman Produk untuk Produsen

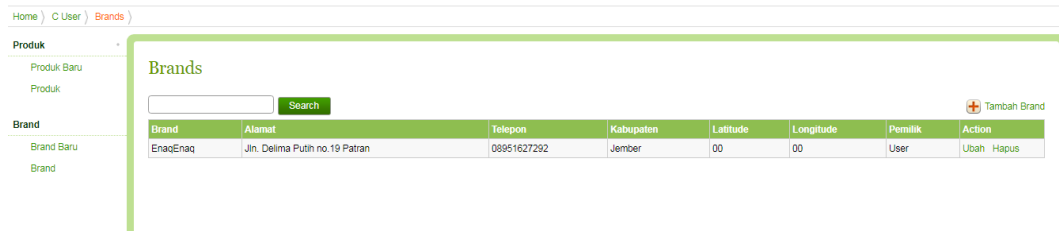
Pada gambar 4.11 merupakan tampilan dari halaman produk untuk user berupa produsen yang sudah diaplikasikan pada website.

2) *Feedback* ProdukTabel 4.27 *Feedback* Produk

No.	Skenario Pengujian	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1.	Memasukkan data dengan benar, klik simpan		Sistem sukses menyimpan kemudian menampilkan data		Valid
2.	Mengupdate data produk dengan benar, klik simpan		Sistem sukses mengupdate data dan menampilkan data produk		Valid
3.	Menghapus data produk dengan mengeklik tombol delete		Sistem sukses menghapus dan menampilkan data	 	Valid

Pada Tabel 4.27 merupakan data hasil pengujian dari halaman produk yang telah dilakukan.

3) *Deployment Delivery Brand*



Gambar 4.12 Halaman Brand untuk Produsen

Pada gambar 4.12 merupakan tampilan dari halaman brand yang sudah aplikasikan pada *website*.

4) *Feedback Brand*

Tabel 4.28 *Feedback Brand*

No.	Skenario Pengujian	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1.	Memasukkan data dengan benar,klik simpan	*ID USER 9 *Nama Brand Sempurna *Alamat jln kalmalang *Contact Person 098xxxxxxx *Kabupaten banyuwangi *Latitude xxxxxx *Longitude xxxxxx Simpan	Sistem sukses menyimpan, menampilkan data	Brand Ditambahkan ! Brand Alamat Lazis jln.pasundan no b jln.kalimantan 7 sedap jln mastrip Sempurna jln kalmalang	Valid
2.	Mengupdate data produk dengan benar, klik simpan	*ID USER 9 *Nama Brand Mantab *Alamat jln.pasundan no ! *Telepon 0333789678 Kabupaten BANYUWANGI *Latitude -2.5523685 *Longitude 120.3205464 Simpan	Sistem sukses mengupdate dan menampilkan data produk	Brand Mantab b sedap Sempurna	Valid
3.	Menghapus data produk	Brand Mantab b sedap Sempurna	Sistem sukses menghapus menampilkan data	Produk Berhasil Dihapus Brand Alamat b jln.kalimantan sedap jln mastrip Sempurna jln kalmalang	Valid

Pada tabel 4.28 merupakan data hasil dari pengujian halaman brand yang telah dilakukan.

4.4.3 Iterasi 3

Pada iterasi 3 dilakukan pembuatan aplikasi yang meliputi pengelolaan produk dan pengelolaan login dan register. Berikut adalah tahapan prototype pada pembuatan proses mengelola produk dan mengelola brand.

a. *Communication*

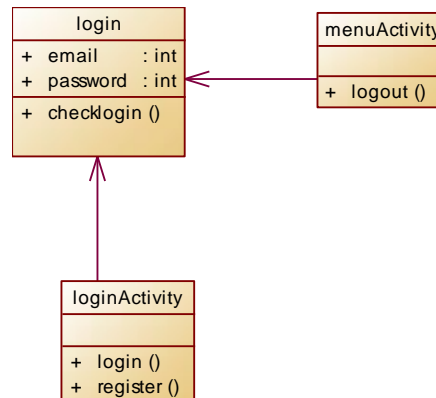
Pada tahap *Communication*, usecase yang digunakan adalah proses yang dapat dilakukan dari sisi user produsen. Use Case yang dilakukan adalah login dan register.

b. *Quick Plan*

Pada tahap *Quick Plan*, macam-macam usecase yang digunakan adalah:

- 1) login : Skenario usecase mengacu pada tabel 4.9
- 2) Register : Skenario usecase mengacu pada tabel 4.8

Sementara *class diagram* mengacu pada gambar 4.13 berikut ini.



Gambar 4.13 *Class Diagram Login*

Pada Gambar 4.13 merupakan gambaran dari perintah maupun data yang digunakan pada proses login yaitu dengan menggunakan *email* yang berfungsi sebagai *username* dan *password*.

c. *Quick Design*

Tahap ini merupakan proses dari pembuatan *mockup* atau design tampilan dari halaman mengelola produk dan mengelola brand. Design mockup dapat dilihat pada gambar 4.14 dan 4.15

Gambar 4.14 *Mockup* Halaman Login

Pada gambar 4.14 diatas merupakan design dari tampilan login.

