***APPLIKASI PENGELOLA KEUANGAN PRIBADI MAHASISWA (ATURDUIT) BERBASIS MOBILE***

**Diajukan untuk memenuhi tugas UTS Peraktikum Rekayasa Perangkat Lunak Lanjut**

[*https://github.com/namaAkunGit/namaRepositoryProject*](https://github.com/namaAkunGit/namaRepositoryProject)



Oleh:

Nama : Yusman Abdurohman Hilmansyah

Nim : 1137050224

Kelas : IF/A

**Teknik Informatika**

**UIN Sunan Gunung Djati Bandung**

**2015**

# LATAR BELAKANG

Sikap konsumtif yang tinggi ini dikalangan mahasiswa dan generasi muda lainnya, menyebabkan pengelolaan keuangan menjadi sesuatu yang tidak mudah. Selain dari sikap tersebut, beberapa penelitian juga menunjukan bahwa generasi muda banyak yang belum memiliki pengetahuan akan pengelolaan keuangan. Jika generasi muda belum memiliki pengetahuan akan mengelola keuangan pribadi, maka mereka tidak dapat merencanakan dan mengendalikan penggunaan uang untuk pencapain tujuan individu mereka.

Pada cara konvensional kita sering memanajemen keuangan secara manual menggunakan jurnal keuangan, hal ini dirasa tidak efisien karena kita tidak mungkin membawa jurnal keuangan kita kemanapun kita berada dan juga kita tidak bisa secara cepat mencatat pengeluaran yang kita lakukan saat itu juga sehingga terkadang kita akan lupa akan pengeluaran kecil yang kita lakukan.

Berdasarkan dari latar belakang di atas, maka penulis mencoba mengembangkan teknologi aplikasi ponsel yang berbasis Android dengan membuat suatu aplikasi managemen keuangan. Dengan adanya aplikasi ini, para pengguna akan lebih mudah memanajemen keuangannya, sehingga pemasukkan dan pengeluaran keuangan dapat dimonitor setiap hari, bulan bahkan tahun. maka dengan aplikasi ponsel ini hal tersebut akan sangat mudah dilakukan karena semua hal tersebut bisa dilakukan hanya dalam satu genggaman. Sehingga dengan aplikasi mobile berbasis Android ini diharapkan mampu memanajemen keuangan kita secara lebih simple, cepat dan efisien.

# RUMUSAN MASALAH

Berdasarkan uraian di atas, rumusan masalah yang akan dipecahkan adalah bagaiamana merancang dan membuat sebuah applikasi berbasis mobile untuk manajemen keuangan pribadi yang sederhana, mudah digunakan, efektif, dan efisien.

# 3.TUJUAN

Tujuan dari pembuatan software ini adalah :

* 1. Untuk memenuhi salah satu tugas mata kuliah RPLL dan Praktimuk RPLL di Universitas Islam Negeri Sunan Gunung Djati Bandung.
  2. Mempelajari dan mengenal lebih dalam tentang pembuatan applikasi pengelola uang pribadi mahasiswa.
  3. Mengetahui lebih dalam tentang bahasa pemrograman java yang dipakai dalam pembuatan pengelola uang pribadi mahasiswa.
  4. Menghasilkan aplikasi manajemen keuangan pribadi dimana para pengguna dapat memantau kegiatan finansial mereka.

# 4. BATASAN MASALAH

Dalam penyusunan tugas besar ini, penulis membatasi ruang lingkup masalah yaitu :

* 1. Aplikasi ini berbasis mobile dan berjalan pada sistem opersai android (minimum V.4.42)
  2. Aplikasi ini bersifat offline
  3. Applikasi ini memiliki lima menu yaitu pemasukan, pengeluaran, utang , piutang dan laporan.

