

**COLLEGE OF ENGINEERING**

**SCHOOL OF COMPUTER SCIENCE & ENGINEERING**

**CZ2006: Software Engineering   
Academic Year 2016/17**

**Project Documentation**

Revision Date: 24/10/2016

Version: 1.6

Students’ Name: Thomas Lim Jun Wei (U1521407L)

Lim Zi Yang (U1522218E)

Huang Jian Wei (U1521567A) Lim Hao Zhe (U1521589B)

Sim Long Siang (U1522053H)

Genevieve Lam (U1521863H)

**Table of Contents**

**Table of Contents**

**Revision History**

**1. Introduction**

1.1 Purpose

1.2 Document Conventions

1.3 Intended Audience and Reading Suggestions

1.4 Product Scope

1.5 References

**2. Overall Description**

2.1 Product Perspective

2.2 Product Functions

2.3 User Classes and Characteristics

2.4 Operating Environment

2.5 Design and Implementation Constraints

2.6 User Documentation

2.7 Assumptions and Dependencies

**3. External Interface Requirements**

3.1 User Interfaces

3.2 Hardware Interfaces

3.3 Software Interfaces

3.4 Communications Interfaces

**4. System Features**

4.1 View Available BTO Flats

4.2 View Amenities

4.3 Query Grant

4.4 Calculate Monthly Repayment

**5. Other Non-functional Requirements**

5.1 Performance Requirements

5.2 Safety Requirements

5.3 Security Requirements

5.4 Software Quality Attributes

5.5 Business Rules

**6. Other Requirements**

6.1 Class Diagram

6.2 Sequence Diagram

6.3 State Machine Diagram

6.4 System Architecture

**7. Testing**

7.1 Black box Testing

7.1.1 Repayment

7.2 White Box Testing

7.2.1 View Flat

7.2.2 Grant Query

7.2.3 Repayment Calculator

**Appendix A: Glossary**

**Appendix B: UI Mockup**

**Appendix C: Use Case Diagram & Description**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Huang Jian Wei | 04 Sep | Compiling of information | V1.1 |
| Thomas Lim | 15 Sep | Updating of information | V1.2 |
| Sim Long Siang | 26 Sep | Inserting of Information | v1.3 |
| Thomas Lim | 01 Oct | Addition of diagrams | V1.4 |
| Sim Long Siang | 14 Oct | Addition of Test cases | V1.5 |
| Genevieve Lam | 28 Oct | Updates on diagrams | V1.6 |
| Thomas Lim | 08 Nov | Documentation tidy-up | V1.7 |
| Sim Long Siang | 09 Nov | Updates on diagrams | V1.8 |

# Introduction

## 1.1 Purpose

The purpose of this document is to present a detailed description of the software functional and non-functional requirements for release of version 1.0 of BTO@SG. It will explain the purpose of the system and features of the system, the interfaces of the system, what the system will do, constraint that which is imposed and how the system reacts to external stimuli. UI mock up and dialog maps will also be illustrated to provide a greater understanding of the system.

## 1.2 Document Conventions

The conventions used to prepare the document are as followed:

Font – Times new Roman, size 12

Main Heading – Times new Roman, size 18, bold

Sub Heading – Times new Roman, size 14, bold

Sub-sub Heading – Times new Roman size 12, bold

Unless specified, all requirement statements are high priority and committed to the release of BTO@SG.

## 1.3 Intended Audience and Reading Suggestions

This document is intended to be used by the members of the project team that will implement and verify the correct functioning of the system. The brief overview of the system would be focused in section 1 (Introduction) as well as section 2 of the document (Overall description). Section 3 consist of external interface requirement highlighting the local characteristics of each interface between the software products the users discussed. Section 4 gives a breakdown of system features with their functional requirements to describe major service provided by the intended system. Finally, the specification is concluded with the non-functional requirement on which this document is based on.

It is recommended that the reader starts with the overview of the system, followed by the system feature and non-functional requirements. Lastly, the reader can read on the external interface requirement to gain a better understanding of system.

## 1.4 Product Scope

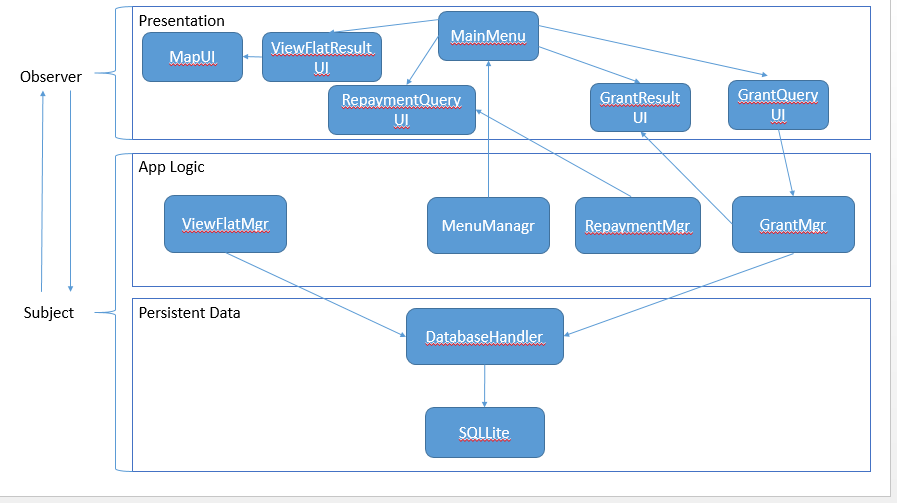
GJ Enterprise will develop a mobile application that will aid citizens who are in the process of selecting a Build-To-Order (BTO) flat for purchase in Singapore. The application has a full package functionality which allows users to view nearby amenities of the flat of interest, calculate debt repayment and query available grants, hence facilitating users in the process of making a decision about their future flat purchase. In short, BTO@SG is a one stop application to cater convenience for target users with adequate, organised housing information in the entire of Singapore.

# Overall Description

## 2.1 Product Perspective

The product is a newly developed standalone product that will be publish into the android market. The mobile application is implemented based on the model-view-controller model (MVC) model. BTO@SG provides relevant information for users to make an informed choice before making a decision for their BTO flats.

**2.1.1 System Architecture**

****

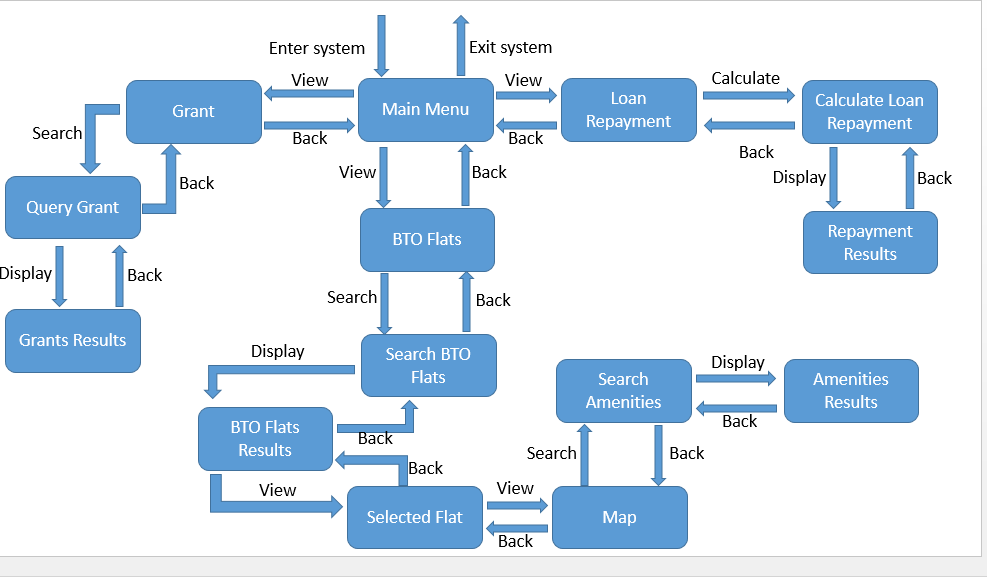
***Figure 1. A top-down view of the application implementation and individual components***

## 2.2 Product Functions

The bullet list below are the main functionality of what the application provides:

* Query grants: Allows users to view all grants available based on combined household income
* Repayment Calculator: Allow users to calculate the amount needed per month or year to finish the loan amount.
* View available flats and surrounding amenities: Allow users to search for available BTO flats in a specific area or region and filter out amenities according to their choices.

**2.2.1 Dialog map**

****

***Figure 1.1. A dialog map showing all possible user interaction with the system***

## 2.3 User Classes and Characteristics

The target audience for our mobile application are mainly couples with BTO enquiries, or anyone seeking BTO housing information in Singapore. The development of this application will be closely linked to the governing bodies of housing development board (HDB).

