**DPPL-02**

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

Aplikasi Hemat.com

untuk:

Membantu Pengguna Mengatur Keuangannya

Dipersiapkan oleh:

Farhan Alfariqi/1301161770

Shella Gustiana/1301164693

Rima Rizita I/1301160781

Refka M. Furqon/ 1301162748

Muhammad Derian Akbar/1301164722

Program Studi Teknik Informatika

FIF – Telkom University

Jl. Telekomunikasi, Bandung 40257

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TU-logo-primer-memusat | **Prodi S1- Teknik Informatika**  **Universitas Telkom** | Nomor Dokumen | | Halaman |
| ***DPPL-02*** | | **58** |
| Revisi | *0* | *25 November 2018* |

DAFTAR PERUBAHAN

|  |  |
| --- | --- |
| Revisi | Deskripsi |
| A |  |
| B |  |
| C |  |
| D |  |
| E |  |
| F |  |
| G |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INDEX  TGL | - | A | B | C | D | E | F | G |
| Ditulis oleh |  |  |  |  |  |  |  |  |
| Diperiksa oleh |  |  |  |  |  |  |  |  |
| Disetujui oleh |  |  |  |  |  |  |  |  |

Daftar Halaman Perubahan

|  |  |  |  |
| --- | --- | --- | --- |
| Halaman | Revisi | Halaman | Revisi |
|  |  |  |  |

Daftar Isi

1 Pendahuluan 6

1.1 Tujuan Penulisan Dokumen 6

1.2 Lingkup Masalah 6

1.3 Definisi dan Istilah 6

1.4 Aturan Penamaan dan Penomoran 7

1.5 Referensi 7

1.6 Ikhtisar Dokumen 7

2 Deskripsi Perancangan Global 8

2.1 Rancangan Lingkungan Implementasi 8

2.2 Deskripsi Arsitektural 8

2.3 Deskripsi Komponen 9

3 Perancangan Rinci 10

3.1 Realisasi Use Case 10

3.1.1 Use Case Login 10

3.1.1.1 Identifikasi Kelas 10

3.1.1.2 Sequence Diagram 10

3.1.1.3 Diagram Kelas 11

3.1.2 Use Case Register 12

3.1.2.1 Identifikasi Kelas 12

3.1.2.2 Sequence Diagram 12

3.1.2.3 Diagram Kelas 12

3.1.3 Use Case Input Pemasukan 13

3.1.3.1 Identifikasi Kelas 13

3.1.3.2 Sequence Diagram 13

3.1.3.3 Diagram Kelas 13

3.1.4 Use Case Input Pengeluaran 14

3.1.4.1 Identifikasi Kelas 14

3.1.4.2 Sequence Diagram 14

3.1.4.3 Diagram Kelas 14

3.1.5 Use Case View Budget 15

3.1.5.1 Identifikasi Kelas 15

3.1.5.2 Sequence Diagram 15

3.1.5.3 Diagram Kelas 15

3.1.6 Use Case Input Barang 16

3.1.6.1 Identifikasi Kelas 16

3.1.6.2 Sequence Diagram 16

3.1.6.3 Diagram Kelas 16

3.1.7 Use Case Update Barang 17

3.1.7.1 Identifikasi Kelas 17

3.1.7.2 Sequence Diagram 17

3.1.7.3 Diagram Kelas 17

3.1.8 Use Case Delete Barang 18

3.1.8.1 Identifikasi Kelas 18

3.1.8.2 Sequence Diagram 18

3.1.8.3 Diagram Kelas 18

3.1.9 Use Case View Barang 19

3.1.9.1 Identifikasi Kelas 19

3.1.9.2 Sequence Diagram 19

3.1.9.3 Diagram Kelas 19

3.1.10 Use Case Input Hutang 20

3.1.10.1 Identifikasi Kelas 20

3.1.10.2 Sequence Diagram 20

3.1.10.3 Diagram Kelas 20

3.1.11 Use Case Update Hutang 21

3.1.11.1 Identifikasi Kelas 21

3.1.11.2 Sequence Diagram 21

3.1.11.3 Diagram Kelas 21

3.1.12 Use Case Delete Hutang 22

3.1.12.1 Identifikasi Kelas 22

3.1.12.2 Sequence Diagram 22

3.1.12.3 Diagram Kelas 22

3.1.13 Use Case View Hutang 23

3.1.13.1 Identifikasi Kelas 23

3.1.13.2 Sequence Diagram 23

3.1.13.3 Diagram Kelas 23

3.1.14 Use Case Update System 24

3.1.14.1 Identifikasi Kelas 24

3.1.14.2 Sequence Diagram 24

3.1.14.3 Diagram Kelas 24

3.2 Perancangan Detil Kelas 25

3.2.1 Kelas User 25

3.2.2 Kelas Login 26

3.2.3 Kelas Developer 26

3.2.4 Kelas Budget 27

3.2.5 Kelas Barang 27

3.2.6 Kelas Hutang 28

3.2.7 Kelas Registrasi 28

3.2.8 Kelas Database 29

3.2.9 Kelas Application 29

3.2.10 Kelas budgetController 29

3.2.11 Kelas barangController 30

3.2.12 Kelas loginController 30

3.2.13 Kelas userController 31

3.2.14 Kelas registrasiController 31

3.2.15 Kelas hutangController 32

3.2.16 Kelas developerController 32

3.2.17 Kelas budgetView 32

3.2.18 Kelas barangView 33

3.2.19 Kelas loginView 33

3.2.20 Kelas userView 33

3.2.21 Kelas registrasiView 34

3.2.22 Kelas hutangView 34

3.2.23 Kelas developerView 34

3.3 Diagram Kelas Keseluruhan 35

3.4 Algoritma/Query 36

3.5 Diagram Statechart 40

3.5.1 Statechart Budget 40

3.5.2 Statechart Login 40

3.6 Perancangan Antarmuka 41

3.7 Perancangan Representasi Persistensi Kelas 57

3.8 Perancangan Diagram Deployment 57

4. Matriks Kerunutan 58

# Pendahuluan

## Tujuan Penulisan Dokumen

Dokumen ini berisi penjelasan Deskripsi Perancangan Perangakat Lunak (DPPL) dengan pendekatan berorientasi objek dari perangkat lunak yang akan dibuat. Dokumen ini akan menjadi bahan acuan untuk membangun sistem pengaturan keuangan dan akan menjadi bahan evaluasi untuk sistem yang akan dibuat pada proses pembuatan ataupun pada saat akhir pengembangan sistem yang dibuat. Dengan dibuatnya dokumen ini dapat memudahkan dibangunnya aplikasi ini dan sesuai dengan keinginan pengguna.

## Lingkup Masalah

Hemat.com merupakan perangkat lunak yang memudahkan para user untuk mengelola keuangan user itu sendiri, berorientasi objek untuk monitoring atau memantau setiap transaksi yang telah dilaksanakan agar setiap uang yang masuk dan keluar menjadi lebih rapi dan padu sehingga tidak terjadi lagi kekurangan yang tidak diinginkan dan agar para user bisa memiliki barang impian sesuai dengan yang user inginkan. Sasaran membuat aplikasi ini sendiri lebih kepada pengguna yang sudah cukup umur dan yang sudah berpenghasilan.

## Definisi dan Istilah

|  |  |
| --- | --- |
| Istilah | Penjelasan |
| Berorientasi Objek | Merupakan paradigma pemrograman yang berorientasi objek |
| DPPL | Singkatan dari Deskripsi Perangcangan Perangkat Lunak yaitu dokumen yang mendeskripsikan dan menjabarkan secara tereperinci mengenai perancangan perangkat lunak yang akan dibangun. |
| Use Case | Use Case merupakan merepresentasikan sebuah interaksi antara aktor (user atau sistem lainya) dengan sistem. |
| User | Pengguna aplikasi tersebut. |
| Hemat.com | Merupakan nama aplikasi tersebut. |
| Java | Bahasa pemograman tingkat tinggi yang berorientasi objek dan program java tersusun dari bagian yang disebut kelas. |
| Input | Masukan |
| Oracle | Basis data relasional yang terdiri dari kumpulan data dalam suatu sistem manajemen basis data RDBMS. |
| Controller | Sebuah file class yang berujuan dapat berhubungan dengan sebuah URI. |
| View | Bagian yang mengatur tampilan panggilan ke pengguna. |
| Model | Memanipulasi data |
| Sequence | Serangkaian scene atau shot-shot, yang merupakan suatu kesatuan utuh |
| Deployment | Salah satu model diagram dalam UML untuk mengerahkan artifact dalam node. |
| GUI | Jenis antarmuka pengguna yang menggunakan metode interaksi pada peranti elektronik secara grafis (bukan perintah teks) antara pengguna dan komputer. |

## Aturan Penamaan dan Penomoran

* Penomoran Algoritma adalah Algo- diikuti dengan 3 angka dan diawali dengan Algo-001.
* Penomoran Query adalah Query- diikuti dengan 3 angka dan diawali dengan Query-001.

## Referensi

Dokumen-dokumen yang digunakan sebagai referensi dalam pembuatan DPPL ini adalah sebagai berikut :

1. Template SKPL Fakultas Informatika, Telkom University.
2. [https://www.sinaryuda.web.id/codeigniter/apa-sih-controller-itu.htm*l*](https://www.sinaryuda.web.id/codeigniter/apa-sih-controller-itu.html)
3. <https://situsali.com/konsep-dasar-mvc-model-view-controller-dari-php/>

## Ikhtisar Dokumen

Terdapat empat bagian dari penyusunan dokumen DPPL ini. Dalam bagian pertama terdapat pendahuluan yang berisi tentang dokumen DPPL yang mencakup tujuan dari pembuatan dokumen ini, lingkup masalah, definisi dan istilah yang terpakai dari pembuatan dokumen ini, aturan penamaan dan penomoran, referensi, dan ikhtisar dokumen. Pada bagian kedua terdapat penjelasan secara umum mengenasi perangkat lunak yang akan dibangun yaitu, rancangan lingkungan implementasi, deskripsi arsitektural, dan deskripsi komponen.

Pada bagian ketiga terdapat perancangan rinci yang berisi tentang realiasi use case, perancangan detail kelas, diagram antar kelas keseluruhan, algoritma, diagram state chart, perancangan antarmuka dan rancangan representasi persintensi kelas. Pada bagian terakhir, bagian empat terdapat matriks kerunutan pada dokumen DPPL.