Gambar 4.15 *Mockup* Halaman Register

Pada gambar 4.15 diatas merupakan tampilan dari halaman register. Berikut adalah design dari database yang berupa tabel user.

Tabel 4.29 Tabel User

No	Nama Field	Jenis	Keterangan
1.	id	int(17)	<i>Primary Key</i>
2.	NamaDepan	varchar(20)	
3.	NamaBelakang	Varchar(20)	
4.	email	Varchar(20)	
6.	password	Varchar(20)	
7.	alamat	Varchar(20)	
8.	status	Tinyint(1)	
9.	Level	Tinyint(1)	

Pada Tabel 4.29 diatas merupakan tabel user yang terdapat pada database.

Terdiri dari 9 field dengan ID sebagai *Primary Key* nya.

d. *Construction of Prototype*

Pada tahap ini yaitu mengimplementasikan code sesuai design yang telah dibangun.

```

1) function checkLogin($username, $password) {
2)   $this->db->select('*');
3)   $this->db->where('email', $username);
4)   $this->db->where('password', md5($password));
5)   $this->db->where('status', 1);
6)   $query = $this->db->get($this->table, 1);

7)   if ($query->num_rows() == 1) {
8)     return $query->row_array();
9)   }
10) }
```

Kode Program 4.27 Fungsi *Check Login*

Pada Kode Program 4.27 merupakan fungsi checklogin yang ada pada model. Fungsi cek login dipanggil ketika username(email) dan password di inputkan pada form login.

```

1) function login() {
2) $this->form_validation->set_rules('email', 'email',
   'required|valid_email');
3) $this->form_validation->set_rules('password', 'password',
   'required');
4) $this->form_validation->set_error_delimiters('', '<br/>');
5) if ($this->form_validation->run() == TRUE) {
6) $email = $this->input->post('email');
7) $password = $this->input->post('password');
8) $user = $this->User->checkLogin($email, $password);
9) if (!empty($user)) {
10) $sessionData['id'] = $user['id'];
11) $sessionData['email'] = $user['email'];
12) $sessionData['NamaDepan'] = $user['NamaDepan'];
13) $sessionData['NamaBelakang'] = $user['NamaBelakang'];
14) $sessionData['level'] = $user['level'];
15) $sessionData['is_login'] = TRUE;
16) $this->session->set_userdata($sessionData);
17) $this->User->updateLastLogin($user['id']);
18) if ($this->session->userdata('level') == 1) {
19) redirect('admin/dashboard');
20) } else {
21) redirect('users/beranda');
22) }
23) }
24) $this->session->set_flashdata('error', 'Login Gagal!, Email
   Atau Password Tidak Cocok ');
25) redirect('users/login');
26) }
27) $data['content'] = 'user/login';
28) $this->load->view($this->template, $data);
29) }

```

Kode Program 4.28 Fungsi *Login*

Pada Kode Program 4.28 diatas merupakan fungsi login pada controller. Inputan email dan password di validasi terlebih dahulu kemudian dilakukan pengecekan level yang ada pada database. Level berfungsi membedakan administrator dengan user produsen biasa.


```

1) function register() {
2) $this->form_validation->set_rules('NamaDepan', 'Nama Depan',
   'trim|required');
3) $this->form_validation->set_rules('NamaBelakang', 'Nama
   Belakang', 'trim|required');
4) $this->form_validation->set_rules('email', 'Email',
   'trim|required|valid_email|is_unique[users.email]');
5) $this->form_validation->set_rules('password', 'Password',
   'trim|required|min_length[6]|max_length[16]');
6) $this->form_validation->set_rules('repass', 'Re-Password',
   'trim|required|matches[password]');
7) $this->form_validation->set_rules('phone', 'Nomor Telp',
   'trim|required|');
8) $this->form_validation->set_rules('alamat', 'Alamat',
   'trim|required|');
9) if ($this->form_validation->run() == TRUE) {
10) $this->User->create();
11) $this->session->set_flashdata('success', 'Pendaftaran Sukses
   !');
12) redirect('users/login');
13) }
14) $data['content'] = 'user/register';
15) $this->load->view($this->template, $data);
16) }

```

Kode Program 4.29 Fungsi Register

Pada Kode Program 4.29 diatas merupakan fungsi register pada controller. Apabila validasi berhasil maka registrasi akan berhasil. Apabila data yang diisi salah atau kurang, maka proses registrasi akan gagal.

```

1) function create() {
2) $data = array(
3) 'NamaDepan' => $this->input->post('NamaDepan'),
4) 'NamaBelakang' => $this->input->post('NamaBelakang'),
5) 'email' => $this->input->post('email'),
6) 'password' => md5($this->input->post('password')),
7) 'phone' => $this->input->post('phone'),
8) 'alamat' => $this->input->post('alamat'),
9) 'level' => 0,
10) 'status' => 1
11) );
12) $this->db->insert($this->table, $data);
13) }