## 2.4 Operating Environment

BTO@SG must operate in an android smartphone with minimum operating system KitKat 4.4 and above. The application do require internet connection for certain feature (view amenities) and is able to perform other function (grant query & repayment calculator) offline. There is also no external hardware requirement or software dependencies to run the application. Currently as of October 25 2016, there are no existing conflict with any existing mobile application in the market.

## 2.5 Design and Implementation Constraints

The application will be developed using android studios hence developers must be familiar with Java programming language. As it is run on a mobile platform, developers will be limited to smaller scale database, such as SQLite. Developers have to maintain and update the data every month to ensure that users are provided with the latest information.

## 2.6 User Documentation

We provide helpdesk services to our users, when encountered with any problem or if users would like to get more information they could always click on to the help button.

## 2.7 Assumptions and Dependencies

Users are assumed to be equipped with basic navigation of an application. It is also assumed that users are familiar and have prior knowledge of how BTO works. Information provided to the users are based on latest data updated to data.gov hence any changes made to the original data will be affecting the one existing in our application. The application also incorporated Map API from Google to display a map showing the available BTO flats to the user.

# 3.1 External Interface Requirements

## User Interfaces

## 3.1.1 Home page

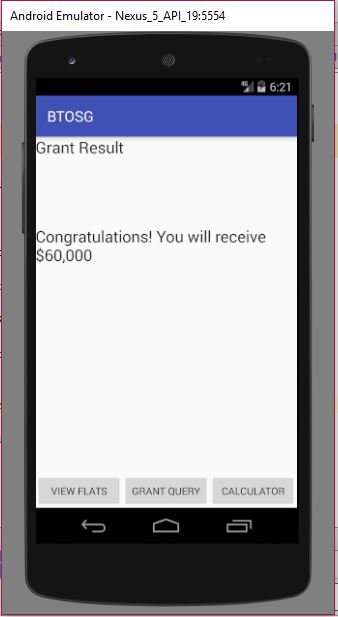
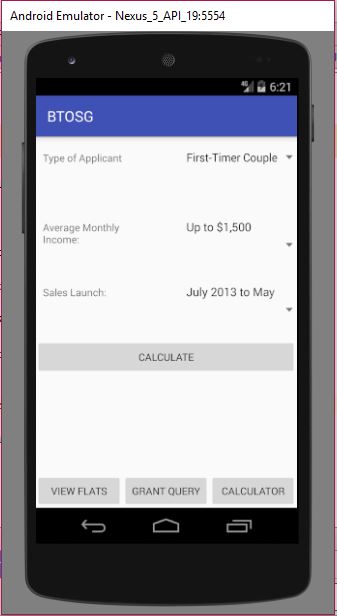
Where a user opens the application, he/she can view the different features provided by the app. They are View Flats, Grant Enquires and Calculator; those features all come in buttons where users can manoeuvre themselves around the app. See Figure 2.1



***Figure 2.1-Home page***

3.1.2 Grant Query

This is the page user will be directed to after pressing the Grant Query button. In this page, user should pick the different options in the drop-down list, provided in the following fields: Type of application, Average monthly income and Sales Launch. Users must select the appropriate options (Section 5.1.2 – Applicable Grant (Functional Requirements)) to the fields above without leaving any of them empty. See Figure 2.2.

After the user, has selected the right input accordingly and tap on the calculated button. And based on those criterial chosen by the user, the result will be generated; and display the result; if user is eligible for the grant offered. See Figure 2.3.

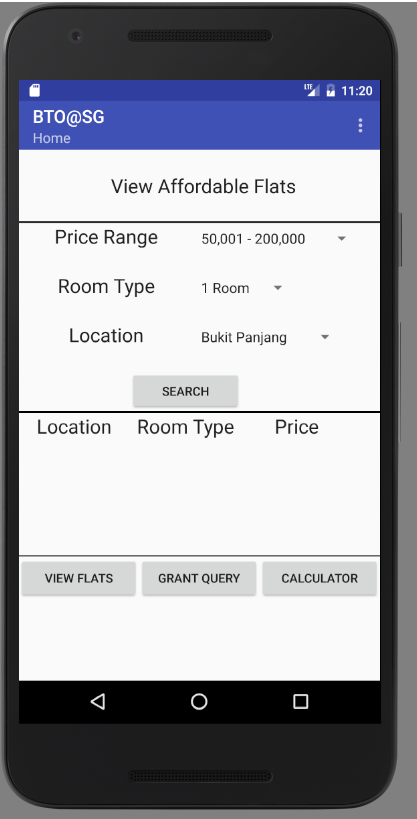
***Figure 2.3-Grant Query II***

***Figure 2.2-Grant Query I***

3.1.3 View Flats

In this page, user should pick the different options in the drop-down list, provided in the following fields: Price Range, Room type and Location. Users must select the appropriate options (Section 5.1.2 – Applicable Grant (Functional Requirements)) to the fields above without leaving any of them empty. See Figure 2.4.

After the user, has selected the right input accordingly and tap on the calculated button. And based on those criterial chosen by the user; results will be generated and displayed on the affordable flats. See Figure 2.5.

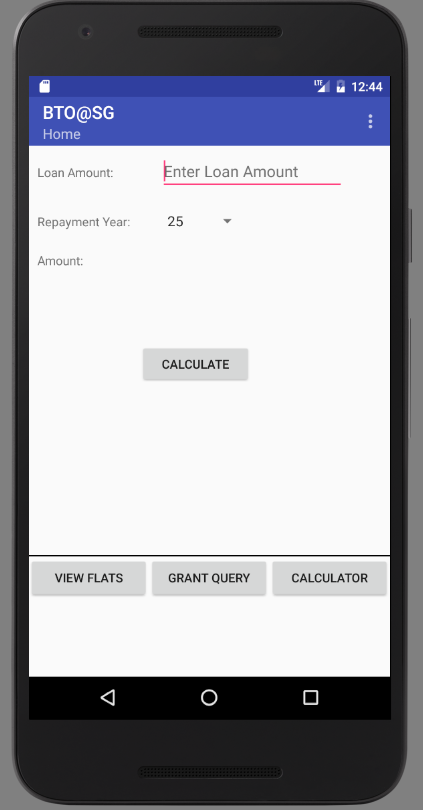
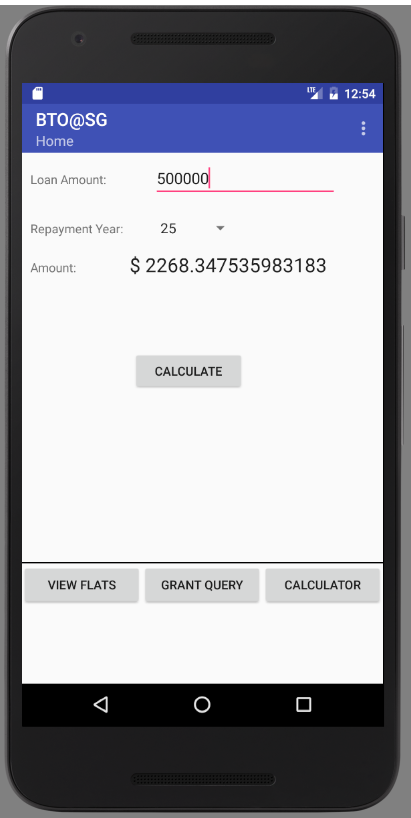


***Figure 2.4- View Flat I***

3.1.4 Calculate Repayment

In this page, users should input the amount they intend to loan in the Loan amount text box and select the desired years of loan in a yearly basis from the drop-down list then tap on the calculate button. Users must select the appropriate options or fill in (Section 5.1.2 – Applicable Grant (Functional Requirements)) the fields above without leaving any of them empty. See Figure 2.7.

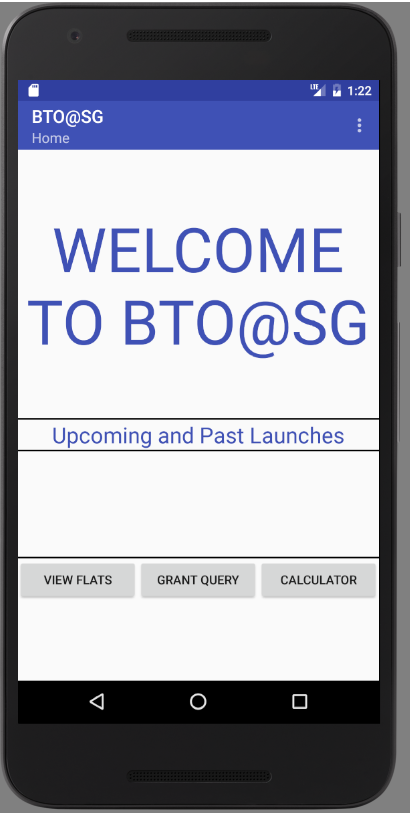
Based on the different criterial chosen by the user, the result will be generated, displaying the monthly instalment amount. See Figure 2.8.



