

# APLIKASI PENGELOLAAN KEUANGAN DESA CIPAGALO BERBASIS WEB

## WEB BASED FINANCIAL MANAGEMENT APPLICATION OF CIPAGALO VILLAGE

Latifa Nur Fitriana<sup>1</sup>, M. Barja Sanjaya, S.T., M.T., OCA.<sup>2</sup>, Dahliar Ananda, S.T., M.T.<sup>3</sup>

<sup>1,2,3</sup>Prodi D3 Manajemen Informatika, Fakultas Ilmu Terapan, Universitas Telkom

<sup>1</sup>[latifanurfitriana@student.telkomuniversity.ac.id](mailto:latifanurfitriana@student.telkomuniversity.ac.id),

<sup>2</sup>[mbarjasanjaya@tass.telkomuniversity.ac.id](mailto:mbarjasanjaya@tass.telkomuniversity.ac.id), <sup>3</sup>[ananda@tass.telkomuniversity.ac.id](mailto:ananda@tass.telkomuniversity.ac.id)

---

### Abstrak

Desa Cipagalo merupakan perangkat desa yang ada di Kabupaten Bandung Selatan Jawa Barat, desa tersebut mempunyai pendapatan dana desa dan pengeluaran dana desa. Pendapatan dana tersebut didapat dari pemerintahan, Pendapatan Asli Daerah (PAD), dan Anggaran Pendapatan dan Belanja Daerah (APBD). Pada saat ini Kantor Kepala Desa Cipagalo mengalami beberapa kesulitan diantaranya adalah pencatatan pendapatan dana desa dan pengeluaran dana desa yang masih menggunakan buku kas, transaksi keuangan yang tidak rapi, dan pembuatan laporan dikerjakan selama berminggu-minggu karena data yang tidak rapi. Dengan permasalahan tersebut dibuatlah aplikasi yang diharapkan dapat membantu Kantor Kepala Desa Cipagalo dalam penyusunan keuangan kantor desa Cipagalo seperti pendapatan dana desa, pengeluaran dana desa, barang masuk kantor desa, barang keluar kantor desa dan laporan keseluruhannya. Pembuatan aplikasi ini menggunakan metode SDLC *Waterfall*, bahasa pemrograman PHP dengan *framework codeigniter*, dan *database MySql*. Sedangkan pengujian aplikasi ini menggunakan *blackbox testing*.

**Kata Kunci:** pendapatan dana, pengeluaran dana, SDLC *Waterfall*, PHP with CI *framework*, *MySql*, *blackbox testing*.

---

### Abstract

*The village of Cipagalo is an existing village device of South Bandung Regency of West Java, the village has income of village funds and the expenditure of village funds. Revenues of funds are obtained from the government, Pendapatan Asli Daerah (PAD), and Anggaran Pendapatan dan Belanja Daerah (APBD). At present, the Cipagalo Village Head Office is experiencing some difficulties such as the recording of village fund revenues and the disbursement of village funds that still use cash books, untidy financial transactions, and reporting done for weeks on unkempt data.*

**Keywords:** *fund revenues, expenses, SDLC Waterfall, PHP with CI framework, MySql, blackbox testing.*

---

## 1. Pendahuluan

### 1.1 Latar Belakang

Desa Cipagalo merupakan kelurahan yang terdapat di Kecamatan Bojongsoang, Kabupaten Bandung Selatan [1]. Desa Cipagalo merupakan perangkat daerah suksesnya penyelenggaraan di pemerintahan terutama dalam pelayanan masyarakat, karena desa Cipagalo adalah suatu instansi pemerintah yang maju dengan berhubungan langsung pada masyarakat dalam melaksanakan pekerjaannya [2]. Desa Cipagalo memiliki tugas penting yaitu melakukan dan mengkoordinir penyelenggaraan pemerintah daerah tingkat desa [3]. Desa Cipagalo diharuskan melakukan sebuah inovasi yang mendasar, terutama berhubungan dengan aspek

bentuk, struktur, fungsi desa Cipagalo bukan wilayah administrasi lagi, melainkan sebagai wilayah kerja perangkat daerah [4].

Kantor Desa Cipagalo adalah kantor yang mengolah dana desa berupa uang atau barang. Kantor Desa Cipagalo mengolah penerimaan dana, pengeluaran dana, barang masuk dan barang keluar, yang dana tersebut didapat dari pemerintah, Pendapatan Asli Daerah (PAD), dan Anggaran Pendapatan dan Belanja Daerah (APBD), dan instansi non pemerintahan.

Pembangunan aplikasi pengelolaan keuangan Desa Cipagalo berbasis *web* merupakan solusi dari permasalahan yang ada di kantor Desa Cipagalo yang belum menerapkan sistem komputer. Dengan di bangunnya aplikasi ini, diharapkan kesalahan dalam mencatat dan menyimpan data dapat teratasi karena adanya aplikasi yang akan membantu pegawai Desa Cipagalo dalam melaksanakan pekerjaannya. Aplikasi pengelolaan keuangan Desa Cipagalo akan ditambahkan fitur-fitur yang mampu mengingatkan pegawai apabila terjadi kesalahan dalam memasukkan data. Kemudahan dalam menggunakan aplikasi ini, juga akan menjadi sebuah kelebihan dalam aplikasi pengelolaan keuangan Desa Cipagalo berbasis *web* karena akan mempengaruhi proses kinerja pegawai kantor Desa Cipagalo.

Dalam mengolah penerimaan dana, pengeluaran dana, pemasukan barang dan pengeluaran barang, kantor Desa Cipagalo membuat laporan transaksi yang masih menggunakan buku kas. Bukti transaksi disimpan dengan tidak rapi dan tidak sesuai waktunya, sehingga menyebabkan bukti transaksi tersebar dan membutuhkan waktu beberapa hari untuk mengumpulkan bukti transaksi pada saat akan melakukan pencatatan. Laporan dibuat selama beberapa minggu dikarenakan pegawai harus meneliti kembali antara bukti transaksi dengan buku kas. Sumber daya manusia yang mengerjakan laporan lebih dari satu orang bahkan melibatkan selain bendahara, sehingga terjadi ketidaksesuaian laporan ketika diperiksa Kepala Desa [5].

Aplikasi pengelolaan keuangan Desa Cipagalo berfungsi untuk mencatat pendataan penerimaan dana, pengeluaran dana, pemasukan barang, pengeluaran barang kantor Desa Cipagalo. Didalam aplikasi ini juga akan ditambahkan fitur pengajuan laporan akhir penggunaan dana Desa Cipagalo.

### 1.2 Rumusan Masalah

Rumusan masalah yang dapat dipaparkan pada Proyek Akhir ini adalah sebagai berikut.

1. Bagaimana membangun sistem yang dapat membantu pegawai Kantor Desa Cipagalo dalam mengelola dokumen yang berkaitan dengan penerimaan dana, pengeluaran dana, pemasukan barang, dan pengeluaran barang?
2. Bagaimana membangun sistem yang dapat memberikan pemberitahuan berupa laporan akhir penggunaan dana kantor desa Cipagalo?

### 1.3 Tujuan

Adapun tujuan proyek akhir ini adalah membangun aplikasi sebagai berikut:

1. Membantu pegawai desa Cipagalo dalam mengelola data penerimaan dana, pengeluaran

dana, pemasukan barang, dan pengeluaran barang di Desa Cipagalo.