# DESKRIPSI PERANGKAT LUNKAK

5.1. Deskripsi Umum

*Applikasi* AturDuit merupakan sebuah *applikasi* yang berjalan pada platform android dan dikembangkan menggunakan tools eclipse. *Applikasi* ini cenderung seperti buku catatan transakasi finansial seseorang yang dikemas kedalam sebuah smartphone. Sehinga menggunaanya dirasa lebih mudah dan lebih praktis karena dewasa ini hampir semua orang tidak bisa lepas dari yang namanya smartphone.

Pada applikasi AturDuit ini terdapat 5 menu yang ditampilkan yaitu

* menu pencatatan pengeluaran
* menu pemasaukan
* menu utang
* menu piutang dan,
* menu laporan.

Pada menu pengeluaran user bisa menambah, mengedit dan menghapus catatan pengeluaran. Begitupun pada menu pemasukan, utang dan piutang, user bisa menambah, mengedit dan menghapus data yang dimasukan sesuai dengan transaksinya. Pada menu laporan, user bisa melihat catatan transaksi yang sudah dilakukan pada ke 4 menu sebelumnya. User bisa melihat catatan dengan opsi seminggu terahir, dua minggu terahir ataupun satu bulan terahir sesuai opsi yang diinginkan user.

## Pengguna

Semua pengguna perangkat lunak AturDuit hanya berperan sebagai user, tidak ada pengguna yang memiliki otorisasi khusus terhadap sistem. Sehingga semua user bersifat umum. Tidak ada keahlian khusus yang harus dipenuhi oleh pengguna selain dasar-dasar pengetahuan dalam mengoperasikan smartphone Android khususnya penggunaan keypad .

| **Kategori Pengguna** | **Tugas** | **Hak Akses ke aplikasi** |
| --- | --- | --- |
| User | Membuka Applikasi. | Menggunakan fungsi “open” |
| sigup | Menggunakan fungsi “sigup” |
| Log in | Menggunakan fungsi “login” |
| Melihat isi dari *Help Contents* yang berisi informasi tentang user guide applikasi. | Menggunakan fungsi “help” |
| Memilih menu pengeluaran | Menggunakan fungsi “pilih menu pengeluaran” |
| Memilih menu pemasukan | Menggunakan fungsi “pilih menu pemasukan” |
| Memilih menu utang | Menggunakan fungsi “pilih menu utang” |
| Memilih menu piutang | Menggunakan fungsi “pilih menu piutang” |
| Memasukan data | Menggunakan fungsi “tambah” |
| Mengedit data | Menggunakan fungsi “Edit” |
| Menghapus data | Menggunakan fungsi “Hapus” |
| Melihat Laporan Pencatatan | Menggunakan fungsi “laporang” |
|  | Menutup Applikasi | Menggunakan fungsi “exit” |