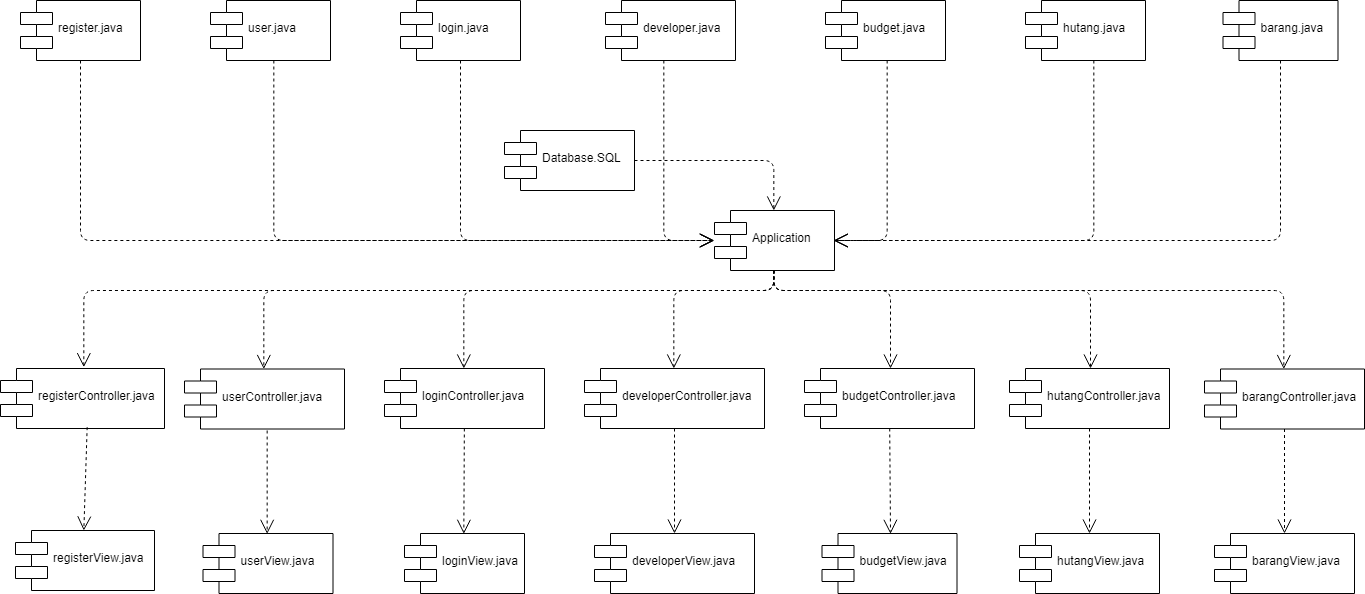
# Deskripsi Perancangan Global

## Rancangan Lingkungan Implementasi

Sistem ini menggunakan lingkup implementasi diantaranya :

1. Operating system : Android versi Marshmallow
2. DBMS : Oracle
3. Development tools : Android Studio
4. Bahasa pemrograman : Java

## Deskripsi Arsitektural



## Deskripsi Komponen

| **No** | **Nama Komponen** | **Keterangan** |
| --- | --- | --- |
|  | user.java | File kelas model user |
|  | login.java | File kelas model login |
|  | developer.java | File kelas model developer |
|  | budget.java | File kelas model budget |
|  | hutang.java | File kelas model hutang |
|  | barang.java | File kelas model barang |
|  | register.java | File kelas model register |
|  | database.SQL | File database |
|  | application.java | File kelas application/kumpulan model |
|  | userController.java | File controller user |
|  | loginController.java | File controller login |
|  | developerController.java | File controller developer |
|  | budgetController.java | File controller budget |
|  | hutangController.java | File controller hutang |
|  | barangController.java | File controller barang |
|  | registerController.java | File controller register |
|  | userView.java | File view java |
|  | loginView.java | File view login |
|  | developerView.java | File view developer |
|  | budgetView.java | File view budget |
|  | hutangView.java | File view hutang |
|  | barangView.java | File view barang |
|  | registerView.java | File view register |

# Perancangan Rinci

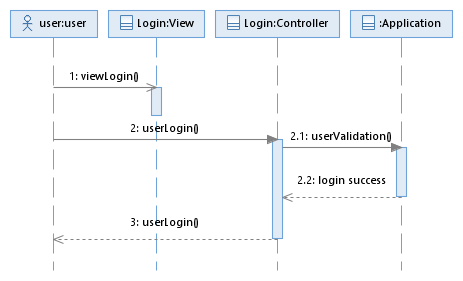
## Realisasi Use Case

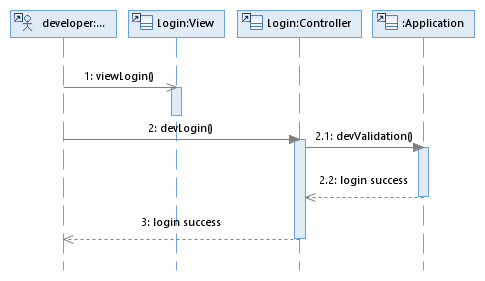
### Use Case Login

#### Identifikasi Kelas

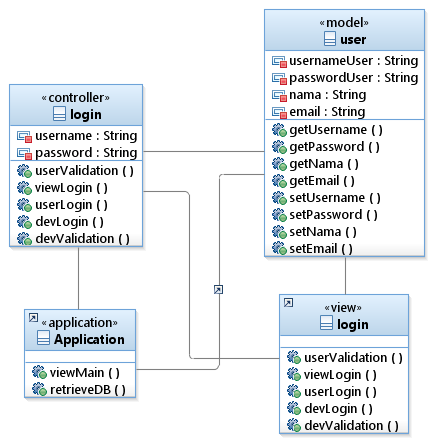
|  |  |  |
| --- | --- | --- |
| *No* | *Nama Kelas Perancangan* | *Nama Kelas Analisis Terkait* |
| *1* | *Application* |  |
| *2* | *controllerLogin* |  |
| *3* | *viewLogin* |  |
| *4* | *user* |  |

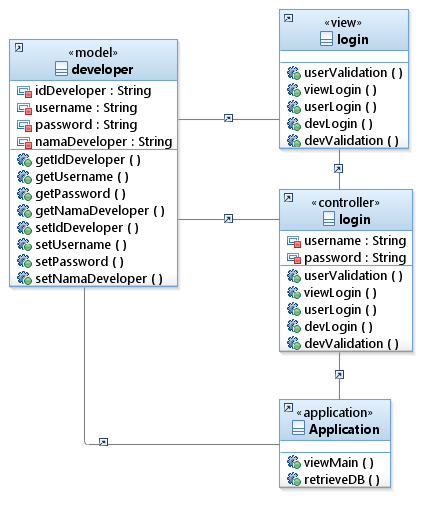
#### Sequence Diagram





#### Diagram Kelas



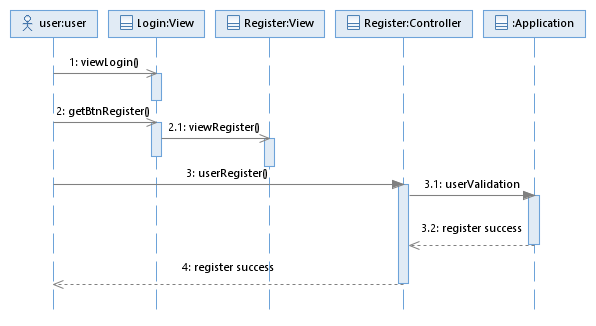


### Use Case Register

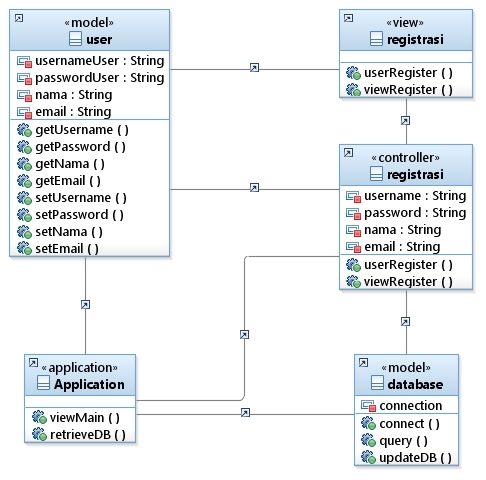
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Kelas Perancangan* | *Nama Kelas Analisis Terkait* |
| *1* | *Application* |  |
| *2* | *user* |  |
| *3* | *viewLogin* |  |
| *4* | *viewRegister* |  |
| *5* | *controllerRegister* |  |

#### Sequence Diagram



#### Diagram Kelas

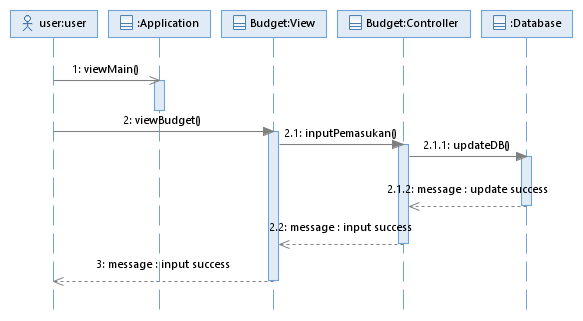


### Use Case Input Pemasukan

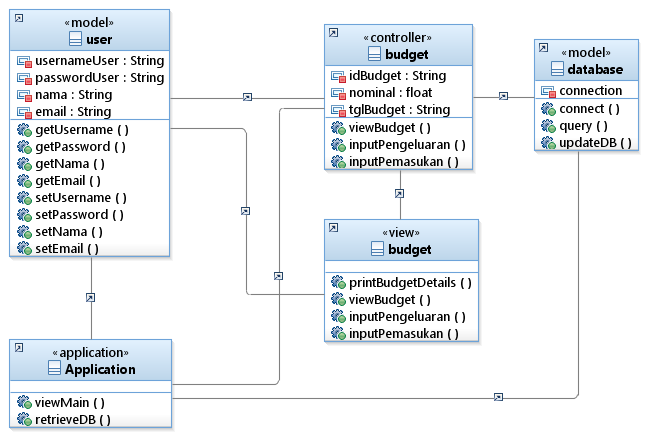
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Kelas Perancangan* | *Nama Kelas Analisis Terkait* |
| *1* | *Application* |  |
| *2* | *user* |  |
| *3* | *viewBudget* |  |
| *4* | *controllerBudget* |  |
| *5* | *Database* |  |

#### Sequence Diagram



#### Diagram Kelas

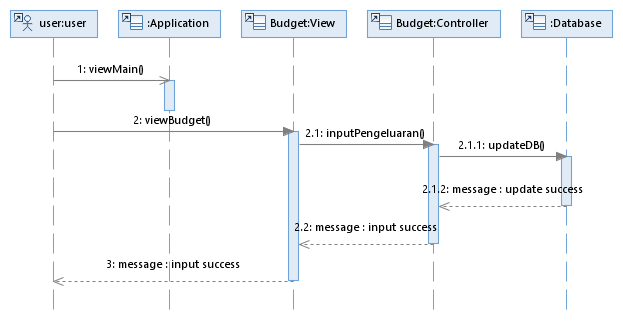


### Use Case Input Pengeluaran

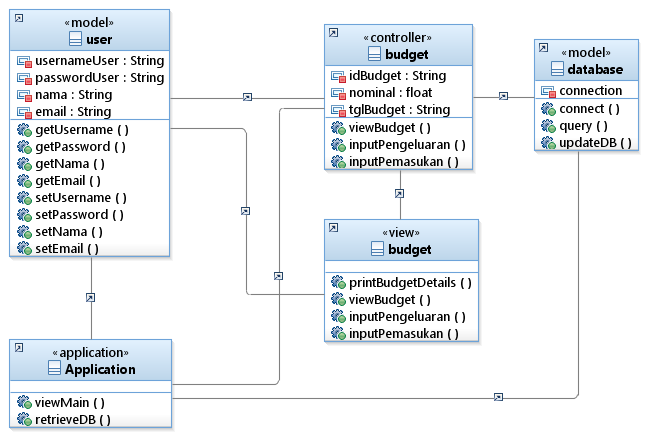
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Kelas Perancangan* | *Nama Kelas Analisis Terkait* |
| *1* | *Application* |  |
| *2* | *user* |  |
| *3* | *viewBudget* |  |
| *4* | *controllerBudget* |  |
| *5* | *Database* |  |

