

**APLIKASI SISTEM INFORMASI KEUANGAN
BERBASIS WEB MENGGUNAKAN KONSEP MVC
(*MODEL, VIEW, CONTROL*) DENGAN
FRAMEWORK LARAVEL UNTUK PENCATATAN
KEUANGAN DI KELOMPOK PETANI SAYUR
HIDROPONIK ASRI 12 KAUMAN
YOGYAKARTA.**



Disusun Oleh :

SAFRI ADAM (12018175)

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS AHMAD DAHLAN
YOGYAKARTA**

2016

LEMBAR PERSETUJUAN

SKRIPSI

**APLIKASI SISTEM INFORMASI KEUANGAN BERBASIS
WEB MENGGUNAKAN KONSEP MVC (*MODEL, VIEW,
CONTROL*) DENGAN FRAMEWORK LARAVEL UNTUK
PENCATATAN KEUANGAN DI KELOMPOK PETANI
SAYUR HIDROPONIK ASRI 12 KAUMAN
YOGYAKARTA.**

Dipersiapkan dan disusun oleh :

SAFRI ADAM

12018175

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS AHMAD DAHLAN
YOGYAKARTA**

Telah disetujui oleh:

Pembimbing

Arfiani Nur Khusna, S.T, M.Kom.

HALAMAN PENGESAHAN

SKRIPSI

APLIKASI SISTEM INFORMASI KEUANGAN BERBASIS WEB MENGGUNAKAN KONSEP MVC (*MODEL, VIEW, CONTROL*) DENGAN FRAMEWORK LARAVEL UNTUK PENCATATAN KEUANGAN DI KELOMPOK PETANI SAYUR HIDROPONIK ASRI 12 KAUMAN YOGYAKARTA.

Yang dipersiapkan dan disusun oleh

Safri Adam

12018175

Telah dipertahankan di depan Dewan Penguji
Pada tanggal 16 Januari 2017
Dan dinyatakan telah memenuhi syarat

Susunan Dewan Penguji :

Ketua	: Arfiani Nur Khusna, S.T., M.Kom.
Penguji I	: Sri Handayaningsih, S.T., M.T.
Penguji II	: Drs.Wahyu Pujiono, M.Kom.

Yogyakarta, 16 Januari 2017

Dekan

Fakultas Teknologi Industri

Universitas Ahmad Dahlan

Kartika Firdausy, S.T., M.T.

HALAMAN PERSEMBAHAN

Assalamualaikum.wr.wb

Alhamdulillah puji syukur saya panjatkan kepada Allah.S.W.T yang telah memberi saya nikmat yang luas atas rampungnya skripsi ini. Skripsi ini saya persembahkan untuk kedua orang tua saya, Suminar K dan Achmad Hamka yang telah mendukung saya secara lahir dan batin tidak hanya saat menyusun skripsi ini, tetapi sejak awal saya dilahirkan hingga saat ini.

Terima kasih yang sebesar- besarnya saya ucapkan kepada :

- Teman - teman Sedulur kelas D Informatika UAD 2012 yang mana telah memberi saya semangat dan motivasi selama menyusun skripsi. Terima kasih atas kepercayaan teman-teman mengangkat saya sebagai kepala suku di kelas.
- Keluarga Besar Asrama Mahasiswa Kalimantan Barat “RAHADI OSMAN 1” Yogyakarta yang telah memberi saya pengalaman, tempat tinggal dan teman selama menempuh pendidikan di kota Yogyakarta.
- Kedua kakak saya Ayu Umyana dan Fitri Hani Pratiwi yang banyak sekali berkontribusi terhadap terselesainya tugas akhir ini. Terima kasih atas semangat dan ilmu yang telah diberikan untuk adikmu ini.
- Ketua Kelompok Petani Hidroponik ASRI 12 Kauman Yogyakarta, Ibu Sumarni atas kerjasamanya dalam penyusunan tugas akhir ini. Terima kasih telah menerima saya sebagai partner kerjasama.

Serta semua pihak yang terkait dalam penyusunan skripsi ini, semoga kita semua selalu dalam lindungan Allah Azza wa jalla, Aamiin.

Jazakallahu Khairan, Wassalamualaikum.wr.wb.

MOTTO

وَمَا خَلَقْتُ الْجِنَّ وَالْإِنْسَ إِلَّا لِيَعْبُدُونِ ﴿٥٦﴾

Wa ma - khalaqtul-jinna wal-insa illa- liya'budu-n(i)

“Aku tidak menciptakan jin dan manusia melainkan agar mereka beribadah kepada-Ku.”

(Q.S. Az-Zariyat Ayat 56)

KATA PENGANTAR



Assalamu'alaikum Wr. Wb.

Dengan mengucapkan *Alhamdulillahirobbil'alamin*, penulis memanjatkan puji dan syukur kehadiran Allah SWT atas rahmat, nikmat dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi dengan judul “Aplikasi Sistem Informasi Keuangan Berbasis Web Menggunakan Konsep MCV Dengan Framework Laravel Untuk Pencatatan Keuangan Di Kelompok Petani Sayur Hidroponik ASRI 12 Kauman Yogyakarta.” dengan baik. Skripsi ini disusun untuk memenuhi sebagian persyaratan mencapai derajat Sarjana Teknik di Teknik Informatika Fakultas Teknologi Industri di Universitas Ahmad Dahlan Yogyakarta.

Selama penulisan skripsi ini telah banyak pihak yang terlibat memberikan bantuannya, sehingga pada kesempatan kali ini penulis mengucapkan terima kasih yang sebesar besarnya kepada :

1. Ibu Kartika Firdausy, S.T., M.T., selaku Dekan Fakultas Teknologi Industri Universitas Ahmad Dahlan.
2. Ibu Sri Winiarti, S.T., M.Cs., selaku Kaprodi Teknik Informatika Universitas Ahmad Dahlan Yogyakarta.
3. Ibu Sri Handayaningsih, S.T.,M.T., dan Bapak Drs.Wahyu Pujiono, M.Kom. selaku dosen penguji yang telah menyetujui, menerima dan memberikan pengarahan pada laporan Tugas Akhir ini.
4. Ibu Arfiani Nur Khusna, S.T., M.Kom selaku dosen pembimbing yang telah memberikan motivasi, dukungan, pengarahan, pelajaran dan dengan sabar dalam memberikan bimbingan.

Harapan penulis, semoga doa dan bantuan yang sangat berharga tersebut mendapat imbalan dari Allah SWT, amin ya robal alamin.

Jazakallahu Khairan, Wassalamualaikum.wr.wb.

Yogyakarta, 4 Januari 2017

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERSETUJUAN	ii
HALAMAN PENGESAHAN	iii
HALAMAN PERSEMBAHAN	iv
MOTTO	v
KATA PENGANTAR.....	vi
DAFTAR ISI	vii
DAFTAR GAMBAR DAN RUMUS	x
DAFTAR TABEL	xiii
DAFTAR LISTING	xv
HALAMAN PERNYATAAN	xvi
ABSTRAK.....	xvii

BAB I PENDAHULUAN

A. Latar Belakang Masalah	1
B. Identifikasi Masalah	5
C. Batasan Masalah	5
D. Rumusan Masalah	6
E. Tujuan Penelitian	6
F. Manfaat Penelitian.....	7

BAB II KAJIAN PUSTAKA

A. Kajian Penelitian Terdahulu	8
B. Landasan Teori	14
1. Sistem Informasi	14
2. Laporan Laba Rugi.....	16
3. Harga Pokok.....	17

4. Sistem Informasi Keuangan	18
5. PHP.....	20
6. MySQL.....	22
7. Framework Laravel	23

BAB III METODOLOGI PENELITIAN

A. Subyek Penelitian.....	25
B. Metode Pengumpulan Data	25
1. Metode Wawancara	26
2. Survey Lokasi	26
C. Alat dan Bahan.....	26
D. Analisis Kebutuhan Pengguna.....	27
E. Analisis Kebutuhan Sistem.....	28
1. Kebutuhan Fungsional	28
2. Kebutuhan Non-Fungsional	28
F. Desain Sistem (Pemodelan)	28
G. Implementasi	30
1. Model	30
2. Controller	30
3. View	31
H. Pengujian Sistem.....	31

BAB IV HASIL DAN PEMBAHASAN

A. Analisa Kebutuhan Pengguna	33
B. Analisis Kebutuhan Sistem.....	34
1. Kebutuhan Fungsional	34
2. Kebutuhan Non-Fungsional	36
C. Desain Sistem (Pemodelan)	51

D. Implementasi	87
E. Pengujian Sistem.....	102

BAB V KESIMPULAN DAN SARAN

A. Kesimpulan.....	106
B. Saran	106
DAFTAR PUSTAKA.....	108
LAMPIRAN	110

DAFTAR GAMBAR DAN RUMUS

Gambar 2.0 gambaran umum sistem informasi	14
Rumus (1) rumus laporan laba rugi.....	16
Rumus (2) rumus laba	16
Rumus (3) rumus rugi	16
Rumus (4) harga pokok	17
Rumus (5) menentukan profit	17
Gambar 4.1 <i>use case diagram</i>	38
Gambar 4.2 activity diagram login	51
Gambar 4.3 <i>activity diagram</i> tambah pengguna	52
Gambar 4.4 <i>Activity Diagram</i> tambah Anggaran	53
Gambar 4.5 <i>Activity Diagram</i> cetak anggaran	54
Gambar 4.6 <i>Activity Diagram</i> tambah realisasi	55
Gambar 4.7 <i>Activity Diagram</i> tampil realisasi	56
Gambar 4.8 <i>Activity Diagram</i> cetak realisasi	57
Gambar 4.9 <i>Activity Diagram</i> tambah pemasukan	58
Gambar 4.10 <i>Activity Diagram</i> tampil pemasukan	59
Gambar 4.11 <i>Activity Diagram</i> cetak pemasukan	60
Gambar 4.12 <i>Activity Diagram</i> tampil laporan laba-rugi	61
Gambar 4.13 <i>Activity Diagram</i> tampil laporan keuangan bulanan	62
Gambar 4.14 <i>Activity Diagram</i> cetak laporan laba-rugi	63

Gambar 4.15 <i>Activity Diagram</i> cetak laporan keuangan bulanan	64
Gambar 4.16 <i>Activity Diagram</i> edit harga produk.....	65
Gambar 4.17 <i>Activity Diagram</i> tampil anggaran	66
Gambar 4.18 <i>Activity Diagram</i> subsistem pengguna	67
Gambar 4.19 <i>Activity Diagram</i> edit profil pengguna	68
Gambar 4.20 <i>Activity Diagram</i> logout	69
Gambar 4.21 ERD (<i>Entity Relationship Diagram</i>).....	70
Gambar 4.22 <i>Mapping Table</i>	71
Gambar 4.23 struktur menu sistem informasi keuangan	77
Gambar 4.24 desain halaman login	77
Gambar 4.25 desain halaman <i>dashboard</i>	78
Gambar 4.26 desain halaman pengguna	78
Gambar 4.27 desain halaman tambah pengguna	79
Gambar 4.28 desain halaman Anggaran	80
Gambar 4.29 desain halaman tambah Anggaran	81
Gambar 4.30 desain halaman pemasukan	81
Gambar 4.31 desain halaman tambah pemasukan	82
Gambar 4.32 desain halaman realisasi	83
Gambar 4.33 desain halaman tambah realisasi	83
Gambar 4.34 desain halaman laporan.....	84

Gambar 4.35 desain halaman laporan laba-rugi	85
Gambar 4.36 desain halaman laporan keuangan bulanan	85
Gambar 4.37 desain halaman harga jual produk	86
Gambar 4.38 asritektur sistem aplikasi sistem informasi keuangan	87
Gambar 4.39 tampilan halaman pengguna	94
Gambar 4.40 tampilan tambah anggaran	95
Gambar 4.41 tampilan halaman tambah realisasi	98
Gambar 4.42 tampilan halaman tambah pemasukan	99
Gambar 4.43 tampilan halaman laporan laba rugi	100
Gambar 4.44 tampilan halaman harga jual produk	102

DAFTAR TABEL

Tabel 2.1. tabel perbandingan kajian penelitian terdahulu.....	11
Tabel 4.1. tabel SRS Fungsional.....	36
Tabel 4.2 tabel <i>use case scenario</i> login	39
Tabel 4.3 tabel <i>use case scenario</i> input pengguna	39
Tabel 4.4 tabel <i>use case scenario</i> input Anggaran.....	40
Tabel 4.5 tabel <i>use case scenario</i> cetak Anggaran	40
Tabel 4.6 tabel <i>use case scenario</i> tambah realisasi	41
Tabel 4.7 tabel <i>use case scenario</i> tampil realisasi	41
Tabel 4.8 tabel <i>use case scenario</i> cetak realisasi	42
Tabel 4.9 tabel <i>use case scenario</i> tambah pemasukan	42
Tabel 4.10 tabel <i>use case scenario</i> tampil pemasukan.....	43
Tabel 4.11 tabel <i>use case scenario</i> cetak pemasukan	43
Tabel 4.12 tabel <i>use case scenario</i> tampil laporan laba-rugi.....	44
Tabel 4.13 tabel <i>use case scenario</i> tampil laporan keuangan bulanan	44
Tabel 4.14 tabel <i>use case scenario</i> tampil laporan keuangan tahunan.....	45
Tabel 4.15 tabel <i>use case scenario</i> cetak laporan laba-rugi	45
Tabel 4.16 tabel <i>use case scenario</i> cetak laporan keuangan bulanan.....	46
Tabel 4.17 tabel <i>use case scenario</i> tampil harga pokok	46

Tabel 4.18 tabel <i>use case scenario</i> tampil Anggaran	47
Tabel 4.19 tabel <i>use case scenario</i> tambah bibit, nutrisi, bahan lain	47
Tabel 4.20 tabel <i>use case scenario</i> edit harga produk	48
Tabel 4.21 tabel <i>use case scenario</i> logout	48
Tabel 4.22 tabel <i>use case scenario</i> non-aktifkan pengguna	49
Tabel 4.23 tabel <i>use case scenario</i> aktifkan pengguna	49
Tabel 4.24 tabel <i>use case scenario</i> edit profil pengguna	50
Tabel 4.25 tabel <i>use case scenario</i> reset password pengguna	50
Tabel 4.26 tabel users	72
Tabel 4.27 tabel anggarans	72
Tabel 4.28 tabel pemasukans.....	73
Tabel 4.29 tabel pengeluaranans.....	73
Tabel 4.30 tabel transakasis.....	74
Tabel 4.31 tabel labarugis	74
Tabel 4.32 tabel sayurans	75
Tabel 4.33 tabel bibits.....	75
Tabel 4.34 tabel hasil <i>black box test</i>	103
Tabel 4.34 tabel hasil pengujian <i>alpha test</i>	104

DAFTAR LISTING

Script 4.1 Model user.php.....	88
Script 4.2 Model Anggaran.....	88
Script 4.3 Model Realisasi	89
Script 4.4 Model Pemasukan	89
Script 4.5 Model Laba-Rugi	90
Script 4.6 Model Transaksi.....	90
Script 4.7 Model Harga Pokok.....	91
Script 4.8 Controller tambah, ubah, non-aktifkan, <i>reset password</i> pengguna	92
Script 4.9 Controller tambah Anggaran	94
Script 4.10 <i>javascript auto Calculate</i>	95
Script 4.11 controller tambah realisasi	96
Script 4.12 <i>javascript</i> kondisi membandingkan anggaran dan realisasi	97
Script 4.13 Controller Tambah Pemasukan	98
Script 4.14 Controller laporan Laba-Rugi	99
Script 4.15 Controller harga jual produk.....	100
Script 4.16 <i>javascript</i> hitung harga jual produk	101

SURAT PERNYATAAN

Yang bertanda tangan dibawah ini :

Nama : SAFRI ADAM

NIM : 12018175

Prodi : Teknik Informatika

Judul TA/ Skripsi : Aplikasi Sistem Informasi Keuangan Berbasis Web
Menggunakan Konsep MVC (Model, View,
Control) Dengan Framework Laravel Untuk
Pencatatan Keuangan Di Kelompok Petani Sayur
Hidroponik ASRI 12 Kauman Yogyakarta.

Dengan ini saya menyatakan bahwa Laporan Tugas Akhir in tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan disuatu Perguruan Tinggi. Dan sepanjang sepengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Yogyakarta, 4 Januari 2017

Mengetahui,
Dosen pembimbing

Yang menyatakan,
Mahasiswa

(Arfiani Nur Khusna, S.T, M.Kom)

(Safri Adam)

**Aplikasi Sistem Informasi Keuangan Berbasis Web Menggunakan Konsep
MVC (*Model, View, Control*) Dengan Framework Laravel Untuk Mengelola
Keuangan Di Kelompok Petani Sayur Hidroponik ASRI 12 Kauman
Yogyakarta**

Safri Adam

12018175

ABSTRAK

Kelompok petani sayur hidroponik di RW 12 kauman, Yogyakarta merupakan kelompok tani yang memiliki masalah dalam pengelolaan keuangan. Sehingga dampaknya terlihat ketika melihat hasil penjualan yang selalu merugi. Hal ini diakibatkan tidak adanya sistem yang menangani masalah keuangan di kelompok petani hidroponik ini.

Sistem yang akan dibuat berbasis web dan dibangun menggunakan *framework* laravel dengan konsep MVC (*Model, View, Control*). Fitur yang ada pada sistem ini antara lain dapat merancang anggaran bulanan, mendokumentasikan pengeluaran, dan pemasukan, serta dapat mencetak laporan keuangan bulanan dan laba-rugi setiap masa tanam yang telah ditentukan.

Dari penelitian yang dilakukan menghasilkan sebuah perangkat **aplikasi sistem informasi keuangan berbasis web menggunakan framework laravel menggunakan metode MVC (*Model, View, Control*)**. Hasil pengujian sistem menunjukkan bahwa aplikasi layak digunakan sesuai kebutuhan pengguna

Kata kunci : Sistem Informasi, Web, Keuangan, hidroponik , sayur hidroponik.

Bab I

Pendahuluan

A. Latar Belakang Masalah

Memasuki tahun 2016 dimana MEA secara menyeluruh akan diterapkan, MEA (Masyarakat Ekonomi ASEAN) merupakan sebuah gagasan dari pemimpin ASEAN dan seluruh Negara-negara ASEAN yang bertujuan untuk menciptakan pembangunan di negara ASEAN dengan melakukan integrasi ekonomi yaitu aliran bebas barang, jasa, investasi, dan tenaga kerja terdidik antar negara ASEAN. Indonesia, dengan penduduknya yang berjumlah 252 juta jiwa lebih akan menjadi sasaran perdagangan karena sangat berpotensi menjadi basis konsumsi yang besar. Oleh karena itu, jika Indonesia memiliki strategi yang baik, MEA 2016 dapat menjadi peluang besar Indonesia untuk membangun perekonomiannya.

Salah satu cara Indonesia untuk membangun perekonomiannya adalah dengan menambah sektor UMKM. UMKM (Usaha Mikro, Kecil dan Menengah) memiliki peran yang sangat besar dalam perekonomian Indonesia. Jumlah usaha mikro kecil dan menengah (UMKM) di Indonesia saat ini sekitar 55 juta, dan menyerap 97% tenaga kerja Indonesia. Meski secara kuantitas sangat besar dan menyerap banyak tenaga kerja, pangsa dalam pendapatan nasional masih sekitar 57%. Keadaan ini disebabkan oleh masalah internal yang dihadapi UMKM yaitu: rendahnya kualitas SDM UMKM dalam manajemen, organisasi, penguasaan teknologi, dan pemasaran, lemahnya kewirausahaan dari para pelaku UMKM, dan terbatasnya akses UMKM terhadap permodalan,

informasi, teknologi dan pasar, serta faktor produksi lainnya.(Anggarani, Prasetyoning, & Safitri, 2015)

Salah satu masalah internal terhambatnya pertumbuhan UMKM yaitu rendahnya penguasaan SDM terhadap manajemen dan penguasaan teknologi. Dari sisi kelemahan inilah dapat dibangun pondasi kekuatan untuk menyelesaikan kendala terhadap manajemen. Salah satu manajemen yang terpenting adalah manajemen keuangan. Yang terpenting dalam menjalankan usaha adalah manajemen keuangan yang baik. Salah satu caranya yaitu mengelola keuangan dengan ilmu akuntansi yang kemudian disinergikan dengan penguasaan teknologi informasi.

Kauman merupakan sebuah kampung yang terletak di pusat kota Yogyakarta, berada di kelurahan Ngupasan dan kecamatan Gondomanan, selatan Malioboro dan barat Kraton Ngayogyakarta. Saat ini warga Kauman yang meliputi 7 RT sedang menggalakkan budidaya tanaman menggunakan hidroponik meliputi sayuran selada, kangkung, sawi dan seledri. Setiap RW di Kauman memiliki beberapa kelompok tani yang masing-masing memiliki nama dan keunggulan dalam hal bercocok tanam menggunakan hidroponik. Salah satu kelompok petani hidroponik yang sudah sukses adalah kelompok tani Sakinah di RW 11. Bercermin kepada kelompok petani yang sudah sukses, maka membuat kelompok tani lain ingin memajukan kelompok tani masing-masing. Salah satunya yaitu kelompok tani RW 12.

Kelompok petani sayur hidroponik di RW 12 Kauman, Yogyakarta merupakan kelompok tani yang berdiri atas gagasan kelompok ibu-ibu PKK

yang berinisiatif menjadikan usaha sayur hidroponik sebagai UMKM bersama. Untuk memulai bertanam sayur hidroponik kelompok petani melakukan iuran untuk mengumpulkan modal membeli item keperluan bertanam sayur hidroponik. Setelah membeli item untuk keperluan bertanam, kelompok tani tidak mencatat secara detail pengeluaran yang ada. Hanya sebatas pertanggung jawaban melalui nota pembelian bahkan tidak semua item yang dibeli memiliki nota pembelian. Hal ini menyulitkan bendahara dalam mendokumentasikan setiap pengeluaran. Sesudah melewati masa tanam, berkisar 3 sampai 4 bulan, sayuran siap dipanen. Kelompok tani RW 12 selama bercocok tanam dengan hidroponik sudah 2 kali panen, namun hasil panen belum dapat dipasarkan secara optimal, pemasaran hanya di warung sekitar dengan harga kekeluargaan sehingga harga yang didapat sangat ekonomis atau murah.

Setelah mendapat pemasukan dari penjualan produk sayur hidroponik, uang yang didapat langsung masuk kedalam kas tanpa melalui pencatatan terlebih dahulu. Sehingga menyebabkan kelompok petani tidak mengetahui untung atau rugi dari penjualan produk sayuran mereka. Maka untuk membuka masa tanam baru, kelompok harus mengumpulkan modal lagi dengan iuran anggota kelompok bukan menggunakan uang hasil penjualan sebelumnya. Hal itu disebabkan belum adanya perencanaan, pengelolaan dan pencatatan dana yang sudah dikeluarkan untuk modal usaha.

Berdasarkan masalah diatas , maka kelompok tani hidroponik RW 12 memerlukan sebuah perencanaan dan pengelolaan keuangan yang nantinya

diharapkan akan menyelesaikan masalah diatas. Mengingat pada tahun ini sudah diterapkan MEA (Masyarakat Ekonomi Asean) yang diharapkan masyarakat menguasai teknologi informasi dalam bersaing dalam MEA, maka dibangunlah sebuah system informasi keuangan terkomputerisasi yang kemudian diharapkan dengan sistem ini dapat mempermudah kelompok petani dalam pencatatan keuangan. Dengan sistem ini petani juga dapat mengevaluasi *history* keuangan kelompok sebagai acuan perencanaan di masa tanam yang akan datang. Selain itu dengan sistem ini juga akan menambah daya saing keunggulan dibidang teknologi informasi.

Salah satu teknologi yang berkembang pesat saat ini adalah internet. Mengingat kebutuhan akan internet saat ini sudah mendekati kebutuhan primer di masyarakat kota khususnya, maka dibangun sistem informasi keuangan berbasis website untuk kelompok petani hidroponik RW 12 Kauman. Dengan demikian, diharapkan kelompok tani ini dapat berkembang pesat dan siap menuju persaingan Masyarakat Ekonomi Asean 2016. Untuk membangun sebuah website dibutuhkan sebuah *framework* yang mana *framework* ini fungsinya sebagai kerangka kerja yang dapat mempermudah *programmer* dalam membangun *website*. *Framework* yang digunakan adalah *Laravel* karena proses penulisan PHP akan lebih ringkas dan struktur yang dibuat mengacu pada prosedur MVC (Model, View, Control) yang digunakan oleh *framework* lainnya sehingga diharapkan pengembang (*developer*) atau pemrograman lainnya yang akan memperbaiki kesalahan (*bug*) dapat mengetahui struktur yang telah baku. (Ariefin & Ilhamsyah, 2015).

B. Identifikasi Masalah

Berdasarkan latar belakang masalah diatas didapat beberapa masalah yang ada di kelompok tani RW 12 Kauman ini antara lain :

1. Kelompok petani hidroponik tidak melakukan perencanaan secara detail terhadap modal yang akan dikeluarkan untuk memulai masa tanam sayuran hidroponik.
2. Kelompok petani tidak melakukan pencatatan terhadap pengeluaran dan pemasukan keuangan.
3. Kelompok petani kesulitan mengetahui untung atau rugi dari hasil penjualan produk sayur hidroponik.

C. Batasan Masalah

Batasan masalah pada penelitian ini adalah merancang dan membangun sistem informasi keuangan berbasis web menggunakan *framework* laravel dengan batasan sebagai berikut :

1. Data anggaran dalam bentuk perencanaan item yang akan dibeli berdasarkan masa tanam sayuran serta tidak ada pembagian presentase saldo untuk anggaran.
2. Data realisasi sesuai dengan data perencanaan anggaran yang telah dibuat sebelumnya berdasarkan masa tanam sayuran.
3. Penyajian laporan laba-rugi dari data pengeluaran dan pemasukan yang telah ada sebelumnya berdasarkan periode bulanan.