***Figure 2.7- Calculate Repayment I***

***Figure 2.8- Calculate Repayment II***

3.1.4 Settings

In this page, on the top right hand corner of icon is the setting page. After user tap on it, they can choose the following options:. See Figure 2.9

***Figure 2.9- Settings I***

## Hardware Interfaces

The hardware interface that are incorporate to BTO@SG include the mobile device that has android operating system KitKat 4.4 and above installed. Data and information will displayed to users both in plain text as well as graphical map representation. To update existing data, there must be internet connection to the mobile device so that downloading can take place.

# 4. System Features

## 4.1 View available BTO Flats

* 1. User must be able to search for BTO flats available within Singapore.
     1. User must select available BTO flats based on location area from a drop-down selection box.
     2. Users must select number of room not less than 2 and not more than 5.
     3. Users must select a preferred HDB block among the region.
     4. The application must display a list of BTO flats available together with its price range and room size.

**4.2 View Amenities**

* 1. User must be able to view all amenities within 10km away from the selected flat.
     1. User must be able to view all public school under Ministry of Education the selected flat.
     2. User must be able to view all swimming complex and gyms surrounding the selected flat.
     3. User must be able to view all clinics available surrounding the selected flat.
     4. User must be able to view all parks surrounding the selected flat.
     5. User must be able to click and drag map to the direction he or she wants to explore.
     6. User must be able to exit current view with ease by clicking on the “x” button at the top right hand corner of the map.
  2. Application must show different amenities indicated by pin tacks on the on-screen map
     1. Application must show the name of the category as well as its category.
     2. User must be able to selectively choose what category of amenities he or she choose to view.

**4.4 Query Grant**

4.1 User must be able to view all eligible grant.

4.2 Application must be able to display a list of eligible grant based on user applicant type and user monthly salary from a drop-down box.

4.3 Application must be able to display error message if he or she click search without either of the drop-down list.

**4.5 Calculate Repayment**

5.1 Application must be able to show repayment details according to user loan amount and years of repayment.

5.2 Users must choose amount of year he or she wants to complete repayment from a drop-down selection box.

5.3 Users must be able to choose the percentage between cash and Central Provident Fund (CPF) he or she wish to pay for the repayment using a slider.

5.4 Application must show the result of user preferred repayment plan and the amount required per year for the repayment for both cash and CPF category.

# 5. Other Non-functional Requirements

1. **Usability**

The application is catered mainly for Singaporean citizen planning to apply for BTO flats.

* 1. The application interface elements (e.g. menu) must be easy to understand and consistent in its layout.
  2. The application must offer informative feedbacks for each of user action.
  3. The application must be responsive with regards to its output mapping to user activity.
  4. The application must be design in a way user cannot make a serious error. Most user input are defined by the application itself.
  5. The application must be able to handle invalid inputs and prompt error message if necessary.
  6. The application must allow user to undo their action to relieve anxiety.

1. **Reliability**

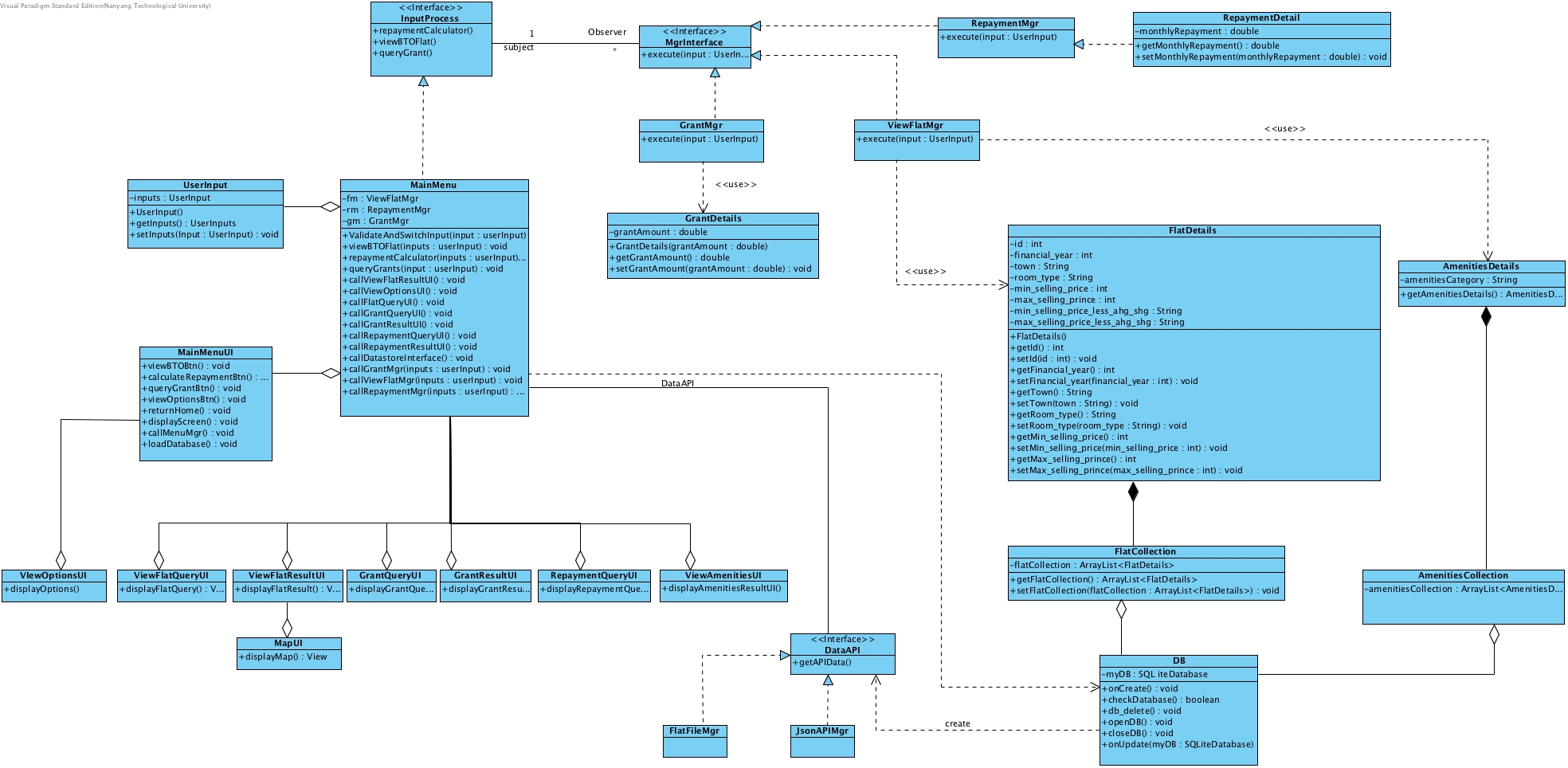
The application must have the ability to behave consistently in a user-acceptable manner when operating within the environment for which the application is intended.

* 1. The application must maintain a 95% uptime.
  2. The application must yield consistent search result for a specific query.

1. **Performance**
   1. The application must provide a feedback upon a user activity within 2 seconds.
   2. The application must be able to perform with previous retrieved data as backup when there is no internet connection.
   3. The application must be able to perform as usual when there is mobile device under low power.
2. **Supportability**
   1. The application must be able to receive future version updates.
   2. The application must be able to support both Android and Apple platform.
   3. The application must be able to support multiple languages.
   4. The application must be installed with ease.