#### Sequence Diagram



#### Diagram Kelas

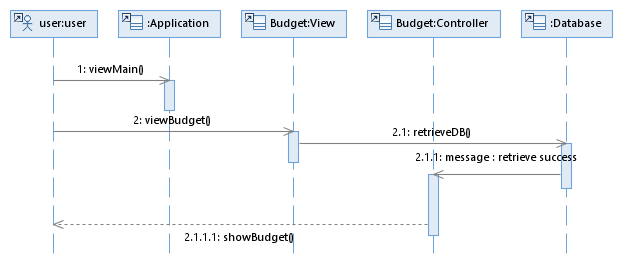


### Use Case View Budget

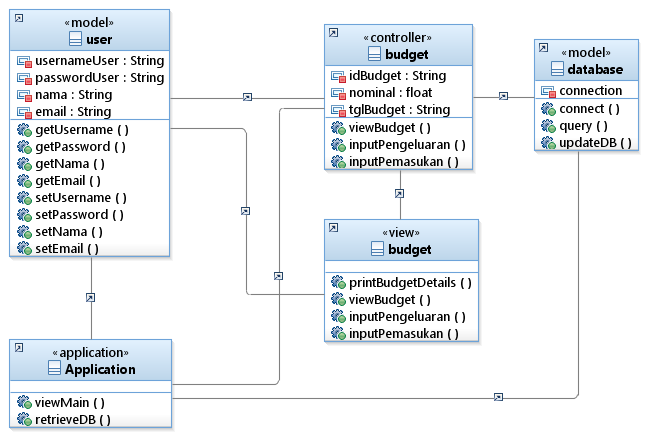
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Kelas Perancangan* | *Nama Kelas Analisis Terkait* |
| *1* | *Application* |  |
| *2* | *user* |  |
| *3* | *viewBudget* |  |
| *4* | *controllerBudget* |  |
| *5* | *Database* |  |

#### Sequence Diagram



#### Diagram Kelas

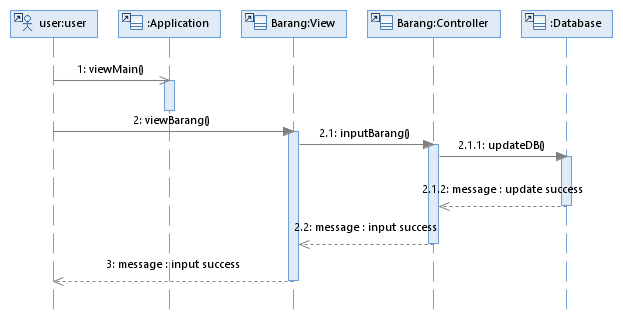


### Use Case Input Barang

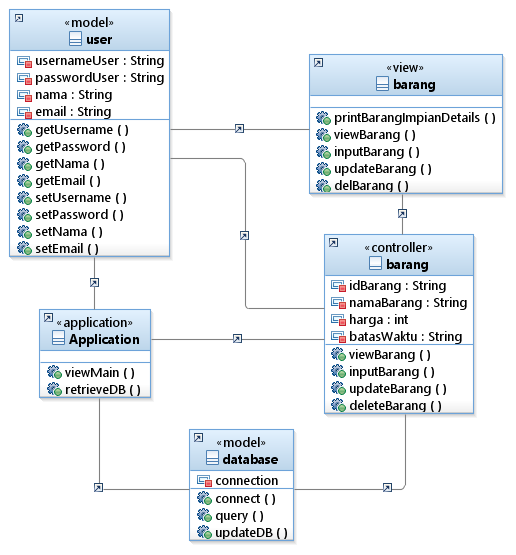
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Kelas Perancangan* | *Nama Kelas Analisis Terkait* |
| *1* | *Application* |  |
| *2* | *user* |  |
| *3* | *viewBarang* |  |
| *4* | *controllerBarang* |  |
| *5* | *Database* |  |

#### Sequence Diagram



#### Diagram Kelas

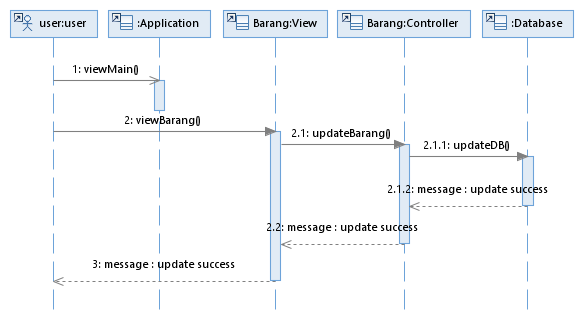


### Use Case Update Barang

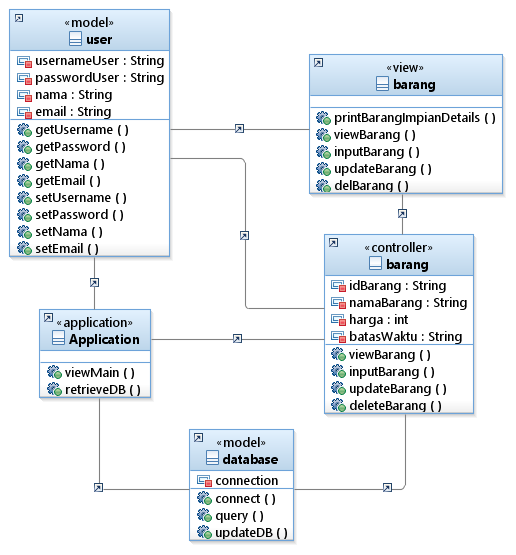
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Kelas Perancangan* | *Nama Kelas Analisis Terkait* |
| *1* | *Application* |  |
| *2* | *user* |  |
| *3* | *viewBarang* |  |
| *4* | *controllerBarang* |  |
| *5* | *Database* |  |

#### Sequence Diagram



#### Diagram Kelas

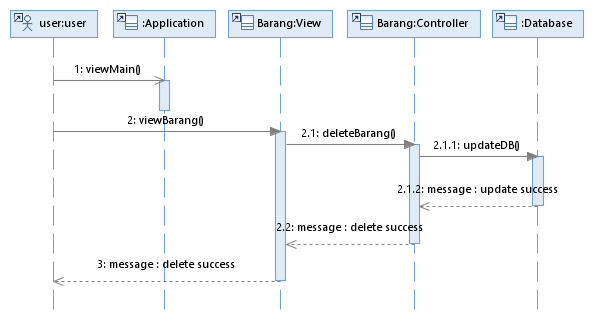


### Use Case Delete Barang

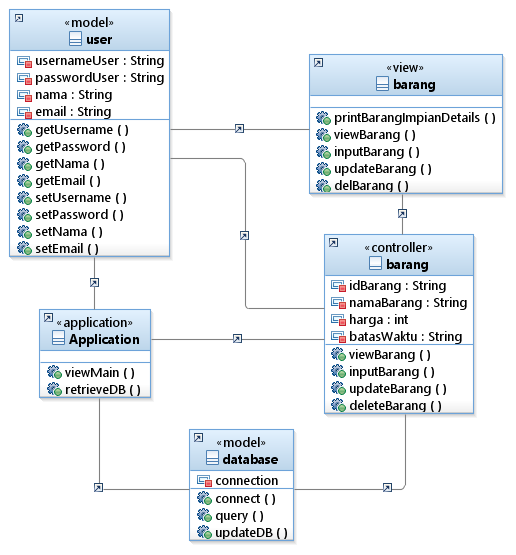
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Kelas Perancangan* | *Nama Kelas Analisis Terkait* |
| *1* | *Application* |  |
| *2* | *user* |  |
| *3* | *viewBarang* |  |
| *4* | *controllerBarang* |  |
| *5* | *Database* |  |

#### Sequence Diagram



#### Diagram Kelas

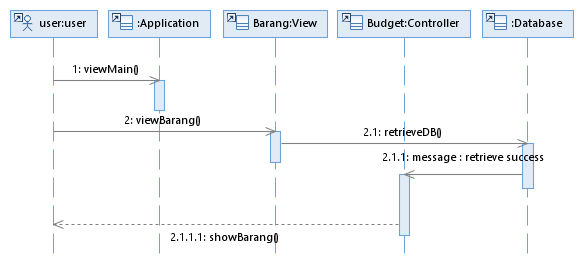


### Use Case View Barang

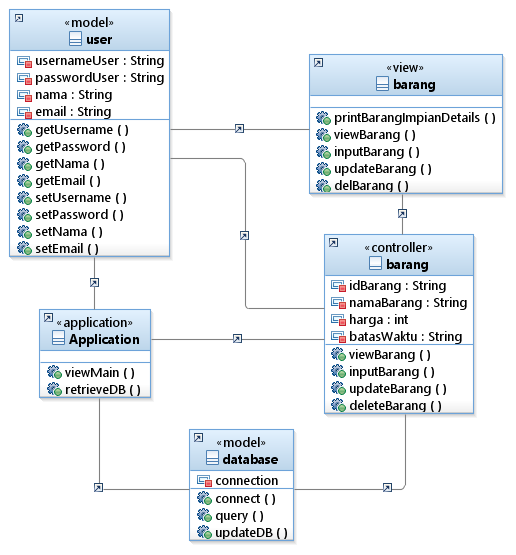
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Kelas Perancangan* | *Nama Kelas Analisis Terkait* |
| *1* | *Application* |  |
| *2* | *user* |  |
| *3* | *viewBarang* |  |
| *4* | *controllerBarang* |  |
| *5* | *Database* |  |

#### Sequence Diagram



#### Diagram Kelas

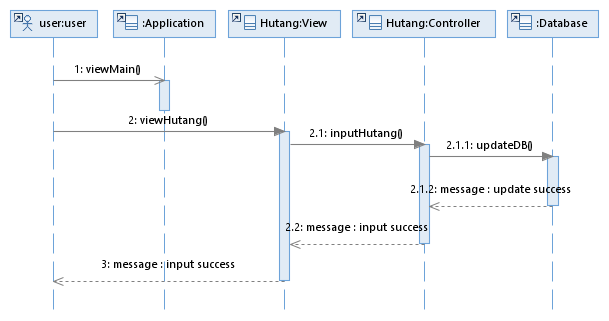


### Use Case Input Hutang

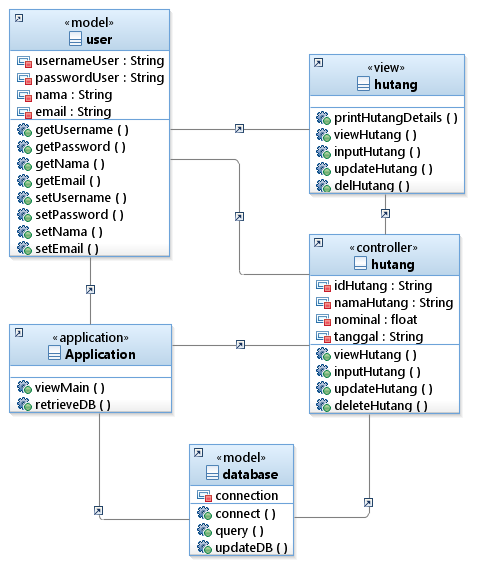
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Kelas Perancangan* | *Nama Kelas Analisis Terkait* |
| *1* | *Application* |  |
| *2* | *user* |  |
| *3* | *viewHutang* |  |
| *4* | *controllerHutang* |  |
| *5* | *Database* |  |