4. Penyajian laporan keuangan berdasarkan bulan dan tahun.
5. Perhitungan harga jual produk sayur hidroponik dari data pengeluaran

D. Rumusan Masalah

Melihat latar belakang masalah tersebut, serta melihat referensi mengenai Framework Laravel dan sistem informasi keuangan melalui media digital dan media non-digital, maka didapat rumusan masalah antara lain:

1. Bagaimana merancang dan membangun sebuah system informasi keuangan berbasis web yang dapat digunakan untuk pencatatan keuangan kelompok tani hidroponik RW 12 Kauman.
2. Bagaimana mengimplementasikan perhitungan laporan laba-rugi penjualan sayur hidroponik ke dalam sistem informasi keuangan kelompok petani hidroponik ASRI 12 Kauman.

E. Tujuan Penelitian

1. Merancang dan membangun sebuah sistem informasi keuangan berbasis web yang dapat digunakan untuk pencatatan keuangan kelompok tani hidroponik ASRI 12 Kauman.
2. Mengimplementasikan perhitungan laporan laba-rugi penjualan sayuran hidroponik ke dalam sistem informasi pencatatan keuangan kelompok petani hidroponik ASRI 12 Kauman.

F. Manfaat Penelitian

Manfaat bagi kelompok petani hidroponik yaitu:

1. Dapat mendokumentasikan perencanaan, pengeluaran dan pemasukan dana dengan sistem yang telah terkomputerisasi. Sehingga dapat memantau *history* keuangan kelompok petani.
2. Dapat memberi informasi kepada kelompok petani terkait laba dan rugi hasil penjualan sayuran hidroponik
3. Dapat memeriksa *history* perolehan keuntungan dari waktu ke waktu.

Bab II

Kajian Pustaka

A. Kajian Penelitian Terdahulu

Kajian terdahulu melihat dari penelitian A. Nugroho, Beeh, dan Astuningdyas (2008) tentang Perancangan Aplikasi Rencana Anggaran Biaya (RAB). Mengatakan bahwa Penyusunan Rencana Anggaran Biaya (RAB) suatu proyek adalah kegiatan yang harus dilakukan sebelum proyek dilaksanakan. RAB adalah banyaknya biaya yang dibutuhkan baik upah maupun bahan dalam sebuah pekerjaan proyek konstruksi. Daftar ini berisi volume, harga satuan, serta total harga dari berbagai macam jenis material dan upah tenaga yang dibutuhkan untuk pelaksanaan proyek tersebut. (Nugroho et al., 2008)

Penelitian yang dilakukan oleh Kabuhung (2013) tentang sistem informasi akuntansi penerimaan dan pengeluaran kas untuk perencanaan dan pengendalian keuangan pada organisasi nirlaba keagamaan mengatakan bahwa tanpa adanya sistem informasi akuntansi yang mengawasi aktivitas-aktivitas yang berlangsung, perusahaan atau organisasi akan mengalami kesulitan untuk menentukan seberapa baik kinerjanya dan juga akan mengalami kesulitan dalam menelusuri bagaimana pengaruh-pengaruh dari berbagai aktivitas atas sumberdaya-sumberdaya yang ada dibawah pengawasannya. Oleh karena itu, sistem informasi akuntansi yang efektif

sangatlah penting bagi keberhasilan jangka panjang organisasi manapun.(Kabuhung, 2013)

Bedasarkan penelitian dari Satya (2015) tentang pembangunan sistem informasi keuangan ,mengatakan bahwa organisasi yang besar pasti membutuhkan sistem keuangan yang besar pula. Hal ini disebabkan karena proses transaksi yang terjadi sangat sering dan melibatkan nominal yang besar. Sistem yang besar mempunyai prosedur dan proses bisnis yang banyak dan rumit. Oleh karena itu, dibutuhkan suatu model untuk membantu kinerja keuangan menjadi lebih baik dan dapat terorganisir dengan rapi. Model tersebut dapat disebut dengan sistem informasi keuangan. (Satya, 2015)

Sugiantina (2011) dalam penelitiannya yaitu penerapan sistem informasi dalam praktek laporan keuangan sederhana menyatakan terkait dengan metode analisis laporan keuangan dan penerapan teknologi informasi dalam berbagai analisis keungan, hal ini merupakan suatu tolak ukur bagi kemajuan pelaporan keuangan. Terutama terhadap upaya setiap institusi bisnis, untuk menyeimbangkan perkembangan teknologi Informasi, yang selaras dengan perkembangan analisis pelaporan keuangan. Dengan perkembangan tersebut, institusi bisnis dituntut untuk bisa lebih cepat dan kreatif dalam bekerja khususnya dalam pelaporan keuangan. Hal ini dikarenakan adanya shopware sistem informasi yang sudah menjadi trend, dalam dunia bisnis. Oleh sebab itu, sekarang telah menjadi trend dalam pembuatan laporan keuangan berbasis web internet hal ini dapat memberi kemudahan dalam hal financial

controlling dan Manajemen bagi semua lapisan level manajemen intitusi bisnis maupun secara terbuka bagi pelanggan atau masyarakat.(Sugiantina, 2011)

Bedasarkan penelitian yang dilakukan Andreas Handojo dkk (2004) tentang pembuatan sistem informasi akuntansi terkomputerisasi atas siklus pembelian dan penjualan pada cv. x. Mengatakan bahwa Sistem akuntansi pada CV. X masih menggunakan cara manual yaitu pencatatan piutang pada kartu piutang dan pencatatan hutang pada kartu hutang. Perhitungan total penjualan dan keuntungan dilakukan manual, sehingga membutuhkan waktu yang lama untuk mendapatkan informasi yang diinginkan. Oleh karena itu dibutuhkan sistem informasi akuntansi yang memadai, untuk memproses pencatatan secara otomatis dan menghasilkan infomasi yang cepat dan tepat. Sehingga setelah sistem diterapkan kepuasan pengguna terhadap keseluruhan aplikasi sebesar 72.5%, dan dapat disimpulkan bahwa aplikasi sudah memenuhi dan sesuai dengan kebutuhan sistem, tapi masih ada beberapa kelemahan yang perlu untuk diperbaiki. Aplikasi ini membantu proses pencatatan dengan melakukan otomatisasi penjurnalan dan posting serta membantu mempercepat proses pembuatan laporan laba rugi perusahaan. (Andreas Handojo et al., 2004)

perbandingan dari keseluruhan penelitian dapat dilihat pada tabel 2.1

Tabel 2.1: Tabel Perbandingan Kajian Penelitian Terdahulu

Peneliti	Kajian 1	Kajian 2	Kajian 3	Kajian 4	Kajian 5	Topik Yang Akan Dikembangkan
Judul	PERANCANGAN APLIKASI RENCANA ANGGARAN BIAYA (RAB)	PEMBUATAN SISTEM INFORMASI AKUNTANSI TERKOMPUTERISASI ATAS SIKLUS PEMBELIAN DAN PENJUALAN PADA CV. X	PEMBANGUNAN SISTEM INFORMASI KEUANGAN STUDI KASUS SMK MARSUDI LUHUR I YOGYAKARTA	PENERAPAN SISTEM INFORMASI DALAM PRAKTEK LAPORAN KEUANGAN SEDERHANA	SISTEM INFORMASI AKUNTANSI PENERIMAAN DAN PENGELUARAN KAS UNTUK PERENCANAAN DAN PENGENDALIAN KEUANGAN PADA ORGANISASI NIRLABA KEAGAMAAN	APLIKASI SISTEM INFORMASI LABA RUGI USAHA BAGI KELOMPOK PETANI SAYUR HIDROPONIK DI RW 12 KAUMAN YOGYAKARTA.
Teknologi	Desktop Application menggunakan bahasa pemograman java	Desktop Application (Borland Delphi 7.0.)	Desktop Application (C#)	Web based application (PHP)	Computer Based System	Web based application (PHP) laravel framework
Database	mySQL	SQL server 7.0.	DBMS SQL Server 2008 R2	MySQL		MySql
Fitur Umum	Manajemen pengguna, tahun anggaran, kategori material tenaga,	modul penjualan, modul pembelian dan akuntansi	Pengelolaan mata akun pemasukan, pengelolaan mata akun pengeluaran,	1. Pembuatan User Account oleh Administrator.		Mini POS (Point Of sales) penjualan sayur, Pengelolaan buku harian (debet-

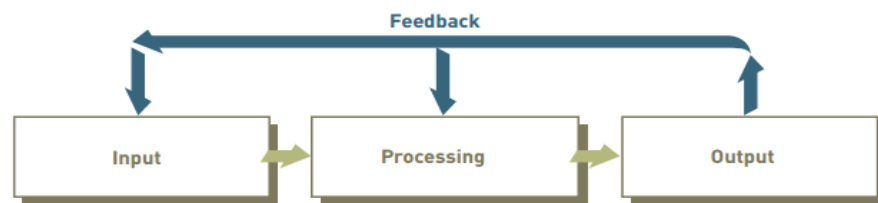
	manajemen satuan, material tenaga kerja, proyek, <i>time schedule</i>	(pencatatan). Penjualan meliputi pembuatan sales order, pengecekan inventori dan pengecekan kredit limit pelanggan. Pembelian meliputi pembuatan purchase order sampai penerimaan purchase invoice. Pencatatan meliputi penjurnalan, posting dan pembuatan laporan laba rugi.	pengelolaan anggaran pemasukan, pengelolaan anggaran pengeluaran, pengelolaan transaksi, pembuatan reporting.	2. Log in oleh Direktur Utama, untuk entry Data Master. 3. Log in oleh Operator, untuk entry Data Transaksi. Sebagai bahan pembuatan laporan keuangan. 4. Log in oleh Komisaris, untuk melihat Laporan		kredit), pencetakan laporan laba rugi per periode, pencetakan laporan arus kas,
Tempat Penelitian	DINAS PEKERJAAN UMUM KOTA SALATIGA	CV. X	SMK Marsudi Luhur I Yogyakarta		GMIM Nafiri Malayang	Kelompok tani hidroponik RW 12 Kauman yogyakarta.

Hasil	perancangan aplikasi yang mampu menjabarkan RAB untuk pembangunan dan pemeliharaan/ rehabilitasi gedung menjadi daftar kebutuhan material dan tenaga dengan teliti, cepat, akurat, dan cermat.	Sebuah system informasi akuntansi berbasis desktop untuk CV.X	Sebuah system informasi keuangan yang membantu petugas TU dalam mengelola keuangan	Sebuah system informasi keuangan sederhana berbasis web agar dapat diakses dari setiap unit usaha.		
-------	--	---	--	--	--	--

B. Landasan Teori

1. Sistem Informasi

Sistem informasi (SI) adalah seperangkat elemen yang saling terkait atau komponen yang mengumpulkan (input), memanipulasi (proses), menyimpan, dan menyebarkan (output) data dan informasi, dan memberikan reaksi korektif (mekanisme umpan balik) untuk memenuhi tujuan sesuai yang tertera pada gambar 2.0 . Mekanisme umpan balik merupakan komponen yang membantu organisasi mencapai tujuan mereka, seperti meningkatkan keuntungan atau meningkatkan layanan pelanggan (Ralph M. Stair, 2010)



(Gambar 2.0) Gambaran umum sistem informasi

a. INPUT

Dalam sistem informasi, input adalah kegiatan pengumpulan dan menangkap data mentah. Dalam memproduksi gaji misalnya, jumlah jam setiap karyawan bekerja harus dikumpulkan sebelum gaji dapat dihitung atau dicetak. Dalam sistem peringkat universitas, instruktur harus menyerahkan nilai siswa sebelum

ringkasan nilai untuk satu semester dapat dikompilasi dan dikirim ke siswa.(Ralph M. Stair, 2010)

b. PROSES

Dalam sistem informasi, proses berarti mengkonversi atau mengubah data menjadi output yang berguna. proses dapat melibatkan membuat perhitungan, membandingkan data dan mengambil tindakan alternatif, dan menyimpan data untuk penggunaan kedepannya. Pengolahan dapat dilakukan secara manual atau dengan bantuan komputer. Setelah perhitungan dan perbandingan yang dilakukan, hasilnya biasanya disimpan. Penyimpanan termasuk menjaga data dan informasi yang tersedia untuk digunakan di masa depan, termasuk output, dibahas berikutnya.(Ralph M. Stair, 2010)

c. OUTPUT

Dalam sistem informasi, output melibatkan produksi informasi yang berguna, biasanya dalam bentuk dokumen dan laporan. Output dapat mencakup gaji bagi karyawan, laporan untuk manajer, dan informasi yang diberikan kepada pemegang saham, bank, instansi pemerintah, dan kelompok-kelompok lainnya. Dalam beberapa kasus, output dari satu sistem dapat menjadi masukan bagi yang lain. Misalnya, output dari sistem yang memproses order

penjualan dapat digunakan sebagai masukan untuk penagihan pelanggan sistem. (Ralph M. Stair, 2010)

d. UMPAN BALIK

Dalam sistem informasi, umpan balik adalah informasi dari sistem yang digunakan untuk membuat perubahan kegiatan masukan atau pengolahan. Misalnya, kesalahan atau masalah yang perlu diperbaiki dbagian input data atau mengubah suatu proses. (Ralph M. Stair, 2010)

2. Laporan Laba Rugi

Dalam literatur akuntansi, laporan laba rugi diturunkan dari istilah *profit and loss statement, earning statement, operations statement, atau income statement*. Setiap jangka waktu tertentu, perusahaan perlu memperhitungkan hasil usaha perusahaan yang dituangkan dalam bentuk laporan laba rugi. Hasil usaha didapat dengan cara membandingkan penghasilan dan biaya selama jangka waktu tertentu. Besarnya laba atau rugi akan diketahui dari hasil perbandingan tersebut. (Jumingan, 2005)

Rumus sederhana untuk menentukan laba rugi yaitu :

Laba rugi = pendapatan – biaya (1)

Laba = Pendapatan > biaya(2)

Rugi = Pendapatan < biaya(3)

(1) Rumus laporan laba rugi. (2) Rumus laba. (3) Rumus rugi.

3. Harga Pokok

Untuk menjamin bahwa suatu kegiatan usaha akan menghasilkan nilai keluaran yang lebih tinggi dari pada nilai masukan diperlukan alat untuk mengukur nilai masukan yang dikorbankan dalam menghasilkan keluaran tersebut. Informasi biaya merupakan alat yang berfungsi untuk mengukur pengorbanan nilai masukan, guna menghasilkan informasi yang bermanfaat untuk mengukur apakah kegiatan usahanya akan menghasilkan laba atau tidak. Adapun biaya - biaya yang merupakan pengorbanan sumber ekonomi untuk memperoleh aktiva disebut dengan istilah Harga Pokok. (Sarfi'ah, 2013)

Salah satu tujuan dari penentuan harga pokok adalah untuk menentukan biaya yang seharusnya dikeluarkan untuk memproduksi satu satuan produk. Dengan demikian perusahaan dapat menetapkan harga jual produk per unitnya, setelah mempertimbangkan biaya produksi per unit serta biaya-biaya lain yang di bebarkannya yaitu dengan rumus :

$$\text{Total biaya produksi} \div \text{Total Hasil Produksi} + \text{Profit} = \text{Harga Pokok} \dots\dots(4)$$

(4) Rumus Harga pokok.

Sedangkan untuk menentukan profit menggunakan rumus :

$$\text{Total biaya produksi} \div \text{Total Hasil Produksi} * \text{Presentase yang diinginkan} = \text{profit} \dots\dots(5)$$

(5) rumus menentukan profit

4. Sistem Informasi Keuangan

Laporan keuangan berkaitan erat dengan bidang akuntansi. Kegiatan akuntansi pada dasarnya merupakan kegiatan mencatat, menganalisis, menyajikan, dan menafsirkan data keuangan dari lembaga perusahaan dan lembaga lainnya dimana aktifitasnya berhubungan dengan produksi dan pertukaran barang dan jasa. Bagi lembaga yang bertujuan memperoleh keuntungan, akuntansi memberikan metode untuk menentukan apakah lembaga tersebut memperoleh keuntungan atau kerugian sebagai hasil dari transaksi yang dilakukannya.

Akuntansi dapat memberi informasi tentang kondisi keuangan dari hasil operasi perusahaan seperti tercermin pada laporan keuangan yang bersangkutan. Oleh karena itu, akuntansi (laporan keuangan) dapat dipakai sebagai alat berkomunikasi dengan pihak pihak yang berkepentingan dengan data keuangan perusahaan. (Jumingan, 2005)

a. Peran Komputer dalam Akuntansi

Proses pengolahan data akuntansi dapat dilakukan dengan lebih cepat bila digunakan komputer. Hal ini dapat terjadi karena kemampuan komputer untuk mengolah data jauh melebihi kecepatan manusia. Dengan adanya perkembangan teknologi komputer, semakin banyak perusahaan yang menggunakan jasa komputer untuk memproses data akuntansinya. Disatu pihak, komputer merupakan alat bantu yang sangat bermanfaat dalam

sistem informasi akuntansi. Akan tetapi, di pihak lain diperlukan teknik-teknik pengawasan yang berbeda dengan yang digunakan dalam cara manual untuk menjamin ketelitian dan keamanan dalam memproses data . Beberapa tahapan proses pengolahan data yang memperoleh manfaat besar dari penggunaan komputer adalah :

1) Verifikasi

Komputer dapat mengecek kebenaran maupun kelayakan angka-angka yang menjadi input dalam suatu proses. Misalnya pengecekan kebenaran kode yang digunakan, pengecekan kelayakan jumlah rupiah dari transaksi, dan lain-lainnya.

2) Sortir

Komputer memungkinkan untuk dilakukannya pensortiran data ke dalam beberapa klasifikasi yang berbeda dengan cepat. Misalnya, kumpulan faktur penjualan dapat disortir ke dalam klasifikasi langganant jenis produk, daerah penjualan dan lain sebagainya.

3) Transmission

Komputer dapat memindahkan lokasi data dari suatu tempat ke tempat lainnya dengan cepat. Misalnya , data dari suatu file dipindahkan ke file lainnya.

4) Perhitungan

Dengan komputer, perhitungan-perhitungan dapat dilakukan dengan cepat. Misalnya, menghitung saldo rekening sesudah adanya posting, menghitung jumlah sekelompok transaksi, dan lain sebagainya. (Tata Sutabsi, 2003)

5. PHP

PHP adalah kependekan dari *PHP Hypertext Preprocessor*. Bahasa interpreter yang mirip dengan bahasa C dan Perl yang dimiliki kesederhanaan dalam perintah. PHP dapat digunakan bersama dengan HTML. Sehingga memudahkan dalam pembangunan aplikasi *web* dengan cepat. PHP dapat digunakan untuk meng-*update* basis data dan menciptakan basis data. (Iswanto, 2006)

Seperti halnya program open source lainnya, PHP dibuat dibawah lisensi GNU, *General Public License*, yang dapat di-*download* gratis melalui situs <http://www.php.net>.

a. Penulisan PHP

Penulisan *script* PHP diawali dan diakhiri dengan sintaks khusus. Ada beberapa cara yang dapat digunakan untuk menuliskan *script* PHP yaitu :

1. Dengan sintaks “<?php” dan “?>”; sintaks tersebut adalah yang paling umum dari PHP.

2. Dengan sintaks “<%” dan “%>”, sintaks tersebut sebenarnya adalah sintaks dari ASP, tetapi dapat digunakan sebagai sintaks PHP apabila konfigurasi PHP pilihan `asp_tag` diaktifkan.

Dengan sintaks “<SCRIPT LANGUAGE = “php” dan diakhiri dengan </SCRIPT>; sintaks itu digunakan untuk mendeklarasikan sebuah perintah pada html.

b. Variabel pada PHP

Variabel merupakan suatu “tempat” di dalam memori komputer yang dialokasikan untuk menyimpan data. Pada PHP, variabel ditandai dengan tanda dollar (\$) dan diikuti nama variabelnya. Penamaan variabel mengikuti aturan sebagai berikut:

1. Harus diawali dengan huruf atau *underscore* (_), dapat diikuti oleh huruf, angka, *underscore*, atau sembarang karakter ASCII dari 127 hingga 255.
2. Huruf besar dan kecil dibedakan (*case sensitive*).
3. Tidak boleh mengandung spasi.

Variabel dalam PHP tidak harus di deklarasikan terlebih dahulu sebelum digunakan, tidak seperti bahasa pemrograman tertentu yang harus mendeklarasikan variabel terlebih dahulu.

c. Fungsi pada PHP

Fungsi adalah konstruksi pemograman untuk melakukan suatu proses. Dalam pemograman, kita biasa melakukan proses secara berulang kali. Oleh karena itu, fungsi sangat membantu *programmer* untuk menangani hal tersebut supaya penulisan kode programnya lebih efisien.

Bentuk umum fungsi:

Function nama_fungsi (argumen){

Kode perintah }

Fungsi harus didefinisikan terlebih dahulu sebelum digunakan. Pengembalian nilai dari suatu fungsi menggunakan perintah `return`. Semua jenis data bisa digunakan sebagai nilainya, termasuk *list* dan objek.

4. My SQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: database management system) atau DBMS yang multithread, multi-user, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis di bawah lisensi GNU General Public License (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL. (Solichin, 2010)

Fitur-fitur MySQL antara lain :

- a. Relational Database System. Seperti halnya software database lain yang ada di pasaran, MySQL termasuk RDBMS.
- b. Mengenal perintah SQL standar. SQL (Structured Query Language)
- c. Merupakan suatu bahasa standar yang berlaku di hampir semua software database. MySQL mendukung SQL versi SQL:2003.
- d. Mendukung Sub Select. Mulai versi 4.1 MySQL telah mendukung select dalam select (sub select).
- e. Mendukung Triggers. MySQL mendukung trigger pada versi 5.0 namun masih terbatas. Pengembang MySQL berjanji akan meningkatkan kemampuan trigger pada versi 5.1.
- f. Tersedia fungsi GIS.

5. Framework LARAVEL

Framework laravel adalah framawork yang dikembangkan oleh komunitas PHP yang menggunakan konsep MVC (Model, View, Contoller). Ketika melihat di dunia pengembangan saat ini, konsep MVC digunakan dimana saja. ASP.NET, dan PHP Frameworks seperti *CakePHP* dan *Code Igniter* semua menggunakannya. Jadi yang membuat MVC penting dalam sebuah framework adalah pemisahan Logika dari layer presentasi (Views / HTML). (Dangar, 2013)

MVC memungkinkan Anda untuk menulis kode yang dapat dibagi atas dasar tiga hal:

a. Model

Model adalah cara dimana aplikasi dapat berinteraksi dengan data. Ini adalah *layer* antara data dan aplikasi. Data yang disimpan bisa dalam sistem database seperti MySQL, MSSQL, atau files XML

b. Controller

Controller adalah hubungan antara Model dan Views. Tanggung jawab utama Controller adalah untuk menangani permintaan dan melewatkan data dari Model untuk views.

c. View

Views adalah representasi visual dari aplikasi . view juga memungkinkan untuk menulis representasi dari logika proses bisnis yang ada pada aplikasi.

BAB III

METODOLOGI PENELITIAN

A. Subjek Penelitian

Subjek penelitian pada tugas akhir yang berjudul “Aplikasi Sistem Informasi Keuangan Berbasis Web Menggunakan Konsep MVC (Model, View, Control) dengan Framework Laravel Untuk Mengelola Keuangan di Kelompok Petani Sayur Hidroponik di RW 12 Kauman Yogyakarta ” adalah kelompok petani hidroponik di Kauman , Yogyakarta. Bendahara kelompok petani di UMKM ini adalah pengguna utama dari aplikasi sistem informasi keuangan untuk membantu mengelola keuangan di UMKM kelompok tani RW 12 Kauman ini. Sehingga kelompok UMKM petani hidroponik di RW 12 mampu mengelola keuangan dengan baik, dapat memantau keadaan keuangan pada periode tertentu sehingga dapat menentukan harga yang pantas untuk menjual hasil panen agar tidak merugi. Aplikasi nantinya dapat digunakan oleh UMKM lain karena sistem yang dibuat adalah sistem pengelolaan keuangan yang ada pada umumnya.

B. Metode Pengumpulan Data

Dalam mengambil beberapa data yang dibutuhkan dalam pembuatan Aplikasi Sistem Informasi Kuangan maka dalam melakukan pengumpulan data digunakan beberapa metode sebagai berikut :

1. Metode Wawancara

Metode wawancara dilakukan dengan mewawancarai ketua kelompok petani sayur hidroponik terkait permasalahan yang akan diteliti. Karena ketua kelompok banyak mengetahui informasi tentang permasalahan yang ada pada kelompoknya. Serta wawancara dilakukan kepada bendahara kelompok. Wawancara kepada bendahara dilakukan karena mengetahui permasalahan keuangan yang ada pada kelompok serta memberi saran terkait fitur-fitur yang akan dibuat pada sistem informasi keuangan ini. Dalam wawancara berisi 9 pertanyaan yang mendasar. Pertanyaan tiap butirnya menggali informasi tentang tingkat penguasaan anggota kelompok terhadap penggunaan komputer maupun *smartphone* . serta mengukur seberapa besar manfaat sistem informasi keuangan yang akan dibuat pada penelitian ini untuk kelompok petani sayur hidroponik RW 12 Kauman.

2. Survey Lokasi

Peninjauan dilakukan dilokasi penanaman sayur hidroponik milik kelompok tani RW 12 Kauman. Untuk mengetahui lokasi penanaman sayuran hidroponik.

C. Alat dan Bahan

1. Perangkat keras (*Hardware*)

Perangkat keras atau *hardware* yang digunakan untuk membangun aplikasi sistem informasi keuangan ini antara lain :

- a. Laptop dengan spesifikasi :

- 1) Intel® Core i5-450M Processor 2.4GHz, 3MB L3 Cache
- 2) RAM 4Gb DDR3
- 3) 500Gb SATA HDD
- 4) Intel® HD Graphics
- 5) 14" HD LED LCD

2. Perangkat Lunak (*Software*)

Perangkat lunak yang digunakan untuk membuat aplikasi sistem informasi keuangan ini antara lain :

- a. Sistem Operasi : Windows 10
- b. Bahasa pemrograman : *PHP v5.6.15, HTML 5.*
- c. *Case Tools* : Sublime text 3, Framework LARAVEL 5, MySQL 4.5.1.

D. Analisis kebutuhan pengguna

Analisis kebutuhan pengguna ini merupakan tahapan untuk mengumpulkan spesifikasi kebutuhan sebuah perangkat lunak dari semua stakeholder yang terlibat. Analisis bisa diperoleh dari wawancara atau investigasi terhadap proses bisnis yang berjalan. Analisis kebutuhan pengguna sistem informasi keuangan ASRI 12 antara lain *user* atau pengguna yang terlibat di kelompok tani adalah ketua, bendahara , dan anggota kelompok. Masa tanam sayuran hidroponik adalah 3 bulan , jadi perencanaan keuangan setiap sayuran adalah per 3 bulan. Pemasukan di kelompok tani hidroponik terdapat 3 jenis pemasukan yaitu : pemasukan hasil penjualan, iuran anggota dan sumbangan dari kelompok PKK.