```

Kode Program 4.30 Fungsi Tambah User

Pada Kode Program 4.30 diatas merupakan fungsi dari create user yang ada pada model. Fungsi create inilah yang nantinya akan berperan menambah data pada database apabila registrasi berhasil dilakukan.

e. *Deployment Delivery and Feedback*

Pada tahap ini merupakan tahapan pengujian pada iterasi login dan register, apabila pengguna tidak sesuai dengan hasil yang di harapkan maka pengembang memperbaiki sistem tersebut sehingga aplikasi atau sistem yang di buat sesuai dengan yang di inginkan oleh pengguna pengujian ini menggunakan black box.

1) *Deployment Delivery login*

Gambar 4.16 Halaman Login

Pada gambar 4.16 diatas merupakan tampilan dari halaman login yang sudah di aplikasikan pada *website*.

2) *Feedback login*

Tabel 4.30 *Feedback Login*

No	Skenario Pengujian	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1.	Memasukkan data username dan password dengan benar kemudian klik login	Email <input type="text" value="admin@gmail"/> Password <input type="password" value="*****"/> Lupa Password ? <input type="button" value="Login"/>	Data yang dimasukkan benar dan menampilkan halaman menu		Valid

Pada Tabel 4.30 merupakan data hasil dari pengujian proses login yang telah dilakukan.

3) *Deployment Delivery* Register

FORMREGISTER

Nama Depan

Nama Belakang

Email

Password

Confirm Password

Phone

Alamat

Daftar

Gambar 4.17 Halaman Register

Pada gambar 4.17 diatas merupakan tampilan dari halaman register yang sudah di aplikasikan pada *website*.

4) *Feedback* Register

Tabel 4.31 *Feedback* Register

No	Skenario Pengujian	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1.	Memasukkan data dengan benar kemudian klik daftar	Nama Depan Ahmad Nama Belakang Zikro Email zikro@gmail.com Password ***** Confirm Password ***** Phone 0965768890 Alamat babuwang	Data yang dimasukkan benar, tampil halaman login	FORMLOGIN Valid Jika Anda Mempunyai Akun S Pendaftaran Sukses ! Email <input type="text"/> Password <input type="password"/> Lupa Password ? Login	Valid

Pada Tabel 4.31 merupakan data hasil dari pengujian proses register yang telah dilakukan.

4.4.4 Iterasi 4

Pada iterasi 4 dilakukan pembuatan aplikasi yang meliputi tampilan halaman utama dan menggunakan filter kategori. Berikut adalah tahapan prototype pada pembuatan proses mengelola halaman utama dan mengelola filter.

a. *Communication*

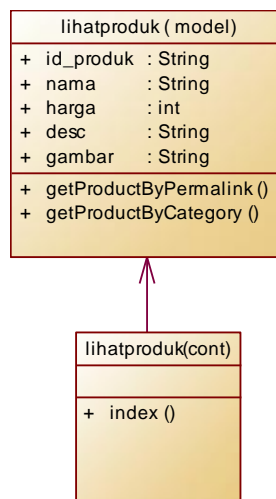
Pada tahap *Communication*, usecase yang digunakan adalah proses yang dapat dilakukan dari sisi user konsumen. Use Case yang dilakukan adalah melihat produk dan menggunakan filter.

b. *Quick Plan*

Pada tahap *Quick Plan*, macam-macam usecase yang digunakan adalah:

- 1) melihat iklan : Skenario usecase mengacu pada tabel 4.21
- 2) Menggunakan Filter: Skenario usecase mengacu pada tabel 4.22

Sementara *class diagram* mengacu pada gambar 4.18 berikut ini.

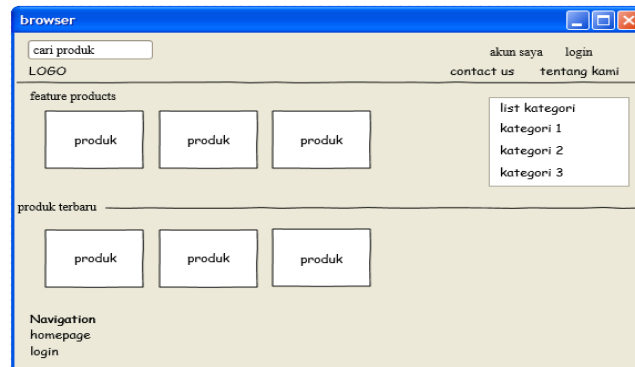


Gambar 4.18 *Class Diagram* Melihat Iklan

Pada Gambar 4.48 merupakan gambaran dari data dan perintah yang digunakan pada tahap ini. Karena proses ini dilakukan oleh konsumen umum maka tidak diperlukan data apapun dari pihak pengguna (konsumen). Terdapat perintah `getProductByPermalink()` untuk melihat detail produk dan `getProductByCategory()` untuk mencari produk berdasarkan kategorinya.