#### Sequence Diagram



#### Diagram Kelas

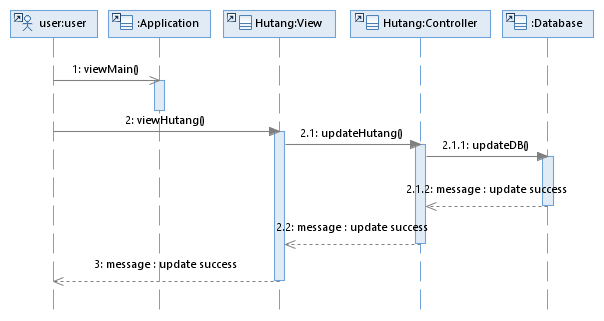


### Use Case Update Hutang

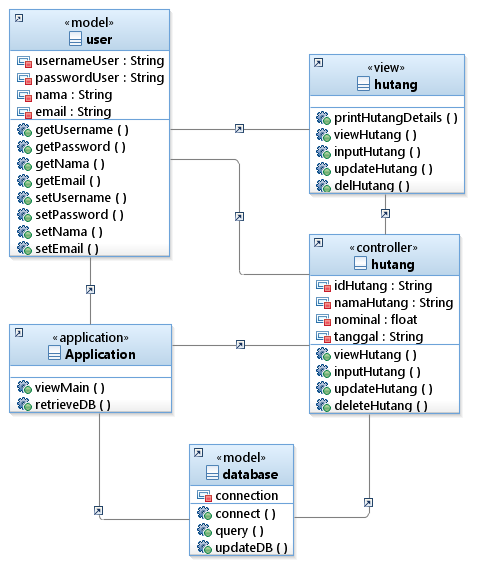
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Kelas Perancangan* | *Nama Kelas Analisis Terkait* |
| *1* | *Application* |  |
| *2* | *user* |  |
| *3* | *viewHutang* |  |
| *4* | *controllerHutang* |  |
| *5* | *Database* |  |

#### Sequence Diagram



#### Diagram Kelas

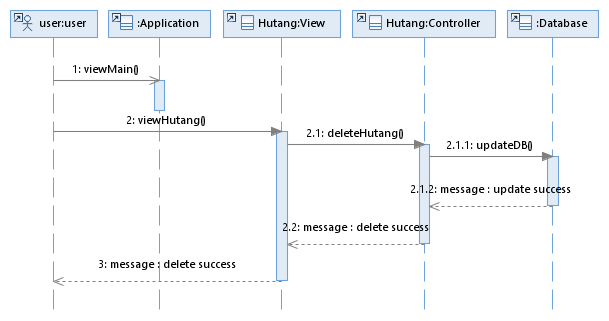


### Use Case Delete Hutang

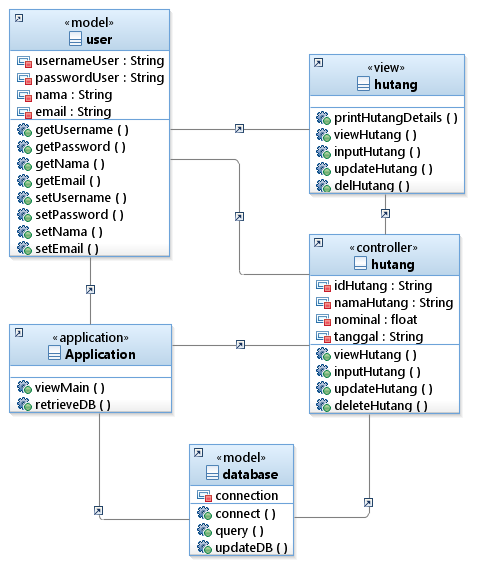
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Kelas Perancangan* | *Nama Kelas Analisis Terkait* |
| *1* | *Application* |  |
| *2* | *user* |  |
| *3* | *viewHutang* |  |
| *4* | *controllerHutang* |  |
| *5* | *Database* |  |

#### Sequence Diagram



#### Diagram Kelas

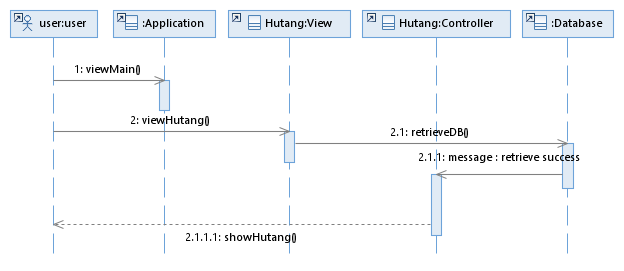


### Use Case View Hutang

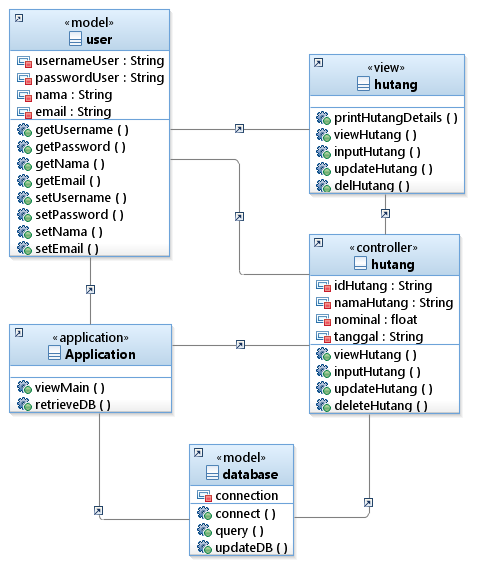
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Kelas Perancangan* | *Nama Kelas Analisis Terkait* |
| *1* | *Application* |  |
| *2* | *user* |  |
| *3* | *viewHutang* |  |
| *4* | *controllerHutang* |  |
| *5* | *Database* |  |

#### Sequence Diagram



#### Diagram Kelas

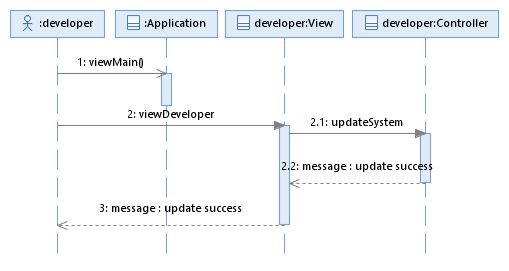


### Use Case Update System

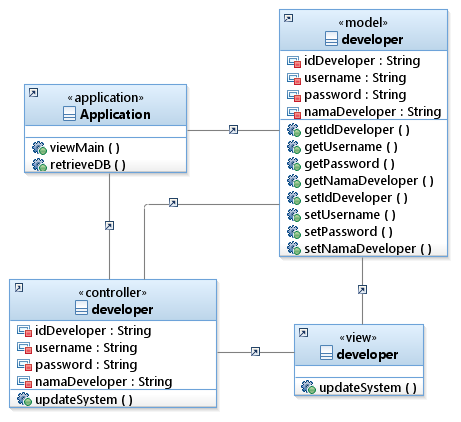
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Kelas Perancangan* | *Nama Kelas Analisis Terkait* |
| *1* | *Application* |  |
| *2* | *developer* |  |
| *3* | *controllerDeveloper* |  |
| *4* | *viewDeveloper* |  |

#### Sequence Diagram



#### Diagram Kelas



## Perancangan Detil Kelas

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Kelas Perancangan* | *Nama Kelas Analisis Terkait* |
| *1* | user | user |
| *2* | login | login |
| *3* | developer | developer |
| *4* | budget | budget |
| *5* | hutang | hutang |
| *6* | barang | barang |
| *7* | register | register |
| *8* | database |  |
| *9* | application |  |
| *10* | userController |  |
| *11* | loginController |  |
| *12* | developerController |  |
| *13* | budgetController |  |
| *14* | hutangController |  |
| *15* | barangController |  |
| *16* | registerController |  |
| *17* | userView |  |
| *18* | loginView |  |
| *19* | developerView |  |
| *20* | budgetView |  |
| *21* | hutangView |  |
| *22* | barangView |  |
| *23* | registerView |  |

### Kelas User

*Nama Kelas : user*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| setUsername() | Public | Method Set |
| getUsername() | Public | Method Get |
| setPassword() | Public | Method Set |
| getPassword() | Public | Method Get |
| setNama() | Public | Method Set |
| getNama() | Public | Method Get |
| setEmail() | Public | Method Set |
| getEmail() | Public | Method Get |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| Username | Private | String |
| password | Private | String |
| Nama | Private | String |
| Email | Private | String |

### Kelas Login

*Nama Kelas : login*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| setUsername() | Public | Method Set |
| getUsername() | Public | Method Get |
| setPassword() | Public | Method Set |
| getPassword() | Public | Method Get |
| setNama() | Public | Method Set |
| getNama() | Public | Method Get |
| setEmail() | Public | Method Set |
| getEmail() | Public | Method Get |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| usernameUser | Private | String |
| passwordUser | Private | String |
| Nama | Private | String |
| Email | Private | String |

### Kelas Developer

*Nama Kelas : developer*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| setIdDeveloper() | Public | Method Set |
| getIdDeveloper() | Public | Method Get |
| setUsername() | Public | Method Set |
| getUsername() | Public | Method Get |
| setPassword() | Public | Method Set |
| getPassword() | Public | Method Get |
| setNamaDeveloper() | Public | Method Set |
| getNamaDeveloper() | Public | Method Get |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| idDeveloper | Private | String |
| username | Private | String |
| password | Private | String |
| namaDeveloper | Private | String |

### Kelas Budget

*Nama Kelas : budget*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| setIdPemasukan() | Public | Method Set |
| getIdPemasukan() | Public | Method Get |
| setIdPengeluaran() | Public | Method Set |
| getIdPengeluaran() | Public | Method Get |
| setTglBudget() | Public | Method Set |
| getTglBudget() | Public | Method Get |
| setNominal() | Public | Method Set |
| getNominal() | Public | Method Get |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| idPemasukan | Private | String |
| idPengeluaran | Private | String |
| tglBudget | Private | Date |
| nominal | Private | Integer |