E. Analisis Kebutuhan Sistem

Analisis kebutuhan sistem ini merupakan tahapan untuk menentukan spesifikasi kebutuhan sistem (*system requirement spesification*). Pada tahap ini didefinisikan kebutuhan yang harus dipenuhi oleh program yang akan dibangun. Dalam analisis kebutuhan sistem, terdapat kebutuhan fungsional dan kebutuhan non-fungsional.

1. Kebutuhan fungsional

Kebutuhan fungsional adalah pernyataan layanan sistem yang harus disediakan, bagaimana sistem bereaksi pada input tertentu dan bagaimana perilaku sistem pada situasi tertentu.

2. Kebutuhan non-fungsional

Kebutuhan non – fungsional adalah batasan layanan atau fungsi yang ditawarkan sistem seperti batasan waktu, batasan pengembangan proses, standarisasi.

Dari hasil analisis kebutuhan sistem dihasilkan *use case diagram*. Yang bertujuan untuk menggambarkan proses yang dapat dilakukan oleh *user*.

F. Desain Sistem (Pemodelan)

Setelah dibuat gambaran yang jelas tentang sistem yang akan dibangun, selanjutnya dilakukan desain sistem (Pemodelan). Tahap desain sistem ini dilakukan untuk memberikan gambaran umum yang jelas kepada pengguna dan rancang bangun yang lengkap tentang sistem yang akan dikembangkan kepada pihak-pihak yang terlibat dalam pengembangan sistem ini. Adapun perancangan dibagi menjadi :

1. Desain Aktifitas

Desain aktifitas digunakan untuk menggambarkan proses yang terjadi pada aplikasi sistem informasi keuangan . *Activity diagram* digunakan untuk menggambarkan alur proses bisnis dan urutan aktifitas dalam sistem.

2. Desain Data

Desain data digunakan untuk menggambarkan pemodelan basis data yang ada pada aplikasi sistem informasi keuangan. *Entity Relationship Diagram (ERD)* digunakan dalam tahap ini untuk memodelkan struktur data dan hubungan antar tabel yang ada dalam sistem informasi keuangan ini.

3. Desain Antarmuka

Perancangan antarmuka bertujuan untuk memberikan gambaran aplikasi sistem informasi keuangan kepada *user* . perancangan antarmuka meliputi struktur menu, input dan output aplikasi, dan fungsi-fungsi yang ada didalam aplikasi. Hasil dari desain antarmuka ini adalah *prototype* aplikasi sistem informasi keuangan.

4. Arsitektur Jaringan

Arsitektur aplikasi sistem informasi keuangan ini memanfaatkan internet untuk menjalankan fungsinya seperti : mengakses data, menginput data serta melihat informasi hasil dari pengolahan sistem. Basis data yang disimpan di server digunakan untuk menyimpan seluruh data dan dapat diakses kapan saja selama terkoneksi internet.

G. Implementasi

Implementasi Aplikasi Sistem Informasi Keuangan ini dibangun menggunakan *framework* laravel 5.1 dengan bahasa pemograman *HTML* dan *PHP* yang berbasis web. Dengan beberapa bahasa pemograman yang menambah daya tarik tampilan web antara lain *Bootstrap*, *Javascript*. Data aplikasi ini disimpan di web server salah satu penyedia jasa hosting. Dan *database* yang dibuat menggunakan MySQL. Pembangunan web dengan *framework* Laravel menggunakan konsep MVC yaitu :

1. Model

Yaitu berisi *class php* yang menghubungkan *controller* dengan *database MySql*. Selain itu model juga dapat memproteksi atau mengizinkan *controller* dalam mengakses database, *query* data dari *database* untuk disampaikan ke *controller*, serta mengatur integritas dan konsistensi data.

2. Controller

Yaitu kode yang bertugas menerima data, validasi data, membuat data baru dan memperbaharui data. Selain itu juga dapat memodifikasi kueri *database*. Semua kode disusun di masing- masing *function* sesuai dengan yang diminta oleh *view*. Di *controller* juga berisi kondisi yang digunakan untuk menyimpan setiap data yang masuk. Adapun controller yang mendukung berjalannya fungsi utama sistem yaitu :

- a. Controller tambah, ubah, non-aktifkan dan *reset password* pengguna.

- b. Controller tambah anggaran beserta script kalkulasi otomatis anggaran.
- c. Controller tambah realisasi beserta kondisi jika realiasi lebih besar dari anggaran.
- d. Controller tambah pemasukan.
- e. Controller laporan laba-rugi dan keuangan bulanan.
- f. Controller harga jual produk.

3. View

Yaitu berisi *script* yang digunakan untuk menerima dan mengirim data dari *controller* dan membuat tampilan aplikasi yang mudah dibaca langsung oleh *user*.

H. Pengujian Sistem

Untuk melakukan pengujian sistem terhadap aplikasi sistem informasi keuangan, metode yang digunakan antara lain :

1. Alpha Test

Alpha test yaitu pengujian program dengan cara mengundang pakar ilmu akuntansi untuk menjalankan aplikasi, kemudian pakar akan diberikan quisioner untuk memberikan penilaian terhadap aplikasi yang dijalankan baik ditinjau dari tampilan antar muka (*user interface*), kemudian dalam pemasukan data, menjalankan operasi yang ada di dalam aplikasi.

Pengujian sistem sangat diperlukan untuk menentukan keberhasilan suatu sistem dan melakukan perbaikan-perbaikan jika masih

terjadi kesalahan atau perubahan fitur sesuai dengan kebutuhan pengguna.

2. Pengujian *Black Box*

Pengujian *black box* dilakukan untuk menguji antarmuka aplikasi, menyediakan input dan memeriksa outputnya, dan untuk menguji fitur-fitur yang ada pada aplikasi sudah berjalan dengan benar. Pengujian ini dilakukan oleh pengguna dengan berinteraksi langsung dengan aplikasi dan menjalankan beberapa fitur. Hal ini bertujuan untuk mengetahui apakah fitur-fitur yang ada sudah bekerja dengan baik dan tepat.

Bab IV

Hasil Dan Pembahasan

A. Analisis Kebutuhan Pengguna

Analisis kebutuhan pengguna ini merupakan tahapan untuk mengumpulkan spesifikasi kebutuhan sebuah perangkat lunak dari semua stakeholder yang terlibat. Dari hasil wawancara dengan *stakeholder* yang bersangkutan, maka di dapat spesifikasi kebutuhan pengguna antara lain :

1. Kelompok petani hidroponik memiliki beberapa jabatan antara lain: ketua, bendahara, sekretaris, dan anggota.
2. Ketua memiliki hak untuk mendaftarkan anggota baru, mengisi daftar perencanaan Anggaran, mengisi daftar pengeluaran, mengisi daftar pemasukan dan melihat laporan keuangan laba-rugi.
3. Bendahara dapat mengisi daftar perencanaan Anggaran, mengisi daftar pengeluaran, mengisi daftar pemasukan dan melihat laporan keuangan, laba-rugi, dan menentukan harga jual sayuran hidroponik.
4. Anggota dapat melihat laporan keuangan, laba-rugi dan daftar harga sayuran hidroponik.
5. sayuran hidroponik yang ditanam antara lain : sawi, selada, tomat. Seledri, sereh dan kangkung. Dan akan ditambah jenisnya sesuai perkembangan dan kemampuan kelompok petani.

6. Setiap sayuran masa tanamnya berkisar 3 sampai 4 bulan siap dipanen.

Sedangkan untuk bibit sayuran dapat dipanen dan dijual saat usia tanam berkisar 2 sampai 3 minggu.

7. Bahan yang dibeli untuk memulai masa tanam antara lain : bibit, nutrisi, *rockwool*, dan tanah pupuk.

B. Analisis Kebutuhan Sistem

Dari hasil pengumpulan data yang dilakukan dengan analisis kebutuhan pengguna maka didapatkan spesifikasi yang diperlukan untuk membangun sistem informasi keuangan berbasis web. spesifikasi yang telah didapat kemudian di analisis dan menghasilkan spesifikasi kebutuhan sistem (*system requirement spesification*). Dalam analisis kebutuhan sistem, terdapat kebutuhan fungsional dan kebutuhan non-fungsional.

1. Kebutuhan fungsional

kebutuhan fungsional adalah kebutuhan yang terkait dengan fungsi sistem, antara lain :

- a. Membatasi pembuatan akun pengguna hanya terdapat jenis 3 hak akses yaitu Ketua, Bendahara, dan Anggota.
- b. Masing masing pengguna dibatasi hanya dapat mengubah nama, email, dan password. Tidak dapat mengubah jabatan.
- c. Jika pengguna lupa password, dibatasi hanya ketua yang dapat *reset password* dengan *password* baru secara *default*.

- d. Membatasi ketua tidak dapat menghapus akun pengguna, tetapi hanya menonaktifkan pengguna .
- e. Membatasi pembuatan anggaran hanya sebagai perencanaan pembelian item yang akan dibeli.
- f. Membatasi pembuatan data realisasi hanya dapat dibuat sesuai dengan data perencanaan anggaran. Sehingga item yang di beli tidak melebihi perencanaan yang telah dibuat.
- g. Membatasi pembuatan data pemasukan keuangan hanya berasal dari 3 sumber yaitu : pemasukan hasil penjualan sayuran, iuran anggota kelompok , dan sumbangan dari kelompok PKK.
- h. Membatasi pembuatan laporan laba rugi hanya dapat menampilkan pengeluaran, pemasukan, dan jumlah laba atau rugi yang didapatkan oleh kelompok petani serta laporan hanya dapat dicetak dalam bentuk *pdf*.
- i. Membatasi pembuatan laporan keuangan hanya dapat dilihat per-bulan dan per-tahun. laporan keuangan bulanan hanya bisa di cetak dalam bentuk *pdf*.
- j. Membatasi pembuatan data harga penjualan produk hanya dapat menampilkan harga rekomendasi yang dihitung berdasarkan jumlah modal yang dikeluarkan dibagi jumlah hasil panen dan di tambah presentase profit. Harga pasar dan harga jual produk hanya dapat diinputkan oleh pengguna.

2. Kebutuhan non-fungsional

- a. Sistem hanya dapat membuat 3 hak akses.
- b. Sistem hanya dapat di akses melalui *browser* di komputer dan *smartphone* yang terkoneksi internet dengan kecepatan minimal 300 Kbps yaitu dengan saluran HSDPA.
- c. Sistem hanya dapat membuat laporan laba-rugi bulanan, laporan keuangan bulanan dan tahunan
- d. Sistem hanya mencetak laporan dalam bentuk *pdf*.

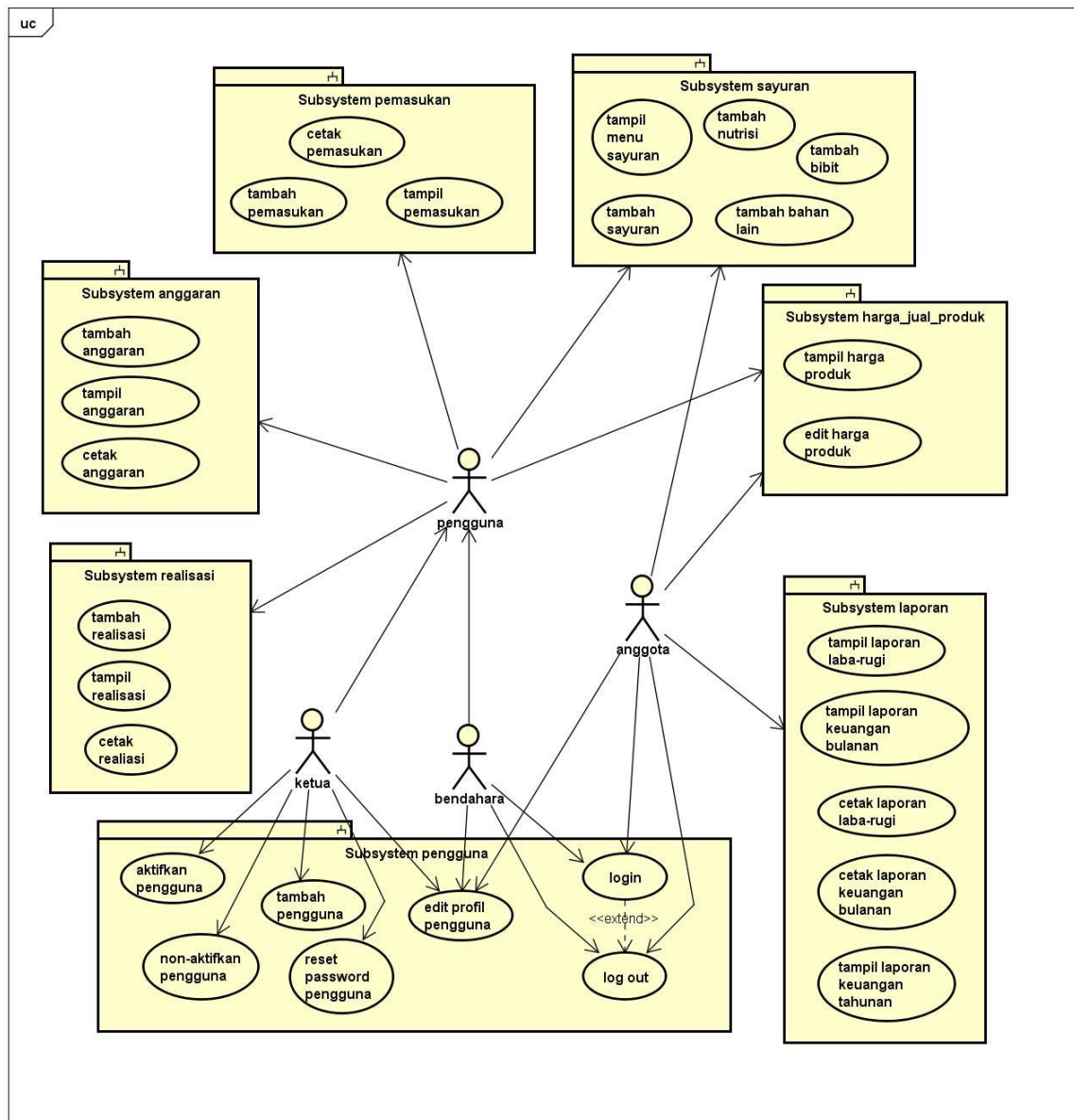
Bedasarkan hasil analisis kebutuhan pengguna di kelompok petani hidroponik ASRI 12 Kauman, diperoleh *system requirement spesification (SRS)* pada tabel 4.1.

Tabel 4.1 SRS Fungsional

Kode SRS	Deskripsi
SRS-F-01	Aplikasi mempunyai 3 hak akses pengguna yaitu Ketua , bendahara, dan Anggota.
SRS-F-02	Aplikasi mampu membuat, menonaktifkan, reset password dan mengaktifkan data pengguna yang hanya bisa di akses oleh ketua.
SRS-F-03	Aplikasi mampu mengubah data profil masing masing pengguna.yang bisa di akses ketua, bendahara, dan anggota.
SRS-F-4	Aplikasi mampu membuat, menampilkan, dan mencetak data rencana anggaran yang bisa di akses ketua dan bendahara.
SRS-F-05	Aplikasi mampu membuat, menampilkan, dan mencetak data realisasi keuangan yang bisa di akses ketua dan bendahara.

SRS-F-06	Aplikasi mampu membuat, menampilkan, dan mencetak data pemasukan keuangan yang bisa di akses ketua dan bendahara.
SRS-F-07	Aplikasi mampu menampilkan dan mencetak laporan laba rugi yang bisa di akses ketua, bendahara, dan anggota.
SRS-F-08	Aplikasi mampu menampilkan dan mencetak laporan keuangan bulanan yang bisa di akses ketua, bendahara, dan anggota.
SRS-F-09	Aplikasi mampu menampilkan dan mengubah daftar harga sayuran yang bisa di akses ketua, bendahara, dan anggota.
SRS-F-10	Aplikasi mampu membuat dan menampilkan jenis sayuran, bibit, nutrisi, dan bahan lain.
SRS-F-11	Aplikasi mampu menampilkan laporan keuangan tahunan yang bisa diakses ketua, bendahara , dan anggota

Setelah menganalisis setiap SRS (*system requirement system*) maka dapat didefinisikan kebutuhan yang harus dipenuhi oleh sistem yang dibangun menggunakan diagram *use case* yang bertujuan untuk menggambarkan proses apa saja yang bisa dilakukan *user*. Diperoleh diagram *use case* pada gambar 4.1.



Gambar 4.1. use case diagram

Bedasarkan *use case* yang telah di buat , maka penjelasan skenario *use case* sebagai berikut :

1. Login

Tabel 4.2 *use case scenario* login

<i>Use Case Name</i>	Login	
<i>Actor</i>	Ketua, Bendahara, Anggota	
<i>Description</i>	Use case login digunakan untuk memeriksa keamanan dan memastikan bahwa yang mengakses aplikasi adalah aktor yang mendapatkan hak akses	
<i>Pre Condition</i>	Admin dan anggota memasukan email dan password	
<i>Trigger</i>	saat pengguna ingin mengakses aplikasi sesuai dengan hak akses	
<i>Typical Course of Events</i>	<i>Actor action</i>	<i>System Response</i>
	<ul style="list-style-type: none"> - Masukkan email dan password 	<ul style="list-style-type: none"> - Cek email dan password - Menampilkan halaman dashboard
<i>Alternate Course</i>	Jika email dan password salah maka aplikasi menampilkan pesan kesalahan dan kembali ke halaman login	
<i>Conclusion</i>	User login ke dalam aplikasi sistem informasi keuangan	
<i>Post Condition</i>	Aplikasi berhasil di akses	

2. tambah pengguna

Tabel 4.3 *use case scenario* input pengguna

<i>Use Case Name</i>	Tambah Pengguna	
<i>Actor</i>	Ketua	
<i>Description</i>	Use case input pengguna menggambarkan saat admin akan menambahkan pengguna aplikasi sistem informasi keuangan.	
<i>Pre Condition</i>	actor menambahkan informasi pengguna	
<i>Trigger</i>	saat actor ingin menambah pengguna sistem	
<i>Typical Course of Events</i>	<i>Actor action</i>	<i>System Response</i>
	<ul style="list-style-type: none"> - Masukkan nama - Masukan email - Memilih jabatan - Klik tombol "tambah" 	<ul style="list-style-type: none"> - Cek form yang kosong - Menampilkan halaman tabel pengguna
<i>Alternate Course</i>	<ul style="list-style-type: none"> - Jika ada form yang kosong, maka akan muncul pesan kesalahan - Jika pada form email tidak diisi format email, maka muncul pesan kesalahan. 	
<i>Conclusion</i>	Pengguna berhasil di tambahkan	
<i>Post Condition</i>	Data pengguna baru disimpan ke tabel pengguna	

3. Tambah Anggaran

Tabel 4.4 *use case scenario* input Anggaran

<i>Use Case Name</i>	tambah anggaran	
<i>Actor</i>	Ketua dan Bendahara	
<i>Description</i>	Use case input anggaran menggambarkan saat pengguna menambahkan data anggaran	
<i>Pre Contidion</i>	Actor memasukan data anggaran	
<i>Trigger</i>	-	
<i>Typical Course of Events</i>	<i>Actor action</i>	<i>System Response</i>
	<ul style="list-style-type: none"> - Memasukan data detail Anggaran - Klik 'simpan data' 	<ul style="list-style-type: none"> - Memeriksa form yang kosong - Menyimpan ke database - Meredirect ke halaman tabel Anggaran
<i>Alternate Course</i>	<ul style="list-style-type: none"> - Jika ada form yang kosong maka akan muncul pesan form diperlukan - Klik 'kembali' jika ingin membatalkan menambah Anggaran 	
<i>Conclusion</i>	Data Anggaran berhasil ditambahkan	
<i>Post Condition</i>	Data Anggaran berhasil di simpan tabel anggaran	

4. Cetak Anggaran

Tabel 4.5 *use case scenario* cetak Anggaran

<i>Use Case Name</i>	Cetak Anggaran	
<i>Actor</i>	Ketua dan Bendahara	
<i>Description</i>	Use case menggambarkan sistem mencetak data Anggaran ke dalam bentuk PDF	
<i>Pre Contidion</i>	Actor memilih periode Anggaran yang ingin di cetak	
<i>Trigger</i>	Jika actor ingin mencetak data Anggaran dalam bentuk PDF	
<i>Typical Course of Events</i>	<i>Actor action</i>	<i>System Response</i>
	<ul style="list-style-type: none"> - Memilih tahun periode - Memilih bulan periode - Klik tombol 'cetak PDF' 	<ul style="list-style-type: none"> - Mendownload data Anggaran dalam bentuk PDF
<i>Alternate Course</i>		
<i>Conclusion</i>	Data Anggaran berhasil dicetak sesuai periode yang dipilih	
<i>Post Condition</i>	Dokumen Anggaran sesuai masa tanam berhasil dicetak	

5. tambah realisasi

Tabel 4.6 *use case scenario* tambah realisasi

<i>Use Case Name</i>	Tambah pengeluaran	
<i>Actor</i>	Ketua dan Bendahara	
<i>Description</i>	<i>Use case</i> tambah realisasi menggambarkan saat pengguna menambahkan data realisasi	
<i>Pre Contidion</i>	Actor memasukan data realisasi	
<i>Trigger</i>	Saat actor ingin menambah data realisasi	
<i>Typical Course of Events</i>	<i>Actor action</i>	<i>System Response</i>
	<ul style="list-style-type: none"> - Memasukan data detail realisasi - Klik 'simpan data' 	<ul style="list-style-type: none"> - Memeriksa form yang kosong - Menyimpan ke tabel pengeluaran, labarugi dan transaksi - Meredirect ke halaman tabel realisasi
<i>Alternate Course</i>	Jika ada form yang kosong maka akan muncul pesan form diperlukan Klik 'kembali' jika ingin membatalkan menambah realisasi	
<i>Conclusion</i>	Data realisasi berhasil ditambahkan	
<i>Post Condition</i>	Data realisasi berhasil di simpan tabel realisasi	

6. Tampil Realisasi

Tabel 4.7 *use case scenario* tampil realisasi

<i>Use Case Name</i>	Tampil realisasi	
<i>Actor</i>	Ketua dan Bendahara	
<i>Description</i>	<i>Use case</i> tampil realisasi menjelaskan saat sistem menampilkan data realisasi sesuai masa tanam yang di pilih	
<i>Pre Contidion</i>	Actor memilih masa tanam	
<i>Trigger</i>	Saat actor ingin menampilkan tabel realiasi pada masa tanam yang diinginkan	
<i>Typical Course of Events</i>	<i>Actor action</i>	<i>System Response</i>
	<ul style="list-style-type: none"> - Memilih tahun masa tanam - Memilih bulan masa tanam - Klik 'tampilkan' 	<ul style="list-style-type: none"> - Menampilkan tabel realisasi sesuai dengan periode yang dipilih.
<i>Alternate Course</i>	Jika actor tidak memilih tahun dan bulan periode, maka sistem menampilkan seluruh data realisasi di tabel pengeluaran.	
<i>Conclusion</i>	Data realisasi berhasil ditampilkan	
<i>Post Condition</i>	Data realisasi berhasil ditampilkan sesuai masa tanam	

7. Cetak realisasi

Tabel 4.8 *use case scenario* cetak realisasi

<i>Use Case Name</i>	cetak realisasi	
<i>Actor</i>	Ketua , Bandahara	
<i>Description</i>	<i>Use case</i> cetak realisasi menjelaskan saat sistem mencetak data realisasi sesuai masa tanam yang di pilih	
<i>Pre Contidion</i>	Actor memilih masa tanam	
<i>Trigger</i>	Saat actor ingin mencetak tabel realisasi pada masa tanam yang diinginkan	
<i>Typical Course of Events</i>	<i>Actor action</i>	<i>System Response</i>
	<ul style="list-style-type: none"> - Memilih tahun masa tanam - Memilih bulan masa tanam - Klik 'Cetak PDF 	<ul style="list-style-type: none"> - Mendownload data realisasi dalam bentuk PDF
<i>Alternate Course</i>	Jika actor tidak memilih tahun dan bulan masa tanam, maka sistem mencetak realisasi pada periode sekarang.	
<i>Conclusion</i>	Data realisasi berhasil dicetak	
<i>Post Condition</i>	Data realisasi berhasil dicetak sesuai masa tanam	

8. tambah pemasukan

Tabel 4.9 *use case scenario* tambah pemasukan

<i>Use Case Name</i>	tambah pemasukan	
<i>Actor</i>	Ketua, Bendahara	
<i>Description</i>	<i>Use case</i> tambah pemasukan menggambarkan saat pengguna menambahkan data pemasukan	
<i>Pre Contidion</i>	Actor memasukan data pemasukan	
<i>Trigger</i>	Saat actor ingin menambah data pemasukan	
<i>Typical Course of Events</i>	<i>Actor action</i>	<i>System Response</i>
	<ul style="list-style-type: none"> - Memasukan data pemasukan - Memilih jenis pemasukan - Klik 'simpan data' 	<ul style="list-style-type: none"> - Memeriksa form yang kosong - Jika jenis pemasukan = hasil penjualan, maka data juga disimpan ke tabel laba-rugi - Menyimpan ke tabel pemasukan dan transaksi - Meredirect ke halaman tabel pemasukan
<i>Alternate Course</i>	Jika ada form yang kosong maka akan muncul pesan form diperlukan Klik 'kembali' jika ingin membatalkan menambah data pemasukan	
<i>Conclusion</i>	Data pemasukan berhasil ditambahkan	
<i>Post Condition</i>	Data pemasukan berhasil disimpan tabel pemasukan	