# REQUIREMENT

## Functional Requirement

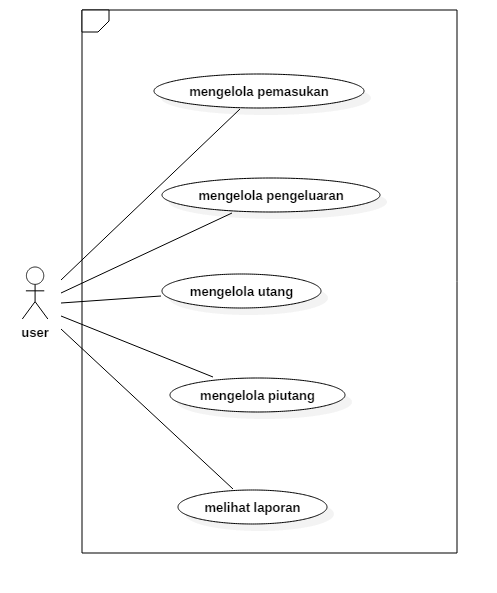
| **ID** | **Kebutuhan** | **Penjelasan** |
| --- | --- | --- |
| SKPL-AD.KF.001 | System dapat memulai applikasi | Sistem akan meminta semua data yang berhubungan dengan applikas, mulai dari grafik, suara, database dan lainnya. Semua data tersebut akan diproses dan ditampilkan langsung pada layar monitor |
| SKPL-AD.KF.002 | System dapat menampilkan form sig in | Applikasi AD dapat menampilkan form sig in, setelah user membuka applikasi, user harus memasukan user name dan password |
| SKPL-AD.KF.003 | System dapat menampilkan form sigup | Applikasi dapat menampilkan form sig up, jika user belum memiliki user\_id, maka harus membuat dulu pada form sig up. |
| SKPL-AD.KF.004 | Sistem dapat menampilkan pilihan menu. | Applikasi AD dapat menampilkan pilihan menu utama. |
| SKPL-AD.KF.005 | *Sys*tem dapat menampilkan *User Guide / Informasi Petunjuk penggunaan applikasi* | Applikasi AD dapat menampilkan pilihan bantuan. Dapat diakses user dari menu yang tersedia sebelum pencatatan dimulai. Jika user merasa bingung tentang penggunaan applikasi AD |
| SKPL-AD.KF.006 | *Sys*tem dapat menampilkanform pencatatan pengeluaran | Applikasi AD dapat menampilkan halaman pencatatan transaksi pengeluaran. Dapat diakses user setelah memilih menu applikasi. |
| SKPL-AD.KF.007 | *Sys*tem dapat menampilkanform pencatatan pengeluaran | Applikasi AD dapat menampilkan halaman pencatatan transaksi pengeluaran. Dapat diakses user setelah memilih menu applikasi. |
| SKPL-AD.KF.008 | *Sys*tem Dapat MenampilkanForm Pencatatan Pemasukan | Applikasi AD dapat menampilkan halaman pencatatan transaksi pemasukan. Dapat diakses user setelah memilih menu applikasi. |
| SKPL-AD.KF.009 | *Sys*tem dapat menampilkanform pencatatan utang | Applikasi AD dapat menampilkan halaman pencatatan transaksi utang. Dapat diakses user setelah memilih menu applikasi. |
| SKPL-AD.KF.0010 | *Sys*tem dapat menampilkanform pencatatan piutang | Applikasi AD dapat menampilkan halaman pencatatan transaksi piutang. Dapat diakses user setelah memilih menu applikasi. |
| SKPL-AD.KF.0011 | *Sys*tem dapat menampilkanform laporan | Applikasi AD dapat menampilkan halaman untuk melihat laporan. Dapat diakses user setelah memilih menu applikasi. |
| SKPL-AD.KF.0012 | *System dapat menambah data pencatatan transaksi* | Applikasi AD dapat menyimpan penambahan data transaksi, baik transaksi pengeluaran, pemasukan, utang maupun piutang. |
| SKPL-AD.KF.0013 | *System dapat mengedit data pencatatan transaksi* | Applikasi AD dapat menyimpan perubahan data transaksi, baik transaksi pengeluaran, pemasukan, utang maupun piutang. |
| SKPL-AD.KF.0014 | *System dapat menghapus data pencatatan transaksi* | Applikasi AD dapat menghapus data transaksi, baik transaksi pengeluaran, pemasukan, utang maupun piutang. |

* 1. Non Functional Requirement

| ID | Parameter | Kebutuhan |
| --- | --- | --- |
| SKPL-AD.KNF.001 | Availability | *harus bisa beroperasi 7 har per minggui, 24 jam per hari tanpa error* |
|  | Reliability | N/A |
|  | Ergonomy | N/A |
|  | Portability | N/A |
| SKPL-AD.KNF.002 | Memory | Ram Minimal 512 mb |
| SKPL-AD.KNF.003 | Response time | Applikasi harus mampu menampilkan laporan dalam 5 detik. |
|  | Safety | N/A |
| SKPL-AD.KNF.004 | Security | Applikasi tidak membatasi jumlah kesalahan pada saat login |
|  |  |  |
| SKPL-AD.KNF.005 | Others 1: Bahasa komunikasi | Bahasa komunikasi yang digunakan adalah bahasa inggris |
|  |  |  |
|  |  |  |