### Kelas Barang

*Nama Kelas : barang*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| setIdBarang() | Public | Method Set |
| getIdBarang() | Public | Method Get |
| setNamaBarang() | Public | Method Set |
| getNamaBarang() | Public | Method Get |
| setHargat() | Public | Method Set |
| getHarga() | Public | Method Get |
| setbatasWaktu() | Public | Method Set |
| getbatasWaktu() | Public | Method Get |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| idBarang | Private | String |
| namaBarang | Private | String |
| Harga | Private | Integer |
| batasWaktu | Private | Date |

### Kelas Hutang

*Nama Kelas : hutang*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| setIdHutang() | Public | Method Set |
| getIdHutang() | Public | Method Get |
| setNamaHutang() | Public | Method Set |
| getNamaHutang() | Public | Method Get |
| setNominal() | Public | Method Set |
| getNominal() | Public | Method Get |
| setTanggal() | Public | Method Set |
| getTanggal() | Public | Method Get |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| idHutang | Private | String |
| namaHutang | Private | String |
| Nominal | Private | Integer |
| Tanggal | Private | Date |

### Kelas Registrasi

*Nama Kelas : registrasi*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| setUsername() | Public | Method Set |
| getUsername() | Public | Method Get |
| setPassword() | Public | Method Set |
| getPassword() | Public | Method Get |
| setNama() | Public | Method Set |
| getNama() | Public | Method Get |
| setEmail() | Public | Method Set |
| getEmail() | Public | Method Get |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| Username | Private | String |
| Password | Private | String |
| Nama | Private | String |
| Email | Private | String |

### Kelas Database

*Nama Kelas : database*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| setConnect() | Public | Method Set |
| getConnect() | Public | Method Get |
| setQuery() | Public | Method Set |
| getQuery() | Public | Method Get |
| setUpdateDB() | Public | Method Set |
| getUpdateDB() | Public | Method Get |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| Connection | Private | Boolean |

### Kelas Application

*Nama Kelas : application*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| viewMain() | Public | Menampilkan Halaman Utama Aplikasi |
| RetrieveDB() | Public | Mengambil data dari Database |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *-* | *-* | *-* |

### Kelas budgetController

*Nama Kelas : budgetController*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| viewBudget() | Public | Menampilkan data Budget |
| inputPengeluaran() | Public | Menambahkan data Pengeluaran |
| inputPemasukan() | Public | Menambahkan data Pemasukan |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| idBudget | Private | String |
| Nominal | Private | Integer |
| tglBudget | Private | Date |

### Kelas barangController

*Nama Kelas : barangController*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| viewBarang() | Public | Menampilkan data Barang |
| inputBarang() | Public | Menambahkan data Barang |
| updateBarang() | Public | Melakukan perubahan pada data Barang |
| deleteBarang() | Public | Melakukan penghapusan pada data Barang |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| idBarang | Private | String |
| namaBarang | Private | String |
| Harga | Private | Integer |
| bataswaktu | Private | String |

### Kelas loginController

*Nama Kelas : loginController*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| userValidation() | Public | Melakukan pengecekan data User |
| viewLogin() | Public | Menampilkan form Login |
| userLogin() | Public | Melakukan proses login (user) |
| devLogin() | Public | Melakukan proses login (developer) |
| devValidation() | Public | Melakukan pengecekan data Developer |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| Username | Private | String |
| Password | Private | String |

### Kelas userController

*Nama Kelas : userController*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| viewBudget() | Public | Menampilkan data Budget |
| inputPengeluaran() | Public | Menambahkan data Pengeluaran |
| inputPemasukan() | Public | Menambahkan data Pemasukan |
| inputBarang() | Public | Menambahkan data Barang |
| updateBarang() | Public | Melakukan perubahan pada data Barang |
| viewBarang() | Public | Menampilkan data Barang |
| delBarang() | Public | Melakukan penghapusan data Barang |
| inputBarang() | Public | Menambahkan Data |
| updateBarang() | Public | Melakukan Perubahan Pada Data Barang |
| viewHutang() | Public | Menampilkan Data Hutang |
| delHutang() | Public | Melakukan Penghapusan Data Hutang |
| viewApplicationmenu() | Public | Menampilkan Halaman Utama Aplikasi |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| usernameUser | Private | String |
| passwordUser | Private | String |
| Nama | Private | String |
| Email | Private | String |

### Kelas registrasiController

*Nama Kelas : registrasiController*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| userRegister() | Public | Melakukan proses Registrasi (user) |
| viewRegister() | Public | Menampilkan form Registrasi |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| Username | Private | String |
| Password | Private | String |
| Nama | Private | String |
| Email | Private | String |

### Kelas hutangController

*Nama Kelas : hutangController*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| viewHutang() | Public | Menampilkan data Hutang |
| inputHutang() | Public | Menambahkan data Hutang |
| updateHutang() | Public | Melakukan Perubahan Pada data Hutang |
| deleteHutang() | Public | Melakukan Penghapusan Pada data Hutang |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| idHutang | Private | String |
| namaHutang | Private | String |
| Nominal | Private | Float |
| Tanggal | Private | Date |

### Kelas developerController

*Nama Kelas : developerController*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| updateSystem() | Public | Melakukan perubahan Pada System Aplikasi |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| idDeveloper | Private | String |
| Username | Private | String |
| Password | Private | String |
| namaDeveloper | Private | String |

### Kelas budgetView

*Nama Kelas : budgetView*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| printBudgetDetails() | Public | Menampilkan informasi Tentang Budget |
| viewBudget() | Public | Menampilkan data Budget |
| inputPengeluaran() | Public | Menambahkan data Pengeluaran |
| inputPemasukan() | Public | Menambahkan data Pemasukan |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *-* | *-* | *-* |

### Kelas barangView

*Nama Kelas : barangView*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| printBarangImpianDetails() | Public | Menampilkan informasi mengenai Barang Impian |
| viewBarang() | Public | Menampilkan data Barang |
| inputBarang() | Public | Menambahkan data Barang |
| updateBarang() | Public | Melakukan Perubahan pada data Barang |
| delBarang() | Public | Melakukan Penghapusan data Barang |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *-* | *-* | *-* |

### Kelas loginView

*Nama Kelas : loginView*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| userValidation() | Public | Melakukan pengecekan Akun User |
| viewLogin() | Public | Menampilkan from Login |
| userLogin() | Public | Melakukan Login (user) |
| devLogin() | Public | Melakukan Login (developer) |
| devValidation() | Public | Melakukan pengecekan Akun Developer |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *-* | *-* | *-* |

### Kelas userView

*Nama Kelas : userView*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| printUserDetail() | Public | Menampilkan informasi Tentang User |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *-* | *-* | *-* |

### Kelas registrasiView

*Nama Kelas : registrasiView*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| userRegistrasi() | Public | Melakukan Registrasi |
| viewRegistrasi() | Public | Menampilkan form Registrasi |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *-* | *-* | *-* |

### Kelas hutangView

*Nama Kelas : hutangView*

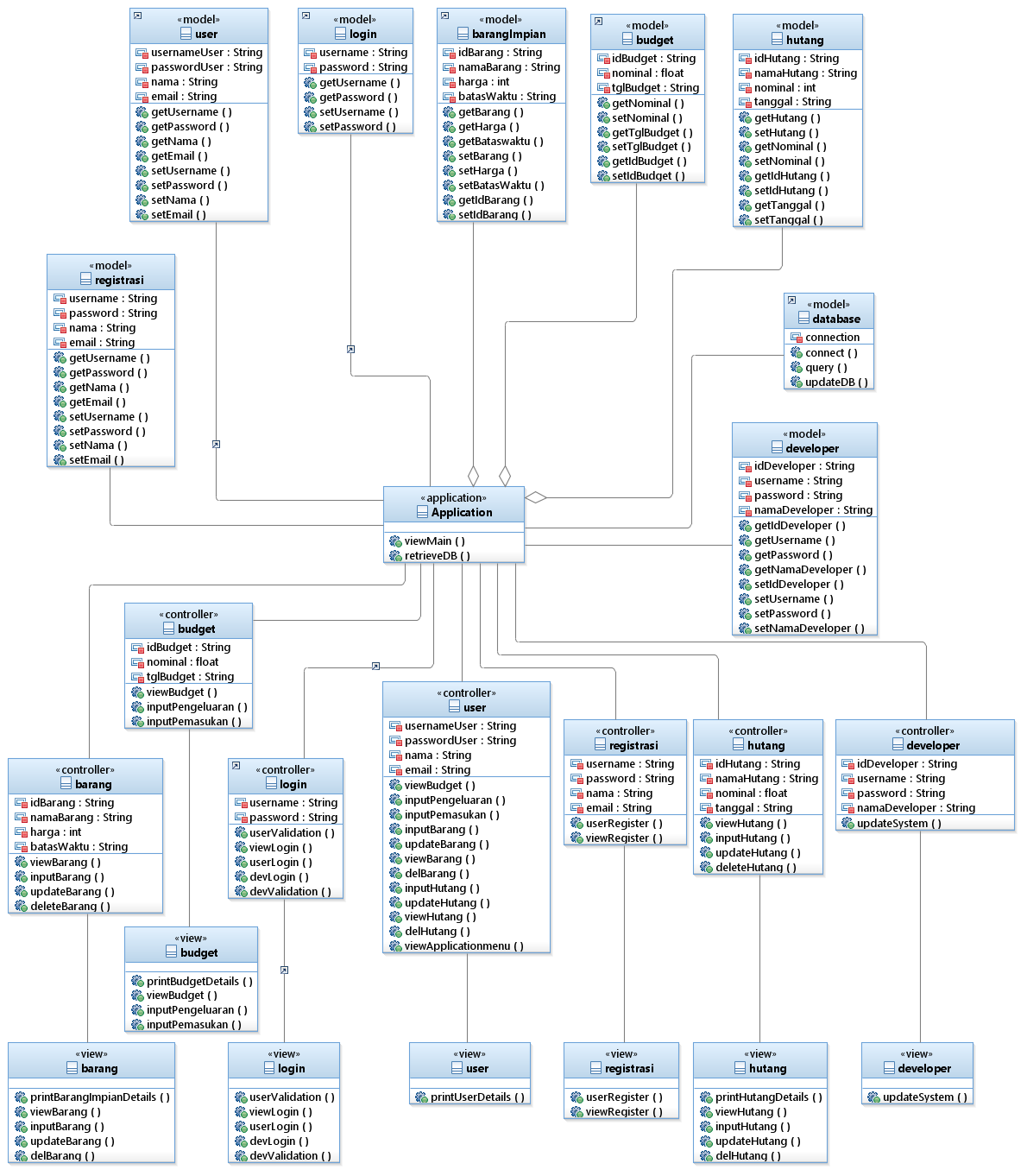
|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| printHutangDetails() | Public | Menampilkan informasi Hutang |
| viewHutang() | Public | Masuk kelas viewHutang |
| inputHutang() | Public | Menambahkan data Hutang |
| updateHutang() | Public | Melakukan perubahan pada data Hutang |
| delHutang() | Public | Melakukan penghapusan data Hutang |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *-* | *-* | *-* |

### Kelas developerView

*Nama Kelas : developer view*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| updateSystem() | Public | Melakukan perubahan pada System Aplikasi |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *-* | *-* | *-* |