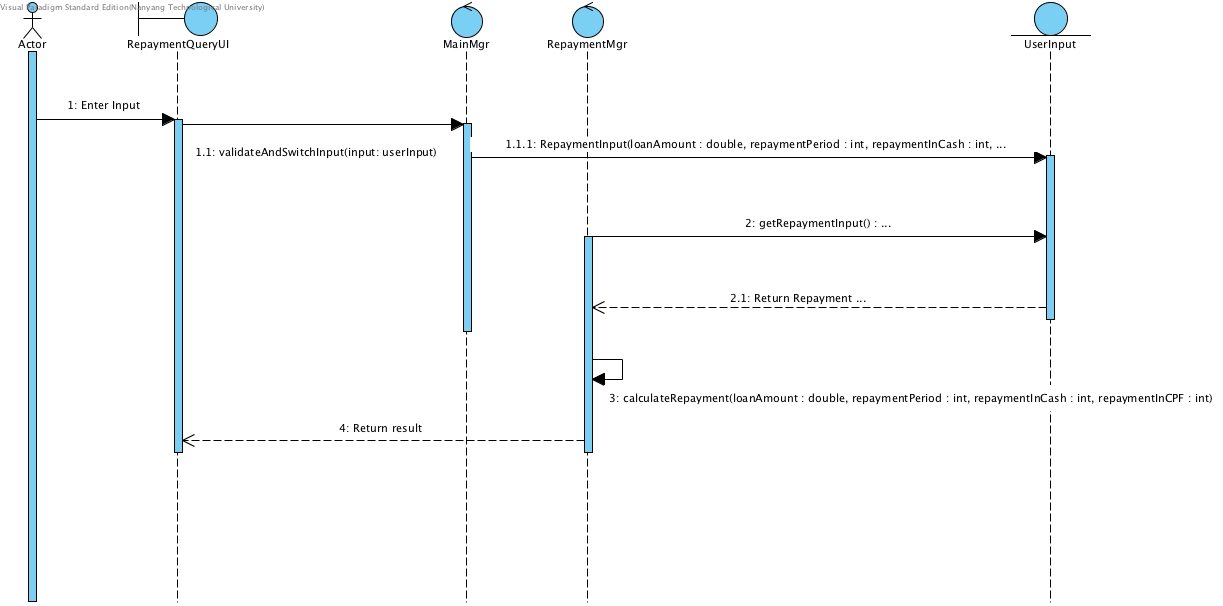
# 6. Other Requirements

**6.1 Class Diagram**



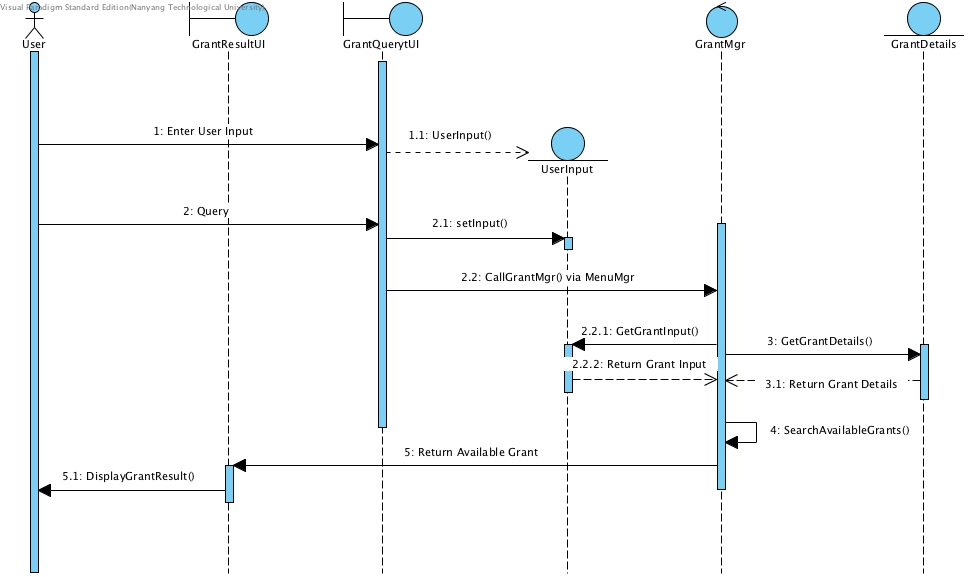
**6.2 Sequence Diagram**

**6.2.1 Repayment Calculator**

****

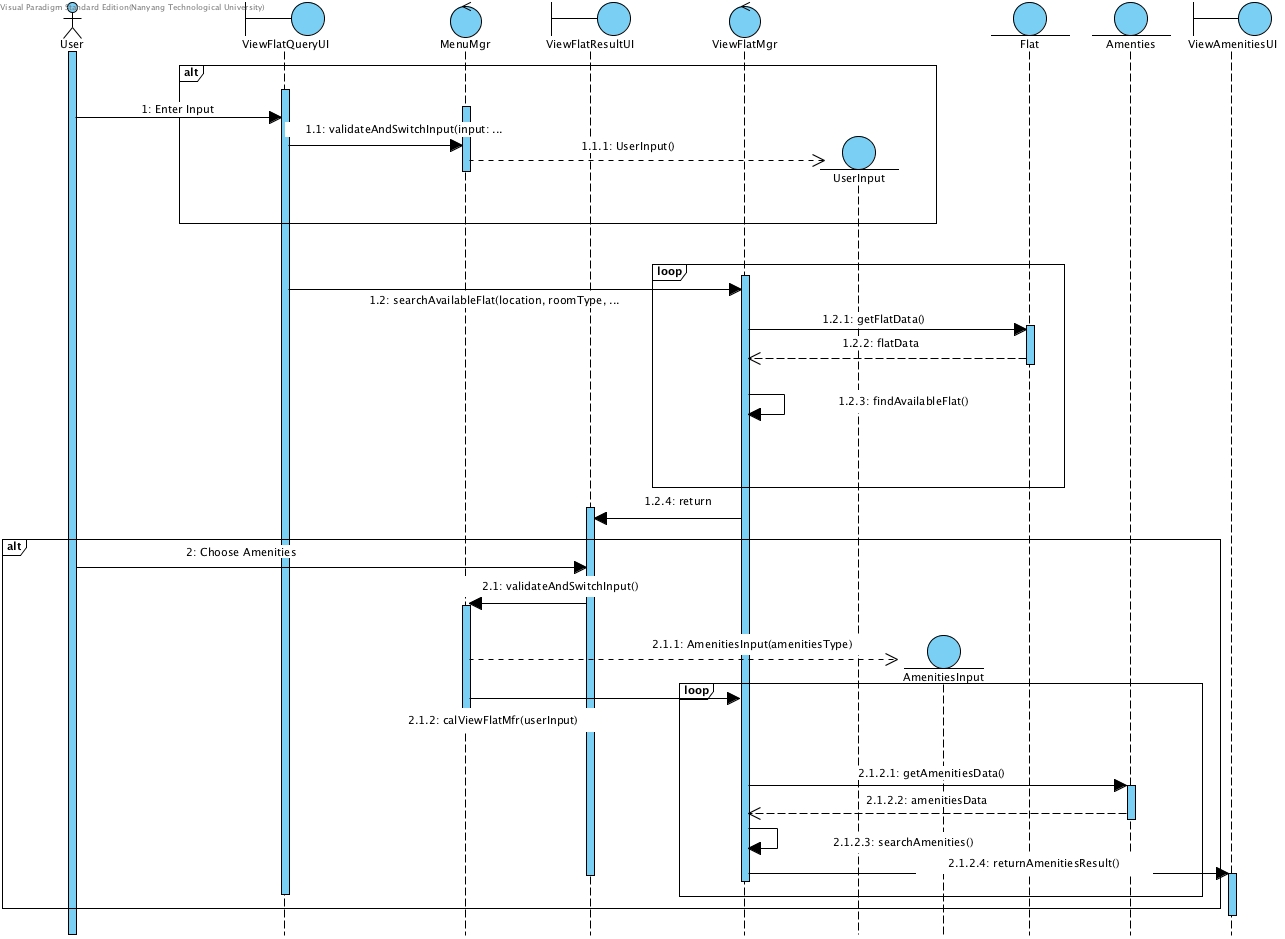
***Figure 2. Repayment calculator function***