9. Tampil pemasukan

Tabel 4.10 *use case scenario* tampil pemasukan

<i>Use Case Name</i>	Tampil pemasukan	
<i>Actor</i>	Ketua , Bendahara	
<i>Description</i>	<i>Use case</i> tampil pemasukan menjelaskan saat sistem menampilkan data pemasukan sesuai periode yang di pilih	
<i>Pre Contidion</i>	Actor memilih masa tanam	
<i>Trigger</i>	Saat actor ingin menampilkan tabel pemasukan pada masa tanam yang diinginkan	
<i>Typical Course of Events</i>	<i>Actor action</i>	<i>System Response</i>
	<ul style="list-style-type: none"> - Memilih tahun masa tanam - Memilih bulan masa tanam - Klik 'tampilkan' 	<ul style="list-style-type: none"> - Menampilkan tabel pemasukan sesuai dengan periode yang dipilih.
<i>Alternate Course</i>	Jika actor tidak memilih tahun dan bulan masa tanam, maka sistem menampilkan seluruh data pemasukan di tabel pemasukan masa tanam saat ini.	
<i>Conclusion</i>	Data pemasukan berhasil ditampilkan	
<i>Post Condition</i>	Data pemasukan berhasil ditampilkan sesuai masa tanam	

10. Cetak pemasukan

Tabel 4.11 *use case scenario* cetak pemasukan

<i>Use Case Name</i>	cetak pemasukan	
<i>Actor</i>	Ketua, Bendahara	
<i>Description</i>	<i>Use case</i> cetak pemasukan menjelaskan saat sistem mencetak data pemasukan sesuai periode yang di pilih	
<i>Pre Contidion</i>	Actor memilih masa tanam	
<i>Trigger</i>	Saat actor ingin mencetak tabel pemasukan pada masa tanam yang diinginkan	
<i>Typical Course of Events</i>	<i>Actor action</i>	<i>System Response</i>
	<ul style="list-style-type: none"> - Memilih tahun masa tanam - Memilih bulan masa tanam - Klik 'Cetak PDF 	<ul style="list-style-type: none"> - Mendownload data pemasukan dalam bentuk PDF
<i>Alternate Course</i>	Jika actor tidak memilih tahun dan bulan periode, maka sistem mencetak pemasukan pada masa tanam sekarang.	
<i>Conclusion</i>	Data pemasukan berhasil dicetak	
<i>Post Condition</i>	Data pemasukan berhasil dicetak sesuai masa tanam	

11. Tampil laporan laba-rugi

Tabel 4.12 *use case scenario* tampil laporan laba-rugi

<i>Use Case Name</i>	Tampil laporan laba-rugi	
<i>Actor</i>	Ketua, Bendahara, Anggota	
<i>Description</i>	<i>Use case</i> tampil laporan laba-rugi menjelaskan saat sistem menampilkan tabel laba-rugi sesuai periode yang dipilih	
<i>Pre Contidion</i>	Sudah ada data di pengeluaran dan pemasukan yang berjenis 'produksi' sebelumnya.	
<i>Trigger</i>	Saat actor ingin melihat laporan laba-rugi setiap periode	
<i>Typical Course of Events</i>	<i>Actor action</i>	<i>System Response</i>
	<ul style="list-style-type: none"> - Pilih tahun periode - Pilih bulan periode - Klik 'tampilkan' 	<ul style="list-style-type: none"> - Menampilkan tabel laba rugi sesuai periode yang di pilih - Menampilkan jumlah laba-rugi di form laba-rugi
<i>Alternate Course</i>	Jika actor tidak memilih tahun dan bulan periode, maka sistem akan menampilkan tabel laba rugi pada periode sekarang	
<i>Conclusion</i>	data laba-rugi berhasil ditampilkan	
<i>Post Condition</i>	data laba-rugi berhasil ditampilkan sesuai periode yang di pilih	

12. Tampil laporan keuangan bulanan

Tabel 4.13 *use case scenario* tampil laporan keuangan bulanan

<i>Use Case Name</i>	Tampil laporan keuangan bulanan	
<i>Actor</i>	Ketua Bendahara, Anggota	
<i>Description</i>	<i>Use case</i> tampil laporan keuangan bulanan menjelaskan saat sistem menampilkan tabel keuangan bulanan sesuai periode yang dipilih	
<i>Pre Contidion</i>	Sudah ada data pengeluaran dan pemasukan sebelumnya.	
<i>Trigger</i>	Saat actor ingin melihat laporan keuangan bulanan setiap periode	
<i>Typical Course of Events</i>	<i>Actor action</i>	<i>System Response</i>
	<ul style="list-style-type: none"> - Pilih tahun periode - Pilih bulan periode - Klik 'tampilkan' 	<ul style="list-style-type: none"> - Menampilkan tabel keuangan bulanan sesuai periode yang di pilih - Menampilkan jumlah saldo saat ini.
<i>Alternate Course</i>	Jika actor tidak memilih tahun dan bulan periode, maka sistem akan menampilkan tabel keuangan bulanan pada periode sekarang	
<i>Conclusion</i>	data keuangan bulanan berhasil ditampilkan	
<i>Post Condition</i>	data keuangan bulanan berhasil ditampilkan sesuai periode yang di pilih	

13. Tampil Laporan Keuangan Tahunan

Tabel 4.14 *use case scenario* tampil laporan keuangan bulanan

<i>Use Case Name</i>	Tampil laporan keuangan tahunan	
<i>Actor</i>	Ketua, Bendahara, Anggota	
<i>Description</i>	<i>Use case</i> tampil laporan keuangan tahunan menjelaskan saat sistem menampilkan tabel keuangan tahunan sesuai tahun yang dipilih	
<i>Pre Contidion</i>	Sudah ada data pengeluaran dan pemasukan sebelumnya.	
<i>Trigger</i>	Saat actor ingin melihat laporan keuangan tahunan	
<i>Typical Course of Events</i>	<i>Actor action</i>	<i>System Response</i>
	<ul style="list-style-type: none"> - Pilih tahun periode - Klik 'tampilkan' 	<ul style="list-style-type: none"> - Menampilkan tabel keuangan tahunan sesuai tahun yang di pilih - Menampilkan jumlah saldo akhir tahun.
<i>Alternate Course</i>	Jika actor tidak memilih tahun, maka sistem akan menampilkan tabel keuangan tahunan pada tahun sekarang	
<i>Conclusion</i>	data keuangan tahunan berhasil ditampilkan	
<i>Post Condition</i>	data keuangan tahunan berhasil ditampilkan sesuai periode yang di pilih	

14. Cetak laporan laba-rugi

Tabel 4.15 *use case scenario* cetak laporan laba-rugi

<i>Use Case Name</i>	cetak laporan laba-rugi	
<i>Actor</i>	Ketua, Bendahara, Anggota	
<i>Description</i>	<i>Use case</i> cetak laporan laba-rugi menjelaskan saat sistem mencetak data laporan laba-rugi sesuai periode yang di pilih	
<i>Pre Contidion</i>	Actor memilih periode	
<i>Trigger</i>	Saat actor ingin mencetak tabel laporan laba-rugi pada periode yang diinginkan	
<i>Typical Course of Events</i>	<i>Actor action</i>	<i>System Response</i>
	<ul style="list-style-type: none"> - Memilih tahun periode - Memilih bulan periode - Klik 'Cetak PDF 	<ul style="list-style-type: none"> - Mendownload data pemasukan dalam bentuk PDF
<i>Alternate Course</i>	Jika actor tidak memilih tahun dan bulan periode, maka sistem mencetak laporan laba-rugi pada periode sekarang.	
<i>Conclusion</i>	Data laporan laba-rugi berhasil dicetak	
<i>Post Condition</i>	Data laporan laba-rugi berhasil dicetak sesuai periode	

15. Cetak laporan keuangan bulanan

Tabel 4.16 *use case scenario* cetak laporan keuangan bulanan

<i>Use Case Name</i>	cetak laporan keuangan bulanan	
<i>Actor</i>	Admin, Anggota	
<i>Description</i>	Use case cetak laporan keuangan bulanan menjelaskan saat sistem mencetak data laporan keuangan bulanan sesuai periode yang di pilih	
<i>Pre Contidion</i>	Actor memilih periode	
<i>Trigger</i>	Saat actor ingin mencetak tabel laporan keuangan bulanan pada periode yang diinginkan	
<i>Typical Course of Events</i>	<i>Actor action</i>	<i>System Response</i>
	<ul style="list-style-type: none"> - Memilih tahun periode - Memilih bulan periode - Klik 'Cetak PDF 	<ul style="list-style-type: none"> - Mendownload data pemasukan dalam bentuk PDF
<i>Alternate Course</i>	Jika actor tidak memilih tahun dan bulan periode, maka sistem mencetak laporan keuangan bulanan pada periode sekarang.	
<i>Conclusion</i>	Data laporan keuangan bulanan berhasil dicetak	
<i>Post Condition</i>	Data laporan keuangan bulanan berhasil dicetak sesuai periode	

16. Tampil harga produk

Tabel 4.17 *use case scenario* tampil harga produk

<i>Use Case Name</i>	Tampil harga produk	
<i>Actor</i>	Ketua, Bendahara, Anggota	
<i>Description</i>	Use case tampil harga pokok menjelaskan proses menampilkan harga produk	
<i>Pre Contidion</i>	Sudah ada data di tabel pengeluaran berjenis 'produksi'	
<i>Trigger</i>	Saat actor ingin menampilkan harga pokok produk yang akan di jual sesuai dengan pengeluaran produksi	
<i>Typical Course of Events</i>	<i>Actor action</i>	<i>System Response</i>
	- Klik 'menu harga jual produk	- Tampil harga jual produk
<i>Alternate Course</i>	Jika actor tidak memilih masa tanam pengeluaran produksi, maka sistem menampilkan berdasarkan harga jual produk berdasarkan pengeluaran produksi pada masa tanam sekarang.	
<i>Conclusion</i>	Harga pokok berhasil ditampilkan	
<i>Post Condition</i>	Harga pokok berhasil ditampilkan	

17. Tampil Anggaran

Tabel 4.18 *use case scenario* tampil Anggaran

<i>Use Case Name</i>	Tampil Anggaran	
<i>Actor</i>	Ketua, Bendahara, Anggota	
<i>Description</i>	<i>Use case</i> tampil anggaran menjelaskan proses menampilkan data Anggaran sesuai periode yang diinginkan	
<i>Pre Contidion</i>	Sudah ada data di tabel Anggaran	
<i>Trigger</i>	Saat actor ingin menampilkan Anggaran per periode	
<i>Typical Course of Events</i>	<i>Actor action</i>	<i>System Response</i>
	<ul style="list-style-type: none"> - Memilih tahun periode - Memilih bulan periode - Klik 'tampilkan' 	<ul style="list-style-type: none"> - Merequest database untuk menampilkan data anggaran sesuai tahun dan bulan - Menampilkan data Anggaran
<i>Alternate Course</i>	Jika actor tidak memilih tahun dan bulan periode anggaran, maka sistem akan menampilkan data Anggaran pada periode sekarang.	
<i>Conclusion</i>	Data Anggaran berhasil ditampilkan	
<i>Post Condition</i>	Data Anggaran berhasil ditampilkan sesuai periode.	

18. Tambah sayuran, bibit, nutrisi dan Bahan lain

Tabel 4.19 *use case scenario* Tambah sayuran, bibit, nutrisi dan Bahan lain

<i>Use Case Name</i>	Tambah sayuran, bibit, nutrisi dan Bahan lain	
<i>Actor</i>	Ketua, Bendahara, Anggota	
<i>Description</i>	<i>Use case</i> Tambah sayuran, bibit, nutrisi dan Bahan lain menjelaskan proses menambah data sayuran , bibit, Nutrisi dan Bahan lain	
<i>Pre Contidion</i>	Tampil data tabel Sayuran, Bibit, Nutrisi, dan Bahan Lain	
<i>Trigger</i>	Saat actor ingin tambah data tabel Sayuran, Bibit, Nutrisi, dan Bahan Lain	
<i>Typical Course of Events</i>	<i>Actor action</i>	<i>System Response</i>
	<ul style="list-style-type: none"> - Klik menu sayuran, - Pilih salah satu yang ingin ditambahkan. - Klik "tambah" 	<ul style="list-style-type: none"> - Menambah data sayuran, bibit, nutrisi, dan bahan lain - Menyimpan data terakhir di tambahkan
<i>Alternate Course</i>		
<i>Conclusion</i>	data tabel Sayuran, Bibit, Nutrisi, dan Bahan Lain berhasil ditambah	
<i>Post Condition</i>	Data Tampil data tabel Sayuran, Bibit, Nutrisi, dan Bahan Lain	

19. edit harga produk

Tabel 4.20 *use case scenario* edit harga produk

<i>Use Case Name</i>	Edit harga produk	
<i>Actor</i>	Ketua, Bendahara, Anggota	
<i>Description</i>	<i>Use case</i> edit harga pokok menjelaskan proses menampilkan harga pokok	
<i>Pre Contidion</i>	Sudah ada data di tabel pengeluaran berjenis 'produksi'	
<i>Trigger</i>	Saat actor ingin mengedit harga pokok produk yang akan di jual sesuai dengan pengeluaran produksi	
<i>Typical Course of Events</i>	<i>Actor action</i>	<i>System Response</i>
	<ul style="list-style-type: none"> - Memilih jenis sayur yang akan dihitung harganya. - Mengisi form total hasil panen - Memilih profit. - Klik 'tampil harga pokok' 	<ul style="list-style-type: none"> - Menghitung harga pokok denga rumus penentuan harga pokok - Menampilkan harga pokok di form harga pokok
<i>Alternate Course</i>	Jika actor tidak memilih tahun dan bulan periode pengeluaran produksi, maka sistem menghitung berdasarkan harga pokok berdasarkan pengeluaran produksi pada periode sekarang.	
<i>Conclusion</i>	Harga pokok berhasil edit	
<i>Post Condition</i>	Harga pokok berhasil edit	

20. Logout

Tabel 4.21 *use case scenario* logout

<i>Use Case Name</i>	Logout	
<i>Actor</i>	Admin, Anggota	
<i>Description</i>	<i>Use case</i> logout digunakan untuk keluar dari aplikasi dan memastikan pengguna diluar hak akses tidak bisa mengakses aplikasi	
<i>Pre Contidion</i>	Actor dalam keadaan login	
<i>Trigger</i>	Saat pengguna ingin mengakses aplikasi sesuai dengan hak akses	
<i>Typical Course of Events</i>	<i>Actor action</i>	<i>System Response</i>
	<ul style="list-style-type: none"> - Klik 'logout' 	<ul style="list-style-type: none"> - Menutup akses aplikasi - Redirect ke halaman login
<i>Alternate Course</i>	-	
<i>Conclusion</i>	Actor keluar dari aplikasi	
<i>Post Condition</i>	Actor keluar dari aplikasi	

21. Non-aktifkan pengguna

Tabel 4.22 *use case scenario* Non-aktifkan pengguna

<i>Use Case Name</i>	Non-aktifkan pengguna	
<i>Actor</i>	Ketua	
<i>Description</i>	Use case non-aktifkan pengguna digunakan untuk menonaktifkan pengguna yang tidak berhak mengakses aplikasi	
<i>Pre Contidion</i>	Pengguna dalam keadaan aktif	
<i>Trigger</i>	Saat pengguna ketua ingin menonaktifkan salah satu pengguna	
<i>Typical Course of Events</i>	<i>Actor action</i>	<i>System Response</i>
	<ul style="list-style-type: none"> - Klik menu pengguna - Klik "non-aktifkan" 	<ul style="list-style-type: none"> - Mengubah hak akses pengguna menjadi no aktif - Tombol berubah menjadi "Aktifkan"
<i>Alternate Course</i>	-	
<i>Conclusion</i>	Actor yang di nonaktifkan tidak dapat login.	
<i>Post Condition</i>	Actor keluar dai hak akses	

22. Aktifkan pengguna

Tabel 4.23 *use case scenario* Aktifkan pengguna

<i>Use Case Name</i>	Aktifkan pengguna	
<i>Actor</i>	Ketua	
<i>Description</i>	Use case aktifkan pengguna digunakan untuk mengaktifkan pengguna yang tidak berhak mengakses aplikasi	
<i>Pre Contidion</i>	Pengguna dalam keadaan aktif	
<i>Trigger</i>	Saat pengguna ketua ingin mengaktifkan salah satu pengguna	
<i>Typical Course of Events</i>	<i>Actor action</i>	<i>System Response</i>
	<ul style="list-style-type: none"> - Klik menu pengguna - Klik "aktifkan" 	<ul style="list-style-type: none"> - Mengubah hak akses pengguna menjadi aktif - Tombol berubah menjadi "non-Aktifkan"
<i>Alternate Course</i>	-	
<i>Conclusion</i>	Actor yang di aktifkan dapat login.	
<i>Post Condition</i>	Actor yang di aktifkan dapat login	

23. Edit profil pengguna

Tabel 4.24 *use case scenario* edit profil pengguna

<i>Use Case Name</i>	Edit profil pengguna	
<i>Actor</i>	Ketua, bendahara, anggota	
<i>Description</i>	Use case edit profil pengguna digunakan untuk mengubah nama, email dan password pengguna.	
<i>Pre Contidion</i>	Pengguna dalam keadaan aktif dan login	
<i>Trigger</i>	Saat pengguna ketua ingin mengaktifkan salah satu pengguna	
<i>Typical Course of Events</i>	<i>Actor action</i>	<i>System Response</i>
	<ul style="list-style-type: none"> - Klik nama pengguna - Pilih profil - Mengganti nama - Mengganti email - Masukan password - Masukan <i>confirm password</i> - Klik "simpan" 	<ul style="list-style-type: none"> - <i>Update</i> nama, email, dan password pengguna - Muncul pesan profil berhasil di ubah.
<i>Alternate Course</i>	Klik kembali jika ingin mambatalkan edit profil	
<i>Conclusion</i>	Profil pengguna telah berubah	
<i>Post Condition</i>	Profil pengguna telah berubah	

24. Reset password pengguna

Tabel 4.25 *use case scenario* reset password pengguna

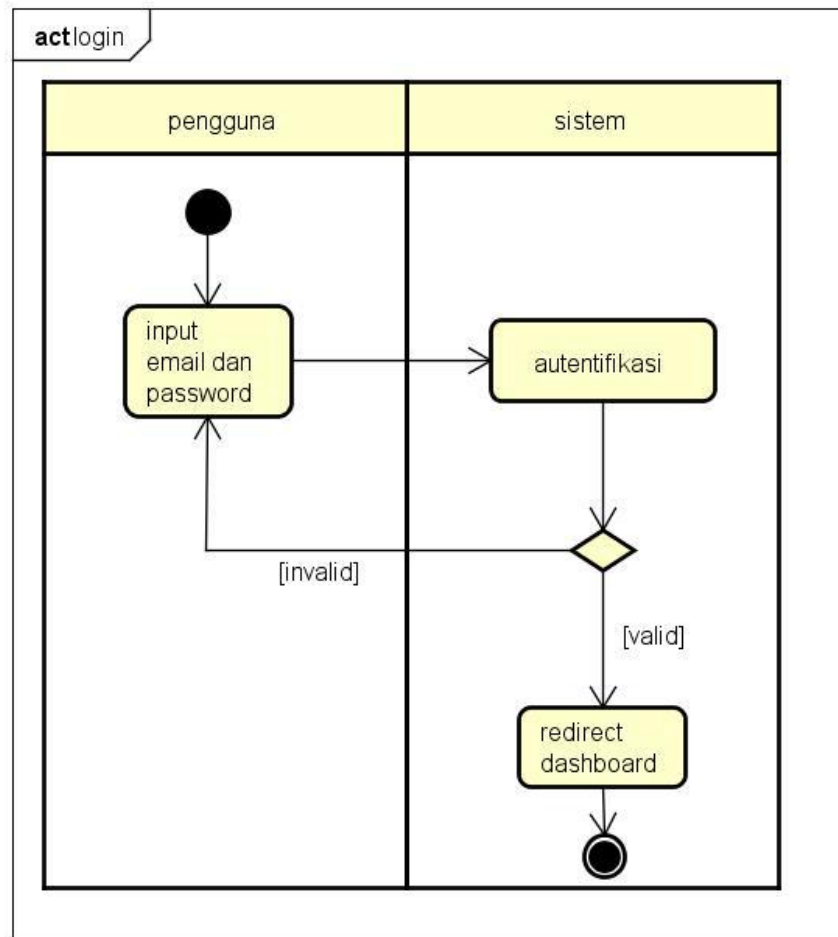
<i>Use Case Name</i>	Reset password pengguna	
<i>Actor</i>	Ketua	
<i>Description</i>	Use case reset password pengguna digunakan untuk mengubah password pengguna ke password default.	
<i>Pre Contidion</i>	Pengguna dengan status aktif	
<i>Trigger</i>	Saat pengguna ketua ingin mereset salah satu password pengguna	
<i>Typical Course of Events</i>	<i>Actor action</i>	<i>System Response</i>
	<ul style="list-style-type: none"> - Klik menu pengguna - Klik "aktifkan" 	<ul style="list-style-type: none"> - Mengubah password pengguna menjadi password default - Muncul pesan sukses reset password.
<i>Alternate Course</i>		
<i>Conclusion</i>	password pengguna telah di reset	
<i>Post Condition</i>	Pengguna yang di reset dapat login dengan password default	

C. Desain Sistem (Pemodelan)

1. Desain Aktivitas

Setelah mendefinisikan setiap *use case* dengan detail, tahap selanjutnya adalah menggambarkan *workflow* atau aliran kerja dengan *activity diagram*. Yaitu diagram yang menggambarkan alur aktivitas setiap *use case* yang dilakukan oleh sistem.

1.1. Activity diagram login

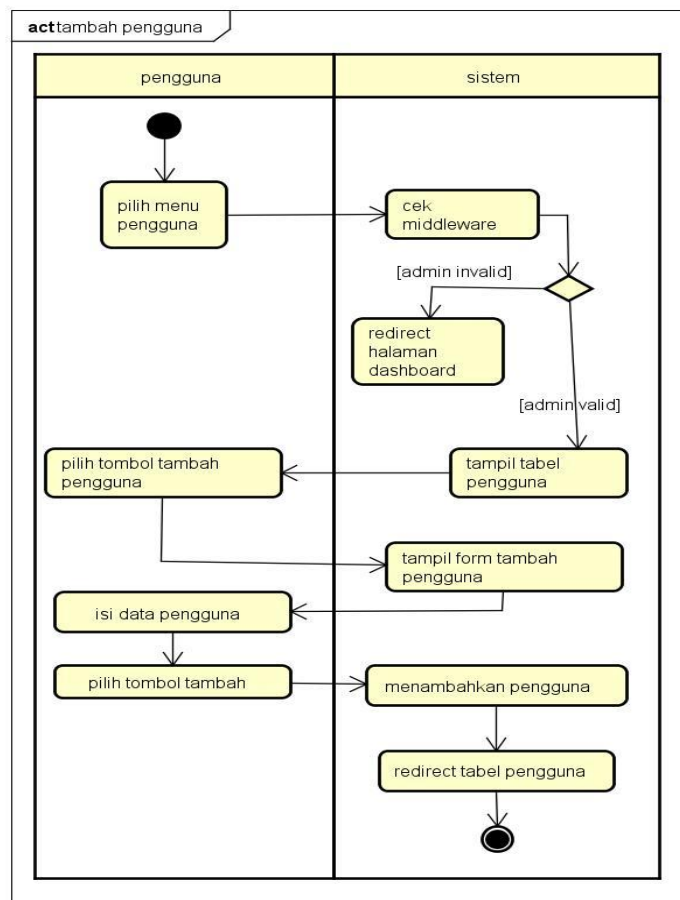


Gambar 4.2. activity diagram login

Aktivitas pada gambar 4.2 dilakukan oleh actor Admin, Anggota melakukan pengisian *email* dan *password* pada menu

login. Kemudian sistem mengecek ketersediaan pengguna di tabel pengguna. Jika valid maka sistem mengarahkan ke halaman *dashboard*, tetapi jika tidak valid maka sistem mengarahkan ke halaman login kembali dan memunculkan pesan kesalahan.

1.2. Activity diagram tambah pengguna

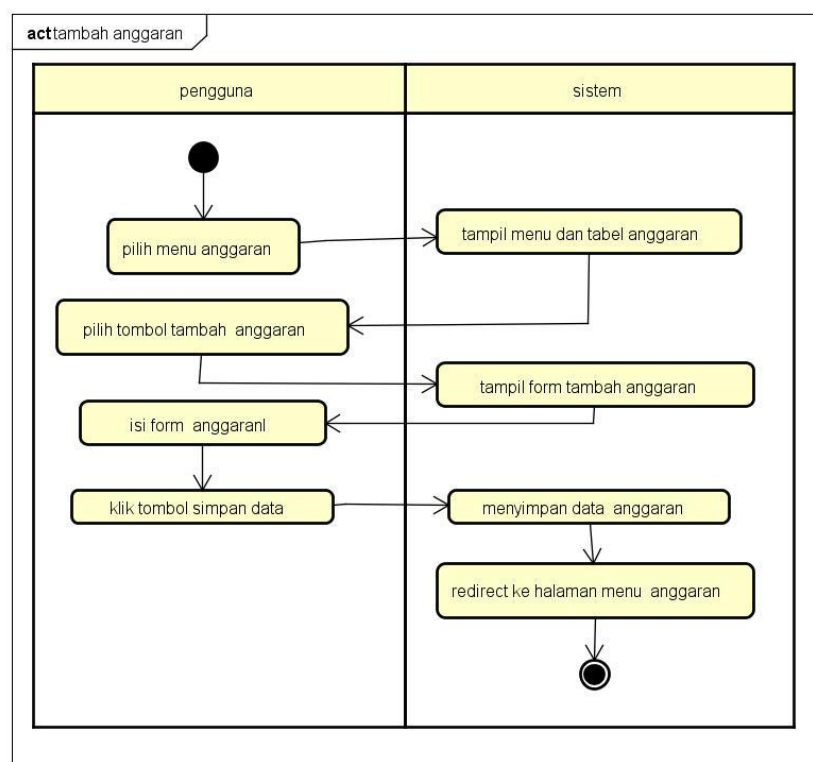


Gambar 4.3 Activity diagram tambah pengguna

Gambar 4.3 Aktivitas tambah pengguna hanya bisa dilakukan oleh admin. Setelah login sebagai admin, pilih menu pengguna kemudian sistem akan mengecek *middleware* atau hak akses apakah login sebagai admin atau bukan. Jika bukan maka

akan kembali ke halaman dashboard. Kemudian klik tombol tambah pengguna dan sistem akan menampilkan form tambah pengguna. Actor kemudian mengisi username, email, dan jabatan. Kemudian setelah selesai klik tombol tambah. Maka pengguna berhasil di tambahkan.