## Diagram Kelas Keseluruhan



## Algoritma/Query

*Nama Kelas : budgetController*

*Nama Operasi : inputPemasukan()*

*Algoritma : (Algo-001)*

budget budget = new Budget(idBudget, nominal, tglBudget)

budget.inputPemasukan(budget)

INSERT INTO budget(idBudget, nominal, tglBudget)

VALUES (budget.getIdBudget(), budget.getNominal(), budget.getTglBudget())

*Nama Kelas : budgetController*

*Nama Operasi : inputPengeluaran()*

*Algoritma : (Algo-002)*

budget budget = new Budget(idBudget, nominal, tglBudget)

budget.inputPengeluaran(budget)

INSERT INTO budget(idBudget, nominal, tglBudget)

VALUES (budget.getIdBudget(), budget.getNominal(), budget.getTglBudget())

*Nama Kelas : barangController*

*Nama Operasi : inputBarang()*

*Algoritma : (Algo-003)*

barang barang = new Barang(idBarang, namaBarang, harga, batasWaktu)

barang.inputBarang(barang)

INSERT INTO barang(idBarang, namaBarang, harga, batasWaktu)

VALUES (barang.getIdBarang(), budget.getNamaBarang(), budget.getHarga(), budget.getBatasWaktu())

*Nama Kelas : barangController*

*Nama Operasi : deleteBarang()*

*Algoritma : (Algo-004)*

i = 0

while (barang.get(i).getIdBarang() != ID AND i<barang.size())

i++

if(barang.get(i).getIdBarang() = ID)

barang.remove(i)

DELETE FROM barang where ID = barang.get(i).getIdBarang()

*Nama Kelas : barangController*

*Nama Operasi : updateBarang()*

*Algoritma : (Algo-005)*

i = 0

while (idBarang != barang.get(i).idBarang AND i < barang.size())

i++

if(idBarang = barang.get(i).idBarang)

barang.get(i).setNamaBarang(nmBrg)

barang.get(i).setHarga(hrg)

barang.get(i).setBatasWaktu(btsWkt)

UPDATE barang SET idBarang = idBrg, namaBarang = nmBrg, harga = hrg, batasWaktu = btsWkt

WHERE idBarang = barang.getIdBarang()

*Nama Kelas : hutangController*

*Nama Operasi : inputHutang()*

*Algoritma : (Algo-006)*

hutang hutang = new Hutang(idHutang, namaHutang, nominal, tanggal)

hutang.inputHutang(hutang)

INSERT INTO hutang(idHutang, namaHutang, nominal, tanggal)

VALUES (hutang.getIdHutang(), hutang.getNamaHutang(), hutang.getNominal(), hutang.getTanggal())

*Nama Kelas : hutangController*

*Nama Operasi : deleteHutang()*

*Algoritma : (Algo-007)*

i = 0

while (hutang.get(i).getIdHutang() != ID AND i<hutang.size())

i++

if(hutang.get(i).getIdHutang() = ID)

hutang.remove(i)

DELETE FROM hutang where ID = hutang.get(i).getIdHutang()

*Nama Kelas : hutangController*

*Nama Operasi : updateHutang()*

*Algoritma : (Algo-008)*

i = 0

while (idHutang != hutang.get(i).idHutang AND i < hutang.size())

i++

if(idHutang = hutang.get(i).hutang)

hutang.get(i).setNamaHutang(nmHtg)

hutang.get(i).setNominal(nmn)

hutang.get(i).setTanggal(tgl)

UPDATE hutang SET idHutang = idHtg, namaHutang = nmHtg, nominal = nmn, tanggal = tgl

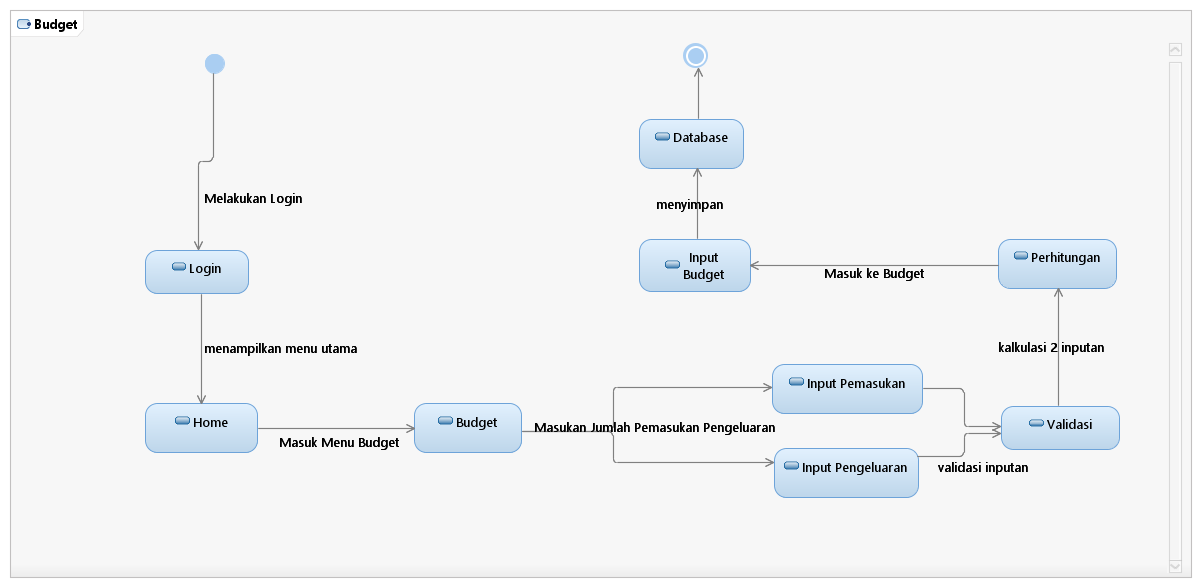
WHERE idHutang = hutang.getIdHutang()

*Query :*

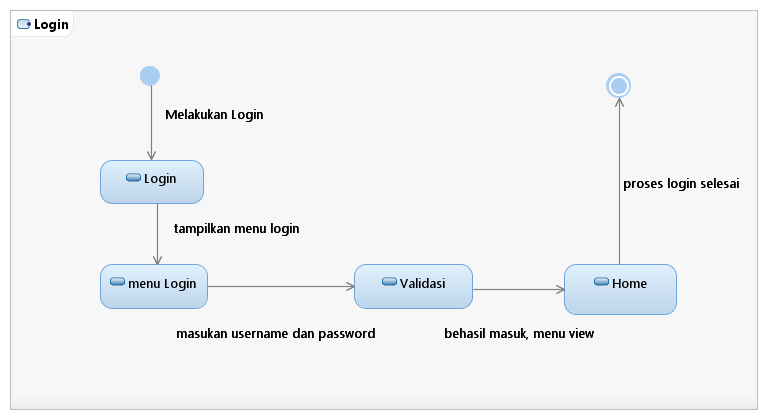
|  |  |  |
| --- | --- | --- |
| *No Query* | *Query* | *Keterangan* |
| *Query-001* | INSERT INTO budget (idBudget, nominal, tglBudget)  VALUES ( budget.getIdBudget(), budget.getNominal(), budget.getTglBudget()) ; | User menambahkan data Budget, dan menyimpannya ke dalam Database |
| *Query-002* | UPDATE budget  SET idBudget = idBgt, nominal = nmn, tglBudget = tglBgt  WHERE idBudget = budget.getIdBudget() ; | User melakukan perubahan data Budget berdasarkan inputan id Budget yang akan dilakukan perubahan |
| *Query-003* | DELETE FROM Budget  WHERE idBudget = Budget.getIdBudget() ; | Melakukan penghapusan data Budget berdasarkan inputan idBudget yang akan dilakukan penghapusan |
| *Query-004* | SELECT budget.getIdBudget(),budget.getNominal(),budget.getTglBudget()  FROM Budget; | Menampilkan data Budget kepada User |
| *Query-005* | INSERT INTO barang (idBarang, namaBarang, harga, batasWaktu)  VALUES (barang.getIdBarang(),barang.getNamaBarang(),barang.getHarga(), barang.getBatasWaktu()) ; | User menambahkan data Barang, dan menyimpannya ke dalam Database |
| *Query-006* | UPDATE barang  SET idBarang= idBrg, namaBarang = nmBrg, harga = hrg, batasWaktu = btsWkt  WHERE idBarang = barang.getIdBarang() ; | User melakukan perubahan data Barang berdasarkan inputan id Barang yang akan dilakukan perubahan |
| *Query-007* | DELETE FROM Barang  WHERE idBarang = barang.getIdBarang() ; | Melakukan penghapusan data Barang berdasarkan inputan idBarang yang akan dilakukan penghapusan |
| *Query-008* | SELECT barang.getIdBarang(), barang.getNamaBarang(), barang.getHarga(), barang.getBatasWaktu()  FROM Budget; | Menampilkan data Barang kepada User |
| *Query-009* | INSERT INTO hutang (idHutang, namaHutang, nominal, tanggal)  VALUES ( hutang.getIdHutang(), hutang.getNamaHutang(), hutang.getNominal(), hutang.getTanggal()) ; | User menambahkan data Hutang, dan menyimpannya ke dalam Database |
| *Query-010* | UPDATE hutang  SET idHutang= idHtg, namaHutang = nmHtg, nominal = nmn, tanggal = tgl  WHERE idHutang = hutang.getIdHutang() ; | User melakukan perubahan data Hutang berdasarkan inputan id Hutang yang akan dilakukan perubahan |
| *Query-011* | DELETE FROM hutang  WHERE idHutang = hutang.getIdHutang() ; | Melakukan penghapusan data Hutang berdasarkan inputan idHutang yang akan dilakukan penghapusan |
| *Query-012* | SELECT hutang.getIdHutang(), hutang.getNamaHutang(), hutang.getNominal(), hutang.getTanggal()  FROM hutang; | Menampilkan data hutang kepada User |

## Diagram Statechart

### Statechart Budget



### Statechart Login



## Perancangan Antarmuka

*Antarmuka : Login*

|  |  |
| --- | --- |
|  | * Jika mengklik button *Entry* maka akan masuk ke UI Login untuk User dan Developer * Jika mengklik button *Register* maka akan masuk ke UI untuk medaftarkan user agar dapat masuk ke aplikasi |
| ukuran 375 x 667 sesuai standar Android dengan jenis font *Mukta* dan *ABeeZee* | |

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *ButtonEntry* | Button | Entry | Jika diklik, akan menampilkan UI Login, masuk kedalam aplikasi |
| *ButtonRegister* | *Button* | *Register* | Jika diklik, akan menampilkan UI Register, daftar kedalam aplikasi |

*Antarmuka : Sign In*

|  |  |
| --- | --- |
|  | * Jika mengklik field *Username* maka harus mengisi username yang telah di miliki * Jika mengklik field *Password* maka harus mengisi password yang telah di miliki * Jika mengklik button *Sign In* maka masuk ke halaman *Home* * Jika mengklik button *Sign Up* maka akan masuk ke halaman *Register* |
| ukuran 375 x 667 sesuai standar Android dengan jenis font *Mukta* dan *AbeeZee* | |