2. Membangun sistem yang disertai dengan pemberitahuan berupa laporan akhir penggunaan dana kantor desa Cipagalo.

### 1.4 Batasan Masalah

Pembuatan Sistem Aplikasi berbasis *web* ini mempunyai batasan-batasan antara lain:

1. sistem yang masih bersifat komunitas eksklusif yang berarti belum bisa diakses secara bebas oleh pihak luar dan hanya fokus pada pengolahan keuangan yang dijalankan kantor desa Cipagalo dan,
2. aplikasi ini hanya bisa diakses dalam *web*

### 1.5 Definisi Operasional

Aplikasi Pengelolaan Keuangan Cipagalo adalah suatu aplikasi yang dibuat dan dirancang bertujuan untuk memberikan kemudahan pada masyarakat maupun pihak Kantor Desa Cipagalo dengan menyediakan fitur untuk memudahkan dalam proses pengajuan laporan pencatatan keuangan. Aplikasi berbasis *web* ini dibangun dengan menggunakan *framework CodeIgniter* yang memanfaatkan teknik pemrograman berorientasi objek atau *object oriented programming* (OOP) dalam PHP. Adapun metode yang digunakan dalam tahapan pengerjaan aplikasi ini yaitu menggunakan metode *Waterfall* yang dimulai dari tahapan analisis kebutuhan, desain sistem, dan pengujian aplikasi. MySQL merupakan *database server* yang digunakan pada pembuatan aplikasi ini. Adapun fitur dari aplikasi ini adalah terdapat menu input dana masuk, dana keluar, barang masuk, barang keluar, laporan keuangan dan laporan barang.

### 1.6 Metode Pengerjaan

Model pengembangan yang digunakan dalam membangun aplikasi ini adalah *Linear Sequential Model* atau *Waterfall Model*. Disebut dengan *Waterfall* karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan. Secara umum tahapan pada model *Waterfall* dapat dilihat pada gambar berikut:

1. Analisis Kebutuhan

Pada tahap ini dilakukan analisis dan didefinisikan kebutuhan yang harus dipenuhi oleh aplikasi yang akan dibangun. Untuk memperoleh informasi tentang proses bisnis dan kebutuhan perusahaan, maka dilakukan wawancara dan survey pada bagian unit administrasi keuangan di kantor desa Cipagalo [6].

2. Desain Sistem

Pada tahap desain dilakukan perancangan aplikasi baru berdasarkan data-data yang telah dikumpulkan pada tahap sebelumnya [6]. Adapun yang didesain diantaranya: diagram proses bisnis dengan BPMN,

UML, *Entity Relationship Diagram* (ERD), Skema Relasi, dan Struktur Tabel serta *Mock-Up*.

### 3. Penulisan Kode Program

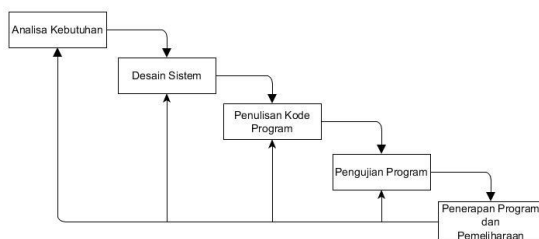
Pada tahap pembuatan kode, pengerjaan aplikasi berdasarkan desain yang telah dibuat. Pada tahap ini dilakukan pembuatan kode untuk membuat aplikasi sesuai dengan desain yang telah dibuat [6]. Juga pada tahapan ini dilakukan implementasi database ke aplikasi *MySQL* beserta konfigurasi lainnya.

### 4. Pengujian Program

Tahapan ini merupakan tahap final dalam perancangan suatu sistem. Hasil dari proses pembuatan kode tersebut digabungkan kemudian dilakukan *testing* untuk menguji kesalahan-kesalahan program maupun fungsi dari sistem sehingga nantinya dapat digunakan oleh *user* [6]. Adapun pengujian yang dilakukan yakni menggunakan implementasi konsep *Black Box Testing*.

### 5. Implementasi

Pada tahap implementasi, aplikasi yang telah layak pengujiannya yang akan digunakan pada Kantor Desa Cipagalo [6].



Gambar 1. Metode Pengerjaan [6]

## 2 Tinjauan Pustaka

### 2.1 Dana Desa

Dana desa adalah dana yang bersumber dari Anggaran Pendapatan dan Belanja Negara (APBN) yang diperuntukkan bagi desa dan desa adat yang ditransfer melalui Anggaran Pendapatan dan Belanja Daerah (APBD) Kabupaten/ Kota dan digunakan untuk membiayai penyelenggaraan pemerintahan, pembangunan, serta pemberdayaan masyarakat, dan kemasyarakatan. Fokus penting dari penyaluran dana ini lebih terkait pada implementasi pengalokasian dana desa agar dapat sempurna gagasan para inisiatornya. Skenario awal dana desa ini diberikan dengan mengganti program pemerintah yang dulunya disebut PNPM, namun dengan berlakunya dana desa ini, dapat menutup kesempatan beberapa pihak asing untuk menyalurkan dana ke daerah di Indonesia dengan program-program yang sebenarnya juga dapat menjadi pemicu pembangunan daerah.

Sesuai dengan amanat Undang-Undang Nomor 6 Tahun 2014 tentang desa, Pemerintah mengalokasikan dana desa melalui mekanisme

transfer kepada Kabupaten/ Kota [2]. Berdasarkan alokasi dana tersebut, maka tiap Kabupaten/ Kota mengalokasikannya kepada tiap desa berdasarkan jumlah desa dengan memperlihatkan jumlah penduduk (30%), luas wilayah (20%), dan angka kemiskinan (50%). Hasil perhitungan tersebut disesuaikan juga dengan tingkat kesulitan geografis masing-masing desa. Alokasi anggaran yang dimaksud adalah bersumber dari belanja pusat dengan mengefektifkan program yang berbasis desa secara merata dan berkeadilan. Besaran alokasi anggaran yang peruntukannya langsung ke desa ditentukan 10% dan dari dana Transfer Daerah secara bertahap.

### 2.2 Pengelolaan

Pengelolaan adalah penyelenggaraan, pengurus atau proses yang membantu merumuskan kebijaksanaan dan tujuan organisasi [8]. Pengelolaan sama halnya dengan manajemen, karena pengelolaan dalam sebuah organisasi memerlukan pelaksanaan tanggung jawab manajerial secara terus menerus. Dan tanggung jawab tersebut secara kolektif sering disebut dengan fungsi manajemen. Manajemen dalam organisasi pada dasarnya hanya dimaksudkan sebagai suatu proses (aktivitas) penentuan dan pencapaian tujuan melalui pelaksanaan empat fungsi dasar : Planning, organizing, actuating dan controlling dalam penggunaan sumber daya organisasi, manajemen memerlukan koordinasi sumber daya manusia dan material ke arah tercapainya tujuan.

### 2.3 Keuangan

Keuangan adalah mempelajari bagaimana individu, bisnis dan organisasi meningkatkan, mengalokasi, dan menggunakan sumber daya moneter sejalan dengan waktu, dan juga menghitung resiko dalam menjalankan proyek mereka. Keuangan berhubungan dengan proses, lembaga, pasar, dan instrument yang terlibat dalam transfer uang diantara individu maupun antara bisnis dan pemerintah [9].

### 2.4 Definisi Aplikasi

Aplikasi adalah suatu unit perangkat lunak yang dibuat untuk melayani kebutuhan akan beberapa aktifitas seperti sistem perniagaan, pelayanan masyarakat, periklanan atau semua proses yang dilakukan manusia [10]. Dengan menggunakan sistem komputerisasi, diharapkan pekerjaan dapat dilakukan dengan cepat. Aplikasi dapat dibagi menjadi dua bagian, yaitu aplikasi yang terhubung pada jaringan komputer (*client server*) dan aplikasi yang berdiri sendiri tidak terhubung dengan jaringan komputer (*stand alone*) [11]

### 2.5 Notasi Perancangan Sistem

Notasi perancangan sistem terdiri dari Diagram Kelas, Diagram Aktivitas, *Usecase Diagram*, *Entity*

*Relationship Diagram* (ERD), dan *Structured Query Language* (SQL). Berikut adalah beberapa bagian dari notasi perancangan sistem.