1.3. Activity diagram tambah Anggaran

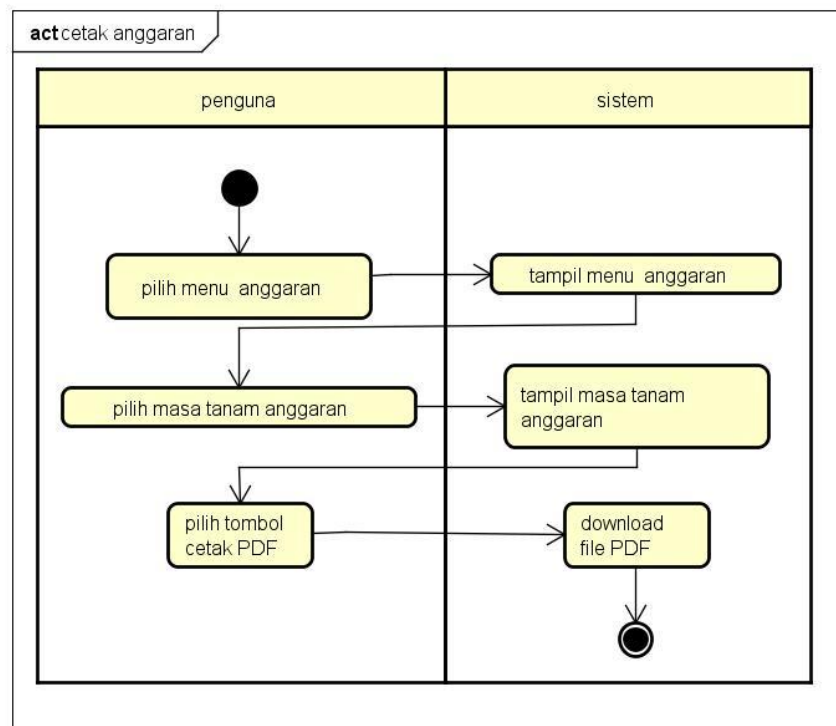


Gambar 4.4 Activity diagram tambah Anggaran

Gambar 4.4 Aktivitas tambah Anggaran dimulai dengan memilih menu Anggaran kemudian pilih tombol tambah Anggaran. Kemudian actor mengisi data Anggaran yaitu masa tanam, pilih bibit, pilih nutrisi, pilih bahan lain, isi anggran bibit,

anggaran nutrisi, anggaran bahan lain dan keterangan. Selanjutnya klik tombol simpan data. Dan sistem akan meredirect ke halaman menu Anggaran.

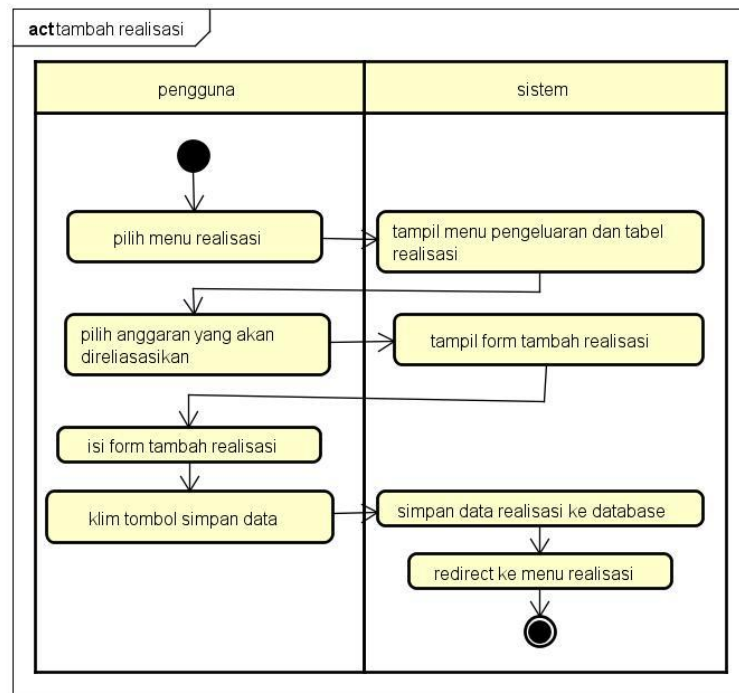
1.4. Activity diagram cetak Anggaran



Gambar 4.5 Activity diagram cetak anggaran

Sesuai gambar 4.5 Aktivitas cetak anggaran menjelaskan alur saat pengguna akan mencetak anggaran. Dimulai dari memilih menu anggaran, kemudian memilih periode anggaran terdiri tahun dan bulan. Kemudian pilih tombol cetak PDF. Maka sistem akan mendownload data anggaran sesuai periode yang dipilih.

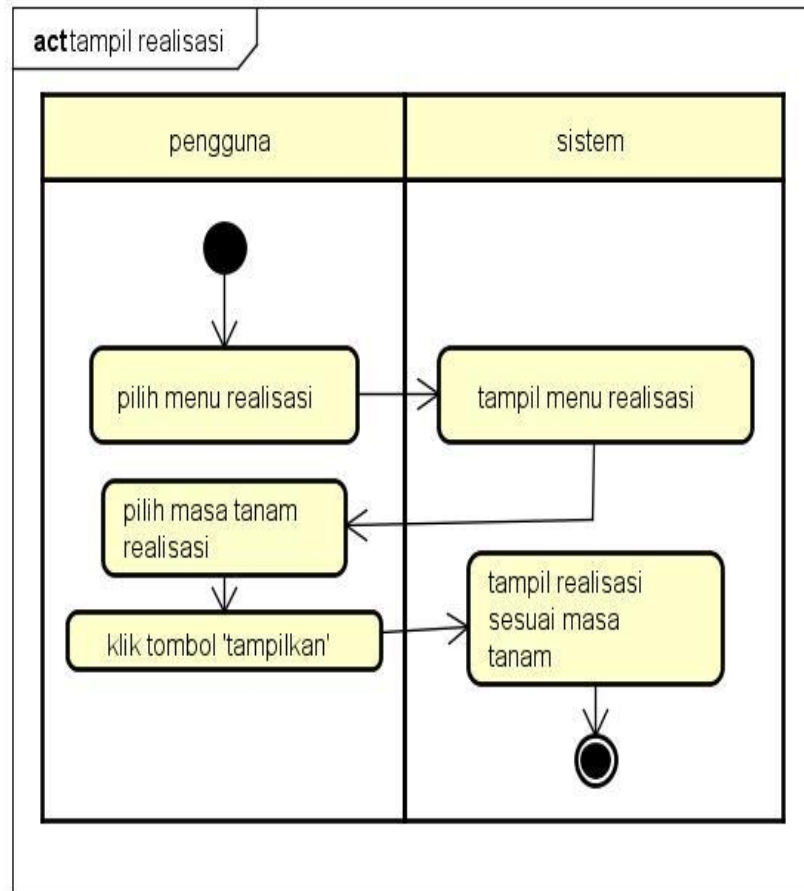
1.5. Activity diagram tambah realisasi



Gambar 4.6 Activity diagram tambah realisasi

Gambar 4.6 menjelaskan aliran kerja aktifitas sistem saat pengguna akan menambahkan data realisasi. Dimulai dari memilih menu pengeluaran, maka sistem akan menampilkan menu anggaran. Kemudian pengguna memilih anggaran manayang akan di realisasikan . kemudaian isi data realisasi. Kemudian pilih tombol 'tambah'. Maka sistem akan menambahkan data realisasi.

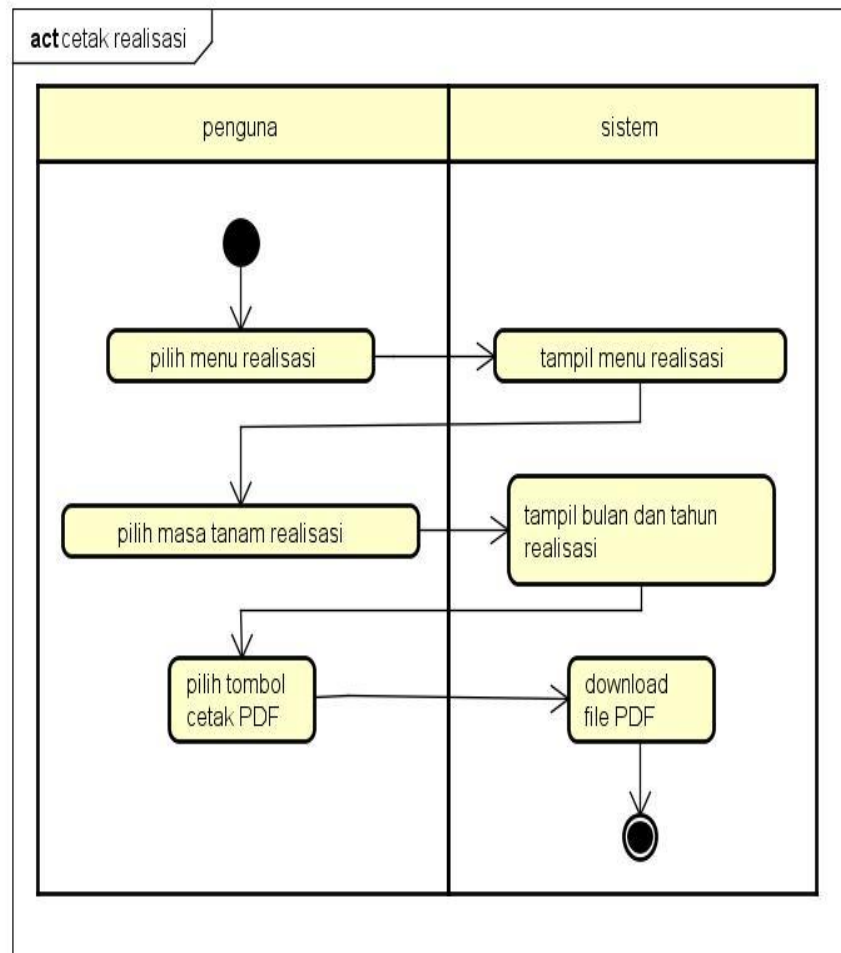
1.6. Activity diagram tampil realisasi



gambar 4.7 Activity diagram tampil realisasi

gambar 4.7 menjelaskan aktifitas tampil realisasi menjelaskan alur sistem dalam menampilkan data realisasi. dimulai dari pengguna memilih menu realisasi, kemudian sistem akan menampilkan menu realisasi. Pengguna memilih masa tanam realisasi terdiri dari tahun dan bulan. Kemudian pilih tombol tampilkan. Maka sistem akan menampilkan data realisasi sesuai masa tanam yang dipilih.

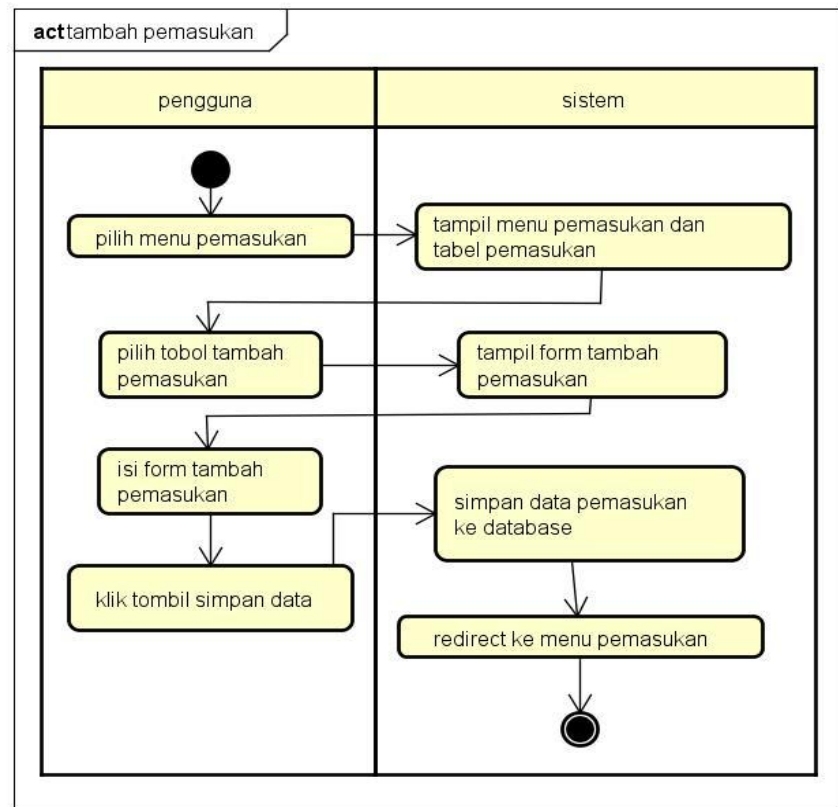
1.7. Activity diagram cetak realisasi



Gambar 4.8 Activity diagram cetak realisasi

Gambar 4.8 aktifitas cetak realisasi menjelaskan alur saat sistem mencetak data pengeluaran dalam bentuk PDF. Dimulai dari pengguna memilih menu realisasi. Kemudian sistem akan menampilkan menu realisasi. Pengguna memilih masa tanam realisasi yang akan dicetak terdiri dari tahun dan bulan. Kemudian pilih tombol 'cetak PDF'. Maka sistem akan men-*download* data realisasi dalam bentuk PDF.

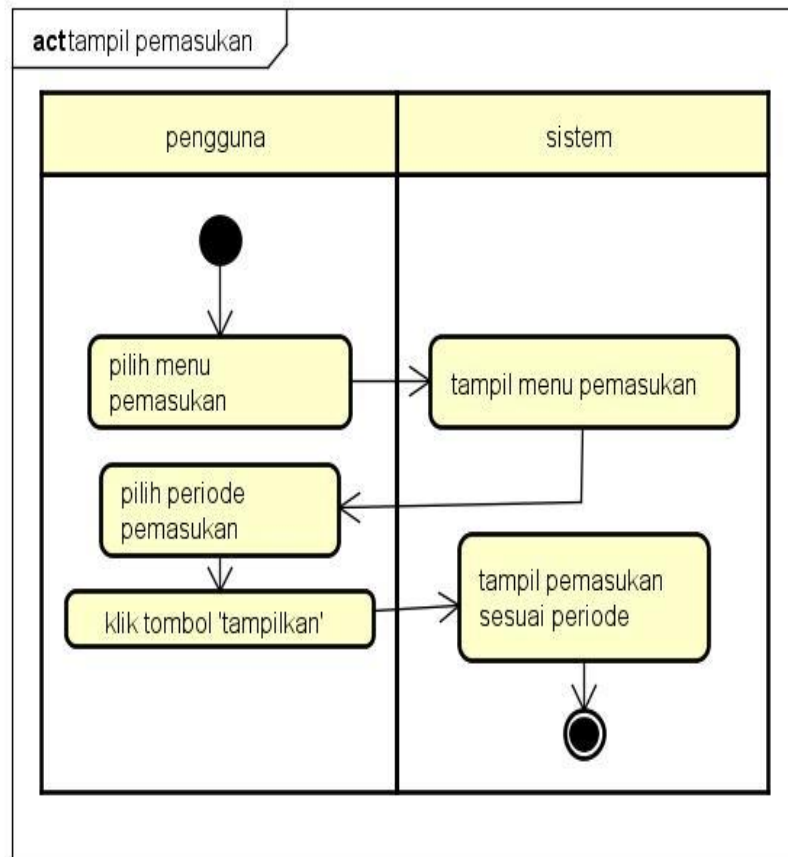
1.8. Activity diagram tambah pemasukan



Gambar 4.9 Activity diagram tambah pemasukan

Sesuai gambar 4.9 aktifitas tambah pemasukan menjelaskan alur untuk menambah data pemasukan. Dimulai dari pengguna memilih menu pemasukan, kemudian sistem akan menampilkan menu pemasukan. Pengguna kemudian pilih tombol 'tambah pemasukan'. Maka sistem akan menampilkan form untuk menambah pemasukan. Terdiri dari tanggal, deskripsi, jumlah pemasukan, dan jenis pemasukan. Kemudian pilih tombol 'simpan data'. Maka sistem akan menyimpan data ke database dan me-*redirect* ke halaman menu pemasukan.

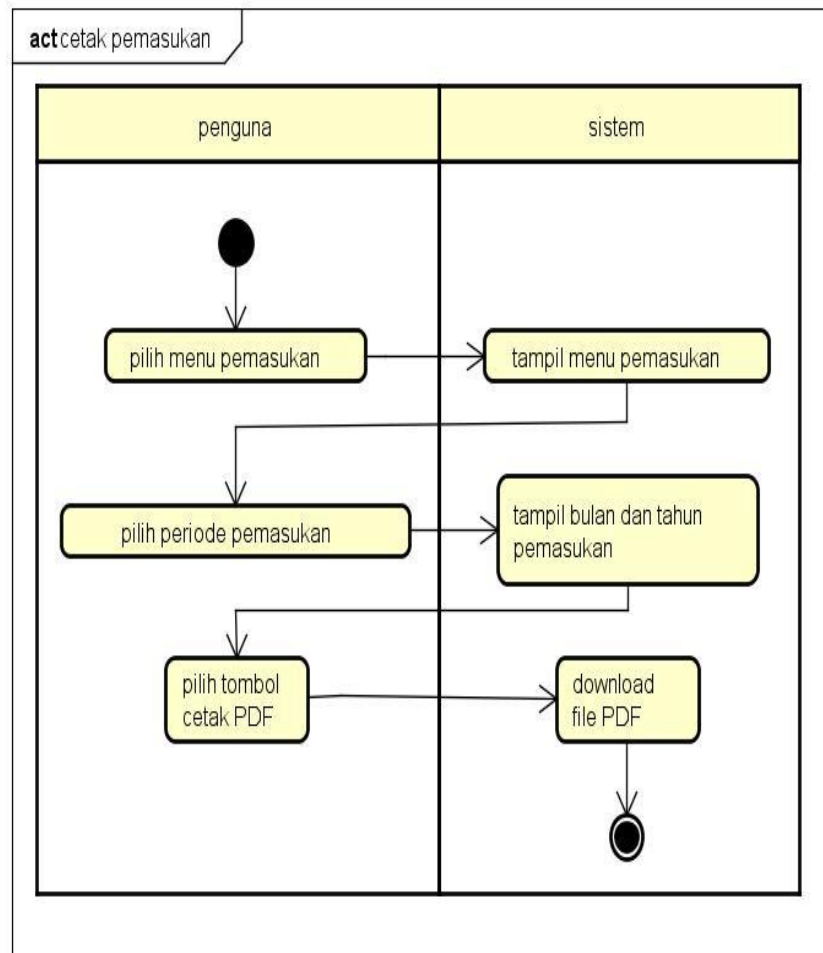
1.9. Activity diagram tampil pemasukan



Gambar 4.10 Activity diagram tampil pemasukan

Sesuai gambar 4.10 aktifitas tampil pemasukan menjelaskan alur sistem dalam menampilkan data pemasukan. Dimulai dari pengguna memilih menu pemasukan, kemudian sistem akan menampilkan menu pemasukan. Pengguna memilih periode pemasukan terdiri dari tahun dan bulan. Kemudian pilih tombol tampilkan. Maka sistem akan menampilkan data pemasukan sesuai periode yang dipilih.

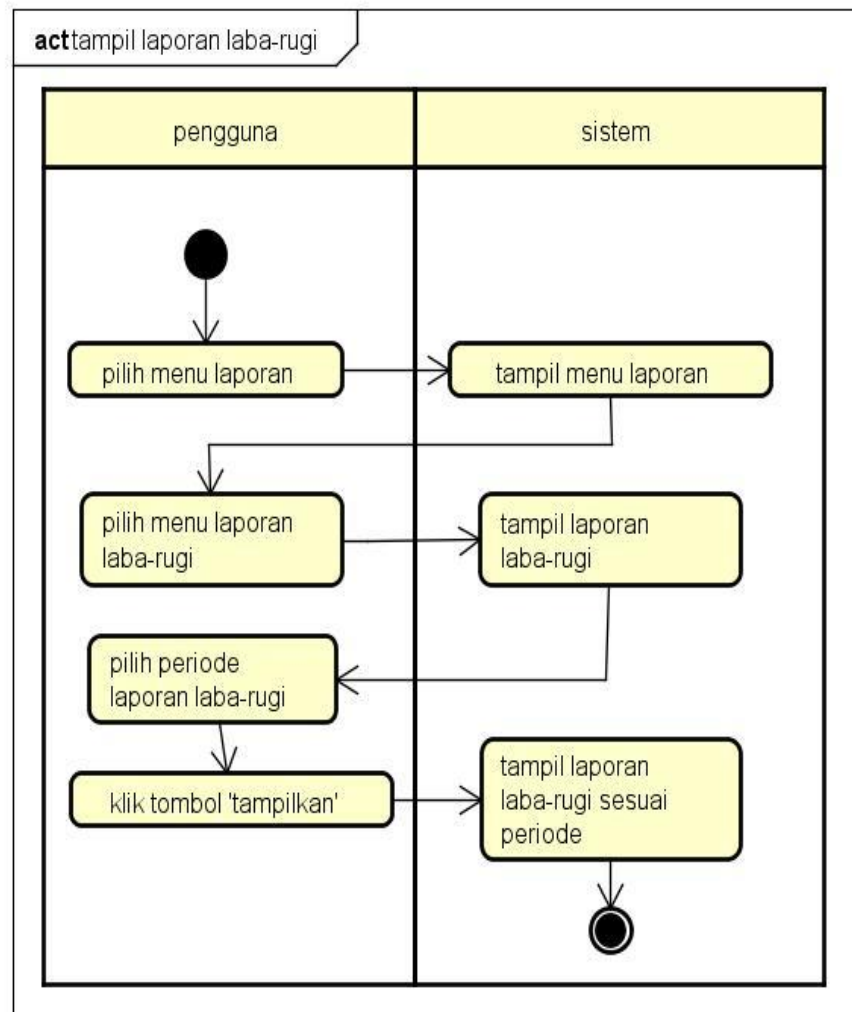
1.10. Activity diagram cetak pemasukan



Gambar 4.11 Activity diagram cetak pemasukan

Gambar 4.11 yaitu aktifitas cetak pemasukan menjelaskan alur saat sistem mencetak data pemasukan dalam bentuk PDF. Dimulai dari pengguna memilih menu pemasukan. Kemudian sistem akan menampilkan menu pemasukan. Pengguna memilih periode pemasukan yang akan dicetak terdiri dari tahun dan bulan. Kemudian pilih tombol 'cetak PDF'. Maka sistem akan *download* data pemasukan

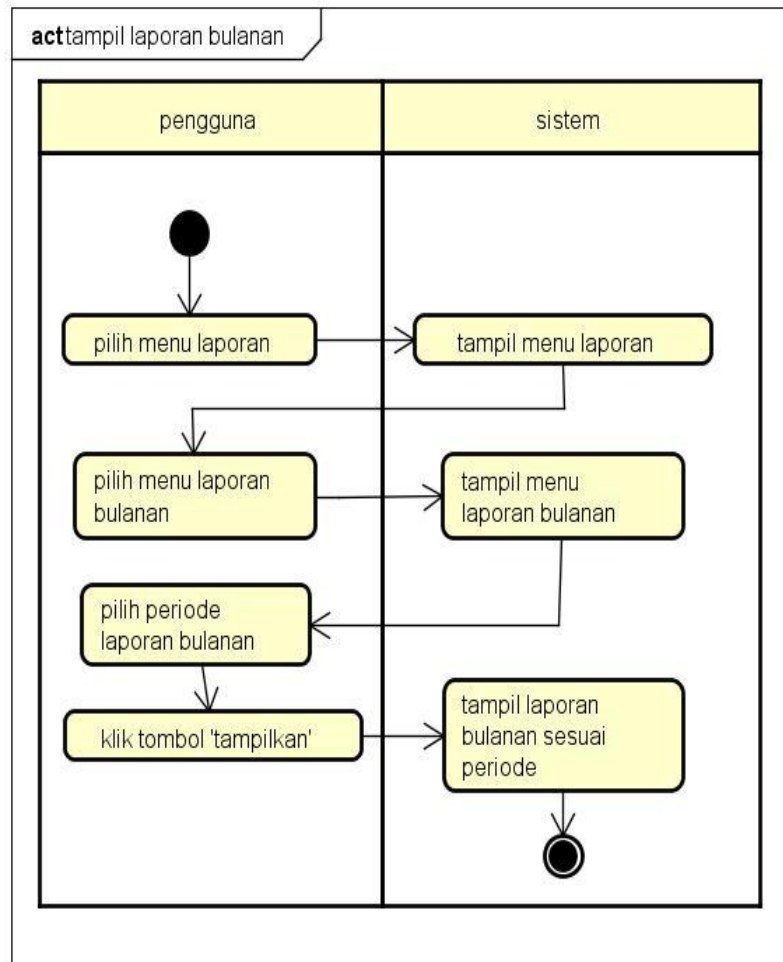
1.11. Activity diagram tampil laporan laba-rugi



Gambar 4.12 Activity diagram tampil laporan laba-rugi

Gambar 4.12 aktifitas tampil laporan laba rugi menjelaskan alur saat sistem menampilkan laporan laba rugi. Dimulai dari pengguna memilih menu laporan, kemudian memilih laporan laba-rugi. Maka sistem akan menampilkan data laba-rugi. Pengguna memilih periode laporan terdiri dari tahun dan bulan. Kemudian pengguna pilih tombol 'tampilkan'. Maka sistem akan menampilkan laporan laba-rugi sesuai periode.