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *UsernameField* | Field | Usernam | Field untuk mengisi username yang telah terdaftar dalam aplikasi |
| *PasswordField* | *Field* | *Password* | Field untuk mengisi password untuk username yang telah terdaftar dalam aplikasi |
| *SignInButton* | *Button* | *Sign in* | Setelah mengisi field username dan passwod, klik sign in untuk masuk ke aplikasi dengan validasi account |
| *SignUpButton* | *Button* | *Sign Up* | Jika di klik maka akan keluar UI register untuk user yang belum memiliki account |

*Antarmuka : Pop Up Login Fail*

|  |  |
| --- | --- |
| Login Fail | * Jika mengklik button *OK* maka akan kembali UI Login untuk User dan Developer. |
| ukuran 375 x 667 sesuai standar Android dengan jenis font *Mukta* dan *ABeeZee* | |

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *ButtonEntry* | Button | Entry | Jika diklik akan menampilkan UI Login kembali |
| *MsgError* | *Message* | *Error! Login fail* | Message akan muncul apabila username dan password tidak cocok dalam database |

*Antarmuka : Register*

|  |  |
| --- | --- |
|  | * Jika mengklik field *Full Name* maka harus mengisi nama * Jika mengklik field *E-Mail* maka harus menginputkan email yang aktif * Jika mengklik field *Username* maka harus meginputkan username yang akan dijadikan untuk masuk ke aplikasi * Jika mengklik field *Password* maka harus mengisi password yang akan dijasikan untuk masuk ke aplikasi * Jika mengklik Sign Up maka data yang telah diisikan untuk menggunakan aplikasi ini disimpan dalam database. Dan siap digunakan user untuk masuk kedalam aplikasi. |
| ukuran 375 x 667 sesuai standar Android dengan jenis font *Mukta* dan *ABeeZee* | |

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *ButtonEntry* | Button | Entry | Jika diklik, akan menampilkan UI Login, masuk kedalam aplikasi |
| *ButtonRegister* | *Button* | *Register* | Jika diklik, akan menampilkan UI Register, daftar kedalam aplikasi |

*Antarmuka : Home*

|  |  |
| --- | --- |
|  | * Jika mengklik button *Budget* maka akan masuk ke UI Budget. * Jika mengklik button *Debt* maka akan masuk ke UI Hutang. * Jika mengklik button *DREAM ITEM* maka akan masuk ke UI Barang Impian. * Jika mengklik button ‘‘ maka akan muncul UI Pop Up Menu. |
| ukuran 375 x 667 sesuai standar Android dengan jenis font *Mukta* dan *ABeeZee* | |

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *ButtonBudget* | Button | Budget | Jika diklik, akan menampilkan UI Budget dan menunya |
| *ButtonDebt* | *Button* | *Debt* | Jika diklik, akan menampilkan UI Hutang dan menunya |
| *ButtonDreamItem* | *Button* | *Dream Item* | Jika diklik, akan menampilkan UI Barang Impian dan menunya |
| *MsgBudget* | *Message* | *Your Balance* | Message yang menunjukan berapa besar budget/keungan yang kita miliki |
| *MsgIncomeOutcome* | *Message* | *Income , Outcome* | Message dan Grafik yang menunjukan seberapa banyak income dan outcome |
| *ButtonMenu* | *Button* | ‘‘ | Jika diklik akan dapat menampilkan UI Pop Up Menu |

*Antarmuka : Pop Up Menu*

|  |  |
| --- | --- |
|  | * Jika mengklik button *Home* maka akan masuk ke UI Main Menu * Jika mengklik button *Budget* maka akan masuk ke UI Budget. * Jika mengklik button *Debt* maka akan masuk ke UI Hutang. * Jika mengklik button *Dream Item* maka akan masuk ke UI Barang Impian. * Jika mengklik button *Help* maka akan keluar pop petunjuk dan cara penggunaan aplikasi. * Jika mengklik button *Setting* maka akan masuk ke UI setting. * Jika mengklik button *Profile* maka akan masuk ke UI Profile. * Jika mengklik button *Sign Out* maka akan keluar dan kembali ke UI login. |
| ukuran 375 x 667 sesuai standar Android dengan jenis font *Mukta* dan *ABeeZee* | |

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *ButtonProfile* | Button | Profile | Jika diklik, akan menampilkan UI Profile, untuk view profil user dan edit profil user |
| *ButtonHome* | *Button* | *Home* | Jika diklik, akan menampilkan UI Home |
| *ButtonBudget* | *Button* | *Budget* | Jika diklik, akan menampilkan UI Budget |
| *ButtonDebt* | *Button* | *Debt* | Jika diklik, akan menampilkan UI Debt |
| *ButtonDreamItem* | *Button* | *Dream Item* | Jika diklik, akan menampilkan UI Barang Impian |
| *ButtonHelp* | *Button* | *Help* | Jika diklik, akan menampilkan pop up cara penggunaan aplikasinya |
| *ButtonSetting* | *Button* | *Setting* | Jika diklik, akan menampilkan UI Setting |
| *ButtonSignOut* | *Button* | *Sign Out* | Jika diklik, akan keluar dan menampilkan UI Login |

*Antarmuka : Edit Profil*

|  |  |
| --- | --- |
|  | * Jika mengklik field *Name* maka dapat mengisi atau mengubah nama yang akan tertera dalam apliakasi * Jika mengklik field *Username* maka dapat mengubah username yang akan digunakan dalam apliakasi * Jika mengklik field *E-mail Address* maka dapat mengisi atau mengubah email yang akan tertera dalam apliakasi * Jika mengklik button *Change Photo* maka user dapat mengubah photo profil user. * Jika mengklik button ‘Checkmark ‘ maka perubahan yang terjadi dalam UI profile akan disimpan. * Jika mengklik button ‘*Close* ‘ maka perubahan yang terjadi dalam UI profile tidak akan disimpan dan kembali ke main menu. |
| ukuran 375 x 667 sesuai standar Android dengan jenis font *Mukta* dan *ABeeZee* | |

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *FieldName* | Field | Name | Jika diklik akan dapat mengisi/mangubah Nama yang tertera dalam aplikasi |
| *FieldUsername* | *Field* | *Username* | Jika diklik akan dapat mengisi/mangubah Username yang digunakan untuk memasuki aplikasi |
| *FieldEmail* | *Field* | *E-mail Address* | Jika diklik akan dapat mengisi/mangubah Email yang tertera dalam aplikasi |
| *ButtonChngPhoto* | *Button* | *Change Photo* | Jika diklik akan dapat mangubah photo yang tertera dalam aplikasi |
| *ButtonAcc* | *Button* | ‘Checkmark‘ | Jika diklik akan dapat menyiman semua perubahan/pengisisan yang terjadi dalam UI porfile |
| *ButtonDec* | *Button* | *‘Close’* | Jika diklik tidak akan dapat menyiman semua perubahan/pengisisan yang terjadi dalam UI porfile |

*Antarmuka : Menu Budget(View Budget)*

|  |  |
| --- | --- |
|  | * Jika mengklik button *Input* maka akan masuk ke UI Input Pemasukan untuk User. * Jika mengklik button *Output* maka akan masuk ke UI Input Pengeluaran untuk User. * Jika mengklik button *Report* maka akan masuk ke UI laporan budget Pemasukan dan Pengeluaran. * Jika mengklik button ‘Checkmark ‘ maka perubahan yang terjadi dalam UI Budget akan disimpan. * Jika mengklik button ‘‘ maka akan muncul UI Pop Up Menu. |
| ukuran 375 x 667 sesuai standar Android dengan jenis font *Mukta* dan *ABeeZee* | |

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *ButtonInput* | Button | INPUT | Jika diklik, akan menampilkan UI Input Pemasukan, untuk menginputkan Pemasukan |
| *ButtonOutput* | *Button* | *OUTPUT* | Jika diklik, akan menampilkan UI Input Pengeluaran, untuk menginputkan Pengeluaran |
| *ButtonReport* | *Button* | *REPORT* | Jika diklik, akan menampilkan UI Laporan Pemasukan dan Pengeluaran |
| *ButtonAcc* | *Button* | ‘Checkmark‘ | Jika diklik akan dapat menyiman semua perubahan/pengisisan yang terjadi dalam UI Budget |
| *ButtonMenu* | *Button* | ‘‘ | Jika diklik akan dapat menampilkan UI Pop Up Menu |

*Antarmuka : Input Pemasukan*

|  |  |
| --- | --- |
|  | * Jika mengklik field *Nominal* maka user dapat mangisi Pemasukan yang terjadi. * Jika mengklik button ‘Checkmark‘ maka perubahan yang terjadi dalam UI Budget akan disimpan. * Jika mengklik button ‘‘ maka akan muncul UI Pop Up Menu. |
| ukuran 375 x 667 sesuai standar Android dengan jenis font *Mukta* dan *ABeeZee* | |

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *FieldNominal* | Field | Nominal | Jika diklik, maka user dapat menginputkan sejumlah uang yang nantinya dinyatakan sebagai Pemasukan |
| *MsgIncome* | *Message* | *Income* | Message dan Grafik yang menunjukan seberapa banyak income atau Pemasukan |
| *ButtonAcc* | *Button* | ‘Checkmark‘ | Jika diklik akan dapat menyiman semua perubahan/pengisisan yang terjadi dalam UI Budget |
| *ButtonMenu* | *Button* | ‘‘ | Jika diklik akan dapat menampilkan UI Pop Up Menu |

*Antarmuka : Input Pengeluaran*

|  |  |
| --- | --- |
|  | * Jika mengklik field *Nominal* maka user dapat mangisi Pengeluaran yang terjadi. * Jika mengklik button ‘Checkmark‘ maka perubahan yang terjadi dalam UI Budget akan disimpan. * Jika mengklik button ‘‘ maka akan muncul UI Pop Up Menu. |
| ukuran 375 x 667 sesuai standar Android dengan jenis font *Mukta* dan *ABeeZee* | |

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *FieldNominal* | Field | Nominal | Jika diklik, maka user dapat menginputkan sejumlah uang yang nantinya dinyatakan sebagai Pengeluaran |
| *MsgOutcome* | *Message* | *Income* | Message dan Grafik yang menunjukan seberapa banyak outcome atau Pengeluaran |
| *ButtonAcc* | *Button* | ‘Checkmark‘ | Jika diklik akan dapat menyiman semua perubahan/pengisisan yang terjadi dalam UI Budget |
| *ButtonMenu* | *Button* | ‘‘ | Jika diklik akan dapat menampilkan UI Pop Up Menu |

*Antarmuka : Budget Report*

|  |  |
| --- | --- |
|  | * Jika mengklik button ‘Checkmark‘ maka akan kembali ke UI Budget. * Jika mengklik button ‘‘ maka akan muncul UI Pop Up Menu. |
| ukuran 375 x 667 sesuai standar Android dengan jenis font *Mukta* dan *ABeeZee* | |