### 2.5.1 Diagram Kelas

Diagram kelas digunakan untuk memodelkan pandangan rancangan struktural statis untuk sistem. Hubungan-hubungan kelas diilustrasikan secara grafik. Diagram kelas memperlihatkan kebutuhan *fungsi* dari sistem dikumpulkan oleh cara analisis dan juga rancangan fisik sistem, tetapi tidak mencerminkan keadaan sistem pada titik waktu tertentu.

Antar-muka dan kolaborasi kelas juga disajikan pada diagram kelas, dan atribut-atribut dan operasi yang dimuat dengan masing-masing kelas juga ditampilkan. Model sistem O-O biasanya terdiri dari jumlah diagram kelas yang melukiskan seksi berbeda dari sistem. Secara kolektif, semua diagram kelas mencerminkan rancangan seluruh sistem [12].

### 2.5.2 Diagram Aktivitas

Diagram aktivitas memperlihatkan aliran aktivitas dalam sebuah proses. Diagram ini berbeda dengan diagram bagan keadaan, yang memfokuskan pada keadaan dan transisi keadaan daripada urutan aktivitas selama operasi. Diagram aktivitas memperlihatkan transisi antara keadaan dan aliran *control* dalam proses [12].

### 2.5.3 Usecase Diagram

*Usecase* diagram adalah diagram yang membantu dalam menyusun *requirement* sebuah sistem, mengkomunikasikan rancangan dengan klien, dan merancang *test case* untuk semua fitur yang ada pada sistem. *Usecase* diagram menggunakan fungsionalitas yang diharapkan dari sebuah sistem. *Usecase* diagram menekankan apa yang dibuat sistem, dan bukan bagaimana. Sebuah *usecase* mempresentasikan sebuah interaksi antara *actor* dengan sistem [13].

### 2.5.4 Entity Relationship Diagram (ERD)

*Entity Relationship Diagram* (ERD) merupakan suatu model data yang dikembangkan berdasarkan objek. ERD digunakan untuk menjelaskan hubungan antar data dalam basis data kepada pengguna secara logis. ERD didasarkan pada suatu persepsi bahwa *real world* terdiri atas objek-objek dasar tersebut. Penggunaan ERD relatif mudah dipahami, bahkan oleh para pengguna yang awam. Bagi perancang atau analis sistem, ERD berguna untuk memodelkan sistem yang nantinya, basis data akan dikembangkan. Model ini juga membantu perancang atau analis sistem pada saat melakukan analisis dan perancangan basis data karena model ini dapat menunjukkan macam data yang dibutuhkan dan kerelasi antar data didalamnya [14]

## 2.6 Structured Query Language (SQL)

MySQL merupakan sistem manajemen *database* (*Database Management System – DBMS*) yang sangat populer di kalangan pemrograman *web*, terutama di lingkungan *Linux* dengan menggunakan *script* PHP dan Perl. MySQL merupakan *database* yang paling populer digunakan untuk membangun aplikasi *web* yang menggunakan *database* sebagai sumber dan pengelola datanya. *Database* MySQL, merupakan *database* yang menjanjikan sebagai alternatif pilihan *database* yang dapat digunakan untuk sistem *database* personal atau organisasi. *Oracle* sebagai *database* besar telah membuat *kit* (modul) untuk memudahkan proses migrasi dari MySQL ke dalam *Oracle*, hal ini dapat menunjukkan bahwa *Oracle* telah memperhitungkan *database* MySQL sebagai *database* alternatif masa depan. Demikian juga dengan pengguna dari *database* MySQL, menunjukkan makin banyaknya perusahaan besar menggunakannya [15].

Berikut contoh membuat *database* pada SQL:

```
CREATE DATABASE pegawai(
Nik int(10) Primary Key,
Nama_lengkap varchar2(50)
Golongan varchar2(10));
SHOW DATABASE;
DROP DATABASE pegawai;
```

## 2.7 Bahasa Pemrograman

Bahasa pemrograman terdiri dari *Hypertext Preprocessor* (PHP), *Codeigniter* (CI), *Cascading Style Sheet* (CSS), *Hyper Text Markup Language* (HTML). Berikut adalah beberapa bagian dari bahasa pemrograman.

### 2.7.1 Hypertext Preprocessor (PHP)

*Hypertext Preprocessor* (PHP) yang merupakan *server-side programming*, yaitu Bahasa pemrograman yang diproses di sisi server. Fungsi utama PHP dalam membangun *website* adalah untuk melakukan pengolahan data pada *database*. Data *website* akan dimasukkan ke *database*, diedit, dihapus dan ditampilkan pada *website* yang diatur oleh PHP [16]. PHP hanya mengeksekusi kode yang ditulis dalam pembatas sebagaimana ditentukan oleh dasar sintaks PHP. Adapun diluar pembatas tidak diproses oleh PHP (meskipun teks PHP masih mengendalikan struktur yang dijelaskan dalam kode PHP. Pembatas yang paling umum adalah “<?php” untuk membuka dan “?>” untuk menutup kode PHP. Tujuan dari pembatas ini adalah untuk memisahkan kode PHP dari kode diluar PHP, seperti HTML dan *Javascript* [17].

Contoh *script* PHP :

```
<?php
echo "Aplikasi Pengelolaan Keuangan Desa
Cipagalo";
```

?>

### 2.7.2 CodeIgniter (CI)

*CodeIgniter* (CI) adalah *framework* pengembangan aplikasi (*Application Development Framework*) dengan menggunakan PHP, suatu kerangka pembuatan program dengan menggunakan PHP. Pemrograman tidak perlu membuat program dari awal (*from scratch*), karena CI menyediakan sekumpulan *library* yang banyak yang diperlukan untuk menyelesaikan pekerjaan yang umum, dengan menggunakan atarmuka dan struktur *logic* yang sederhana untuk mengakses librarinya. Pemrograman dapat memfokuskan diri pada kode yang sama dibuat untuk menyelesaikan suatu pekerjaan [18]. *Codeigniter* terdapat teknik atau konsep yaitu MVC dan dibagi menjadi tiga komponen yaitu *model*, *view*, dan *controller*.

- Model* merupakan bagian penanganan yang berhubungan dengan pengolahan atau manipulasi *database*. Seperti mengambil data dari *database*, menginput dan pengolahan *database* lainnya. Semua intruksi yang berhubungan dengan pengolahan *database* diletakkan didalam *model*.
- View* merupakan bagian yang menangani halaman *user interface* atau halaman yang muncul pada *user*. Tampilan dari *user interface* dikumpulkan pada *view* untuk memisahkan dengan *controller* dan *model* sehingga memudahkan *web designer* dalam melakukan pengembangan tampilan halaman *website*.
- Controller* merupakan kumpulan intruksi aksi yang menghubungkan *model* dan *view*, jadi *user* tidak akan berhubungan dengan *model* secara langsung, dengan arti dari *view* kemudian *controller* yang mengolah intruksi [19].

### 2.7.3 Cascading Style Sheet (CSS)

*Cascading Style Sheet* (CSS) merupakan salah satu Bahasa pemrograman *web* untuk mengendalikan beberapa komponen dalam sebuah *web* sehingga akan lebih terstruktur dengan seragam. CSS dapat mengendalikan ukuran gambar, warna bagian tubuh, pada teks, warna tabel, ukuran *border*, warna *border*, warna *hyperlink*, dan parameter lainnya. CSS adalah bahasa *style sheet* yang digunakan untuk mengatur tampilan dokumen. Dengan adanya CSS memungkinkan untuk menampilkan halaman yang sama dengan format yang berbeda [20].