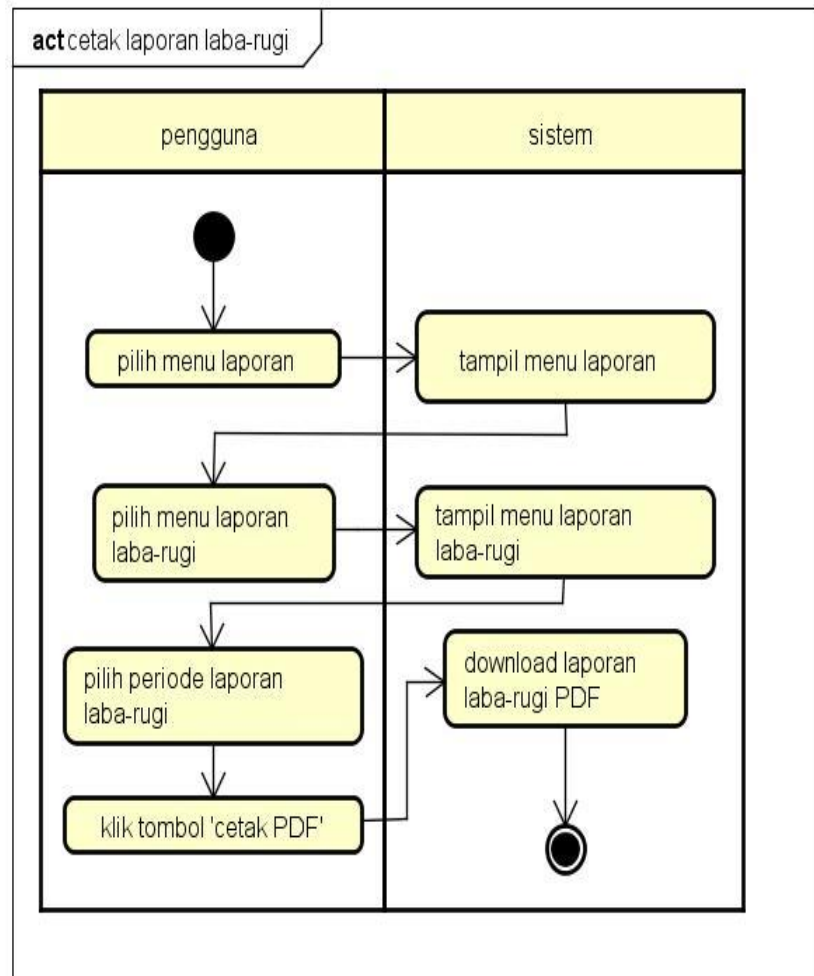
1.12. Activity diagram tampil laporan keuangan bulanan



Gambar 4.13 Activity diagram tampil laporan keuangan bulanan

Gambar 4.13 menjelaskan alur saat sistem menampilkan laporan bulanan. Dimulai dari pengguna memilih menu laporan, kemudian memilih laporan bulanan. Maka sistem akan menampilkan data laporan bulanan. Pengguna memilih periode laporan terdiri dari tahun dan bulan. Kemudian pengguna pilih tombol 'tampilkan'. Maka sistem akan menampilkan laporan bulanan sesuai periode yang dipilih.

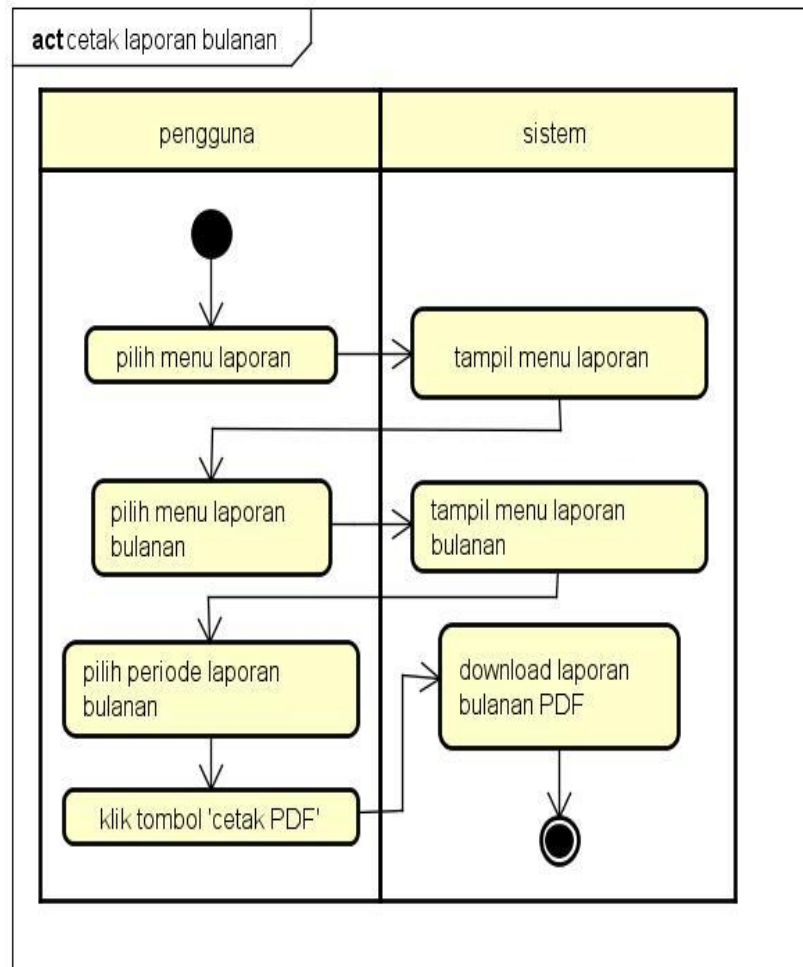
1.13. Activity diagram cetak laporan laba-rugi



Gambar 4.14 Activity diagram cetak laporan laba-rugi

Gambar 4.14 menjelaskan alur saat sistem mencetak data laporan laba-rugi dalam bentuk PDF. Dimulai dari pengguna memilih menu laporan. Kemudian sistem akan menampilkan menu laporan laba-rugi. Pengguna memilih periode laba-rugi yang akan dicetak terdiri dari tahun dan bulan. Kemudian pilih tombol 'cetak PDF'. Maka sistem akan *men-download* data laporan laba-rugi dalam bentuk PDF.

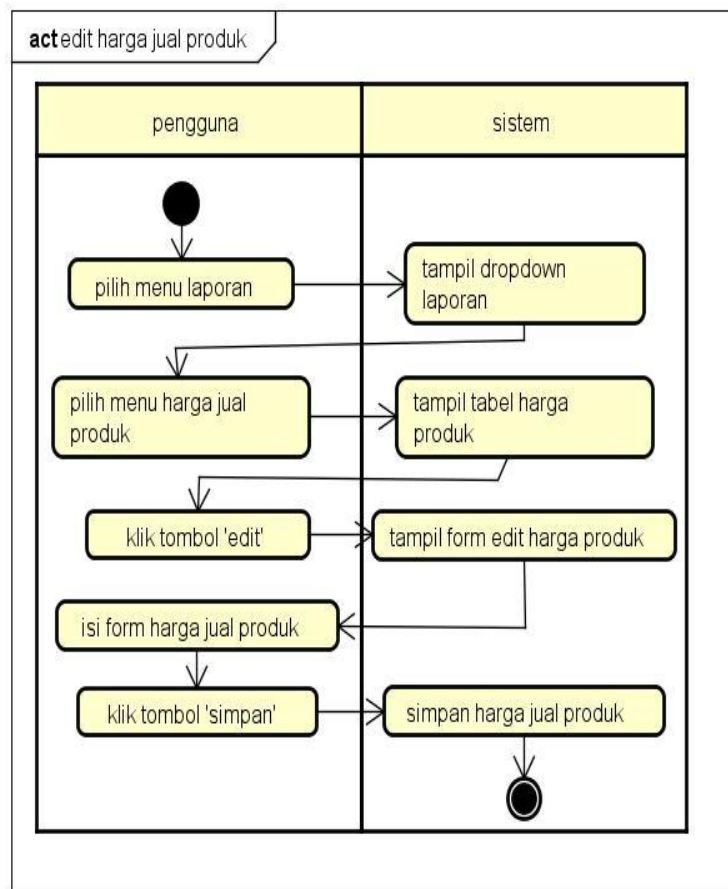
1.14. Activity diagram cetak laporan keuangan bulanan



Gambar 4.15 Activity diagram cetak laporan keuangan bulanan

Gambar 4.15 menjelaskan alur saat sistem mencetak data laporan keuangan bulanan dalam bentuk PDF. Dimulai dari pengguna memilih menu laporan. Kemudian sistem akan menampilkan menu laporan keuangan bulanan. Pengguna memilih periode laporan keuangan bulanan yang akan dicetak terdiri dari tahun dan bulan. Kemudian pilih tombol 'cetak PDF'. Maka sistem akan men-*download* data laporan keuangan bulanan dalam bentuk PDF.

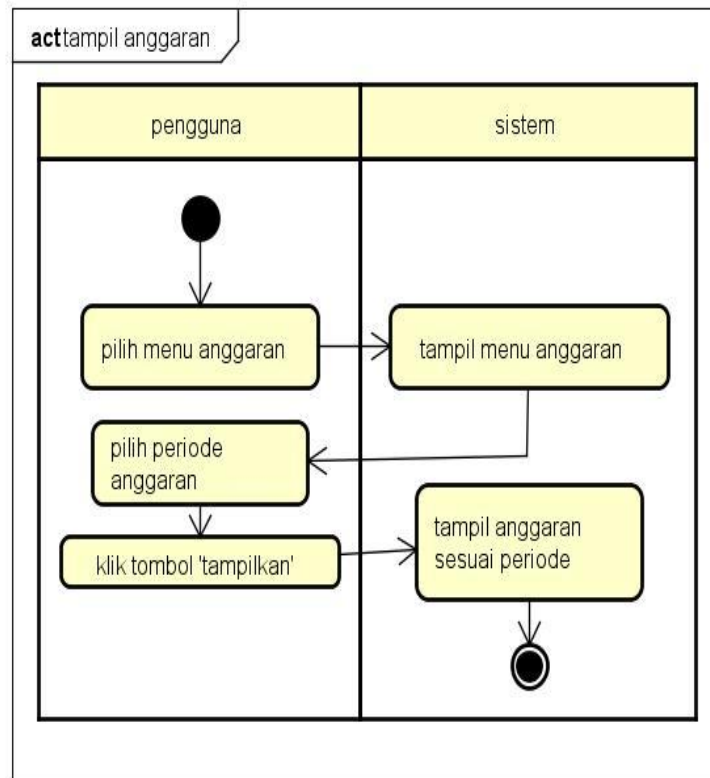
1.15. Activity diagram edit harga produk



Gambar 4.16 Activity diagram edit harga jual produk

Gambar 4.16 menjelaskan alur sistem dalam mengubah harga jual produk sayuran hidroponik. Dimulai dari pengguna memilih menu laporan, sistem akan menampilkan menu *dropdown*. Kemudian pengguna memilih menu harga jual produk. Sistem akan menampilkan form perhitungan harga pokok. Pengguna mengisi form yang terdiri dari jumlah hasil panen, profit keuntungan yang diinginkan dalam persentase, harga pasar, dan harga jual. Pengguna lalu memilih tombol 'simpan'. Maka sistem akan menyimpan harga jual produk di tabel harga jual produk.

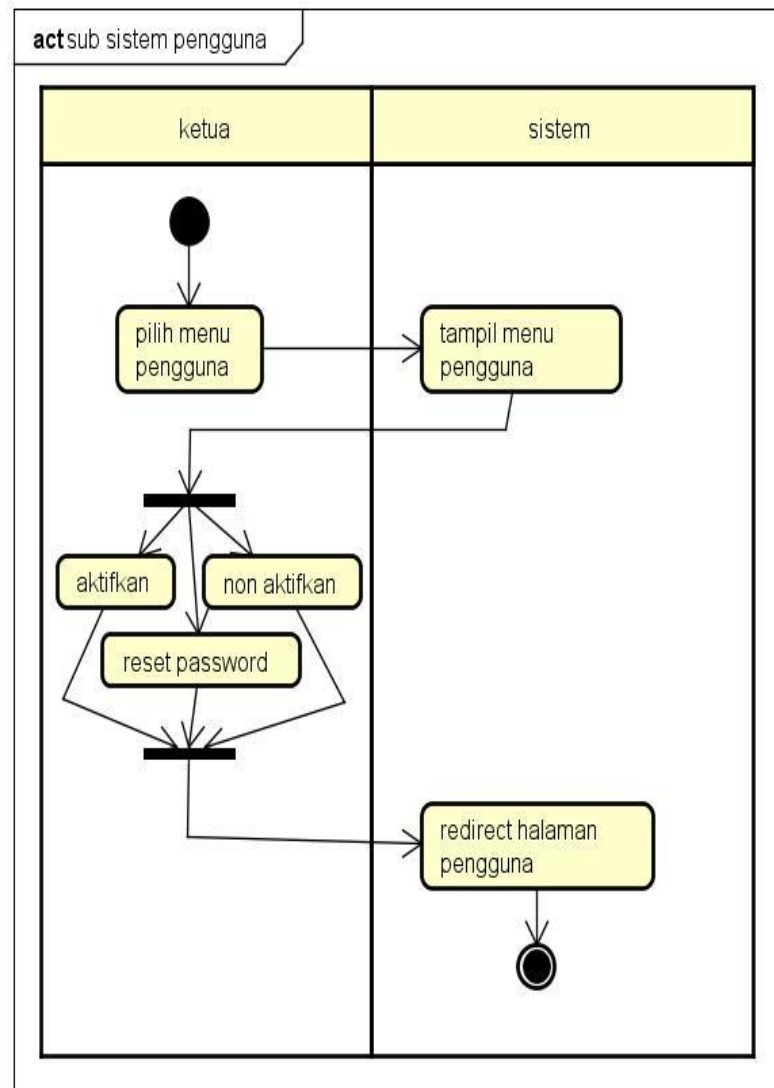
1.16. Activity diagram tampil anggaran



Gambar 4.17 Activity diagram tampil anggaran

Gambar 4.17 menjelaskan alur sistem dalam menampilkan data anggaran. Dimulai dari pengguna memilih menu anggaran, kemudian sistem akan menampilkan menu anggaran. Pengguna memilih periode anggaran terdiri dari tahun dan bulan. Kemudian pilih tombol tampilkan. Maka sistem akan menampilkan data anggaran sesuai periode yang dipilih.

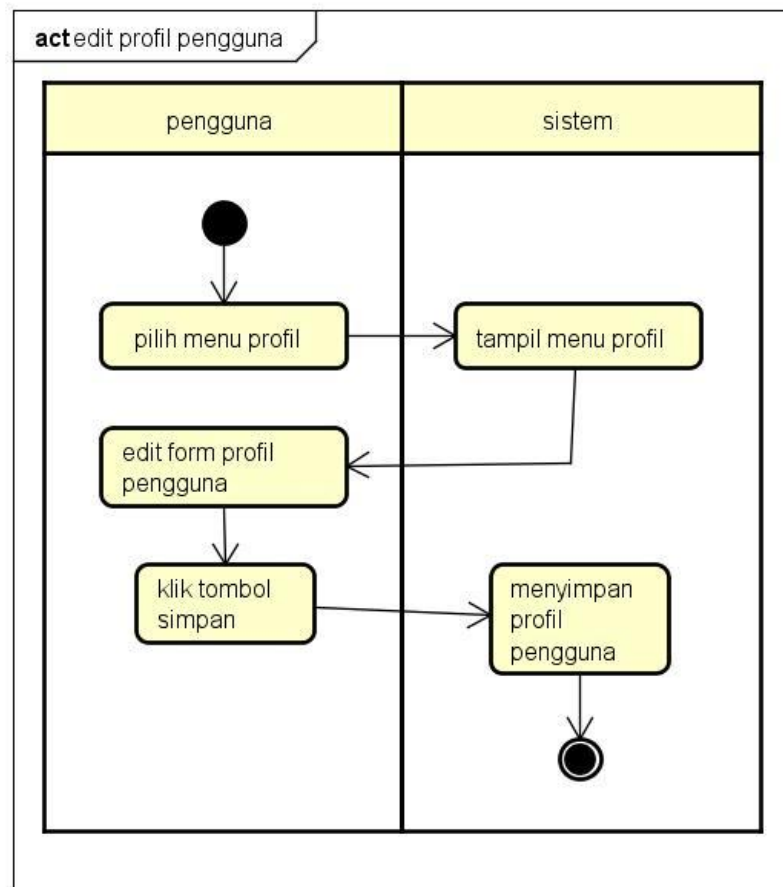
1.17. Activity diagram sub sistem pengguna



Gambar 4.18 Activity diagram sub sistem pengguna

Gambar 4.18 menjelaskan alur saat ketua akan mengaktifkan , meng-nonaktifkan dan mereset password pengguna. Dimulai dari ketua masuk ke menu pengguna, dan pilih salah satu tombol untuk mengaktifkan, menon aktifkan atau mereset password pengguna.

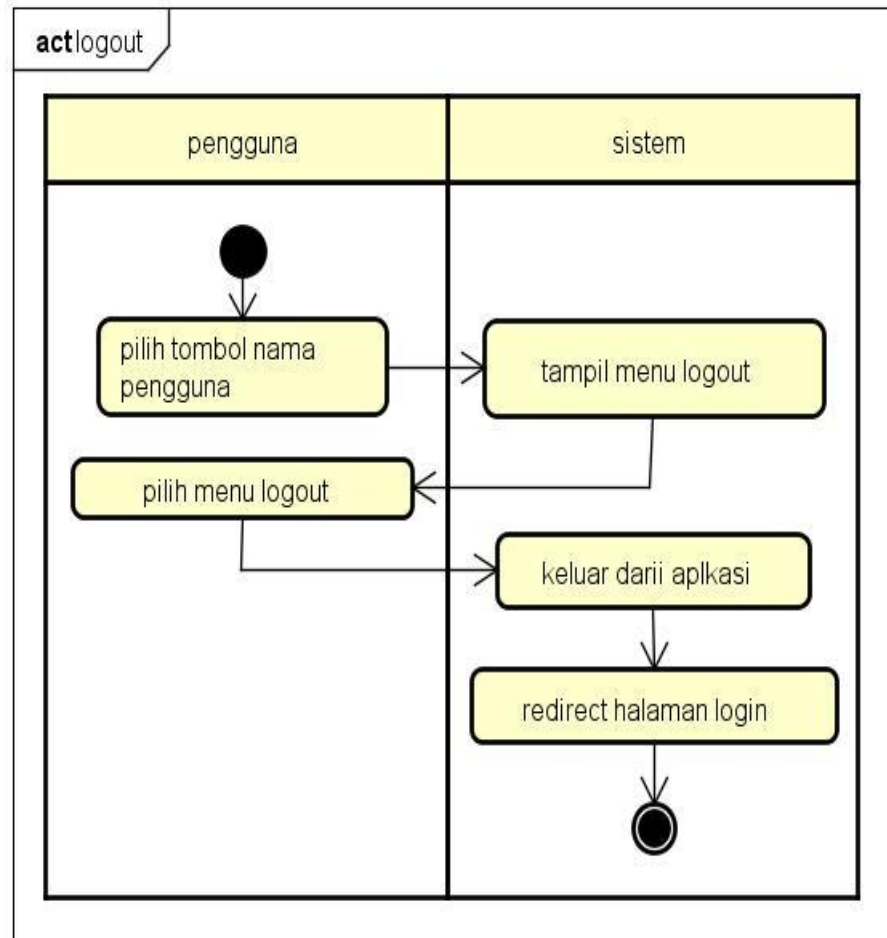
1.18. Activity diagram edit profil pengguna



Gambar 4.19 Activity diagram edit profil pengguna

Gambar 4.19 menjelaskan alur saat pengguna akan mengubah data profil pengguna. Dimulai dari pengguna memilih menu profil, kemudian mengisi data profil nama, email, password dan confirm password. Setelah selesai maka klik tombol simpan. Maka profil pengguna berhasil di simpan.

1.19. Activity diagram logout



Gambar 4.20 Activity diagram logout

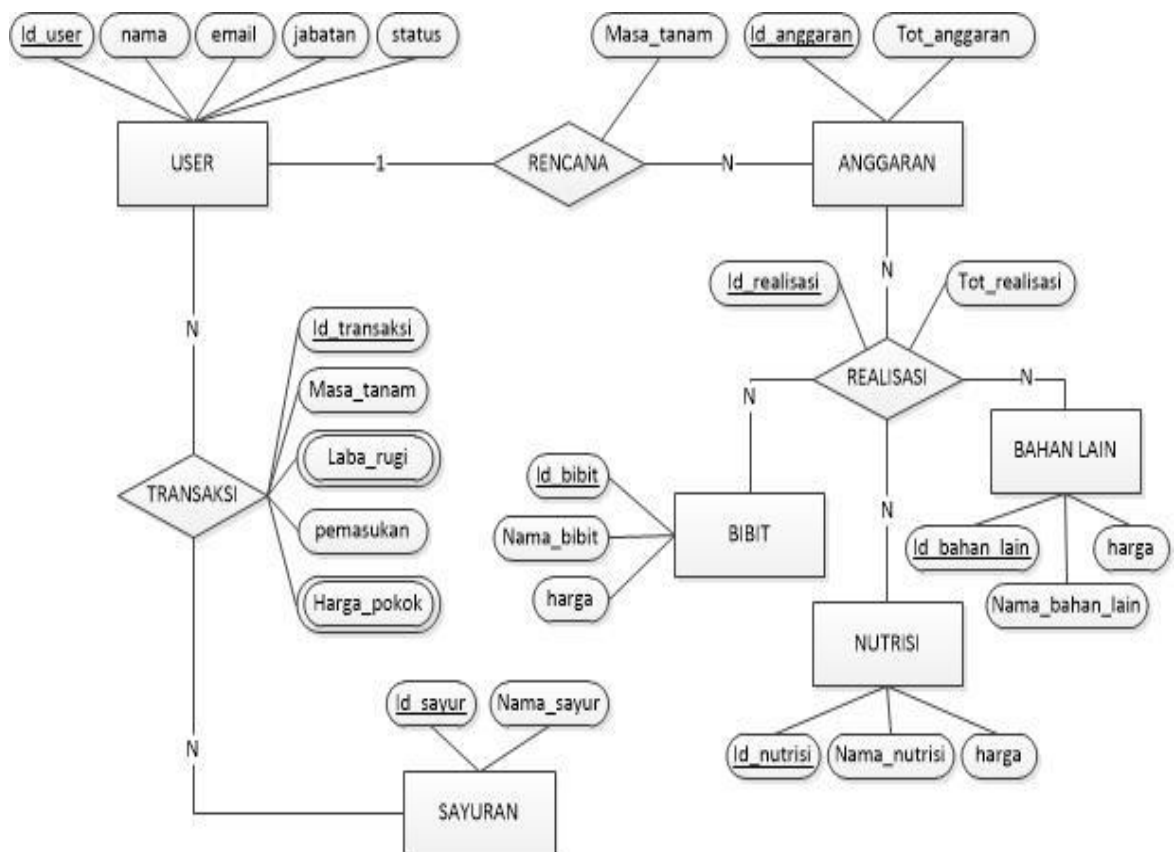
Gambar 4.20 menjelaskan alur saat pengguna ingin logout. Kondisi pengguna harus dalam keadaan login. Kemudian memilih menu nama pengguna yang sedang login, maka muncul menu *dropdown* untuk logout. Pengguna kemudian pilih tombol logout dan sistem akan keluar dari aplikasi dan me-*redirect* ke halaman login.

2. Desain Data

Desain data digunakan untuk menggambarkan pemodelan basis data yang ada pada aplikasi sistem informasi keuangan. *Entity Relationship Diagram (ERD)* digunakan dalam tahap ini untuk memodelkan struktur data yang ada dalam sistem informasi keuangan ini.

ERD dapat dilihat pada gambar 4.21.

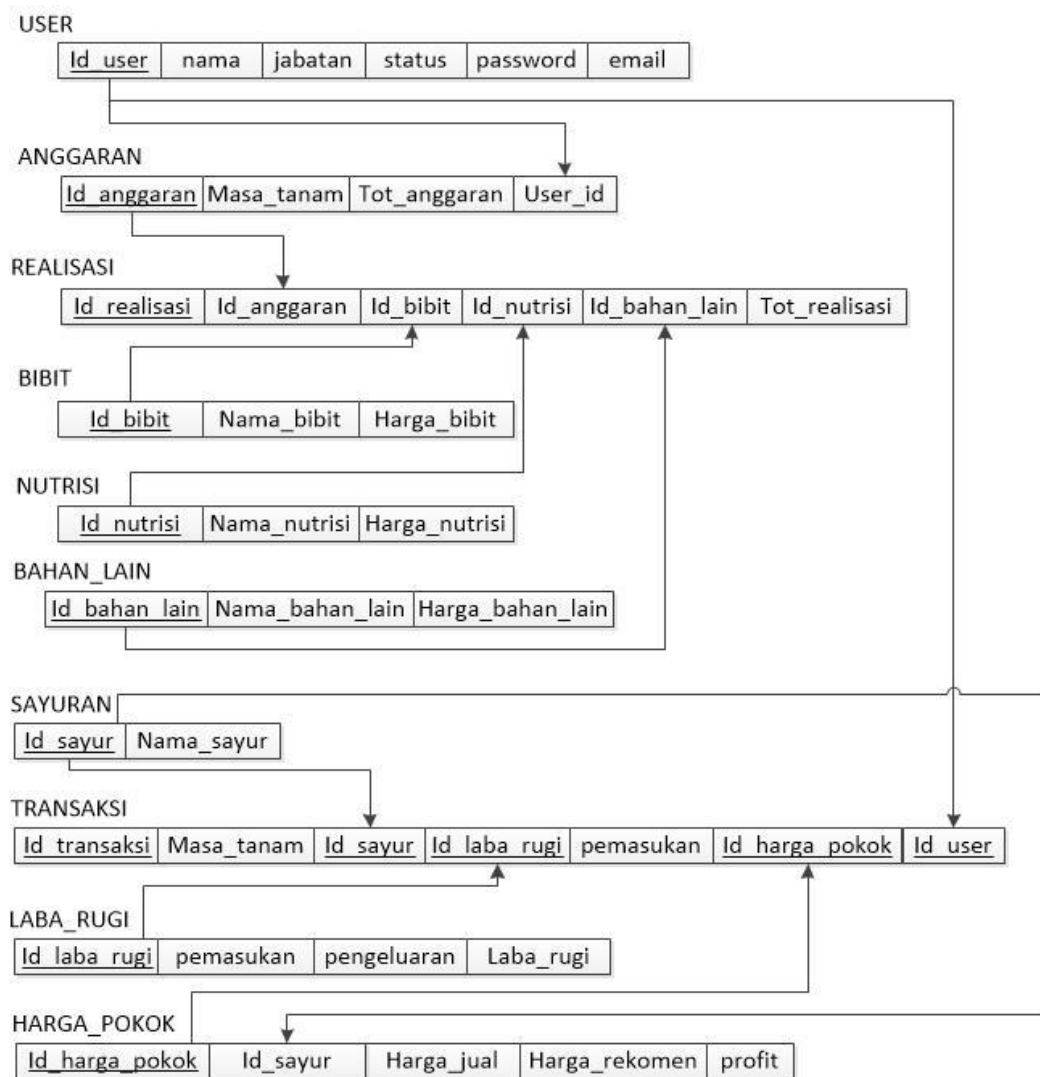
a. ERD (*Entity Relationship Diagram*)



Gambar 4.21. ERD (*Entity Relationship Diagram*)

b. Mapping Table

Setelah membuat ERD, tahap selanjutnya adalah memetakan entitas-entitas ataupun relasi antar tabelnya dalam bentuk *mapping table* seperti pada gambar 4.22. Berikut adalah mapping table berdasarkan ERD yang telah dibuat.