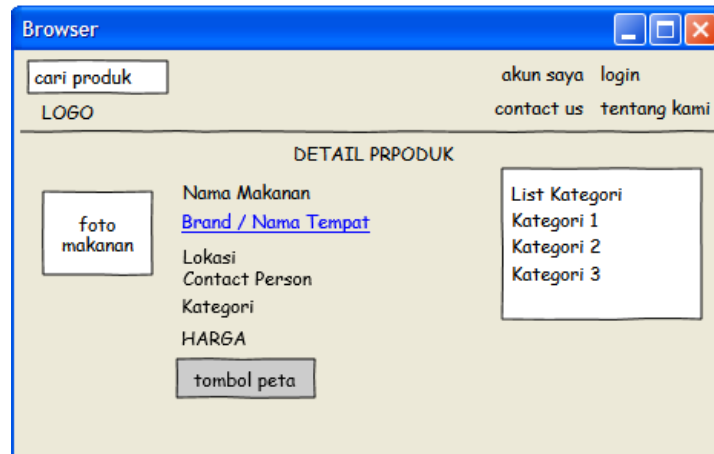
c. *Quick Design*

Tahap ini merupakan proses dari pembuatan *mockup* atau design tampilan dari halaman utama. Design mockup dapat dilihat pada gambar 4.19.



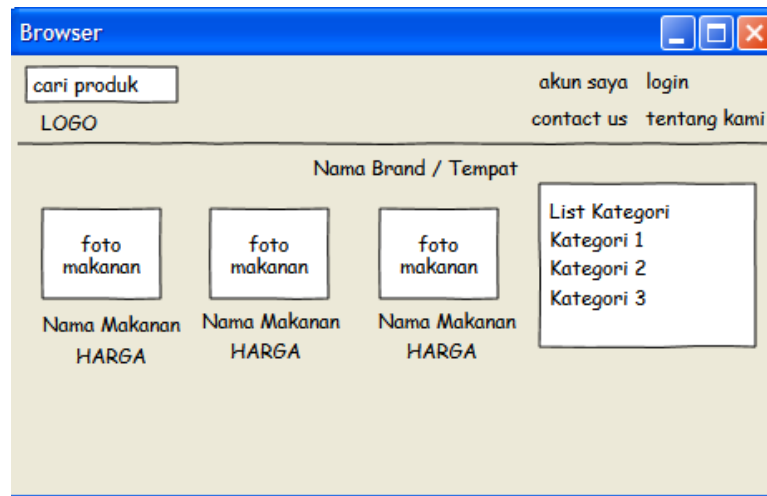
Gambar 4.19 *Mockup* Halaman Utama

Pada gambar 4.19 merupakan tampilan dari halaman utama sistem informasi pengiklanan makanan berbasis *website*.



Gambar 4.20 *Mockup* Halaman Detail Produk

Pada Gambar 4.20 merupakan tampilan dari halaman detail produk. Halaman ini muncul ketika nama makanan yang ada pada halaman utama di klik. Terdapat informasi mengenai nama makanan, nama tempat yang berupa link, lokasi atau alamat, nomor telepon dari tempat makan tersebut dan juga harga. Selain itu terdapat tombol untuk melihat peta dari lokasi tempat makan.



Gambar 4.21 *Mockup* Halaman Brand

Pada Gambar 4.21 merupakan tampilan dari halaman brand atau nama tempat makan. Karena satu brand bisa menjual beberapa makanan, sehingga halaman ini dapat mempermudah pengguna (konsumen) untuk mencari makanan apa saja yang terdapat pada satu brand atau tempat makan. Halaman ini muncul ketika link nama tempat makan yang ada halaman detail produk di klik.