# Usecase

## Usecase Diagram



## Definisi Actor

|  |  |  |
| --- | --- | --- |
| No | Actor | Deskripsi |
| 1 | *User* | *Actor dengan role ini mempunyai wewenang untuk melakukan login, menambah data transaksi, mengedit data transaksi, menghapus data transaksi dan melihat laporan* |

## Definisi Usecase

|  |  |  |
| --- | --- | --- |
| No | Use Case | Deskripsi |
| 1 | *Mengelola pemasukan* | *Sistem menampilkan form menu pemasukan, di dalam menu pemasukan ini user bisa mengelola data seperti menambah, mengedit dan menghapus data transaksi pemasukan* |
| 2 | *Mengelola pengeluaran* | *Sistem menampilkan form menu pengeluaran, di dalam menu pengeluaran ini user bisa mengelola data seperti menambah, mengedit dan menghapus data transaksi pengeluaran* |
| 3 | *Mengelola utang* | *Sistem menampilkan form menu utang, di dalam menu utang ini user bisa mengelola data seperti menambah, mengedit dan menghapus data transaksi utang* |
| 4 | *Mengelola piutang* | *Sistem menampilkan form menu piutang, di dalam menu piutang ini user bisa mengelola data seperti menambah, mengedit dan menghapus data transaksi piutang* |
| 5 | *Melihat laporan* | *Sistem menampilkan form laporan / catatan transaksi yang sudah dilakukan, baik transaksi pemasukan, pengeluaran, utang maupun transaksi piutang* |
|  |  |  |
|  |  |  |

## Skenario Usecase

|  |  |
| --- | --- |
| **Nomor** | 001 |
| **Nama** | Mengolah pemasukan |
| **Tujuan** | Usecase ini digunakan untuk melakukan pengolahan pada pencatatan transaksi pemasukan kas. |
| **Deskripsi** | Pengolahan yang bisa dilakukan pada usecase ini adalah tambah data, edit data, dan hapus data. |
| **Actor** | user |
| **Prekondisi** | user berhasil login |
| **Scenario Utama : input data pemasukan** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. User membuka applikasi |  |
|  | 1. System menampilkan form login/signup |
| 1. User login |  |
|  | 1. System menampilkan form menu |
| 1. User memilih menu pemasukan |  |
|  | 1. System menampilkan form pemasukan |
| 1. User melilih pilihan “tambah” dan memasukan data pemasukan |  |
|  | 1. System menyimpan data ke database |
| **Scenario alternative: edit data pemasukan** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. User membuka applikasi |  |
|  | 1. System menampilkan form login/signup |
| 1. User login |  |
|  | 1. System menampilkan form menu |
| 1. User memilih menu pemasukan |  |
|  | 1. System menampilkan form pemasukan |
| 1. User melilih pilihan “ubah” dan merubah data pemasukan |  |
|  | 1. System menyimpan data ke database |
|  |  |
| **Scenario alternative: hapus data pemasukan** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. User membuka applikasi |  |
|  | 1. System menampilkan form login/signup |
| 1. User login |  |
|  | 1. System menampilkan form menu |
| 1. User memilih menu pemasukan |  |
|  | 1. System menampilkan form pemasukan |
| 1. User melilih pilihan “hapus” dan menghapus data pemasukan |  |
|  | 1. System menyimpan data ke database |
| **Postkondisi** | **System menampilkan notifikasi perubahan data** |