Gambar 4.22. Mapping Table

c. Kamus Data

Aplikasi sistem informasi keuangan ASRI 12 menggunakan beberapa tabel untuk menyimpan data dari aplikasi. Tabel yang digunakan adalah sebagai berikut :

1) tabel users

tabel 4.26 tabel users

kolom	Tipe data	keterangan
Id	Int(10)	<i>Primary key</i>
Name	Varchar(225)	
Email	Varchar(225)	
Jabatan	Enum ("ADMIN","ANGGOTA")	
Status	Enum ("AKTIF","NONAKTIF")	
Password	Varchar(225)	
Created_at	Date	Timestamp
Update_at	Date	timestamp

2) tabel anggarans

tabel 4.27 tabel anggarans

kolom	Tipe data	keterangan
Id	Int(10)	<i>Primary key</i>
Masa_tanam	Date()	
Sayur_id	Int(10)	Foreign key
bibit	Int(5)	
nutrisi	Int(5)	
Bahan_lain	Int(5)	
Tot_anggaran	Int (5)	
Created_at	Date	Timestamp
Update_at	Date	timestamp

3) tabel pemasukans

tabel 4.28 tabel pemasukans

kolom	Tipe data	keterangan
Id	Int(10)	<i>Primary key</i>
Masa_tanam	Date	
Sayur_id	Int(5)	<i>Foreign key</i>
pemasukan	Int(10)	
Jenis_pema	Enum("PENJUALAN, IURAN, SUMBANGAN")	
keterangan	Varchar(100)	
Created_at	Date	Timestamp
Update_at	Date	timestamp

4) Tabel pengeluaranans

Tabel 4.29 tabel pengeluaranans

kolom	Tipe data	keterangan
Id	Int(10)	<i>Primary key</i>
Sayur_id	Int(10)	<i>Foreign key</i>
Id_anggaran	Int(10)	<i>Foreign key</i>
Masa_tanam	Date()	
Real_bibit	Int(5)	
Real_nutrisi	Int(11)	
Real_bahan_lain	Int(11)	
Ket_nutrisi	Varchar(100)	
Ket_bibit	Varchar(100)	
Ket_bahan_lain	Varchar(100)	
Total_realisasi	Int (5)	
Created_at	Date	Timestamp
Update_at	Date	timestamp

5) Tabel transaksi

Tabel 4.30 tabel transaksi

kolom	Tipe data	keterangan
Id	Int(10)	<i>Primary key</i>
Tanggal_transaksi	Date	
Pemasukan_id	Int(10)	<i>Foreign key</i>
Pengeluaran_id	Int(10)	<i>Foreign key</i>
deskripsi	Varchar(100)	
pengeluaran	Int(11)	
pemasukan	Int(11)	
saldo	Int(11)	
Created_at	Date	Timestamp
Update_at	Date	timestamp

6) Tabel labarugis

Tabel 4.31 tabel labarugis

kolom	Tipe data	keterangan
Id	Int(10)	<i>Primary key</i>
periode	Date	
Pemasukan_id	Int(10)	<i>Foreign key</i>
realisasi_id	Int(10)	<i>Foreign key</i>
realisasi	Int(11)	
pemasukan	Int(11)	
labarugi	Int(11)	
Created_at	Date	Timestamp
Update_at	Date	timestamp

7) Tabel sayurans

Tabel 4.32 tabel sayurans

kolom	Tipe data	keterangan
Id	Int(10)	<i>Primary key</i>
Nama_sayur	Varchar(100)	
Created_at	Date	Timestamp
Update_at	Date	timestamp

8) Tabel bibits

Tabel 4.33 tabel bibits

kolom	Tipe data	keterangan
Id	Int(10)	<i>Primary key</i>
Nama_bibit	Varchar(100)	
Created_at	Date	Timestamp
Update_at	Date	timestamp

9) Tabel nutrisi

Tabel 4.34 tabel nutrisi

kolom	Tipe data	keterangan
Id	Int(10)	<i>Primary key</i>
Nama_nutrisi	Varchar(100)	
Created_at	Date	Timestamp
Update_at	Date	timestamp

10) Tabel bahan_lains

Tabel 4.35 tabel bahan_lains

kolom	Tipe data	keterangan
Id	Int(10)	<i>Primary key</i>
Nama_bahan_lain	Varchar(100)	
Created_at	Date	Timestamp
Update_at	Date	timestamp

11) Tabel harga_pokoks

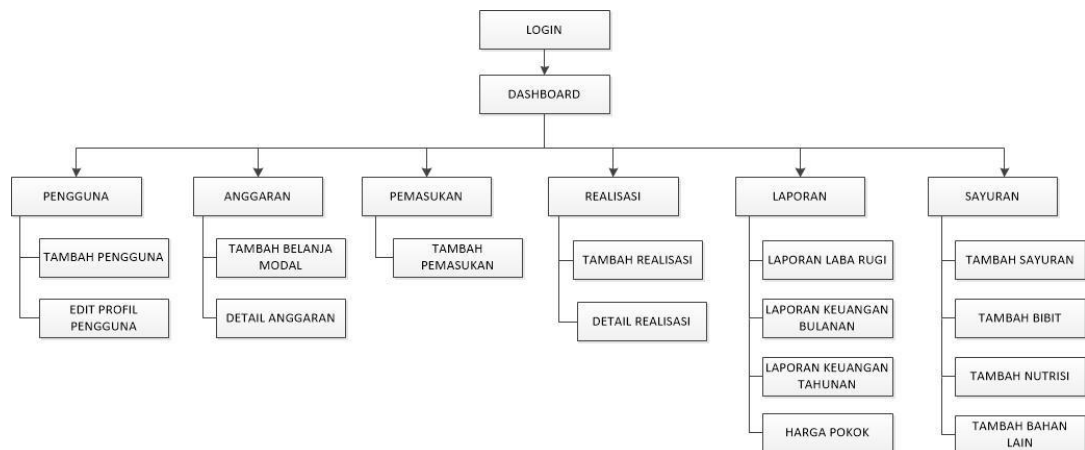
Tabel 4.36 tabel harga_pokoks

kolom	Tipe data	keterangan
Id	Int(10)	<i>Primary key</i>
Sayur_id	Int(10)	<i>Foreign key</i>
Nama_sayur	Varchar(100)	
Masa_tanam	Date()	
Pengeluaran_id	Int(10)	<i>Foreign key</i>
Total_panen	Int (10)	
profit	Int (10)	
Harga_rekomen	Int (10)	
Harga_pasar	Int (10)	
Harga_jual	Int (10)	
Created_at	Date	Timestamp
Update_at	Date	timestamp

3. Desain Antarmuka

Perancangan antarmuka bertujuan untuk memberikan gambaran aplikasi sistem informasi keuangan kepada pengguna . perancangan antarmuka meliputi struktur menu, input dan output aplikasi, dan fungsi-fungsi yang ada didalam aplikasi. Hasil dari desain antarmuka ini adalah prototype aplikasi sistem informasi keuangan.

a. Struktur menu



Gambar 4.23 struktur menu sistem informasi keuangan

b. Login

login

E-mail Address

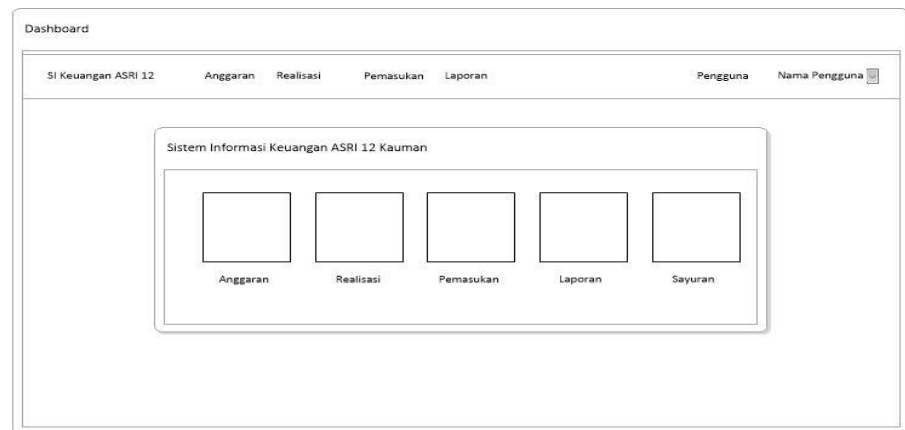
Password

☒ Remember me

Gambar 4.24 desain halaman login

Gambar 4.24 halaman login digunakan untuk menampilkan form email dan password yang diperlukan untuk masuk ke dalam aplikasi.

c. Dashboard



Gambar 4.25 desain halaman dashboard

Halaman *dashboard* pada gambar Gambar 4.25 berisi menu utama dari aplikasi ini. Menampilkan menu utama yang bisa digunakan pengguna seperti Anggaran, pemasukan , pengeluaran dan laporan. Setiap menu utama menggunakan ikon yang gambarnya sesuai dengan nama menu utamanya.

d. Pengguna



Gambar 4.26 desain halaman pengguna

Pada halaman pengguna pada gambar 4.26 menampilkan tabel daftar pengguna yang dapat menggunakan aplikasi. Terdapat tombol

tambah pengguna yang digunakan untuk mengarah ke halaman form tambah pengguna.

e. Tambah pengguna

The screenshot shows a web application interface for adding a new user. The title bar reads 'Tambah Pengguna'. Below it is a navigation menu with tabs: 'Anggaran', 'Pengeluaran', 'Pemasukan', 'Laporan', 'Pengguna', and 'Nama Pengguna'. The main form area contains the following fields:

- Name:** A text input field.
- E-mail Address:** A text input field.
- Password:** A text input field with a default value of 'Default "123456"'. The text is highlighted in grey.
- Jabatan:** A dropdown menu.

At the bottom of the form is a button labeled 'Tambah'.

Gambar 4.27 desain halaman tambah pengguna

Halaman tambah pengguna pada gambar 4.27 berisi form-form yang digunakan untuk menyimpan data pengguna, seperti nama, email, dan jabatan. Sedangkan password secara default . Untuk form jabatan menggunakan menu drop down. Kemudian terdapat tombol tambah yang digunakan untuk menambahkan pengguna.

f. Anggaran

anggaran

SI Keuangan ASRI 12 Anggaran Pengeluaran Pemasukan Laporan Pengguna Nama Pengguna ▾

Saldo saat ini : Jumlah Anggaran : mana tanam

Masa Tanam Cetak Anggaran :

2016 September 2016 September

No	Masa Tanam	Nama Sayur	Bibit	Nutrisi	Bahan Lain	Anggaran Total	Aksi
----	------------	------------	-------	---------	------------	----------------	------

Gambar 4.28 desain halaman Anggaran

Halaman Anggaran pada gambar 4.28 berisi form dan tombol yang mendukung informasi daftar Anggaran. Form saldo saat ini digunakan untuk mengetahui saldo yang ada pada saat itu, sedangkan form total anggaran adalah jumlah anggran pada masa tanam ini. Menu drop down untuk periode digunakan untuk memilih periode yang ingin di tampilkan dan dicetak. Tombol tambah Anggaran digunakan untuk mengarah ke halaman menambah data Anggaran.

g. Tambah Anggaran

Gambar 4.29 desain halaman tambah Anggaran

Gambar 4.29 digunakan untuk menambah data Anggaran, terdapat dropdown masa tanam, nama sayur, bibit, nutrisi, bahan lain dan keterangan.

h. Pemasukan

Gambar 4.30 desain halaman pemasukan

Halaman utama pemasukan pada gambar 4.30 berisi form dan tombol yang mendukung informasi pemasukan. Form saldo saat ini digunakan untuk mengetahui saldo yang ada pada saat itu. Menu drop down untuk masa tanam digunakan untuk memilih masa tanam yang ingin di tampilkan dan dicetak. Tombol tambah pemasukan digunakan untuk mengarah ke halaman menambah data pemasukan.

i. Tambah pemasukan

Gambar 4.31 desain halaman tambah pemasukan

Halaman tambah pemasukan pada gambar 4.31 digunakan untuk menambah data pemasukan yang terdiri dari form tanggal transaksi yang otomatis mengambil tanggal pada hari itu. pemasukan serta menu drop down pilihan jenis pemasukan

j. Realisasi

Gambar 4.32 desain halaman realisasi

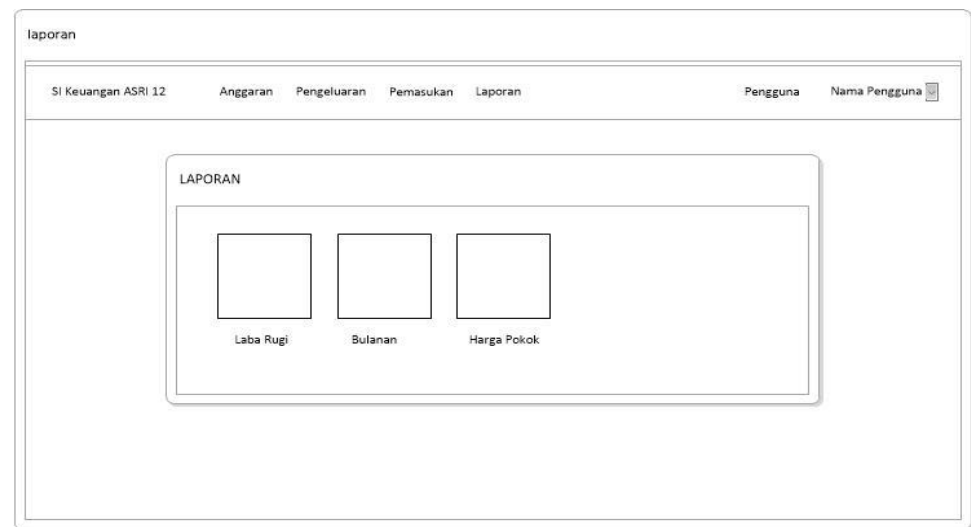
Halaman utama realisasi pada gambar 4.32 berisi form dan tombol yang mendukung informasi realisasi. Form saldo saat ini digunakan untuk mengetahui saldo yang ada pada saat itu. Menu drop down untuk masa tanam digunakan untuk memilih masa tanam yang ingin di tampilkan dan dicetak. Serta terdapat tabel yang menampilkan realisasi masa tanam saat itu atau masa tanam sesuai pilihan pengguna.

k. Tambah Realisasi

Gambar 4.33 desain halaman tambah realisasi

Halaman tambah realisasi pada gambar 4.33 digunakan untuk menambah data realisasi yang terdiri dari form masa tanam, realisasi bibit, nutrisi dan bahan lain dan keterangan. Serta menyediakan form anggaran agar pengguna dapat merealisasikan dana sesuai dengan anggaran.

I. Laporan



Gambar 4.34 desain halaman laporan

Tampilan utama halaman laporan pada gambar 4.34 terdapat 3 menu utama laporan yaitu laba rugi, bulanan, dan harga pokok. Masing-masing menu utama didukung ikon yang terkait dengan menuanya.

m. Laporan laba-rugi

Gambar 4.35 desain halaman laporan laba-rugi

Pada gambar 4.35 yaitu desain halaman laporan laba rugi berisi tombol dan form serta tabel yang mendukung informasi laporan laba rugi. Antara lain pilihan periode, tabel laba rugi dan form laba rugi pada periode itu.

n. Laporan keuangan bulanan

Gambar 4.36 desain halaman laporan keuangan bulanan

Pada gambar 4.36 yaitu halaman laporan bulanan berisi menu yang mendukung informasi laporan keuangan bulanan. Terdapat form saldo saat ini, menu drop down untuk memilih periode, dan tabel pemasukan dan pengeluaran periode saat itu atau sesuai pilihan pengguna.

o. Harga jual produk

Harga Jual Produk

SI Keuangan ASRI 12 Anggaran Realiasi Pemasukan Laporan Pengguna Nama Pengguna

HARGA JUAL PRODUK

Masa Tanam

tahun Bulan Tampilkan Cetak Harga

No	Masa Tanam	Nama Sayur	Total Modal	Total Panen	Profit	Rekomendasi Harga	Harga Pasar	Harga Jual	Aksi
----	------------	------------	-------------	-------------	--------	-------------------	-------------	------------	------

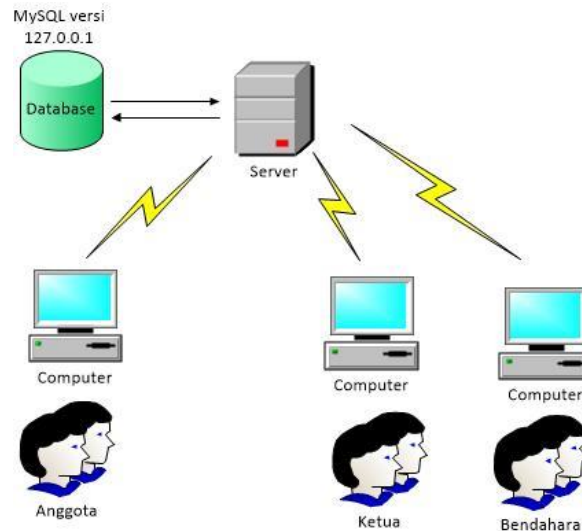
Gambar 4.37 desain halaman harga jual produk

Halaman harga jual produk pada gambar 4.37 digunakan untuk menampilkan harga penjualan produk yang akan dijual di pasaran.

Terdapat tabel yang mendukung perhitungan harga pokok.

4. Arsitektur Sistem

Arsitektur aplikasi sistem informasi keuangan ini memanfaatkan internet untuk menjalankan fungsinya seperti : mengakses data, menginput data serta melihat informasi hasil dari pengolahan sistem. Basis data yang disimpan di server digunakan untuk menyimpan seluruh data dan dapat diakses kapan saja selama terkoneksi internet. Aplikasi dapat diakses di alamat: <http://sikasri12kauman.com/>. Arsitektur sistem dapat dilihat pada gambar 4.38.



Gambar 4.38 arsitektur sistem aplikasi sistem informasi keuangan

D. Implementasi

Implementasi Aplikasi Sistem Informasi Keuangan ini dibangun menggunakan framework laravel 5.1 dengan bahasa pemograman HTML dan PHP yang berbasis web. Dengan beberapa bahasa pemograman yang menambah daya tarik tampilan web antara lain Bootstrap, Javascript. Data aplikasi ini diupload di server penyedia jasa hosting IDHostinger Indonesia. Dengan domain www.siaksri12kauman.com. Dan database yang dibuat menggunakan MySQL.

1. Model

Yaitu berisi *class php* yang menghubungkan *controller* dengan *database MySql*. Selain itu model juga dapat memproteksi atau mengizinkan *controller* dalam mengakses database, *query* data dari

database untuk disampaikan ke *controller*, serta mengatur integritas dan konsistensi data. Model yang terdapat dalam sistem yaitu :

a. Model user

```

b. <?php
c.     namespace App;
d.     use Illuminate\Foundation\Auth\User as Authenticatable;
e.     class User extends Authenticatable
f.     {
g.         protected $fillable = [
h.             'name', 'email', 'password', 'jabatan',
i.         ];
j.         protected $hidden = [
k.             'password', 'remember_token',
l.         ];
1.     }

```

Script 4.1: Model user.php

Model user berisi script yang mengizinkan aplikasi untuk mengakses database dan mengizinkan field apa saja yang bisa di akses.

b. Model anggaran

```

1. <?php
2.     namespace App;
3.     use Illuminate\Database\Eloquent\Model;
4.     use Carbon\Carbon;
5.     use DateTime;
6.     class anggaran extends Model {
7.         protected $fillable = ['nama_item_ang
8.         garan', 'qty', 'harga_satuan', 'anggaran', 'periode'];
9.         public function getCreatedAtAttribute()
10.        {
11.            return Carbon::parse($this->attributes['periode']);
12.        }
13.    }

```

Script 4.2 model Anggaran

Script 4.2 berfungsi menyambungkan aplikasi dengan database anggaran yang berisi data Anggaran. Serta mengizinkan aplikasi untuk menyimpan data ke database.

c. Model Realisasi

```

1. <?php
2.
3. namespace App;
4.
5. use Illuminate\Database\Eloquent\Model;
6. use App\transaksi;
7. use App\labarugi;
8. class pengeluaran extends Model
9. {
10.     protected $fillable = [ 'masa_tanam', 'id_sayur', 'id_anggaran'
11.     'ang_bibit', 'ang_nutrisi', 'ang_bahan_lain', 'anggaran_total', 'real_bibit'
12.     , 'real_nutrisi', 'real_bahan_lain', 'ket_real_bibit', 'ket_real_nutrisi', 'r
13.     eal_bahan_lain', 'total_realisasi'];
14. protected $dates = ['created_at', 'updated_at', 'masa_tanam',];
15. }

```

Script 4.3. model pengeluaran

Script 4.3 model dari pengeluaran yang berfungsi mengizinkan aplikasi dalam mengakses database. Dan mengizinkan aplikasi mengisi data di tabel pengeluaran. Dan mendeklarasikan fungsi yang menyatakan relasi dari tabel pengeluaran.

d. Model pemasukan

```

1. <?php
2.
3. namespace App;
4.
5. use Illuminate\Database\Eloquent\Model;
6. use App\transaksi;
7. use App\labarugi;
8.
9. class pemasukan extends Model {
10.     protected $fillable = [ 'tanggal_transaksi',
11.     'deskripsi',
12.     'pemasukan',
13.     'jenis_pema',
14.     'id'];
15.     public function transaksi() {
16.         return $this->hasOne('App\transaksi');
17.     }
18.     public function labarugi()
19.     {
20.         return $this->hasOne('App\labarugi');
21.     }}

```

Script 4.4 model pemasukan.

Script 4.4 model dari pemasukan yang berfungsi mengizinkan aplikasi dalam mengakses database. Dan mengizinkan aplikasi mengisi data di tabel pemasukan. Dan mendeklarasikan fungsi yang menyatakan relasi dari tabel pemasukan

e. Model laba rugi

```

1. class labarugi extends Model
2. {
3.     protected $fillable = ['tanggal_transaksi', 'deskripsi', 'pengeluaran_id', 'pemasukan_id', 'pemasukan', 'pengeluaran', 'labarugi'];
4.     public function pemasukan()
5.     {
6.         return $this->belongsTo('App\pemasukan');
7.     }
8.     public function pengeluaran()
9.     {
10.        return $this->belongsTo('App\pengeluaran');
11.    }
12. }

```

Script 4.5 model dari laporan laba-rugi

Script 4.5 berisi listing untuk mengizinkan aplikasi dalam mengakses database. Dan mendeklarasikan sebuah relasi terkait dengan tabel laba-rugi.

f. Model transaksi

```

1. <?php
2. namespace App;
3. use Illuminate\Database\Eloquent\Model;
4. use App\pemasukan;
5. use App\pengeluaran;
6. class transaksi extends Model
7. { protected $fillable = ['tgl_transaksi', 'pengeluaran_id', 'pemasukan_id', 'pemasukan', 'pengeluaran', 'saldo', 'deskripsi',];
8.     public function pemasukan()
9.     {
10.        return $this->belongsTo('App\pemasukan');
11.    }
12.     public function pengeluaran()
13.     {
14.        return $this->belongsTo('App\pengeluaran');

```

```

15. }
16. protected $dates = [
17. 'created_at',
18. 'updated_at',
19. 'tgl_transaksi',
20. ];}

```

Script 4.6 model dari transaksi

Script 4.6 berisi listing untuk mengizinkan aplikasi dalam mengakses database. Dan mendeklarasikan sebuah relasi terkait dengan tabel transaksi.

g. Model harga pokok

```

h. <?php
i. namespace App;
j. use Illuminate\Database\Eloquent\Model;
k. class harga_pokok extends Model
l. { protected $fillable = [ 'masa_tanam',
m.                             'sayur_id',
n.                             'nama_sayur',
o.                             'pengeluaran_id',
p.                             'pengeluaran',
q.                             'total_panen',
r.                             'profit',
s.                             'harga_rekomen',
t.                             'harga_jual',
u.                             'harga_pasar',
v.                             ];
w. protected $dates = [
x.     'created_at',
y.     'updated_at',
z.     'masa_tanam',
aa.    ];
bb. }

```

Script 4.7 model dari harga pokok

Script 4.7 berisi listing untuk mengizinkan aplikasi dalam mengakses database. Dan mendeklarasikan sebuah relasi terkait dengan tabel harga pokok.