d. *Construction of Prototype*

Pada tahap ini yaitu mengimplementasikan code sesuai design yang telah dibangun.

```

1) function index($page = null) {
2) $products = $this->Products->getProductsPublished();
3) $config['total_rows'] = $this->db-> count_all('produk');
4) $config['uri_segment'] = 9;
5) $config['per_page'] = $this->general->
   getSetting('paginationLimit');
6) $config['base_url'] = base_url() . 'product/index/lihat';
7) $config['full_tag_open'] = '<div class="pagination
   pagination-small pagination-centered"><ul>';
8) $config['full_tag_close'] = '</ul></div>';
9) $config['first_link'] = false;
10) $config['last_link'] = false;
11) $config['first_tag_open'] = '<li>';
12) $config['first_tag_close'] = '</li>';
13) $config['prev_tag_open'] = '<li>';
14) $config['prev_tag_close'] = '</li>';
15) $config['next_tag_open'] = '<li>';
16) $config['next_tag_close'] = '</li>';
17) $config['last_tag_open'] = '<li>';
18) $config['last_tag_close'] = '</li>';
19) $config['cur_tag_open'] = '<li class="active"><a href="#">';
20) $config['cur_tag_close'] = '</a></li>';
21) $config['num_tag_open'] = '<li>';
22) $config['num_tag_close'] = '</li>';
23) $this->pagination->initialize($config);
24) if ($this->input->get('q')) {
25) $data['products'] = $this->Products-
   >getProductsPublished($config['per_page'],$this->uri-
   >segment(4), $this->input->get('q'));
26) if (empty($data['products'])) {
27) $this->session->set_flashdata('error', 'Maaf Produk yang
   Dicari Tidak ditemukan !');
28) redirect('product');
29) }
30) } else {
31) $data['products'] = $this->Products-
   >getProductsPublished($config['per_page'], $this->uri-
   >segment(4));
32) }
33) $data['pagination'] = $this->pagination->create_links();
34) $data['content'] = 'products/index';
35) $this->load->view($this->template, $data);
36) }

```

Kode Program 4.31 Fungsi Index Halaman Utama

Pada Kode Program 4.31 merupakan fungsi dari index halaman utama. Ketika website dibuka, maka fungsi inilah yang dipanggil.

```

1) function getProductsPublished($perPage='', $offset = 0, $key
   = null) {
2) $this->db->select('p.id, p.nama,p.permalink,
   p.deskripsi,p.harga_net, p.gambar,p.brand,p.enabled,
   c.kategori as categoryName, b.brand as warung');
3) $this->db->join('kategori c', 'c.kat_id = p.kategori');
4) $this->db->join('brand b', 'b.id = p.brand');
5) $this->db->where('p.enabled', 1);
6) $this->db->limit($perPage, $offset);
7) if ($key != null) {
8) $this->db->like('p.nama', $key);
9) $this->db->or_like('p.kategori', $key);
10) }
11) $query = $this->db->get('produk p');
12) if ($query->num_rows() > 0) {
13) return $query->result_array();
14) }
15) }

```

Kode Program 4.31 Fungsi Menampilkan Produk

Pada Kode Program 4.31 merupakan model dari tampilan halaman awal. Fungsi akan memanggil produk yang ada pada database dengan brand dan kategorinya. Apabila kategori atau brand tidak ada, maka produk tidak dapat tampil di halaman utama.

```

1) function getProductsByCategoryId($categoryId, $perPage =3,
   $offset = 0) {
2) $this->db->select('p.id, p.nama,p.permalink,
   p.deskripsi,p.harga_net, p.gambar, p.enabled, c.kategori as
   categoryName');
3) $this->db->join('kategori c','c.kat_id = p.kategori');
4) $this->db->where('p.kategori', $categoryId);
5) $this->db->where('p.enabled', 1);
6) $this->db->limit($perPage, $offset);
7) $query = $this->db->get('produk p');
8) if ($query->num_rows() > 0) {
9) return $query->result_array();
10) }
11) }

```

Kode Program 4.32 Fungsi *getProductById*

Pada Kode Program 4.32 merupakan fungsi yang dipanggil ketika filter kategori digunakan. Produk-produk yang akan tampil di halaman utama hanyalah produk-produk yang sesuai dengan kategori yang dipilih.


```

1) function getProductByPermalink($permalink) {
2) $this->db->select('b.id, p.nama,p.permalink,
3) p.deskripsi,p.kategori,p.harga_net,p.harga,p.diskon,p.gambar,
4) b.cp as cp, b.alamat as alamat,b.brand as warung,c.kategori
5) as categoryName');
6) $this->db->join('kategori c', 'c.kat_id = p.kategori');
7) $this->db->join('brand b', 'b.id = p.brand');
8) $this->db->where('p.permalink', $permalink);
9) $query = $this->db->get('produk p', 1);
10) if ($query->num_rows() == 1) {
11) return $query->row_array();
12) }
13) }

```

Kode Program 4.33 Fungsi *getProductByPermalink*

Pada Kode Program 4.33 diatas merupakan fungsi yang dipanggil ketika ingin melihat detail dari produk yang kita pilih.

```

1) function detail_warung($brand) {
2) $perPage =1000;
3) $offset = 0;
4) $data['detail'] = $this->Products->getProductByBrand($brand,
5) $perPage, $offset)->result();
6) $data['merk'] = $this->Products->getMerk($brand)->result();
7) $data['pagination'] = $this->pagination->create_links();
8) $data['content'] = 'products/detail_warung';
9) $this->load->view($this->template, $data);
10) }