**6.2.2 Query available grants**

****

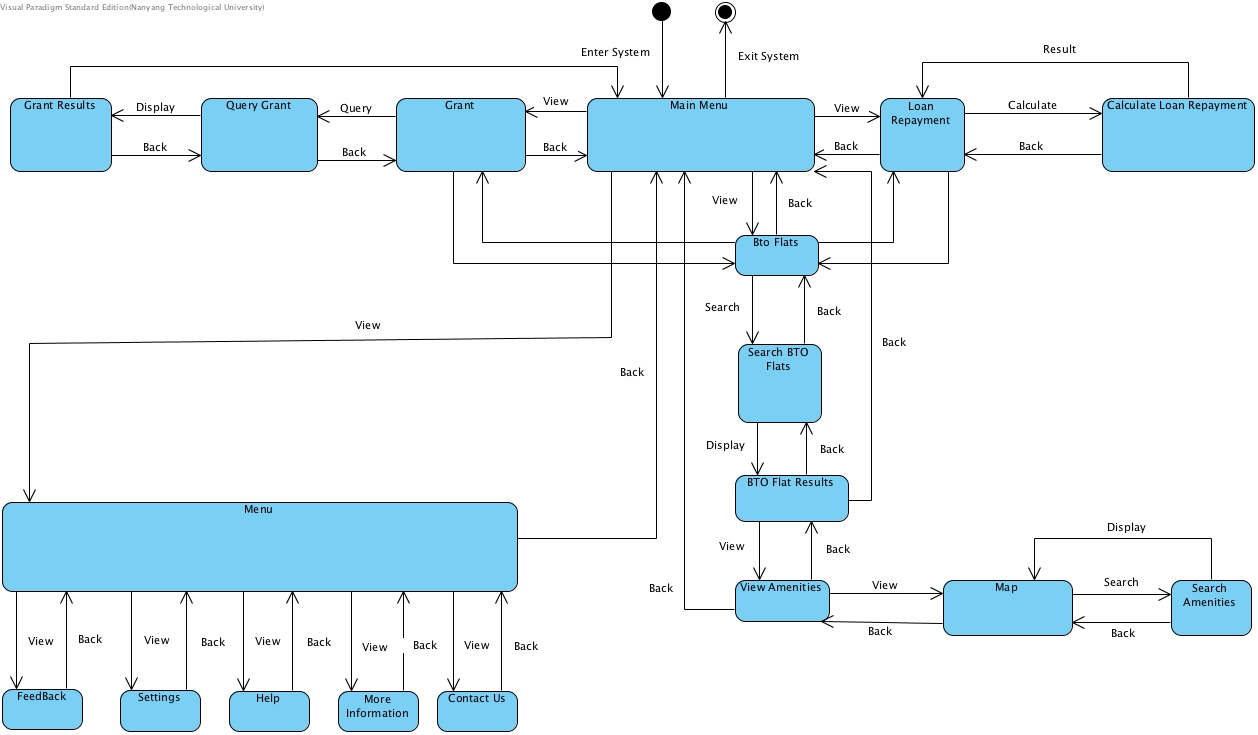
***Figure 2.1 Query available grants***

**6.2.3 View available BTO flats**

****

***Figure 2.2 View available BTO Flats***

**6.4 BTO@SG State Machine Diagram**

****

***Figure 3 State Machine Diagram***

**6.4 System Architecture**

As the core operation of the application is to retrieve data from a data store and update the user interface with newly requested information based on the user input, the Model-View-Controller (MVC) will be the most suitable architecture for BTO@SG mobile application.

In an MVC Architecture, the core functionality of the application occurs between the model and controller layers and the user interface occurs between the controllers and view layers.

As interface components are regularly updated to accommodate changing user’s requirements and technology, it is not wise to tie user interface directly with data store component. By separating the View, Controller and Model, we introduce loose coupling in our system.

In the BTO@SG, the MVC architecture consist of the UI components, service components, data component and widget components (e.g. buttons). The UI components is made up of a package of activities that reflects the View component of the pattern while service and data component(storing and retrieving) represent the model component of the pattern. The widgets component are similar to the Controller it waits for a user input and decide which Model to interact with.

Other advantages of an MVC architecture are speeding up the application development process as different programmers can work on different aspect of the pattern and then integrate it as a whole. Furthermore, modification does not affect the entire model. As components are separated, the application are easier to maintain and scalability and extensibility will not be an issue.

**7. Testing**

**7.1 Black Box Testing**

7.1.1 Repayment Calculator

**Condition assumption**:

Housing Loan simple interest rates at 0.1% per annum

Bank Loan simple interest rates at 10% per annum across all bank institution

Period of repayment available are 1, 5, 10, 15, 20, 25, 30 years.

Maximum/Minimum amount of loan at S$1,000,000/S$10,000

Interest Calculation is based on annual compound rate system.

*Equivalence Classes*

Loan Amount: I <= 9,999 (Invalid), 10,000 <= I <= 1,000,000 (valid), I >= 1,000,001 (Invalid)

*Boundary Values*

Lower Boundary: 9,999, 10,000, ~~10,001~~

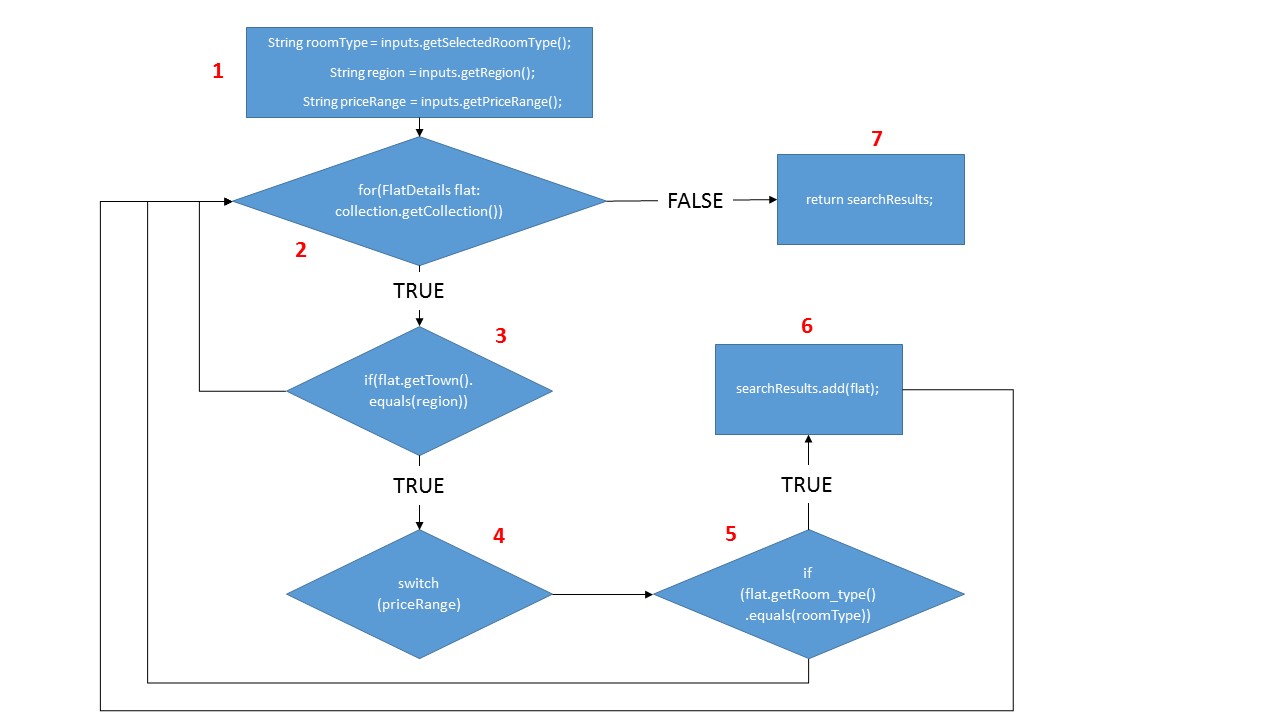
Upper boundary: ~~999,999,~~ 1,000,000, 1,000,001

*Testing*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Loan Amount | Repayment Period | Type of Loan | Expected output | Actual output |
| 10,000 | 1 | Housing | 834 | 834 |
| 10,000 | 1 | Bank | 917 | 917 |
| 10,000 | 30 | Housing | 29 | 29 |
| 10,000 | 30 | Bank | 485 | 485 |
| 9,999 | 1 | Housing | “Amount cannot be less than 10,000” | “Amount cannot be less than 10,000” |
| 9,999 | 1 | Bank | “Amount cannot be less than 10,000” | “Amount cannot be less than 10,000” |
| 9,999 | 30 | Housing | “Amount cannot be less than 10,000” | “Amount cannot be less than 10,000” |
| 9,999 | 30 | Bank | “Amount cannot be less than 10,000” | “Amount cannot be less than 10,000” |
| 1,000,001 | 1 | Housing | “Amount cannot be more than 1,000,000” | “Amount cannot be more than 1,000,000” |
| 1,000,001 | 1 | Bank | “Amount cannot be more than 1,000,000” | “Amount cannot be more than 1,000,000” |
| 1,000,001 | 30 | Housing | “Amount cannot be more than 1,000,000” | “Amount cannot be more than 1,000,000” |
| 1,000,001 | 30 | Bank | “Amount cannot be more than 1,000,000” | “Amount cannot be more than 1,000,000” |