Berikut contoh kode program CSS:

```
.table1
{
Border-collapse:collapse;
Border:1px red;
}
.table2{
Border=1";
}
Table, td, th
```

```
{
Border:1px solid black;
}
```

### 2.7.4 Hyper Text Markup Language (HTML)

HTML berawal dari bahasa *Standart Generalized Markup Language* (SGML) yang penulisannya disederhanakan. HTML dapat dibaca oleh berbagai macam *platform*. HTML juga merupakan bahasa pemrograman yang fleksibel dan dapat digabungkan dengan bahasa pemrograman lain seperti PHP, ASP, JSP, dan *JavaScript*. Beberapa tag dalam dokumen-dokumen HTML menentukan bagaimana teks diformat. Tag-tag yang lain memberitahukan komputer bagaimana menanggapi aksi-aksi yang datang dari pengguna. Kemudian tag lain yang penting adalah link yang mengandung *Uniform Resource Locator* (URL), yang merujuk pada dokumen lain diserver yang sama atau komputer lain yang ada di global jaringan internet [21].

Contoh kode program HTML:

```
<html>
<head>
<title>Aplikasi pengelolaan
keuangan desa cipagalo</title>
</head>
<body>
</html>
```

### 2.8 Pengujian Black-Box

Pengujian *black-box* berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujian *black-box* memungkinkan perekrutan perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsionalitas untuk suatu program. Pengujian *black-box* bukan merupakan alternatif dari teknik *white-box*, tetapi merupakan pendekatan komplementer yang kemungkinan besar mampu mengungkap kelas kesalahan daripada metode *white-box*.

Pengujian *black-box* berusaha menemukan kesalahan dalam kategori sebagai berikut:

- fungsi- fungsi yang tidak benar atau hilang,
- kesalahan *interface*,
- kesalahan dalam struktur data atau akses *database* eksternal,
- kesalahan kinerja dan
- inisialisasi dan kesalahan terminasi [22]

**2.9 User Acceptance Test (UAT)**

UAT singkatan dari *User Acceptance Test* adalah Uji Penerimaan Pengguna adalah suatu proses pengujian oleh pengguna yang dimaksudkan untuk menghasilkan dokumen yang dijadikan bukti bahwa software yang telah dikembangkan telah dapat diterima oleh pengguna, apabila hasil pengujian (testing) sudah bisa dianggap memenuhi kebutuhan dari pengguna.

UAT juga memiliki arti dari suatu proses pengujian oleh pengguna yang bertujuan untuk menghasilkan dokumen yang bisa dijadikan bukti bahwa produk yang dibuat dapat diterima pengguna

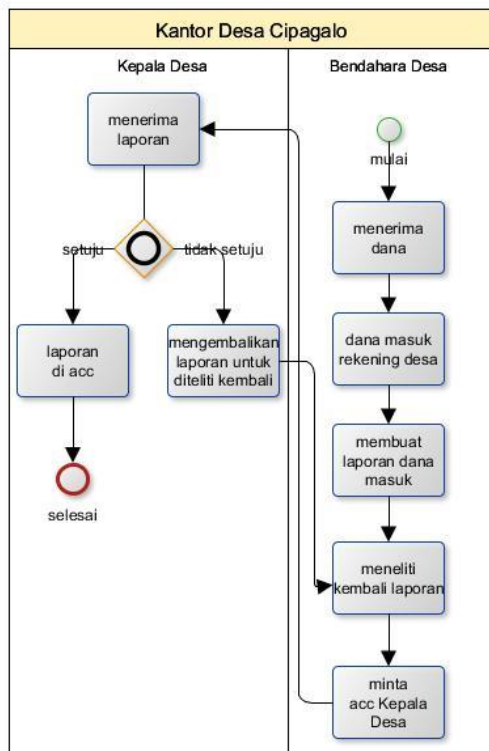
**3 Analisis dan Perancangan Sistem**

**3.1 Gambaran Sistem Saat Ini**

Bagian ini menjelaskan proses bisnis yang berjalan atau sebuah kondisi lapangan, bagaimana jalannya tiap tahapan dalam melaksanakan proses pengelolaan keuangan pada Desa Cipagalo masih dilakukan secara manual. Dalam mengelola keuangan yang masuk ke Desa Cipagalo masih menggunakan Ms Excel dan pembuatan laporan Ms. Word. Proses seperti ini kurang efektif dikarenakan tingkat kehilangan data atau *miss communication* terhadap pihak yang terlibat akan sangat besar.

**3.1.1 Proses Bisnis Penerimaan Dana Saat Ini**

Berikut adalah gambar proses bisnis penerimaan dana saat ini.

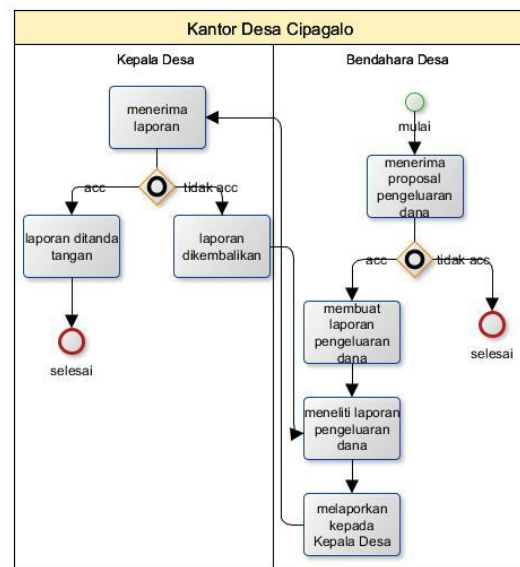


**Gambar 2. Proses Bisnis Penerimaan Dana Saat Ini**

Proses pertama yaitu menerima dana dari pusat, biasanya dana tersebut masuk melalui rekening desa. Kemudian dilanjutkan dengan bendahara desa menulis laporan dana bantuan yang telah masuk, jika sudah selesai dalam penulisan laporan maka bendahara desa harus meneliti kembali hasil laporan tersebut. Jika proses meneliti dirasa sudah cukup, bendahara desa meminta acc kepada Kepala Desa sebagai bukti telah tersusun laporan dana. Kepala Desa akan menerima laporan tersebut, jika dirasa laporan tersebut kurang dalam penyusunannya atau salah dalam penulisan maka Kepala Desa akan mengembalikan laporan tersebut kepada Bendahara Desa. Jika penyusunan dan data sudah benar maka Kepala Desa approve laporan tersebut.

**3.1.2 Proses Bisnis Pengeluaran Dana Saat Ini**

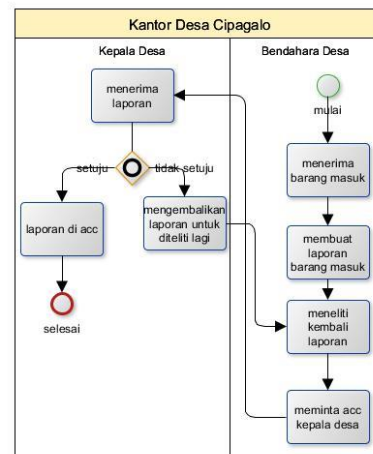
Berikut adalah gambar proses bisnis pengeluaran dana saat ini.



**Gambar 3. Proses Bisnis Pengeluaran Dana Saat Ini**

**3.1.3 Proses Bisnis Barang Masuk Saat Ini**

Berikut adalah gambar proses bisnis barang masuk saat ini.