```

Kode Program 4.34 Fungsi Detail Brand

Pada Kode Program 4.34 merupakan fungsi pada *Controller* yang dipanggil ketika ingin melihat makanan apa saja yang ada dari salah satu brand.

```

1) function getProductByBrand($brand, $perPage, $offset) {
2) $this->db->select('p.id, p.nama,p.permalink, p.deskripsi,
3) p.harga_net, p.gambar,p.brand,p.enabled, c.kategori
4) as categoryName, b.brand as warung');
5) $this->db->join('kategori c', 'c.kat_id = p.kategori');
6) $this->db->join('brand b', 'b.id = p.brand');
7) $this->db->where('p.brand', $brand);
8) $this->db->where('p.enabled', 1);
9) $this->db->limit($perPage, $offset);
10) return $this->db->get('produk p');
11) }

```

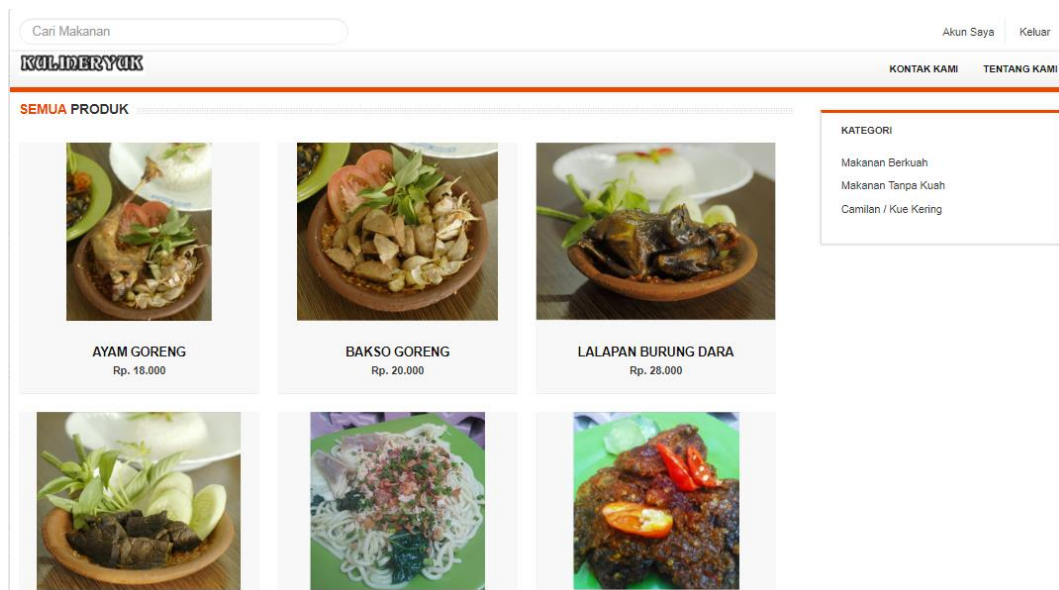
Kode Program 4.35 Fungsi *getProductByBrand*

Pada Kode Program 4.35 merupakan fungsi pada Model yang dipanggil ketika ingin melihat makanan apa saja yang terdapat di salah satu brand atau nama tempat makan. Fungsi inilah yang berperan sebagai pem-filter data-data yang

terdapat di database. Sehingga data-data yang muncul adalah data-data yang sesuai dengan apa yang di inginkan.