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *MsgReport* | Message | REPORT | Menapilkan jumlah nominal budget sekarang |
| *MsgIncomeOutcome* | *Message* | *Income , Outcome* | Message dan Grafik yang menunjukan seberapa banyak income dan outcome |
| *ButtonAcc* | *Button* | ‘Checkmark‘ | Jika diklik akan kembali ke UI Budget |
| *ButtonMenu* | *Button* | ‘‘ | Jika diklik akan dapat menampilkan UI Pop Up Menu |

*Antarmuka : Hutang Menu (View Hutang)*

|  |  |
| --- | --- |
|  | * Jika mengklik button *Input* maka akan keluar pop up UI Input Hutang untuk User. * Jika mengklik button *Edit* maka akan masuk ke UI Edit Hutang untuk User. * Jika mengklik button *Delete* maka akan masuk ke UI Hapus Hutang. * Jika mengklik button ‘Checkmark ‘ maka perubahan yang terjadi dalam UI Hutang akan disimpan. * Jika mengklik button ‘‘ maka akan muncul UI Pop Up Menu. |
| ukuran 375 x 667 sesuai standar Android dengan jenis font *Mukta* dan *ABeeZee* | |

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *ButtonInput* | Button | INPUT | Jika diklik, akan menampilkan pop up UI Input Hutang |
| *ButtonEdit* | *Button* | *EDIT* | Jika diklik, akan menampilkan UI Edit Hutang |
| *ButtonDel* | *Button* | *DELET* | Jika diklik, akan menampilkan UI Hapus Hutang |
| *ButtonAcc* | *Button* | ‘Checkmark‘ | Jika diklik akan dapat menyiman semua perubahan/pengisisan yang terjadi dalam UI Hutang |
| *ButtonMenu* | *Button* | ‘‘ | Jika diklik akan dapat menampilkan UI Pop Up Menu |

*Antarmuka : Input Hutang*

|  |  |
| --- | --- |
|  | * Jika mengklik field *Nominal* maka user dapat mengisikan hutang yang akan diinputkan. * Jika mengklik button ‘Checkmark ‘ maka perubahan yang terjadi dalam UI Hutang akan disimpan. * Jika mengklik button ‘‘ maka akan muncul UI Pop Up Menu. |
| ukuran 375 x 667 sesuai standar Android dengan jenis font *Mukta* dan *ABeeZee* | |

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *FieldInput* | Field | Nominal | Jika diklik, maka user dapat menginput Hutang |
| *ButtonAcc* | *Button* | ‘Checkmark‘ | Jika diklik akan dapat menyiman semua perubahan/pengisisan yang terjadi dalam UI Hutang, jika tidak terjadi perubahan maka tidak akan disimpan dalam aplikasi |
| *ButtonMenu* | *Button* | ‘‘ | Jika diklik akan dapat menampilkan UI Pop Up Menu |

*Antarmuka : Edit Hutang*

|  |  |
| --- | --- |
|  | * Jika mengklik button *Input* maka akan keluar pop up UI Input Hutang untuk User. * Jika mengklik button *Edit* maka akan masuk ke UI Edit Hutang untuk User. * Jika mengklik button *Delete* maka akan masuk ke UI Hapus Hutang. * Jika mengklik button ‘Pencil ‘ maka user dapat melakukan perubahan pada tiap hutang yang dipilih. * Jika mengklik button ‘Checkmark ‘ maka perubahan yang terjadi dalam UI Hutang akan disimpan. * Jika mengklik button ‘‘ maka akan muncul UI Pop Up Menu. |
| ukuran 375 x 667 sesuai standar Android dengan jenis font *Mukta* dan *ABeeZee* | |

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *ButtonInput* | Button | INPUT | Jika diklik, akan menampilkan pop up UI Input Hutang |
| *ButtonDel* | *Button* | *DELET* | Jika diklik, akan menampilkan UI Hapus Hutang |
| *ButtonEdit* | *Button* | ‘Pencil ‘ | Jika diklik user akan dapat melakukan perubahan pada tiap hutang yang dipilih. |
| *ButtonAcc* | *Button* | ‘Checkmark‘ | Jika diklik akan dapat menyiman semua perubahan/pengisisan yang terjadi dalam UI Hutang |
| *ButtonMenu* | *Button* | ‘‘ | Jika diklik akan dapat menampilkan UI Pop Up Menu |

*Antarmuka : Delete Hutang*

|  |  |
| --- | --- |
|  | * Jika mengklik button *Input* maka akan keluar pop up UI Input Hutang untuk User. * Jika mengklik button *Edit* maka akan masuk ke UI Edit Hutang untuk User. * Jika mengklik button *Delete* maka akan masuk ke UI Hapus Hutang. * Jika mengklik button ‘*Close* ‘ maka user dapat melakukan penghapusan pada tiap hutang yang dipilih. * Jika mengklik button ‘Checkmark ‘ maka perubahan yang terjadi dalam UI Hutang akan disimpan. * Jika mengklik button ‘‘ maka akan muncul UI Pop Up Menu. |
| ukuran 375 x 667 sesuai standar Android dengan jenis font *Mukta* dan *ABeeZee* | |

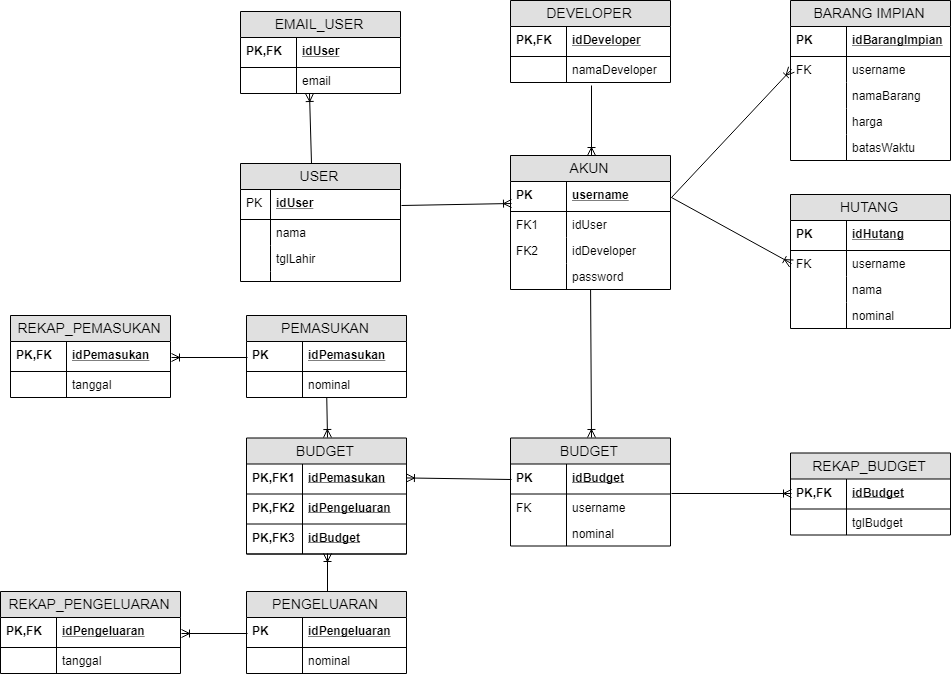
| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *ButtonInput* | Button | INPUT | Jika diklik, akan menampilkan pop up UI Input Hutang |
| *ButtonEdit* | *Button* | *EDIT* | Jika diklik, akan menampilkan UI Edit Hutang |
| *ButtonDelete* | *Button* | ‘*Close* ‘ | Jika diklik user akan dapat melakukan penghapusan pada tiap hutang yang dipilih. |
| *ButtonAcc* | *Button* | ‘Checkmark‘ | Jika diklik akan dapat menyiman semua perubahan/pengisisan yang terjadi dalam UI Hutang |
| *ButtonMenu* | *Button* | ‘‘ | Jika diklik akan dapat menampilkan UI Pop Up Menu |

*Antarmuka : View Barang Impian*

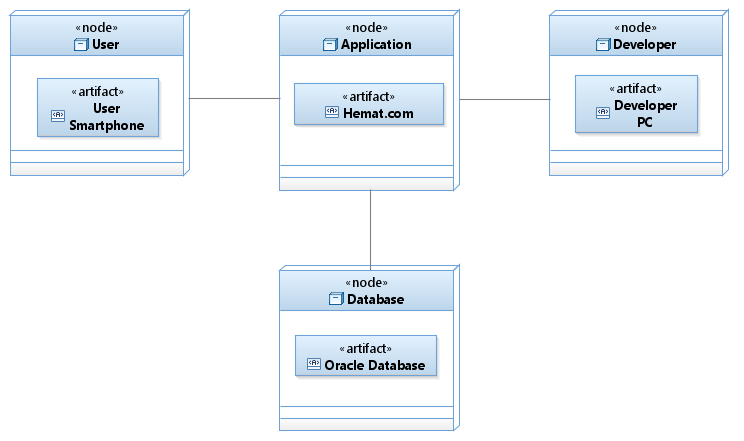
|  |  |
| --- | --- |
|  | * Jika mengklik button *‘ Add’* maka akan keluar pop up UI Input Barang Impian untuk User. * Jika mengklik button *Edit* maka akan masuk ke UI Edit Barang Impian untuk User. * Jika mengklik button *Delete* maka akan masuk ke UI Hapus Barang Impian. * Jika mengklik button ‘ ‘ maka user dapat melakukan pemilihan pada tiap Barang Impian yang dipilih. * Jika mengklik button ‘Checkmark ‘ maka perubahan yang terjadi dalam UI Barang Impian akan disimpan. * Jika mengklik button ‘‘ maka akan muncul UI Pop Up Menu. |
| ukuran 375 x 667 sesuai standar Android dengan jenis font *Mukta* dan *ABeeZee* | |

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| *ButtonInput* | Button | ‘Add’ | Jika diklik, akan menampilkan pop up UI Input Barang Impian |
| *ButtonEdit* | *Button* | *EDIT* | Jika diklik, akan menampilkan UI Edit Barang Impian |
| *ButtonDel* | *Button* | *DELETE* | Jika diklik akan menampilkan UI Delete Barang Impian |
| *ButtonPick* | *Button* | ‘ ‘ | Jika diklik user akan dapat melakukan pemilihan pada tiap Barang Impian yang dipilih. |
| *ButtonAcc* | *Button* | ‘Checkmark‘ | Jika diklik akan dapat menyiman semua perubahan/pengisisan yang terjadi dalam UI Barang Impian |
| *ButtonMenu* | *Button* | ‘‘ | Jika diklik akan dapat menampilkan UI Pop Up Menu |

## Perancangan Representasi Persistensi Kelas



## Perancangan Diagram Deployment



# 4. Matriks Kerunutan

|  |  |
| --- | --- |
| **Kelas** | **Use Case Terkait** |
| Login | Login  User  Developer |
| Register | Register  User |
| User | User |
| Developer | Developer  Memperbaharui Sistem |
| Budget | Menginput Pemasukan  Menginput Pengeluaran  Melihat Budget  User |
| Hutang | Menginput Hutang  Mengubah Hutang  Menghapus Hutang  Melihat Hutang  User |
| Barang | Menginput Barang Impian  Mengubah Barang Impian  Menghapus Barang Impian  User |

# 