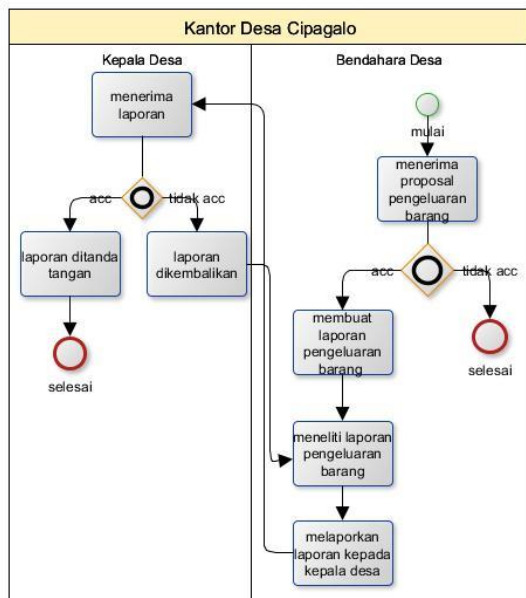


**Gambar 4. Proses Bisnis Barang Masuk Saat Ini**

Proses ketiga yaitu bendahara desa menerima barang masuk kemudian bendahara desa membuat laporan barang masuk, jika laporan sudah dibuat maka bendahara desa meneliti kembali laporan barang masuk. Jika dirasa sudah benar maka bendahara desa meminta tanda-tangan kepala desa. Jika kepala desa sudah menerima laporan barang masuk bendahara desa akan menandatangani laporan, jika tidak setuju kepala desa mengembalikannya dan untuk diteliti kembali.

**3.1.4 Proses Bisnis Barang Keluar Saat Ini**

Berikut adalah gambar proses bisnis barang keluar saat ini.



**Gambar 5. Proses Bisnis Barang Keluar Saat Ini**

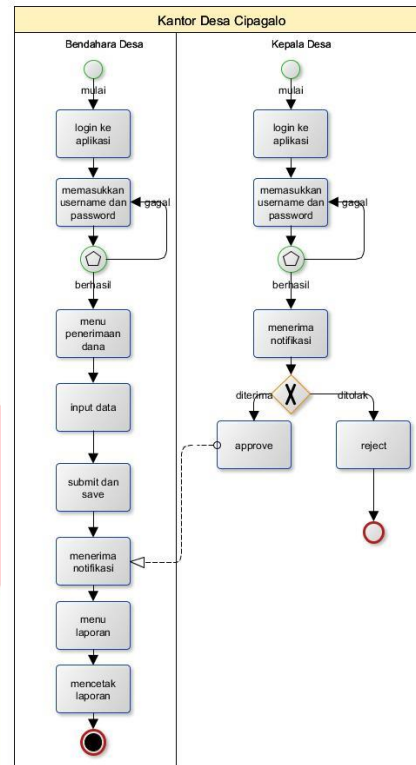
Proses keempat yaitu bendahara desa menerima proposal pengeluaran barang, jika disetujui maka bendahara desa membuat laporan pengeluaran barang jika tidak disetujui maka bendahara desa tidak membuat laporan dan proses selesai. Setelah membuat laporan, bendahara desa meneliti laporan pengeluaran barang kemudian melaporkan laporan kepada kepala desa. Kepala desa akan menerima laporan dari bendahara desa, jika disetujui maka laporan akan ditanda-tangani oleh kepala desa, jika tidak disetujui laporan akan dikembalikan kembali kepada bendahara desa untuk diteliti kembali.

**3.2 Gambaran Sistem Yang Diusulkan**

Pada bagian ini akan menjelaskan tentang sistem proses bisnis yang diusulkan, dimulai dari cara pemakaiannya hingga alur proses bisnis yang baru. Sistem ini akan menggunakan aplikasi berbasis *website* untuk proses dalam mengelola keuangan yang masuk dan keluar dan mengelola barang masuk dan barang keluar di Desa Cipagalo. Beberapa staff diberikan *username* dan *password* untuk mengakses aplikasi *website* ini.

**3.2.1 Proses Bisnis Usulan Penerimaan Dana**

Berikut adalah gambar proses bisnis usulan mengelola penerimaan dana.

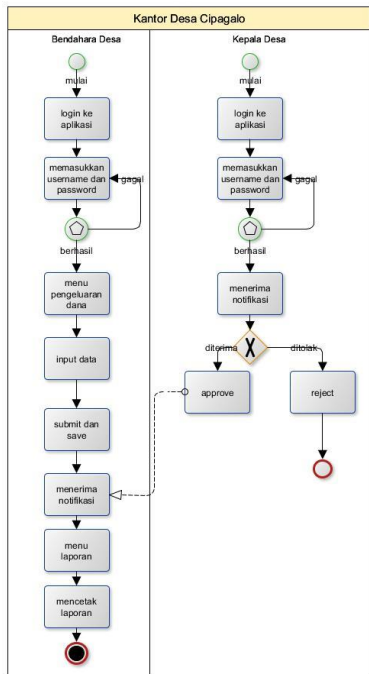


**Gambar 6. Proses Bisnis Usulan Penerimaan Dana**

Sistem yang diusulkan adalah mengelola data penerimaan dana dengan cara Bendahara Desa tidak perlu menginputkan data ke Microsoft Excel secara manual namun dengan langkah *login* terlebih dahulu ke dalam aplikasi dengan memasukkan *username* dan *password*. Jika proses *login* berhasil maka Bendahara Desa lanjut dengan membuka menu kelola data penerimaan dana, jika dalam proses *login* gagal Bendahara Desa harus memasukkan kembali *username* dan *password*. Jika sudah ke menu kelola penerimaan dana atau menu kelola pengeluaran dana bendahara bisa memasukan data kemudian dapat *submit* dan disimpan.

**3.2.2 Proses Bisnis Usulan Pengeluaran Dana**

Berikut adalah gambar proses bisnis usulan mengelola pengeluaran dana.

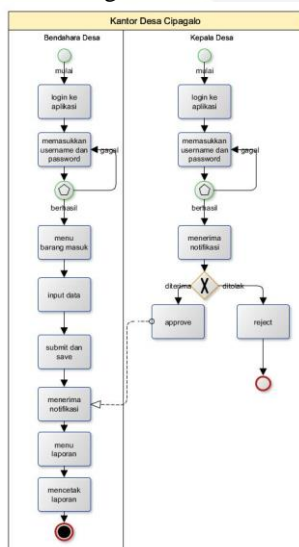


Gambar 7. Proses Bisnis Usulan Pengeluaran Dana

Sistem yang diusulkan adalah mengelola data pengeluaran dana dengan cara Bendahara Desa tidak perlu menginputkan data ke Microsoft Excel secara manual namun dengan langkah *login* terlebih dahulu ke dalam aplikasi dengan memasukkan *username* dan *password*. Jika proses *login* berhasil maka Bendahara Desa lanjut dengan membuka menu kelola data pengeluaran dana, jika dalam proses *login* gagal Bendahara Desa harus memasukkan kembali *username* dan *password*. Jika sudah ke menu kelola penerimaan dana atau menu kelola pengeluaran dana bendahara bisa memasukan data kemudian dapat di-*submit* dan disimpan.

3.2.3 Proses Bisnis Usulan Barang Masuk

Berikut adalah gambar proses bisnis usulan mengelola data barang masuk

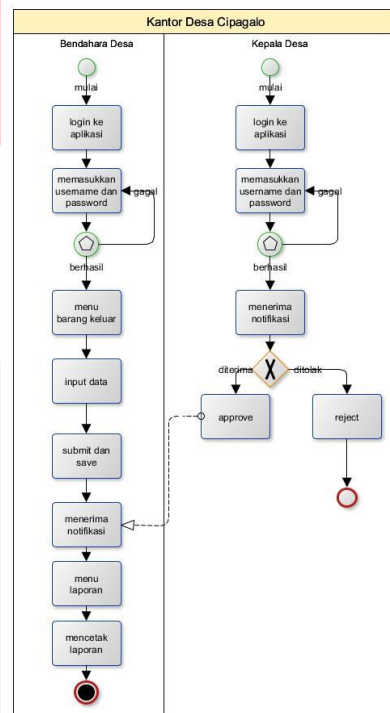


Gambar 8. Proses Bisnis Usulan Barang Masuk

Sistem yang diusulkan adalah mengelola data barang masuk dengan cara Bendahara Desa tidak perlu menginputkan data ke Ms. Excel secara manual namun dengan langkah *login* terlebih dahulu ke dalam aplikasi dengan memasukkan *username* dan *password*. Jika proses *login* berhasil maka Bendahara Desa lanjut dengan membuka menu kelola data barang masuk, jika dalam proses *login* gagal Bendahara Desa harus memasukkan kembali *username* dan *password*. Jika sudah ke menu kelola barang masuk atau menu kelola barang keluar bendahara bisa memasukan data kemudian dapat di-*submit* dan disimpan.