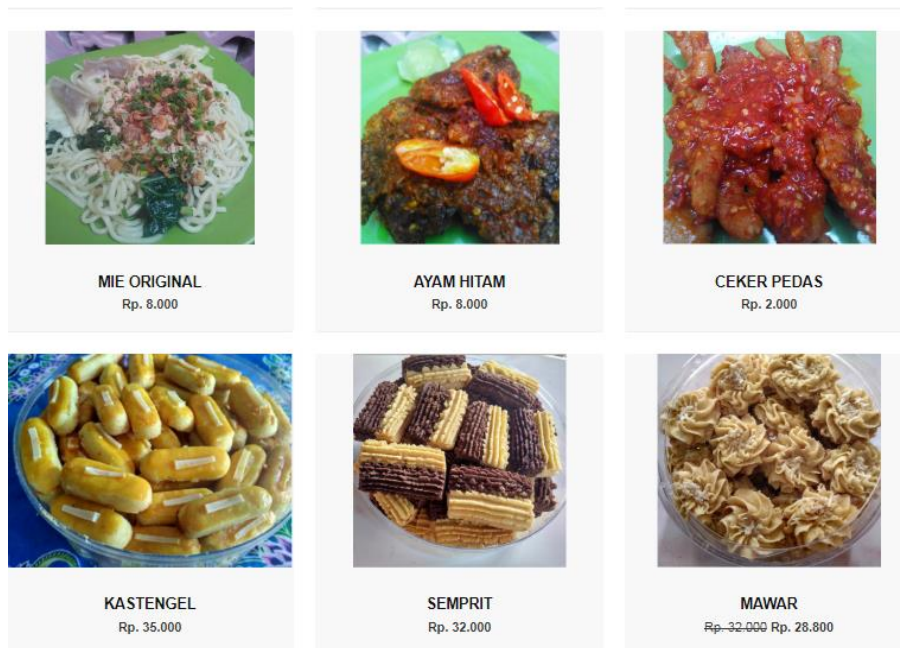
e. Deployment Delivery and Feedback

1) Deployment Delivery



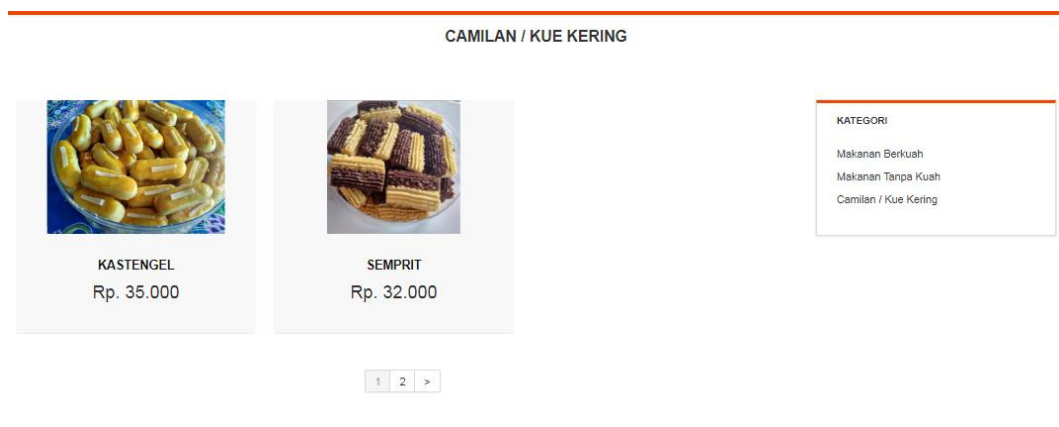
Gambar 4.22 Halaman Utama

Pada gambar 4.22 merupakan tampilan dari halaman utama yang sudah di aplikasikan pada *website*.



Gambar 4.23 Tampilan Diskon

Pada gambar 4.23 diatas merupakan potongan dari halaman utama. Terdapat sebuah produk makanan yang memiliki 2 harga. Gambar diatas merupakan tampilan yang akan muncul apabila produk makanan yang di iklankan sedang memiliki diskon.



Gambar 4.24 Tampilan Filter Kategori

Pada gambar 4.24 merupakan hasil dari penggunaan filter kategori. Ketika kategori di pilih, maka produk-produk yang termasuk kategori tersebut yang akan muncul.



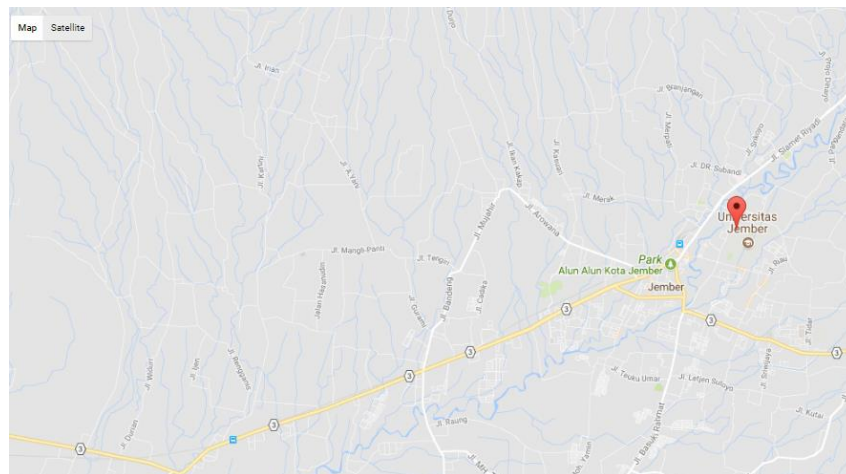
Gambar 4.25 Tampilan Detail Produk

Pada gambar 4.25 diatas merupakan tampilan detail produk. Menampilkan data-data dari alamat, warung/brand dan juga lokasi dari warung tersebut dengan bantuan *google maps*. Selain itu ketika nama brand di klik maka akan muncul halaman dari brand tersebut.



Gambar 4.26 Halaman Brand untuk Konsumen

Pada Gambar 4.26 merupakan tampilan dari halaman brand yang muncul ketika link di halaman detail produk di klik. Halaman ini berisi makanan-makanan yang berasal dari satu brand yang sama.