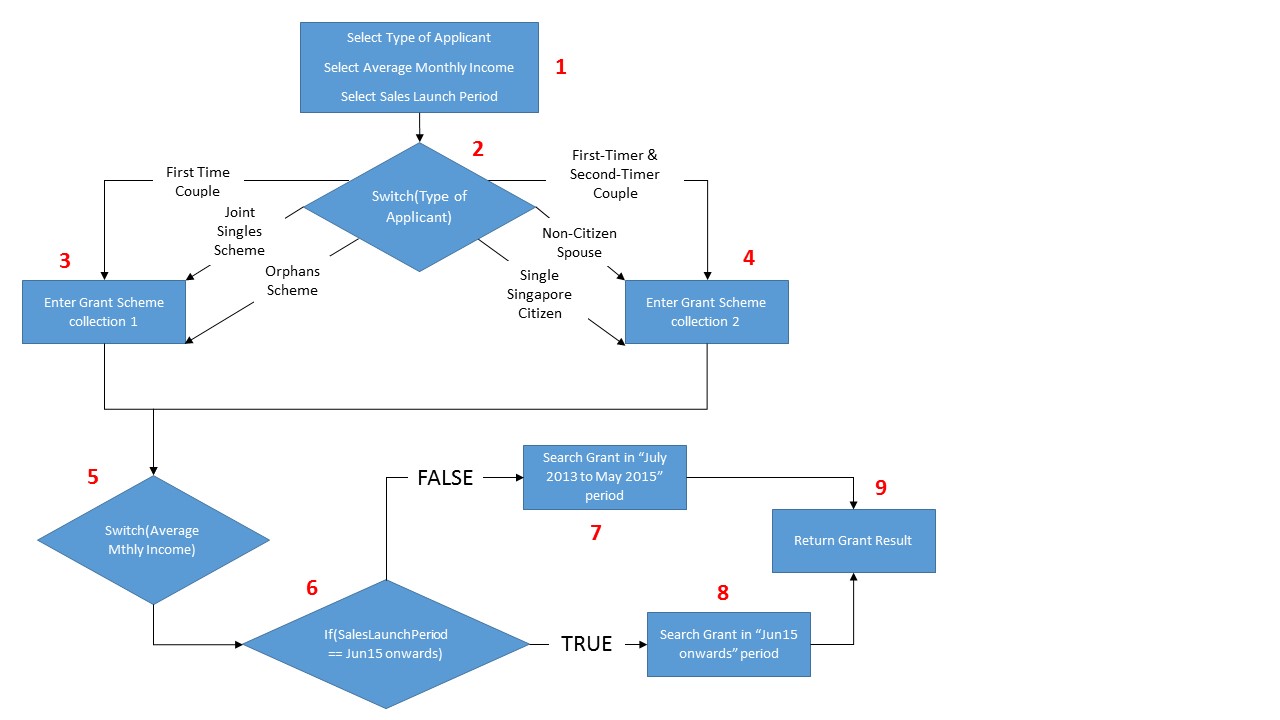
**7. 2 White Box Testing**

**7.2.1 View Flat**

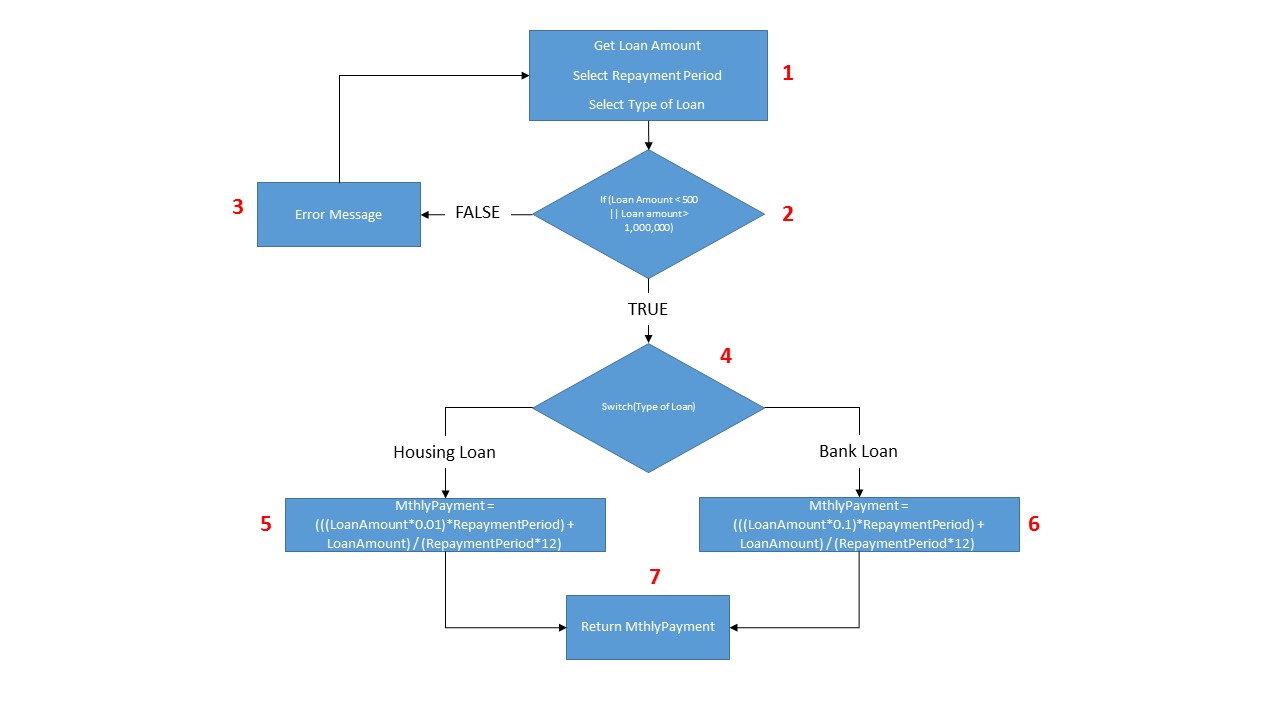
****

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **s/N** | **Basic Path** | **Test Cases** | **Expected Result** | **Test Result** |
| **1** | **1,2,3,4,5,6,2,7** | **Room Type :**  3-Room  **Region :**  Bukit Panjang  **Price Range :**  200,000 – 400,000 | Display result | Display result | |
| **2** | **1,2,3,2,7** | **Room Type :**  2-Room(Flexi)  **Region :**  Punggol  **Price Range :**  > 600,000 | “Error message – No result found, please choose another location” | “Error message – No result found, please choose another location” | |
| **3** | **1,2,3,4,5,2,7** | **Room Type :**  4-Room  **Region :**  Chua Chu Kang  **Price Range :**  400,000 – 600,000 | Display results | Display results | |

**7.2.2 Grant Query**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S/N** | **Basic Path** | **Test Cases** | **Expected Result** | **Test Result** |
| **1** | **1,2,3,5,6,7,9** | **Type of Applicant :**  First Time Couple  **Avg. Monthly Income :**  $3,001 to $3,500  **Sales Launches :**  July 2013 – May 2015 | **S$40,000** | **S$40,000** |
| **2** | **1,2,3,5,6,8,9** | **Type of Applicant :**  Orphans Scheme  **Avg. Monthly Income :**  $7,501 to $8,000  **Sales Launches :**  Jun 2015 onwards | **S$10,000** | **S$10,000** |
| **3** | **1,2,4,5,6,7,9** | **Type of Applicant :**  Non-Citizen Spouse  **Avg. Monthly Income :**  $3,751 to $4,000  **Sales Launches :**  July 2013 – May 2015 | **“Sorry ! you are not eligible for any grant !** | **Sorry ! you are not eligible for any grant !”** |