3.2.3 Proses Bisnis Usulan Barang Keluar

Berikut adalah gambar proses bisnis usulan mengelola data barang keluar.



Gambar 9. Proses Bisnis Usulan Barang Keluar

Sistem yang diusulkan adalah mengelola data barang keluar dengan cara Bendahara Desa tidak perlu menginputkan data ke Ms. Excel secara manual namun dengan langkah *login* terlebih dahulu ke dalam aplikasi dengan memasukkan *username* dan *password*. Jika proses *login* berhasil maka Bendahara Desa lanjut dengan membuka menu kelola data barang keluar, jika dalam proses *login* gagal Bendahara Desa harus memasukkan kembali *username* dan *password*. Jika sudah ke menu kelola barang masuk atau menu kelola barang keluar bendahara bisa memasukan data kemudian dapat di-*submit* dan disimpan.

3.3 Perancangan Basis Data

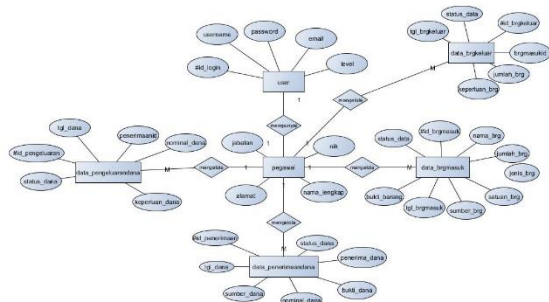
Dalam sub bab ini membahas perancangan basis data yang diantaranya terdapat *entity relationship*



diagram, skema relasi, class diagram dan struktur tabel.

3.3.1 Entity Relationship Diagram (ERD)

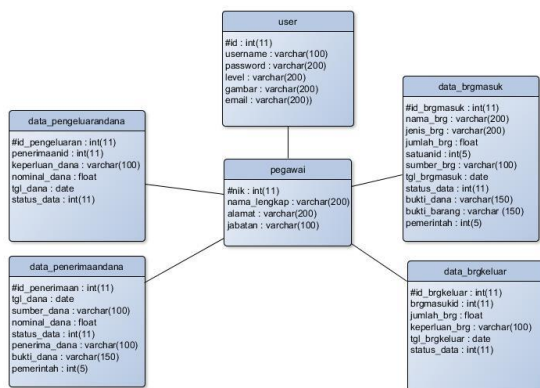
Berikut adalah Entity Relationship Diagram (ERD) dari Aplikasi Pengelolaan Penerimaan Dana Bantuan untuk Desa Cipagalo.



Gambar 10. Entity Relationship Diagram (ERD)

3.3.2 Skema Relasi

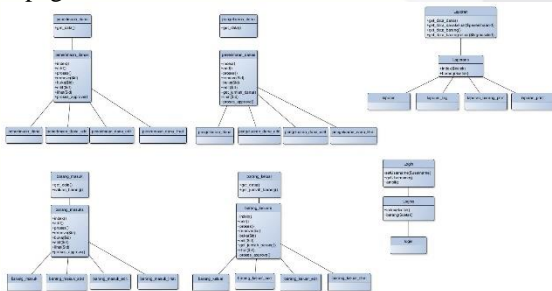
Berikut adalah gambar skema relasi untuk aplikasi pengelolaan keuangan Desa Cipagalo.



Gambar 11. Skema Relasi

3.3.3 Class Diagram

Berikut adalah class diagram dari Aplikasi Pengelolaan Keuangan untuk Desa Cipagalo berdasarkan kebutuhan yang ada di kantor Desa Cipagalo.



Gambar 12. Class Diagram

3.3.4 Struktur Tabel

Struktur tabel pada tabel dibawah menjelaskan tentang tabel Kepala Desa, Bendahara Desa, laporan keuangan, data barang masuk, data barang keluar, data pemasukan dana, dan data pengeluaran dana.

Tabel 1. Struktur Tabel Login

Fields	Tipe Data	Keterangan
Id	Int (11)	Primary Key, merupakan id untuk login
username	Varchar (100)	Not null, merupakan username untuk akun kepala desa dan bendahara desa
Password	Varchar (100)	Not null, merupakan password untuk kepala desa dan bendahara desa
Nama_lengkap	Varchar (200)	Not null, merupakan nama yang harus diisi oleh kepala desa dan bendahara desa
Level	Varchar (200)	Not null, merupakan level diisi oleh kepala desa dan bendahara desa
Gambar	Varchar (200)	Not null, merupakan gambar kepala desa dan bendahara desa
Email	Varchar (200)	Not null, merupakan email kepala desa dan bendahara desa

Tabel 2. Struktur Tabel Data Barang Masuk

Fields	Tipe Data	Keterangan
Id_brgmasuk	int (11)	Primary Key, merupakan id untuk barang masuk
Nama_brg	Varchar (100)	Not null, merupakan nama barang dari data barang masuk.
Jumlah_brg	float	Not null, merupakan jumlah barang dari data barang masuk.
Satuan_brg	Varchar (15)	Not null, merupakan satuan barang dari data barang masuk.

Sumber_brg	Varchar (100)	Not null, merupakan sumber barang dari data barang masuk.
Tgl_brgmasuk	date	Not null, merupakan tanggal barang masuk dari data barang masuk.
Status_data	Int (11)	Not null, merupakan status data dari data barang masuk.
Bukti_dana	Varchar (150)	Not null, merupakan bukti dana dari data barang masuk.

Tabel 3. Struktur Tabel Data Barang Keluar

<i>Fields</i>	<i>Tipe Data</i>	<i>Keterangan</i>
Id_brgkeluar	Int (11)	Primary Key, merupakan id untuk barang keluar
brgmasukid	Int (11)	Not null, merupakan id dari barang masuk dari data barang keluar.
Jumlah_brg	float	Not null, merupakan jumlah barang dari data barang keluar.
keperluan_brg	Varchar (100)	Not null, merupakan keperluan barang dari data barang keluar.
Tgl_brgkeluar	date	Not null, merupakan tanggal barang keluar dari data barang keluar.
Status_data	Int (11)	Not null, merupakan status data dari data barang keluar.

Tabel 4. Struktur Tabel Data Penerimaan Dana

<i>Fields</i>	<i>Tipe Data</i>	<i>Keterangan</i>
Id_penerimaan	Int (11)	Not null, merupakan id dari penerimaan dana.
Sumber_dana	Varchar (100)	Not null, merupakan sumber dana dari penerimaan dana.
Nominal_dana	Float	Not null, merupakan

		nominal dana dari penerimaan dana.
Tgl_dana	date	Not null, merupakan tanggal dari penerimaan dana.
Status_data	Int (11)	Not null, merupakan status data dari data penerimaan dana.
Bukti_dana	Varchar (150)	Not null, merupakan bukti dana dari data penerimaan dana.
Penerima_dana	Varchar (150)	Not null, merupakan penerima dana dari data penerimaan dana.