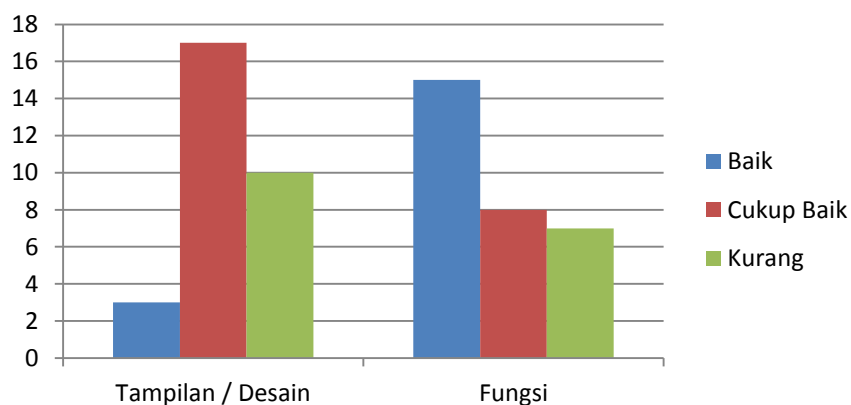
Gambar 4.27 Tampilan Lokasi

Pada gambar 4.27 diatas merupakan tampilan lokasi dari warung/brand berdasarkan produk yang kita pilih.

4.5 Hasil Kuesioner

Untuk mengetahui bagaimana kualitas dari sistem infomasi ini maka developer melakukan pengajuan kuesioner terhadap 30 responden dengan terlebih dahulu mencoba sistem informasi pengiklanan makanan berbasis website ini dengan hasil sebagai berikut :

Jumlah responden : 30 orang



Gambar 4.28 Grafik Hasil Pengujian Sistem Informasi

Pada Gambar 4.28 diatas menunjukkan hasil dari pengujian sistem yang dilakukan dengan menyebar kuesioner pada 30 responden. Responden akan diminta menjalankan sistem informasi tersebut kemudian mengisi kuesioner yang

terdiri dari 2 pertanyaan yaitu bagaimana tampilan atau desain yang telah dibangun dan apakah fungsi dari sistem informasi pengiklanan ini berjalan dengan baik atau tidak.

BAB 5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil pembahasan sistem informasi pengiklanan makanan berbasis *website* maka kesimpulannya adalah :

- a. Sistem informasi ini dapat membantu masyarakat mencari referensi kuliner dengan adanya fitur pencarian berdasarkan nama makanan dan juga pengelompokan makanan berdasarkan kategori-kategori yang ada.
- b. Sistem informasi ini membantu produsen makanan untuk memasarkan produk-produk mereka dan dilengkapi dengan informasi yang lengkap dan jelas seperti nama produk, brand, alamat, nomor telepon yang bisa dihubungi, diskon dan selain itu juga terdapat fitur lokasi yang memudahkan pihak konsumen untuk mencari letak restoran atau penjual tersebut berada.

5.2 Saran

Berdasarkan kesimpulan-kesimpulan yang telah dikemukakan. Dapat diajukan beberapa saran untuk pengembangan lebih lanjut :

- a. Menambahkan fitur lokasi terdekat pada maps sehingga mempermudah pengguna untuk menemukan lokasi kuliner yang terdekat dari lokasi pengguna.
- b. Mengaplikasikan sistem informasi pengiklanan makanan berbasis *website* ini dalam bentuk *mobile* sehingga penggunaannya lebih fleksibel.
- c. Berdasarkan hasil dari kuesioner yang disebar maka perlu adanya pengembangan dari segi desain agar desain lebih rapi dan menarik minat dari masyarakat.

DAFTAR PUSTAKA

- Benner. 2010. Pengertian Sistem Informasi. <http://dosenit.com/kuliah-it/sistem-informasi/pengertian-sistem-informa>. [06 Januari 2017].
- Chandramitra,Citra.2012.*Metode Prototyping Dalam Pengembangan Sistem Informasi*.http://www.academia.edu/10561240/Metode_Prototyping_Dalam_Pengembangan_Sistem_Informasi [04 Mei 2016]
- Dwi,Adi.2010.*Analisis dan Perancangan Sistem Informasi Berbasis Web Sebagai Media Promosi dan Informasi Budi Watch Collecton*.Sistem Informasi Amikom Yogyakarta.<http://repository.amikom.ac.id/> [04 Januari 2017]
- Laudon.2010.Pengertian Sistem Informasi. <http://dosenit.com/kuliah-it/sistem-informasi/pengertian-sistem-informa>. [06 Januari 2017].
- Lee dan Johnson.2007.Periklanan.http://digilib.mercubuana.ac.id/manager/n!@file_skripsi/Isi2786946147784.pdf. [06 Januari 2017].
- Pressman,Roger.2010.*Software Engineering : A Practicioner's Approach Seventh Edition*. Singapore.McGrawHill
- Rahmatullah,Rizky.*Media Informasi dan Periklanan Pasar Tradisional Berbasis Web Study Kasus Pasar Tradisional Kabupaten Bandung*.Teknik Telekomunikasi Universitas Telkom.http://library.telkomuniversity.ac.id/pustaka/files/102554/jurnal_eproc [04 Januari 2017]
- Rossa dan Salahudin.2013.*Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*.Bandung.Informatika Bandung
- Zamroji,Andi.2016. Aplikasi Penentuan Formulasi Ayam Kampung Pedaging. MIF. Politeknik Negeri Jember.