2. Controller

Yaitu kode yang bertugas menerima data, validasi data, membuat data baru dan memperbaharui data. Selain itu juga dapat memodifikasi kueri *database*. Semua kode disusun di masing- masing *function* sesuai dengan yang diminta oleh *view*. Di *controller* juga berisi kondisi yang digunakan untuk menyimpan setiap data yang masuk. Adapun controller yang mendukung berjalannya fungsi utama sistem yaitu :

- a. Controller tambah, ubah, non-aktifkan dan *reset password* pengguna.

```

1. <?php
2. namespace App\Http\Controllers;
3. use Cartalyst\Alerts\Native\Facades\Alert;
4. use Illuminate\Http\Request;
5. use App\pengguna;
6. use App\User;
7. use Validator;
8. use App\Http\Requests;
9. use App\Http\Requests\simpanProfil;
10. class PenggunaController extends Controller
11. {
12.     public function index()
13.     {
14.         $pengguna = User::all();
15.         $data['pengguna'] = $pengguna;
16.         return view('pengguna.index',$data);
17.     }
18.     public function create()
19.     {
20.         return view('pengguna.create');
21.     }
22.     public function store(request $request)
23.     {
24.         $name = $request['name'];
25.         $email = $request['email'];
26.         $password = bcrypt('123456');
27.         $jabatan = $request['jabatan'];
28.         User::create(array('name'=> $name,'email'=>$email,'password
29.         '=>$password,'jabatan'=>$jabatan));
30.         return redirect('pengguna');
31.     }
32.     public function edit($id)
33.     {
34.         $pengguna = User::find($id);
35.         $identitas = Auth()->user()->id;
36.         if ($identitas == $id) {
37.             \Flash::error('Maaf anda tidak dapat menonaktifkan anda send
38.             iri');
39.         }
40.         return redirect('pengguna'); }

```


```

37.         elseif($pengguna->status == 'AKTIF')
38.         {
39.             $pengguna->status = 'NONAKTIF';
40.             $pengguna->save();
41.         }
42.         else
43.         {
44.             $pengguna->status = 'AKTIF';
45.             $pengguna->save();
46.         }
47.
48.         return redirect('pengguna');
49.     }
50.     public function reset($id)
51.     {
52.         $pengguna = User::find($id);
53.         $pengguna->password = bcrypt('123456');
54.         $pengguna->save();
55.         \Flash::success('password '.$pengguna-
>name.' telah direset menjadi: 123456 ');
56.         return redirect('pengguna');
57.     }
58.     public function profil($id)
59.     {
60.         $data['pengguna'] = user::find($id);
61.         $data['nama'] = null;
62.         $data['email'] = null;
63.         $data['jabatan'] = null;
64.         return view('pengguna.profil',$data);
65.     }
66.     public function simpan(simpanProfil $request, $id)
67.     {
68.         $pengguna = User::find($id);
69.         $password = $request['password'];
70.         $pengguna->name = $request['name'];
71.         $pengguna->email = $request['email'];
72.         $pengguna->password = bcrypt($password);
73.         $pengguna->save();
74.         \Flash::success('profil '.$pengguna-
>name.' telah berhasil disimpan');
75.         $data['pengguna'] = user::find($id);
76.         $data['nama'] = null;
77.         $data['email'] = null;
78.         $data['jabatan'] = null;
79.         return view('pengguna.profil',$data);
80.     }
81.     public function destroy($id)
82.     {
83.         $identitas = Auth()->user()->id;
84.         $pengguna = User::find($id);
85.         if ($identitas == $id) {
86.             \Flash::error('Maaf anda tidak dapat menghapus anda se
ndiri');
87.             return redirect('pengguna');
88.         }
89.         else{
90.             $pengguna->delete();
91.         }
92.         return redirect('pengguna');
93.     }}

```

Script 4.8: Controller dari penggunaController.php

Controller pengguna berisi function untuk tambah, ubah, non-aktifkan dan *reset password* pengguna.

 PENGGUNA

[Tambah Pengguna](#)

No	Nama	Email	Jabatan	Status	Aksi	
1	ubay	resty.ubay@gmail.com	KETUA	AKTIF	NON-AKTIFKAN	RESET PASSWORD
2	sumarni	sumarni@gmail.com	KETUA	AKTIF	NON-AKTIFKAN	RESET PASSWORD
3	admin2	admin2@gmail.com	KETUA	NONAKTIF	AKTIFKAN	RESET PASSWORD
4	admin3	admin3@gmail.com	ANGGOTA	AKTIF	NON-AKTIFKAN	RESET PASSWORD
5	bendahara	bendahara@gmail.com	BENDAHARA	AKTIF	NON-AKTIFKAN	RESET PASSWORD

Gambar 4.39 View dari pengguna

Gambar 4.39 adalah halaman pengguna digunakan untuk menambah, menonaktifkan dan mereset password pengguna. Tombol Non-aktifkan akan berubah berwarna hijau dan bertuliskan aktifkan ketika ketua menekan tombol non-aktifkan.

b. Controller tambah anggaran beserta script kalkulasi otomatis anggaran

```

1. public function create()
2. {
3.     $data['y']      = date('Y');
4.     $data['m']      = date('m');
5.     $sayuran        = sayuran::lists('nama_sayur','id');
6.     $bibit           = bibit::lists('nama_bibit','id');
7.     $nutrisi         = Nutrisi::lists('nama_nutrisi','id');
8.     $bahan_lain     = bahan_lain::lists('nama_bahan_lain','id');
9.     $data['sayuran'] = $sayuran;
10.    $data['listbibit'] = $bibit;
11.    $data['listnutrisi'] = $nutrisi;
12.    $data['listbahan'] = $bahan_lain;
13.    return view('anggaran.create',$data);
14. }

```

Script 4.9: Controller tambah anggaran

Script controller tambah anggaran digunakan untuk mengambil data dari database.

```

1. <script type="text/javascript" >
2.     $(document).ready(function() {
3.         //this calculates values automatically
4.         sum();
5.         $("#bibit, #nutrisi, #bahan_lain").on("keydown keyup", function() {
6.             sum();
7.         });
8.     });
9.     function sum() {
10.        var num1 = document.getElementById('bibit').value;
11.        var num2 = document.getElementById('nutrisi').value;
12.        var num3 = document.getElementById('bahan_lain').value;
13.        var result = parseInt(num1) + parseInt(num2) + parseInt(num3);
14.        if (!isNaN(result)) {
15.            document.getElementById('tot_anggaran').value = result;
16.        }
17.    }
18. </script>

```

Script 4.10: *javascript auto-calculate*

Script 4.10 digunakan untuk menjumlahkan anggaran bibit, nutrisi, dan bahan lain secara langsung tanpa *reload* halaman.

+ TAMBAH ANGGARAN

Anggaran Masa Tanam:	2016 ▼	Desember - F ▼		
Nama Sayur	Kangkung ▼			
Pilih Bibit	0	Keterangan		
Pilih Nutrisi	0	Keterangan		
Pilih Bahan Lain	0	Keterangan		
Total Anggaran				
<input type="button" value="Simpan Data"/> <input type="button" value="Kembali"/>				

Gambar 4.40 *view tampilan tambah anggaran*

Gambar 4.40 halaman tambah anggaran menampilkan form yang digunakan untuk menambah anggaran. Untuk form total anggaran otomatis terkalkulasi menggunakan javascript.

- c. Controller tambah realisasi beserta kondisi jika realiasi lebih besar dari anggaran.

```

1. public function update($id, createPengeluaran $request)
2. {
3. $data = $request->all();
4. $pengeluaran = pengeluaran::find($id);
5. $saldo = transaksi::latest()->first();
6. if ($saldo)
7. {
8. $saldo = $saldo->saldo - $data['total_realisasi'];
9. if($saldo>0)
10. {
11. $pengeluaran->update($data);
12. harga_pokok::create(array( 'masa_tanam' =>$data['masa_tanam'],
13.                             'sayur_id'    =>$data['sayur_id'],
14.                             'nama_sayur'  =>$data['nama_sayur'],
15.                             'pengeluaran_id'=>$pengeluaran->id,
16.                             'pengeluaran' =>$data['total_realisasi']
17. ,));
18. pemasukan::create(array( 'masa_tanam' =>$data['masa_tanam'],
19.                           'id_sayur'   =>$data['sayur_id'],
20.                           'nama_sayur' =>$data['nama_sayur'],));
21. transaksi::create(array( 'tgl_transaksi' =>$data['masa_tanam'],
22.                           'deskripsi'    =>'Belanja modal sayur '.$
23. data['nama_sayur'],
24.                           'pengeluaran_id'=>$pengeluaran->id,
25.                           'pengeluaran'  =>$data['total_realisasi']
26. ,
27.                           'saldo'       =>$saldo, ));
28. labarugi::create(array( 'periode'    =>$data['masa_tanam'],
29.                           'deskripsi'  =>'Belanja modal sayur '.$
30. data['nama_sayur'],
31.                           'realisasi_id' =>$pengeluaran->id,
32.                           'pengeluaran' =>$data['total_realisasi']
33. ,));
34. }
35. else
36. {
37. \Flash::error('Maaf saldo tidak cukup untuk realisasi');
38. return redirect('pengeluaran');
39. } }
40. else
41. {
42. \Flash::error('Maaf saldo tidak cukup untuk realisasi');
43. return redirect('pengeluaran');
44. }return redirect('pengeluaran'); }

```

Script 4.11: Controller tambah realiasi

Script 4.11 controller tambah realisasi digunakan untuk menyimpan data realisasi ke tabel pengeluaran dan secara otomatis masuk ke tabel laba rugi.

```

1. <script type="text/javascript">
2. $("#bibit, #nutrisi, #bahan_lain").keyup(function(){
3.     var bibit = $("#bibit").val();
4.     var ang_bibit = $("#ang_bibit").val();
5.     var nutrisi = $("#nutrisi").val();
6.     var ang_nutrisi = $("#ang_nutrisi").val();
7.     var bahan_lain = $("#bahan_lain").val();
8.     var ang_bahan_lain = $("#ang_bahan_lain").val();
9.
10.    if ( parseInt(bibit) > parseInt(ang_bibit) )
11.    {
12.        alert("Maaf, Realisasi tidak boleh lebih dari anggaran")
13.    };
14.    $("#bibit").val("");
15.    else if (parseInt(nutrisi) > parseInt(ang_nutrisi))
16.    {
17.        alert("Maaf, Realisasi tidak boleh lebih dari anggaran"
18.    );
19.    $("#nutrisi").val("");
20.    else if (parseInt(bahan_lain) > parseInt(ang_bahan_lain))
21.    {
22.        alert("Maaf, Realisasi tidak boleh lebih dari anggaran"
23.    );
24.    $("#bahan_lain").val("");
25.    }
26.    else{    }
27.    });
28. </script>

```

Script 4.12 : *javascript* kondisi membandingkan anggaran dan realisasi

Script 4.12 digunakan untuk membandingkan apakah realisasi melebihi anggaran. Jika kondisi terpenuhi, maka akan muncul pesan *alert* bahwa realisasi tidak boleh lebih dari anggaran.

DETAIL REALISASI

Realisasi Masa Tanam:	December - February				
Nama Sayur	Selada				
Anggaran Bibit	7000	Realiasi Bibit	0	Keterangan	
Anggaran Nutrisi	7000	Realiasi Nutrisi	0	Keterangan	
Anggaran Bahan Lain	20000	Realiasi Bahan Lain	0	Keterangan	
Total Anggaran	34000	Total Realiasi	0		
<div> <input type="button" value="simpan data"/> <input type="button" value="Kembali"/> </div>					

Gambar 4.41 view halaman tambah realisasi

Gambar 4.41 adalah gambar tampilan halaman untuk menambah realisasi. Form total realisasi akan dihitung secara otomatis menggunakan *javascript*.

d. Controller tambah pemasukan

```

e. public function update($id, createPemasukan $request)
f. {
g. $data = $request->all();
h. $pemasukan = pemasukan::find($id);
i. $saldo = transaksi::latest()->first();
j. $saldo = $saldo->saldo + $data['pemasukan'];
k. transaksi::create(array(
l.     'tgl_transaksi' =>$data['masa_tanam'],
     'deskripsi'     =>'Hasil penjualan sayur '.$
     data['nama_sayur'],
m.     'pemasukan_id' =>$pemasukan->id,
n.     'pemasukan'    =>$data['pemasukan'],
o.     'saldo'        =>$saldo,));
p. labarugi::create(array(
q.     'periode'      =>$data['masa_tanam'],
     'deskripsi'      =>'Hasil penjualan sayur '
     .$data['nama_sayur'],
r.     'pemasukan_id' =>$pemasukan->id,
s.     'pemasukan'    =>$data['pemasukan'], ));
t.     $pemasukan->update($data);
u.     return redirect('pemasukan');
v. }

```

Script 4.13: Controller tambah pemasukan

Script 4.13 controller tambah pemasukan digunakan untuk menyimpan data pemasukan ke tabel pengeluaran dan secara otomatis masuk ke tabel laba rugi dan transaksi.

+ TAMBAH PEMASUKAN

Pemasukan Masa Tanam:	December - February		
Nama Sayur	Kangkung		
Pemasukan	0	Keterangan	
<input type="button" value="simpan data"/> <input type="button" value="Kembali"/>			

Gambar 4.42 view halaman tambah pemasukan

Gambar 4.42 adalah tampilan halaman tambah pemasukan yaitu pemasukan hasil penjualan. Hanya menampilkan periode masa tanam, nama sayuran dan jumlah pemasukan serta keterangan.

e. Controller laporan laba-rugi dan keuangan bulanan.

```

1. public function labarugi()
2. {
3.     $data['y']          = date('Y');
4.     $data['m']          = date('m');
5.     $labarugi           = labarugi::whereYear('periode', '=', date('Y'
6.     ))
7.     ->whereMonth('periode', '=', date('m'))
8.     ->paginate(40);
9.     $pema               = labarugi::whereYear('periode', '=', date('Y'
10.    ))
11.    ->whereMonth('periode', '=', date('m'))
12.    ->sum('pemasukan');
13.    $penge               = labarugi::whereYear('periode', '=', date('Y'
14.    ))
15.    ->whereMonth('periode', '=', date('m'))
16.    ->sum('pengeluaran');
17.    $data['saldo']        = transaksi::latest()->first();
18.    $saldomt              = labarugi::latest()
19.    ->whereMonth('periode', '=', date('m'))
20.    ->whereYear('periode', '=', date('Y'))
21.    ->first();
22.    $data['saldomt']      = $saldomt;
23.    $data['labarugi']     = $pema - $penge;
24.    $data['laba_rugi']    = $labarugi;
25.    return view('laporan.labarugi',$data); }

```

script 4.14 controller laporan laba-rugi

Script 4.14 adalah script controller untuk mengirimkan data yang diperlukan laporan labarugi dari *database* ke *view*. Didalam controller labarugi juga terdapat rumus menentukan laba-rugi pada yaitu pada baris 8 untuk menjumlahkan pemasukan, baris 11 untuk menjumlahkan pengeluaran, baris 20 adalah rumus laba rugi yaitu pemasukan dikurangi pengeluaran.

 LAPORAN LABA-RUGI

Saldo saat ini: 552,253 Masa Tanam: 2016 Desember - Februari Tampilkan Laba - Rugi :December 450,177 Cetak Laba-Rugi Cetak PDF

No	Masa Tanam	Deskripsi	Pemasukan	Pengeluaran	Aksi
1	December - February	Belanja modal sayur Kangkung	0	4,000	Detail
2	December - February	Hasil penjualan sayur Selada	50,000	0	
3	December - February	Hasil penjualan sayur Kangkung	50,000	0	
4	December - February	Hasil penjualan sayur Kangkung	1,000	0	
5	December - February	Belanja modal sayur Kangkung	0	147,823	Detail

Gambar 4.43 *view* halaman laporan laba-rugi

Gambar 4.43 adalah tampilan halaman laporan laba-rugi. Untuk menginformasikan apakah pada bulan tertentu kelompok petani mengalami untung atau rugi. Terdapat kolom pemasukan dan pengeluaran yang hanya berkategori penjualan. Terdapat tombol detail untuk melihat detail pemasukan.

f. Controller harga jual produk.

```

1. class hargaController extends Controller
2. {
3.     public function index()
4.     {
5.         $harga = harga_pokok::all();
6.         $data['harga'] = $harga;
7.         $data['y'] = date('Y');
8.         $data['m'] = date('m');
9.         return view('hargapokok.index',$data);

```

```

10. }
11. public function edit($id)
12. {
13.     $data['harga'] = harga_pokok::find($id);
14.     return view('hargapokok.edit',$data);
15. }
16. public function update(Request $request, $id)
17. {
18.     $data = $request->all();
19.     $harga = harga_pokok::find($id);
20.     $harga->update($data);
21.     return redirect('harga');
22. }
23. }

```

script 4.15 *controller* harga jual produk

Script 4.15 adalah script untuk menampilkan dan mengubah harga jual produk sayuran hidroponik.

```

1.     <script type="text/javascript">
2.     $("#total_panen, #profit, #harga_rekomen").keyup(function(){
3.         var num1 = $("#total_modal").val();
4.         var num2 = $("#total_panen").val();
5.         var num3 = $("#profit").val();
6.         var sat = (parseInt(num1) / parseInt(num2));
7.         var pro = parseInt(sat) * (parseInt(num3)/100);
8.         var result = sat + pro;
9.         if (!isNaN(parseInt(result))) {
10.             $("#harga_rekomen").val(result);
11.         }
12.     });
13. </script>

```

script 4.16 *script* hitung harga jual produk

Script 4.16 adalah *javascript* untuk menghitung harga jual produk sayuran hidroponik berdasarkan data pengeluaran modal yang telah dikeluarkan, hasil panen, dan profit yang diinginkan menggunakan presentase. Hasil perhitungan secara langsung ditampilkan di form harga rekomendasi tanpa *reload* halaman.

HARGA JUAL PRODUK

Masa Tanam:

2016 ▼

Desember - Februari ▼

Tampilkan

Cetak Harga:

Cetak PDF

No	Masa Tanam	Nama Sayur	Total Modal	Total Panen	Profit	Rekomendasi Harga	Harga Pasar	Harga Jual	Aksi
1	December - February	Kangkung	35,063	55	0	638	9,000	8,800	Edit
2	December - February	Kangkung	836	50	20	20	600	400	Edit
3	December - February	Selada	15,000	20	20	900	5,000	4,500	Edit
4	December - February	Kangkung	30	0	0	0	0	0	Edit

Gambar 4.44 view harga jual produk

Gambar 4.44 adalah tampilan halaman harga jual produk.

Menampilkan harga tiap sayuran berdasarkan total modal, total panen, profit, rekomendasi harga, harga pasar dan harga yang akan dijual dipasaran.

D. Pengujian Sistem

Setelah sistem selesai dibangun, pengguna diminta untuk melakukan test terhadap aplikasi. Dilakukan 2 test yaitu *Black Box test* dan *Aplha Test*.

1. *Black Box Test*

Pengujian dilakukan oleh sdr. D. Harsya Widyatama, SE yaitu mahasiswa Magister Sains Akuntansi FEB UGM. beliau sebagai pakar dibidang akuntansi serta cukup berpengalaman di dunia *software enggining*. Pengujian ini diberikan 9 pertanyaan dengan 2 alternatif jawaban yaitu ya atau tidak. Daftar pada pertanyaan pada pengujian ini dapat dilihat pada tabel 4.28.

Tabel 4.37 Hasil Black Box Test

no	pertanyaan	Pilihan jawaban	
		ya	tidak
1	Apakah sistem dapat menampilkan form login	✓	
2	Apakah aplikasi mampu login dan mengubah profil pengguna	✓	
3	Apakah aplikasi mampu membuat, menonaktifkan data penggunaan dan me-reset password pengguna yang hanya bisa di akses oleh ketua.	✓	
4	Apakah aplikasi mampu membuat, menampilkan, dan mencetak data perencanaan anggaran keuangan yang bisa di akses Ketua dan Bendahara.	✓	
5	Apakah aplikasi mampu membuat, menampilkan, dan mencetak data pemasukan keuangan yang bisa di akses Ketua dan Bendahara.	✓	
6	Apakah aplikasi mampu membuat, menampilkan, dan mencetak data pengeluaran keuangan yang bisa di akses Ketua dan Bendahara.	✓	
7	Apakah aplikasi mampu menampilkan dan mencetak laporan laba rugi yang bisa di akses Ketua, Bendahara dan anggota..	✓	
8	Apakah aplikasi mampu menampilkan laporan keuangan bulanan dan tahunan yang bisa di akses Ketua, Bendahara dan anggota.	✓	
9	Apakah aplikasi mampu menampilkan harga jual produk yang bisa di akses Ketua, Bendahara dan anggota..	✓	
Jumlah		9	

Dari hasil pengujian, diperoleh presentase penilaian jawaban Ya = $9/9 \times 100\% = 100\%$ dan jawaban Tidak = $0/9 \times 100\% = 0\%$. Dari hasil Pengujian tersebut dapat disimpulkan bahwa aplikasi berjalan dengan baik dan proses-proses yang terdapat dalam sistem aplikasi sesuai dengan kebutuhan.

2. Alpha Test

Pengujian dengan *alpha test* dilakukan dengan meminta ketua kelompok petani hidroponik untuk mencoba aplikasi. kemudian pengguna diberikan daftar pertanyaan terkait program yang dijalankan. Pada pengujian ini disertakan 3 pertanyaan dan pilihan jawabannya menggunakan skala likert dimana terdapat 4 kriteria yaitu SS = Sangat Setuju, S = Setuju, TS = Tidak Setuju, STS = Sangat Tidak Setuju. Hasil pengujian dapat dilihat pada tabel 4.39.

Tabel 4.39 hasil pengujian *alpha test*

No	Pertanyaan	Jawaban			
		SS	S	TS	STS
1	Aplikasi sistem informasi keuangan dapat memberikan informasi kepada anggota kelompok petani dalam memantau keadaan keuangan kelompok	√			
2	Aplikasi sistem informasi keuangan dapat memberikan informasi perhitungan harga jual produk sebagai acuan dalam menentukan harga sayur hidroponik	√			
3	Aplikasi sistem informasi keuangan dapat memberikan informasi laporan laba rugi hasil penjualan sayuran hidroponik.	√			

Berdasarkan hasil pengujian diperoleh presentase penilaian sebagai berikut :

STS (Sangat Tidak setuju)	= 0 %
TS (KurangSetuju)	= 0 %
S (Setuju)	= 0 %
SS (Sangat Setuju)	= 100 %

Dari hasil penilaian tersebut dapat disimpulkan bahwa aplikasi sistem informasi keuangan kelompok petani hidroponik dapat mempermudah anggota untuk melakukan pengelolaan keuangan di kelompok petani hidroponik ASRI 12 Kauman.

Bab V

Kesimpulan dan saran

A. Kesimpulan

Bedasarkan hasil penelitian dan pembahasan, didapat beberapa kesimpulan sebagai berikut :

1. Telah dibangun aplikasi sistem informasi keuangan berbasis web menggunakan *framework* laravel untuk kelompok petani hidroponik ASRI 12 Kauman.
2. Aplikasi sistem informasi keuangan dapat memberikan informasi kepada anggota kelompok petani dalam memantau keadaan keuangan kelompok.
3. Aplikasi sistem informasi keuangan dapat memberikan informasi perhitungan harga jual produk sebagai acuan dalam menentukan harga sayur hidroponik.
4. Aplikasi sistem informasi keuangan mampu memberikan informasi laporan laba-rugi hasil penjualan sayuran hidroponik.
5. Hasil pengujian *alpha test* menunjukkan presentase 100% jawaban ya bahwa setiap proses yang ada di dalam aplikasi berjalan dengan baik.

B. Saran

Aplikasi yang dibangun tentunya masih jauh dari sempurna. Maka dari itu terdapat beberapa saran untuk *developer* aplikasi selanjutnya yang ingin mengembangkan sistem ini yaitu :

1. Tampilan antarmuka (*interface*) aplikasi ini masih sangat standar. Sehingga diperlukan tampilan aplikasi yang dikembangkan dengan integrasi *bootstrap*, *google material* dan *framework interface* lainnya agar tampilan aplikasi lebih menarik dan interaktif.
2. Aplikasi sistem informasi keuangan ini berbasis web. Jika diakses di perangkat mobile maka perlu dibuka melalui *browser*. Maka perlu dikembangkan untuk versi *mobile* agar pengaksesan aplikasi lebih mudah dan cepat.

DAFTAR PUSTAKA

- Andreas Handojo, Go Ornella Aquaria, & Sri Maharsi. (2004). Pembuatan Sistem Informasi Akuntansi Terkomputerisasi Atas Siklus Pembelian Dan Penjualan Pada Cv. X. *Jurnal Informatika*, 5(2), 86–94. Retrieved from <http://puslit2.petra.ac.id/ejournal/index.php/inf/article/view/15843>
- Anggarani, A., Prasetyoning, W., & Safitri, V. I. (2015). Penguatan Sektor Umkm Sebagai Strategi Menghadapi Mea 2015, 5.
- Ariefin, M. R. S., & Ilhamsyah. (2015). MEMBANGUN WEBSITE SISTEM INFORMASI SEKOLAH DENGAN MENGGUNAKAN FRAMEWORK PHP LARAVEL PADA SMA NEGERI 4 PONTIANAK, 1–7.
- Dangar, H. (2013). *Learning Laravel 4 Application Development*. BIRMINGHAM - MUMBAI: Packt Publishing Ltd. Livery Place 35 Livery Street Birmingham B3 2PB, UK.
- Iswanto. (2006). Membangun Aplikasi Berbasis PHP 5 dan Firebird 1.5. Yogyakarta.
- Jumingan. (2005). *Analisis Laporan Keuangan (IV)*. Surakarta: Remaja Rosdakarya.
- Kabuhung, M. (2013). Sistem Informasi Akuntansi Penerimaan Dan Pengeluaran Kas Untuk Perencanaan Dan Pengendalian Keuangan Pada Organisasi Nirlaba Keagamaan. *EMBA*, 1(1988), 1–5. <http://doi.org/10.1017/CBO9781107415324.004>
- Lasena, S. R. (2013). ANALISIS PENENTUAN HARGA POKOK PRODUKSI PADA PT. DIMEMBE NYIUR AGRIPRO. *Jurnal EMBA*, 1(3), 585–592.
- Nugroho, A., Beeh, Y. R., & Astuningdyas, H. (2008). Perancangan Aplikasi Rencana Anggaran Biaya (Rab). *JURNAL INFORMATIKA*, 10, 10–18.
- Ralph M. Stair, G. W. R. (2010). *Principles of Information Systems A Managerial Approach* (9th ed.). florida.

- Sarfi'ah, S. nur. (2013). Menghitung harga pokok penjualan, 39(2), 80–90.
- Satya, E. (2015). Pembangunan Sistem Informasi Keuangan Studi Kasus SMK Marsudi Luhur I Yogyakarta. *Universitas Atma Jaya Yogyakarta, Vol 6, No.*
- Solichin, A. (2010). MySQL Dari Pemula Hingga Mahir. Universitas Budi Luhur, Jakarta.
- Sugiantina. (2011). PENERAPAN SISTEM INFORMASI DALAM PRAKTEK LAPORAN KEUANGAN SEDERHANA. *Teknologi, Vol 1, No .*
- Suwardjono. (2005). *Teori Akuntansi Perekayasaan Pelaporan Keuangan* (3rd ed.). Yogyakarta: BPFE-Yogyakarta.
- Tata Sutabsi. (2003). *Sistem Informasi Akuntansi*. Jakarta: Penerbit ANDI.

LAMPIRAN