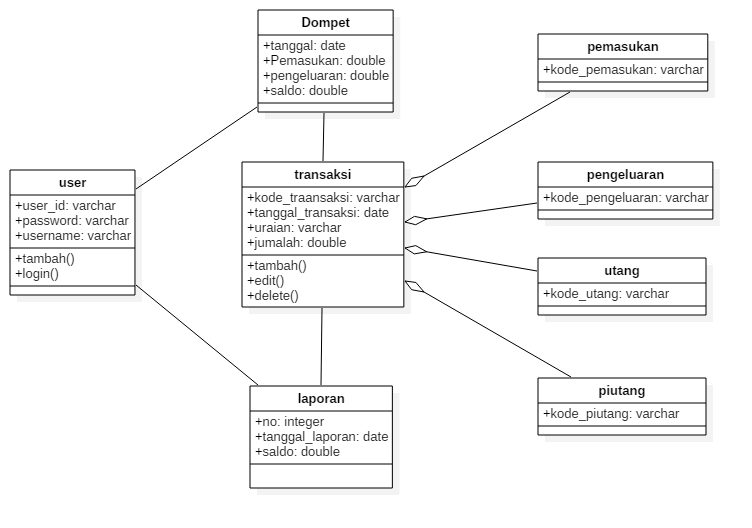
|  |  |
| --- | --- |
| **Nomor** | 002 |
| **Nama** | Mengolah pengeluaran |
| **Tujuan** | Usecase ini digunakan untuk melakukan pengolahan pada pencatatan transaksi pengeluaran kas. |
| **Deskripsi** | Pengolahan yang bisa dilakukan pada usecase ini adalah tambah data, edit data, dan hapus data. |
| **Actor** | user |
| **Prekondisi** | user berhasil login |
| **Scenario Utama : input data pengeluaran** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. User membuka applikasi |  |
|  | 1. System menampilkan form login/signup |
| 1. User login |  |
|  | 1. System menampilkan form menu |
| 1. User memilih menu pengeluaran |  |
|  | 1. System menampilkan form pengeluaran |
| 1. User melilih pilihan “tambah” dan memasukan data pengeluaran |  |
|  | 1. System menyimpan data ke database |
| **Scenario alternative: edit data pengeluaran** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. User membuka applikasi |  |
|  | 1. System menampilkan form login/signup |
| 1. User login |  |
|  | 1. System menampilkan form menu |
| 1. User memilih menu pengeluaran |  |
|  | 1. System menampilkan form pengeluaran |
| 1. User melilih pilihan “ubah” dan merubah data pengeluaran |  |
|  | 1. System menyimpan data ke database |
|  |  |
| **Scenario alternative: hapus data pengeluaran** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. User membuka applikasi |  |
|  | 1. System menampilkan form login/signup |
| 1. User login |  |
|  | 1. System menampilkan form menu |
| 1. User memilih menu pengeluaran |  |
|  | 1. System menampilkan form pengeluaran |
| 1. User melilih pilihan “hapus” dan menghapus data pengeluaran |  |
|  | 1. System menyimpan data ke database |
| **Postkondisi** | **System menampilkan notifikasi perubahan data** |

|  |  |
| --- | --- |
| **Nomor** | 003 |
| **Nama** | Mengolah utang |
| **Tujuan** | Usecase ini digunakan untuk melakukan pengolahan pada pencatatan transaksi utang |
| **Deskripsi** | Pengolahan yang bisa dilakukan pada usecase ini adalah tambah data, edit data, dan hapus data. |
| **Actor** | User |
| **Prekondisi** | user berhasil login |
| **Scenario Utama : input data utang** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. User membuka applikasi |  |
|  | 1. System menampilkan form login/signup |
| 1. User login |  |
|  | 1. System menampilkan form menu |
| 1. User memilih menu utang |  |
|  | 1. System menampilkan form utang |
| 1. User melilih pilihan “tambah” dan memasukan data utang |  |
|  | 1. System menyimpan data ke database |
| **Scenario alternative: edit data utang** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. User membuka applikasi |  |
|  | 1. System menampilkan form login/signup |
| 1. User login |  |
|  | 1. System menampilkan form menu |
| 1. User memilih menu utang |  |
|  | 1. System menampilkan form utang |
| 1. User melilih pilihan “ubah” dan merubah data utang |  |
|  | 1. System menyimpan data ke database |
|  |  |
| **Scenario alternative: hapus data utang** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. User membuka applikasi |  |
|  | 1. System menampilkan form login/signup |
| 1. User login |  |
|  | 1. System menampilkan form menu |
| 1. User memilih menu utang |  |
|  | 1. System menampilkan form utang |
| 1. User melilih pilihan “hapus” dan menghapus data utang |  |
|  | 1. System menyimpan data ke database |
| **Postkondisi** | **System menampilkan notifikasi perubahan data** |