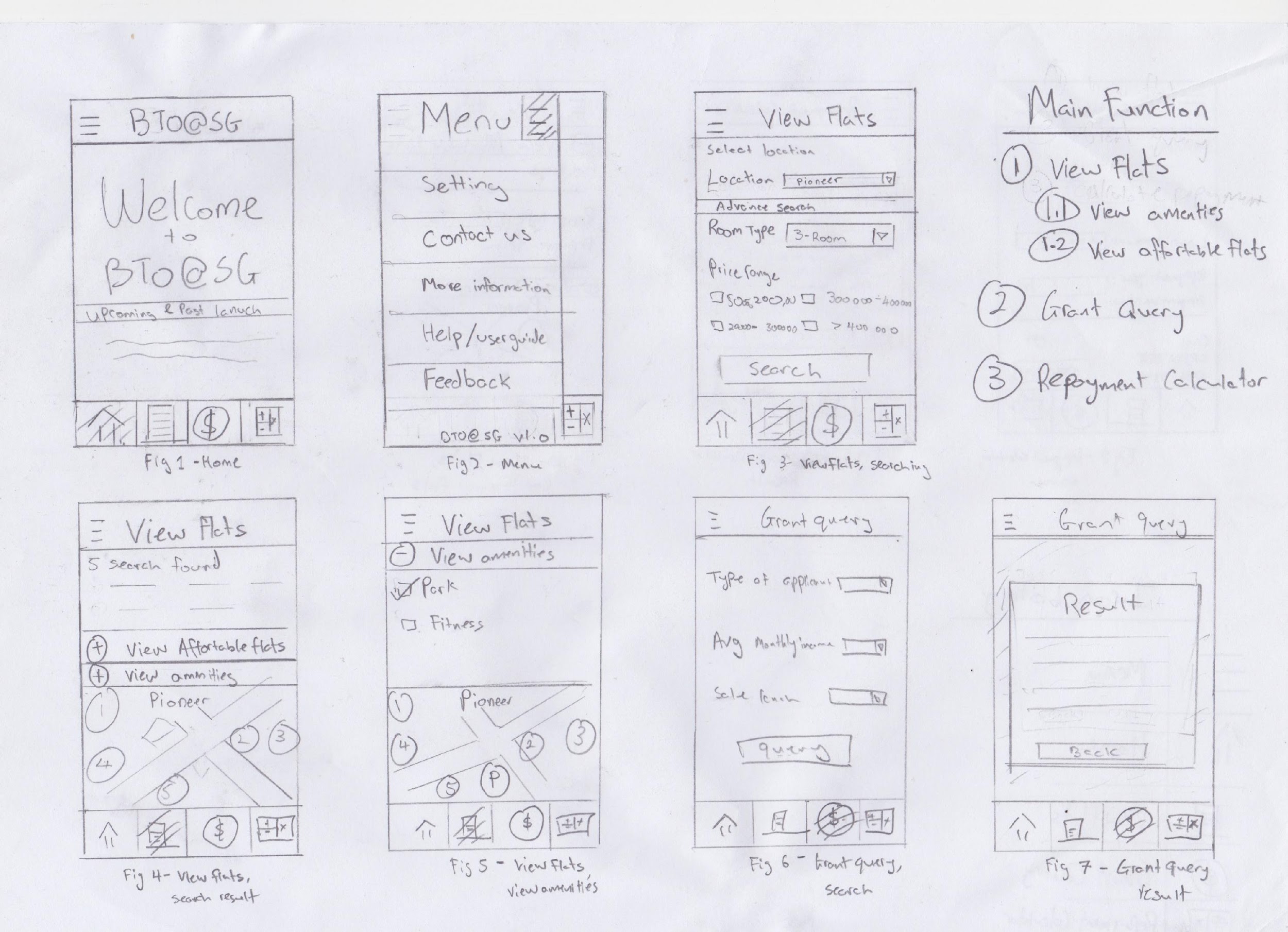
**7.2.3 Repayment Calculator**

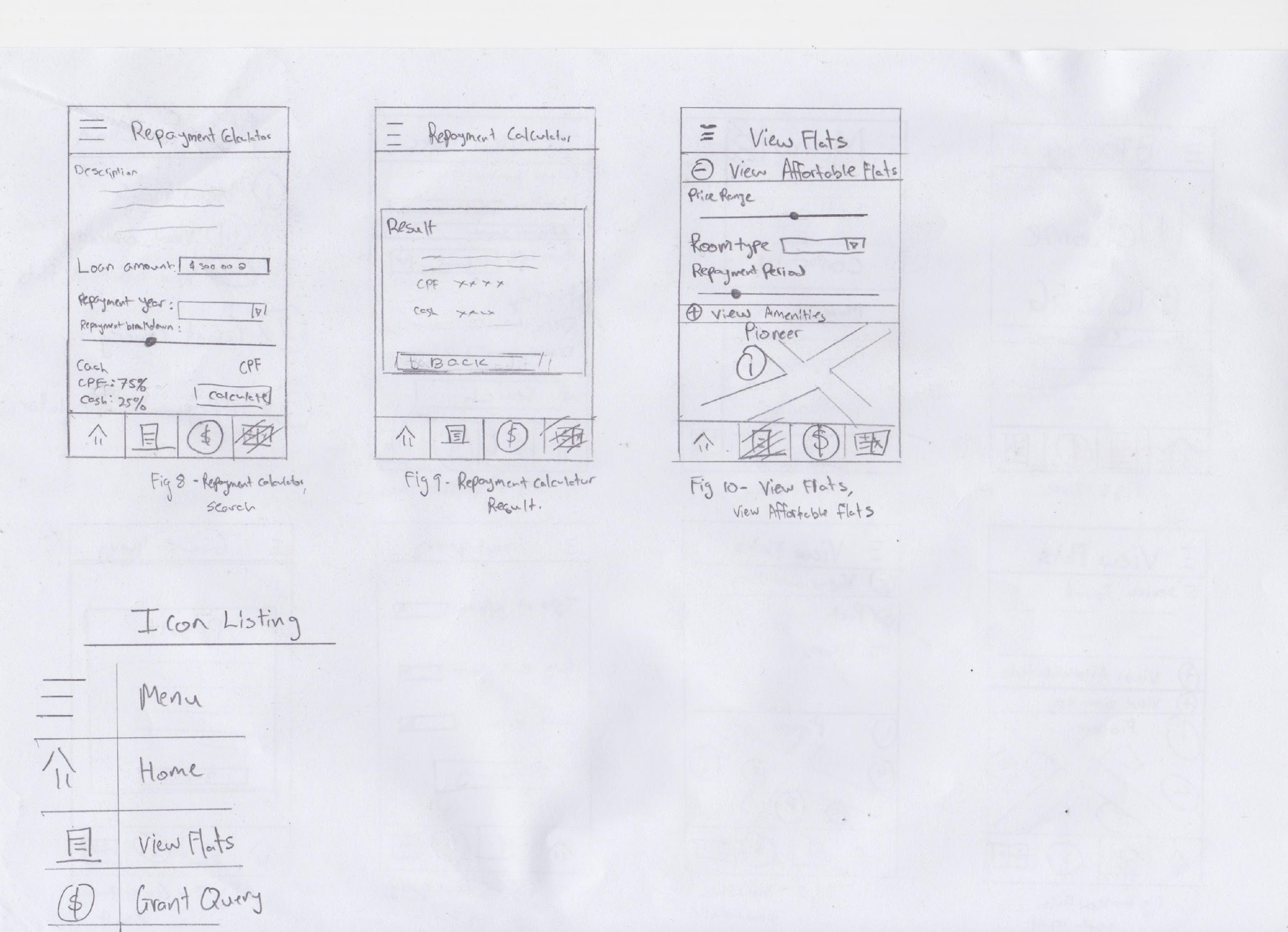
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S/N** | **Basic Path** | **Test Cases** | **Expected Result** | **Test Result** |
| **1** | **1,2,4,5,7** | **Loan Amount :**  16000  **Repayment Period :**  1  **Type of Loan :**  Housing Loan | Display MthlyPayment = 1,335 | Display MthlyPayment = 1,335 |
| **2** | **1,2,3,1,2,4,6,7** | **Loan Amount :**  250  **Repayment Period :**  5  **Type of Loan :**  Bank Loan | Display Error Message = “Amount cannot be less than 10,000” | Display Error Message = “Amount cannot be less than 10,000” |
| **3** | **1,2,4,6,7** | **Loan Amount :** 199,300  **Repayment Period :** 15  **Type of Loan :** Bank Loan | Display MthlyPayment = 4,625 | Display MthlyPayment = 4,625 |

**Appendix A: Data Dictionary**

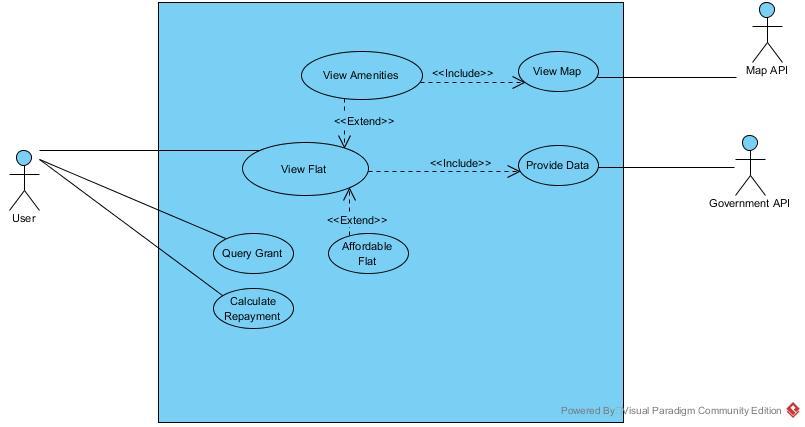
|  |  |
| --- | --- |
| Name | Terms |
| Amenities | Amenities refers to the following infrastructures:  School  Gym  Swimming complex  Clinic |
| Category of Amenity | These categories include Fitness (Swimming Complex & Gym), Health (Clinic) & Education (schools) |
| Central Provident Fund (CPF) | A compulsory comprehensive savings plans for working Singaporeans and permanent residents primarily to fund their retirement, healthcare, and housing needs. The CPF is an employment based savings scheme with employers and employees contributing a mandated amount to the Fund. |
| Flat type | It refers to the type of flat:   * 1-room to 5-room flat * Executive flat * Multi-generation Flat * Adjoined flat * 2-Room Flexi |
| Schemes | This refers to the schemes provided by the government to help buyer reduce cash flow or cash layout (e.g. Fiancé/ Fiancée schemes) |
| Grants | Grants refers to a sum eligible from the government if one meets certain criteria (CPF housing grants, Staggered Down Payment Scheme) |
| Build-To-Order(BTO) | BTO refers to the Housing Development Board’s allocation scheme that offers flexibility in timing and location for owners |
| User | Refers to person who is using the application |
| Affordable Flat | Affordable Flat is defined as the flat(s) deem as the desired monthly installment and repayment time frame chosen by the user |
| Repayment | Repayment is a term that refers to the monthly instalment the user has to pay to afford the flat. |
| Sales Launch | Sales Launch refers to the time frame (quarterly) where new flats available for sale. |
| Principal amount | It refers to the amount borrowed or the amount yet to payoff from the loan |
| Annual Interest Rate | It refers to the yearly interest rates needed to be paid (cost of loan) |
| Loan Period | The period taken to pay off the loan |
| Type of Applicant | Type of applicant refers to the different category of applicants:   * First-Timer Applicants * Second-Timer Applicants * First-Timer and Second-Timer Couple Applicants * Non-Citizen Spouse Applicants * Single Singapore Citizen Applicants * Joint Singles Scheme or Orphans Applicants * Single Mother Applicants |
| Monthly Household Income | It refers to the monthly gross income of the household excluding CPF deduction. |