Tabel 5. Struktur Tabel Data Pengeluaran Dana

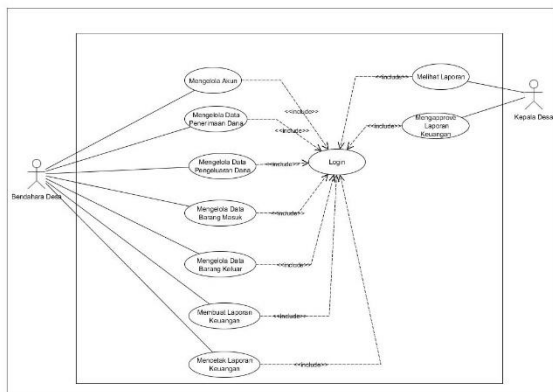
<i>Fields</i>	<i>Tipe Data</i>	<i>Keterangan</i>
Id_pengeluaran	Int (11)	Not null, merupakan id dari pengeluaran dana.
penerimaandanaid	Int (11)	Not null, merupakan id dari penerimaan dana.
Keperluan_dana	Varchar (100)	Not null, merupakan keperluan dari penerimaan dana.
Nominal_dana	Number	Not null, merupakan nominal dana dari pengeluaran dana.
Tgl_dana	Varchar (50)	Not null, merupakan tanggal dari pengeluaran dana.
Status_data	Int (11)	Not null, merupakan status data dari data pengeluaran dana.

### 3.4 Analisis Kebutuhan Sistem

Pada analisis kebutuhan sistem dari Aplikasi Pengelolaan Keuangan Desa Cipagalo terdapat diagram *usecase*, deskripsi aktor, deskripsi *usecase*, dan skenario *usecase*.

#### 3.4.1 Diagram Usecase

Diagram *usecase* ini untuk menjelaskan fungsionalitas yang terdapat dalam aplikasi ini. Berikut adalah *usecase* yang terdapat dalam aplikasi:



Gambar 13. Usecase

#### 3.4.2 Diskripsi Aktor

Berikut adalah tabel yang menjelaskan deskripsi dari masing-masing aktor yang terdapat pada Aplikasi Pengelolaan Keuangan Desa Cipagalo.

Tabel 6. Deskripsi Aktor

Aktor	Deskripsi
Bendahara desa	Aktor ini adalah seorang bendahara di kantor Desa Cipagalo yang bertugas untuk mengelola dana yang masuk, dana yang keluar, barang yang masuk, barang yang keluar, membuat laporan keuangan, dan mengupload laporan keuangan.
Kepala Desa	Aktor ini adalah seorang pimpinan di kantor Desa Cipagalo yang bertugas mengawasi jalannya laporan keuangan yang masuk ke Desa Cipagalo

#### 3.4.3 Diskripsi Usecase

Pada tabel di bawah ini menjelaskan mengenai deskripsi dari masing-masing *usecase* yang ada dalam Aplikasi Pengelolaan Keuangan Desa Cipagalo.

##### 3.4.3.1 Definisi Usecase Bendahara Desa

Berikut adalah tabel definisi *usecase* bendahara desa.

Tabel 7. Definisi Usecase Bendahara Desa

Use Case	Deskripsi
Login	Aktifitas yang dilakukan agar dapat mengakses aplikasi dengan memasukan <i>username</i> dan <i>password</i> .
Mengelola data pemasukan dana	Proses untuk mengelola dana yang masuk ke Desa Cipagalo dengan menginputkan sumber dana, nominal dana, dan tanggal dana.
Mengelola data pengeluaran dana	Proses untuk mengelola dana yang keluar dari Desa Cipagalo dengan menginputkan keperluan dana, nominal dana, dan tanggal dana.
Mengelola data barang masuk	Proses untuk mengelola barang yang masuk ke Desa Cipagalo dengan menginputkan nama barang, jumlah barang, sumber barang, dan jenis barang.
Mengelola data barang keluar	Proses untuk mengelola barang yang keluar dari Desa Cipagalo dengan menginputkan nama barang, jumlah barang, keperluan barang, dan jenis barang.
Membuat laporan keuangan	Proses ini adalah untuk membuat laporan keuangan yang akan diupload oleh bendahara desa untuk kepala desa.

##### 3.4.3.1 Definisi Usecase Kepala Desa

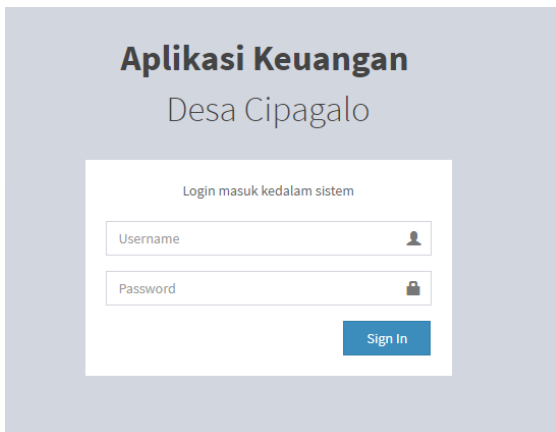
Berikut adalah tabel definisi *usecase* kepala desa.

Tabel 8. Definisi Usecase Kepala Desa

Use Case	Diskripsi
Login	Aktifitas yang dilakukan agar dapat mengakses aplikasi dengan memasukan <i>username</i> dan <i>password</i> .
Mengapprove Laporan Keuangan	Proses dimana kepala desa melakukan proses approve pada laporan keuangan

## 4. Implementasi

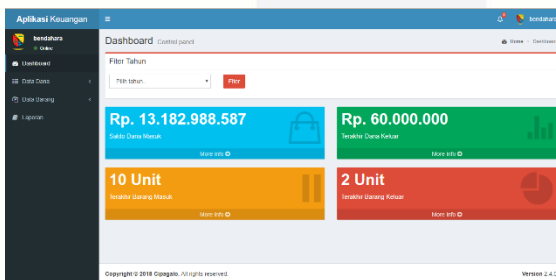
### 4.1 Implementasi Form Login



Gambar 14. Implementasi Form Login

Form *login* tersebut digunakan untuk masuk ke dalam sebuah aplikasi atau sistem. Proses *login* terdiri dari mengisikan nama *username* dan *password*. Jika, user menginputkan *username* dan *password* dengan benar kemudian berhasil maka akan masuk ke dalam halaman utama. Apabila proses *login* tidak berhasil maka akan terdapat pengulangan proses *login*. Di dalam form *login* ini, terdapat tombol “*Sign In*” dengan fungsi apabila sudah menginputkan *username* dan *password* hanya perlu menekan tombol tersebut kemudian masuk ke halaman utama aplikasi.

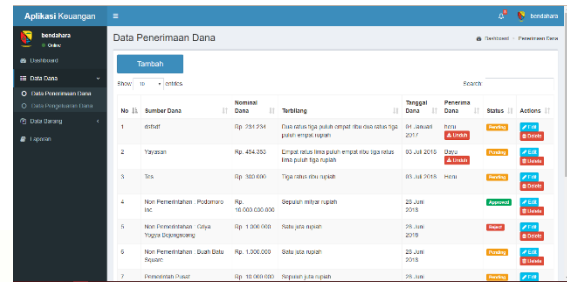
### 4.2 Implementasi Form Dashboard



Gambar 15. Implementasi Form Dashboard

Form *dashboard* adalah sebagai akses menuju menu-menu yang terdapat pada aplikasi tersebut. Menu tersebut antara lain data dana yang terdiri dari penerimaan dana dan pengeluaran dana, kemudian data barang yang terdiri dari data barang masuk dan data barang keluar, dan laporan yang terdiri dari laporan keuangan dan laporan barang. Selain itu, terdapat fitur filter yang berfungsi menyaring data yang ingin ditampilkan berdasarkan tahun penginputannya.