|  |  |
| --- | --- |
| **Nomor** | 004 |
| **Nama** | Mengolah piutang |
| **Tujuan** | Usecase ini digunakan untuk melakukan pengolahan pada pencatatan transaksi piutang |
| **Deskripsi** | Pengolahan yang bisa dilakukan pada usecase ini adalah tambah data, edit data, dan hapus data. |
| **Actor** | User |
| **Prekondisi** | user berhasil login |
| **Scenario Utama : input data piutang** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. User membuka applikasi |  |
|  | 1. System menampilkan form login/signup |
| 1. User login |  |
|  | 1. System menampilkan form menu |
| 1. User memilih menu piutang |  |
|  | 1. System menampilkan form piutang |
| 1. User melilih pilihan “tambah” dan memasukan data piutang |  |
|  | 1. System menyimpan data ke database |
| **Scenario alternative: edit data piutang** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. User membuka applikasi |  |
|  | 1. System menampilkan form login/signup |
| 1. User login |  |
|  | 1. System menampilkan form menu |
| 1. User memilih menu piutang |  |
|  | 1. System menampilkan form piutang |
| 1. User melilih pilihan “ubah” dan merubah data piutang |  |
|  | 1. System menyimpan data ke database |
|  |  |
| **Scenario alternative: hapus data piutang** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. User membuka applikasi |  |
|  | 1. System menampilkan form login/signup |
| 1. User login |  |
|  | 1. System menampilkan form menu |
| 1. User memilih menu utang |  |
|  | 1. System menampilkan form utang |
| 1. User melilih pilihan “hapus” dan menghapus data piutang |  |
|  | 1. System menyimpan data ke database |
| **Postkondisi** | **System menampilkan notifikasi perubahan data** |

|  |  |
| --- | --- |
| **Nomor** | 005 |
| **Nama** | Melihat laporan |
| **Tujuan** | Usecase ini digunakan untuk melihat catatan transaksi yang sudah dilakukan sebelumnya |
| **Deskripsi** | User bisa melihat laporan pencatatan tiap transaksi (pengeluaran, pemasukan, piutang, utang) berdasarkan periode tertentu, missal : satu minggu terahir |
| **Actor** | User |
| **Prekondisi** | User berhasil login |
| **Scenario Utama : melihat laporan** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. User membuka applikasi |  |
|  | 1. System menampilkan form login/signup |
| 1. User login |  |
|  | 1. System menampilkan form menu |
| 1. User memilih menu laporan |  |
|  | 1. System menampilkan form laporan |
| 1. User melilih pilihan jenis transaksi |  |
|  | 1. System menampilkan pilihan periode |
| 1. User memilih pilihan periode |  |
|  | 1. System menampilkan rincian transaksi sesuai kriteria yang dimasukan user |
|  |  |

# Class Diagram

**

|  |  |  |
| --- | --- | --- |
| No | Nama Kelas | Jenis |
| 1 | User |  |
| 2 | Dompet |  |
| 3 | Transaksi |  |
| 4 | Laporan |  |
| 5 | Pemasukan |  |
| 6 | Pengeluaran |  |
| 7 | Utang |  |
| 8 | Piutang |  |