**Appendix B: UI Low-fi Mock-up**





**Appendix C: Use case Diagrams and description**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | **BTO@SG App** | | | |
| Use Case Name: | View flat | | | |
| Created By: | Long Siang | | Last Updated By: | Thomas |
| Date Created: | 04 Sep 2016 | | Date Last Updated: | 15 Sep 2016 |
| Actor: | | - User | | |
| Description: | | - User is able to see the current available Build-To-Order(BTO) flats released by Housing Development Board (HDB) based on location area | | |
| Preconditions: | | - The user has loaded the application | | |
| Postconditions: | | - Interface to list the available BTO flats based on the input given. | | |
| Priority: | | - Very High (Main function of the app) | | |
| Frequency of Use: | | - Most Frequent | | |
| Flow of Events: | | 1. User to click on “View available BTOs” button.  2. Interface will display to the “View Available BTOs” page.  3. The interface will prompt user to select location region of interest from a drop-down selection box, preferred room-type from a drop down box and price range from checkbox.  4. User must choose his/her preferred location region (e.g. Jurong East, Pioneer, etc).  5. User must choose his/her preferred room type (e.g. 3-room flat, 4-room flat, etc).  6. User must select his/her preferred price range.  7. User click on “Search”  8. Interface will show a list of available BTOs block based on his/her preference  9. User to click on preferred block  10. Interface will extend down and show a list of BTOs flat available, together with its price range and room type. | | |
| Alternative Flows: | | AF1 - When user wants to search a new preference combination   1. User click on “back” 2. Return back to step 3. | | |
| Exceptions: | |  | | |
| Includes: | |  | | |
| Special Requirements: | |  | | |
| Assumptions: | |  | | |
| Notes and Issues: | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | **BTO@SG App** | | | |
| Use Case Name: | View Amenities | | | |
| Created By: | Long Siang | | Last Updated By: | Long Siang |
| Date Created: | 04 Sep 2016 | | Date Last Updated: | 15 Sep 2016 |
| Actor: | | - User | | |
| Description: | | - User is able to know available amenities surrounding his/her preferred location area. | | |
| Preconditions: | | - Interface has listed available BTOs flat according to user’s preference | | |
| Postconditions: | | - Interface to show map of preferred location area and pin tacks of nearby amenities | | |
| Priority: | | High | | |
| Frequency of Use: | | Frequent | | |
| Flow of Events: | | 1. User click “View Amenities” button from the list of available BTOs block  2. Application will show a map through a pop-up. The map will have pin tacks, indicating different amenities around the block  3. User to mouse over pin tacks  4. Interface to fade in name and category of amenity (e.g. Chua Chu Kang Polyclinic, Polyclinic) | | |
| Alternative Flows: | | AF1: When User wants to explore another area within 10km of the selected block  - User to click and drag map to the direction he/she wants to explore.  -Pin tacks will pop onto the map if amenities is available on the new area of search.  -Return to step 3.  AF2: When user wants to know if there is a specific amenity available.  - User to click on “+” button beside the “View Amenities” title.  - Interface will extend down to show category of amenities (Checkbox)  - User to check the box of the category of amenities he/she prefer  - Interface will filter the pin tacks on the map to show only the category of amenities the user choose.  -Return to step 3. | | |
| Exceptions: | | EX1: When User wants to search for other BTOs block  - User to click on “X” button at the top right hand corner to close the map  - User will be directed back to the “View Available BTOs” page | | |
| Includes: | |  | | |
| Special Requirements: | |  | | |
| Assumptions: | |  | | |
| Notes and Issues: | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | **BTO@SG App** | | | |
| Use Case Name: | Calculate Repayment | | | |
| Created By: | Long Siang | | Last Updated By: | Thomas |
| Date Created: | 04 Sep 2016 | | Date Last Updated: | 15 Sep 2016 |
| Actor: | | * User | | |
| Description: | | * To assist users by projecting house repayment amount with cash and CPF breakdown. | | |
| Preconditions: | | - The user has loaded the application | | |
| Postconditions: | | - Interface to show repayment details according to user’s loan amount and years of repayment | | |
| Priority: | | Medium | | |
| Frequency of Use: | | Less Frequent (Used by first-time buyers) | | |
| Flow of Events: | | 1. User to click on “Repayment Calculator” button.  2. Interface will switch to the “Calculate repayment” page.  3. User to enter loan amount.  4. User to choose amount of year he/she wishes to complete the repayment from the drop-box selection box.  5. User to adjust the slider to the percentage between cash and Central Provident Fund (CPF) he/she wish to pay for the repayment.  6. User to click “Calculate”.  7. Interface will have a pop-out window that shows the result of the user’s preferred repayment plan and the amount required per year for the repayment for both cash and CPF category. | | |
| Alternative Flows: | | AF1: User wants to do a new calculation.  1. User click on “back”  2. Return to step 3. | | |
| Exceptions: | |  | | |
| Includes: | |  | | |
| Special Requirements: | |  | | |
| Assumptions: | |  | | |
| Notes and Issues: | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | **BTO@SG App** | | | |
| Use Case Name: | View Map | | | |
| Created By: | Zi Yang | | Last Updated By: | Zi Yang |
| Date Created: | 04 Sep 2016 | | Date Last Updated: | 04 Sep 2016 |
| Actor: | | User and Map API | | |
| Description: | | When user searched for flats, he/she can view amenities surrounding the flats. | | |
| Preconditions: | | User must searched for list of available flats. | | |
| Postconditions: | | User will view amenities within 10km of the searched flat. | | |
| Priority: | | High | | |
| Frequency of Use: | | Frequent | | |
| Flow of Events: | | 1. User clicks on ‘View Amenities’ tab.  2. Application displays checkboxes to show categories of amenities.  3. User selects amenities he is interested in.  4. Application display the selected amenities. | | |
| Alternative Flows: | |  | | |
| Exceptions: | | EX1: No amenity was chosen.    The map is updated with no amenity shown. | | |
| Includes: | |  | | |
| Special Requirements: | |  | | |
| Assumptions: | |  | | |
| Notes and Issues: | |  | | |
|  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | **BTO@SG App** | | | |
| Use Case Name: | Provide Data | | | |
| Created By: | Zi Yang | | Last Updated By: | Zi Yang |
| Date Created: | 04 Sep 2016 | | Date Last Updated: | 04 Sep 2016 |
| Actor: | | Government API and User | | |
| Description: | | The data in application database is updated with new data from government API. | | |
| Preconditions: | | The user has loaded the application. | | |
| Postconditions: | | The application is updated with new data retrieved from government API. | | |
| Priority: | | High | | |
| Frequency of Use: | | Frequent | | |
| Flow of Events: | | 1. User starts up application.  2. Application runs in the background to connect to government data API.  3. Government API provides data.  4. Application database will be updated with new data. | | |
| Alternative Flows: | | AF1: No internet connection    1. The application will display the message “Connection error! Check your internet connection and try again.”  2. User will reconnect his device to the internet.  3. Return to step 2. | | |
| Exceptions: | | EX1: No new data    The application will run on existing database. | | |
| Includes: | |  | | |
| Special Requirements: | |  | | |
| Assumptions: | |  | | |
| Notes and Issues: | |  | | |
|  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | **BTO@SG App** | | | |
| Use Case Name: | Query Grant | | | |
| Created By: | Zi Yang | | Last Updated By: | Zi Yang |
| Date Created: | 04 Sep 2016 | | Date Last Updated: | 04 Sep 2016 |
| Actor: | | User | | |
| Description: | | The user can check which grant he/she is eligible for based on his/her average monthly household income. | | |
| Preconditions: | | User must select ‘Query Grant’ icon on the application. | | |
| Postconditions: | | User will view list of eligible grants. | | |
| Priority: | | High | | |
| Frequency of Use: | | Frequent | | |
| Flow of Events: | | 1. User selects ‘Query Grant’ icon.  2. Application will display drop-down box for user to make selection.  3. User selects applicant type from drop-down list.  4. User selects average monthly household income from drop-down list.  5. User selects sale period from drop down list.  6. User clicks ‘Query’ button.  7. Application displays list of grant eligible to user. | | |
| Alternative Flows: | | AF1: User clicks ‘Search’ button without making selection from either or both of the drop-down list    1. Application displays the message “Fields marked with \* cannot be blank.” Please fill in the information and resubmit.”  2. Return to step 2. | | |
| Exceptions: | | EX1: User did not click on ‘Query’ button.    1. Application remains on that page. | | |
| Includes: | |  | | |
| Special Requirements: | |  | | |
| Assumptions: | |  | | |
| Notes and Issues: | |  | | |