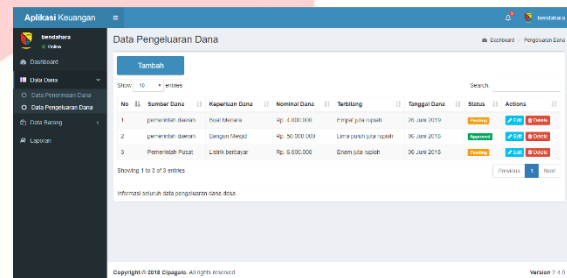
### 4.3 Implementasi Form Data Penerimaan Dana



Gambar 16. Implementasi Form Penerimaan Dana

Form data penerimaan dana digunakan untuk menampilkan dana yang masuk ke desa cipagalo, pada form tersebut terdapat data sumber dana, nominal dana masuk, terbilang, tanggal dana masuk, penerima dana masuk berserta buktinya, status data yang *approve* atau masih *pending*, dan aksi untuk menghapus dan mengedit data.

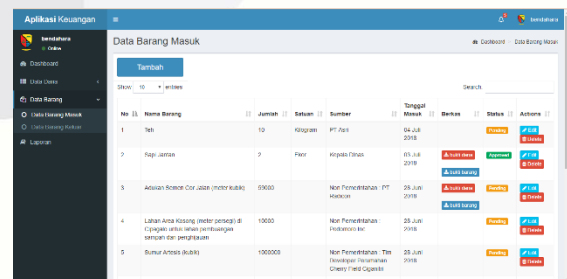
### 4.4 Implementasi Form Pengeluaran Dana



Gambar 17. Implementasi Form Pengeluaran Dana

Form data pengeluaran dana digunakan untuk menampilkan dana yang keluar dari desa cipagalo, pada form tersebut terdapat data sumber dana keluar, keperluan dana keluar, nominal dana keluar, terbilang, tanggal dana keluar, status data yang *approve* atau masih *pending*, dan aksi untuk menghapus dan mengedit data.

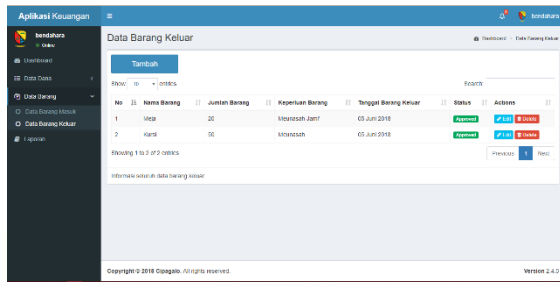
### 4.5 Implementasi Form Barang Masuk



Gambar 18. Implementasi Form Barang Masuk

Form data barang masuk digunakan untuk menampilkan barang yang masuk ke desa cipagalo, pada form tersebut terdapat data nama barang, jumlah barang, satuan barang, sumber barang masuk, tanggal barang masuk, berkas barang masuk, status data yang *approve* atau masih *pending*, dan aksi untuk menghapus dan mengedit data.

#### 4.6 Implementasi Form Barang Keluar



Gambar 19. Implementasi Form Barang Keluar

Form data barang keluar digunakan untuk menampilkan barang yang keluar dari desa cipagalo, pada form tersebut terdapat data nama barang, jumlah barang, keperluan barang, tanggal barang keluar, status data yang *approve* atau masih *pending*, dan aksi untuk menghapus dan mengedit data.

### 5 Kesimpulan dan Saran

#### 5.1 Kesimpulan

Selama melakukan analisis implementasi pada aplikasi dan pengujian dapat disimpulkan :

1. Aplikasi dapat membantu pegawai desa Cipagalo dalam mengelola data penerimaan dana, pengeluaran dana, pemasukan barang, dan pengeluaran barang di Desa Cipagalo.
2. Aplikasi dapat memberikan pemberitahuan berupa laporan akhir penggunaan dana kantor Desa Cipagalo.

#### 5.2 Saran

Berdasarkan hasil dari pembuatan proyek akhir ini, saran yang dapat dipertimbangkan untuk pengembangan kedepannya yaitu memberikan notifikasi kepada Kepala Desa bahwa Bendahara Desa telah melaksanakan tugasnya dan memberikan notifikasi bahwa laporan telah tercetak.

#### Daftar Pustaka:

- [1] "Wikipedia," 4 February 2016. [Online]. Available: [https://id.wikipedia.org/wiki/Cipagalo,\\_Bojongsong,\\_Bandung](https://id.wikipedia.org/wiki/Cipagalo,_Bojongsong,_Bandung).
- [2] "Desa". Indonesia Patent Pasal 1 angka 1 Undang-Undang Nomor 6/2014.
- [3] "Pemerintah Desa," Wikipedia, 10 Maret 2016. [Online]. Available: [https://id.wikipedia.org/wiki/Pemerintah\\_Desa](https://id.wikipedia.org/wiki/Pemerintah_Desa).
- [4] "Diskominfo Kabupaten Bandung," [Online]. Available: <http://www.cipagalo.desa.id/first/statistik/13>.
- [5] Asep, Interviewee, [Interview]. Agsutus 2017.
- [6] Agusdar, 13 April 2013. [Online]. Available: <https://agusdar.wordpress.com/2013/04/13/metode-pengembang-waterfall/>.
- [7] "Apa itu Dana Desa," 2016. [Online]. Available: <https://simperdededemak.wordpress.com/anggaran-dd-2/>.
- [8] D. P. d. Kebudayaan, Kamus besar Indonesia Edisi II, Jakarta: B Pustaka, 1999.
- [9] S. Ridwan, Manajemen Keuangan edisi ke lima, Jakarta: Literat Media, 2003.
- [10] Eko, Kupas Tuntas Microsoft Visual Basic 6.0., Jakarta: Elex M Komputindo, 2004.
- [11] Aswandi, Aplikasi Berbasis Web, Cirebon: Alex Media Komput 2006.
- [12] K. E. Kendall, System Analysis and Design, New Jersey: Prentice 2002.
- [13] M. Riadi, "Kajian Pustaka," 2012. [Online]. Available: <http://www.kajianpustaka.com/2013/12/use-case-diagram.html>.
- [14] Sutanta, Basis Data Dalam Tinjauan Konseptual, 2011.
- [15] B. Sidik, MySQL, Bandung: Informatika Bandung, 2005.
- [16] R. Abdulloh, Easy and Simple Web Programming, Tegal: Elex M Komputindo, 2016.
- [17] A. Saputra, in *Web Tips PHP, HTML5, dan CSS3*, Jasakom, 2011.
- [18] B. Sidik, Framework Codeigniter, Bandung: Informatika, 2012.
- [19] D. A. Hadi, "Pengertian dan Cara Menggunakan CodeIgniter," malasngoding.com, January 2016. [Online]. Available: <http://www.malasngoding.com/pengertian-dan-cara-menggunakan-codeigniter/>.
- [20] W. Sya'ban, Build Your Blogger XML Template, Yogyakarta: P Andi, 2010.
- [21] A. Nugroho, E-Commerce, Bandung: Informatika Bandung, 2000.
- [22] R. Pressman, Rekaya Perangkat Lunak Pendekatan Praktisi, 2000.

- [23] "Dana Bantuan Arti," 11 September 2017. [Online]. Available: <https://www.apaarti.com/dana-bantuan.html>.
- [24] A. Gregorius, Membuat Homepage Interaktif Dengan CGI/ Perl, Jakarta: PT. Elex Media Koputindo, 2000.
- [25] "Sapa Indonesia," Pangriptaloka, [Online]. Available: <http://www.sapa.or.id/b3/11323-kemiskinan-oooooiiip2>.
- [26] Supriyanto, Perancangan Aplikasi, Surabaya: Widyastana, 2005.
- [27] Munawar, Pemodelan Visual dengan UML, Yogyakarta: Graha Ilmu, 2005.
- [28] "Kantor Desa Dukuh," [Online]. Available: <https://kantordesadukuh.wordpress.com/tugas-dan-fungsi/>.
- [29] B. Sidik, "User Acceptance Test," 1 may 2006. [Online]. Available: <https://betha.wordpress.com/2006/05/01/user-acceptance-test/>.