**Software Design**

**for**

**Campus Management Application**

**at**

**World Skill Center (WSC)**

**Design Feature - Infrastructure Module**

Logo

Description automatically generated

**Sustainable Outreach And Universal Leadership (SOUL) Limited**

**Version no 0.1**

**VERSION HISTORY**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Date** | **Version No.** | **Created By** | **Updated Section** | **Updated By** |
| 25-01-2023 | Version 0.1 | Shiv Kaul |  |  |

Contents

[Introduction 6](#_Toc9884)

[Design Overview: 6](#_Toc7487)

[i) Master Screens 6](#_Toc16157)

[1. Land 6](#_Toc17951)

[1.1 Overview 6](#_Toc12662)

[1.2 Description of Problem 6](#_Toc23499)

[1.3 Dependency 7](#_Toc3950)

[1.4 Flow Chart 7](#_Toc13713)

[1.5 Use Case Diagram 7](#_Toc13621)

[1.6 Form Elaboration 8](#_Toc144)

[1.7 Methods 8](#_Toc7767)

[1.8 Screen Authorization 9](#_Toc26987)

[2. Buildings 9](#_Toc28457)

[2.1 Overview 9](#_Toc27759)

[2.2 Description of Problem 9](#_Toc16072)

[2.3 Dependency 9](#_Toc18478)

[2.4 Flow Chart 9](#_Toc19935)

[2.5 Use Case Diagram 9](#_Toc11668)

[2.6 Form Elaboration 10](#_Toc30926)

[2.7 Methods 11](#_Toc25089)

[2.8 Screen Authorization 11](#_Toc26610)

[3. Building Room 11](#_Toc27383)

[3.1 Overview 11](#_Toc22283)

[3.2 Description of Problem 12](#_Toc22611)

[3.3 Dependency 12](#_Toc21795)

[3.4 Flow Chart 12](#_Toc10537)

[3.5 Use Case Diagram 12](#_Toc10340)

[3.6 Form Elaboration 13](#_Toc26818)

[3.7 Methods 14](#_Toc7596)

[3.8 Screen Authorization 14](#_Toc2553)

[3a Building type Room 14](#_Toc7995)

[3a.1 Overview 14](#_Toc13266)

[3a.2 Description of Problem 14](#_Toc31588)

[3a.3 Dependency 14](#_Toc19802)

[3a.4 Flow Chart 15](#_Toc29539)

[3a.5 Use Case Diagram 15](#_Toc11194)

[3a.6 Form Elaboration 15](#_Toc18639)

[3a.7 Screen Authorization 16](#_Toc3832)

[3b Type of Residence 16](#_Toc21814)

[3b.1Overview 16](#_Toc9597)

[3b.2 Description of Problem 16](#_Toc17422)

[3b.3 Dependency 16](#_Toc4313)

[3b.4 Flow Chart 16](#_Toc27886)

[3b.5 Use Case Diagram 17](#_Toc7550)

[3b.6 Form Elaboration 17](#_Toc22734)

[3b.7 Screen Authorization 17](#_Toc11664)

[3c Residence Type 17](#_Toc4647)

[3c.1Overview 17](#_Toc5884)

[3c.2 Description of Problem 18](#_Toc7971)

[3c.3 Dependency 18](#_Toc30573)

[3c.4 Flow Chart 18](#_Toc15936)

[3c.5 Use Case Diagram 18](#_Toc21731)

[3c.6 Form Elaboration 18](#_Toc274)

[3c.7 Screen Authorization 19](#_Toc22513)

[3d Residence Eligibility 19](#_Toc32260)

[3d.1Overview 19](#_Toc5352)

[3d.2 Description of Problem 20](#_Toc22866)

[3d.3 Dependency 20](#_Toc17284)

[3d.4 Flow Chart 20](#_Toc9866)

[3d.5 Use Case Diagram 20](#_Toc212)

[3d.6 Form Elaboration 21](#_Toc28499)

[3d.7 Screen Authorization 21](#_Toc21866)

[ii) Transactional Screens 21](#_Toc20825)

[1. Application for Residence 21](#_Toc18976)

[1.1 Overview 21](#_Toc29824)

[1.2 Description of Problem 21](#_Toc26612)

[1.3 Dependency 21](#_Toc25552)

[1.4 Flow Chart 22](#_Toc9155)

[1.5 Use Case Diagram 22](#_Toc26306)

[1.6 Form Elaboration 23](#_Toc31987)

[1.7 Methods 23](#_Toc9670)

[1.8 Screen Authorization 23](#_Toc3656)

[2. Residence Allotment 24](#_Toc31813)

[2.1 Overview 24](#_Toc2674)

[2.2 Description of Problem 24](#_Toc11117)

[2.3 Dependency 24](#_Toc10787)

[2.4 Flow Chart 24](#_Toc31870)

[2.5 Use Case Diagram 24](#_Toc32677)

[2.6 Form Elaboration 25](#_Toc5756)

[2.7 Methods 26](#_Toc1093)

[2.8 Screen Authorization 27](#_Toc10861)

[3. Employee: Residence Alloted(Child Table) 28](#_Toc5660)

[3.1 Overview 28](#_Toc29325)

[3.2 Description of Problem 28](#_Toc4334)

[3.3 Dependency 28](#_Toc9247)

[3.4 Flow Chart 28](#_Toc17212)

[3.5 Form Elaboration 28](#_Toc12483)

[3.6 Screen Authorization 29](#_Toc18627)

[4. Application for Residence De-Allotment 29](#_Toc23416)

[4.1 Overview 29](#_Toc24048)

[4.2 Description of Problem 29](#_Toc1713)

[4.3 Dependency 29](#_Toc7200)

[4.4 Flow Chart 29](#_Toc19612)

[4.5 Use Case Diagram 30](#_Toc13431)

[4.6 Form Elaboration 30](#_Toc12463)

[4.7 Methods 31](#_Toc21274)

[4.8 Screen Authorization 31](#_Toc20307)

[5. Residence De-Allotment 31](#_Toc30254)

[5.1 Overview 31](#_Toc7482)

[5.2 Description of Problem 32](#_Toc19116)

[5.3 Dependency 32](#_Toc14921)

[5.4 Flow Chart 32](#_Toc9351)

[5.5 Use Case Diagram 32](#_Toc6077)

[5.6 Form Elaboration 33](#_Toc20668)

[5.7 Methods 34](#_Toc13451)

[5.8 Screen Authorization 34](#_Toc18606)

[6. Residence Change request (Workflow) 34](#_Toc18880)

[6.1 Overview 34](#_Toc22919)

[6.2 Description of Problem 35](#_Toc20205)

[6.3 Dependency 35](#_Toc10101)

[6.4 Flow Chart 35](#_Toc16099)

[6.5 Use Case Diagram 35](#_Toc29689)

[6.6 Form Elaboration 36](#_Toc1860)

[6.7 Methods 37](#_Toc1966)

[6.8 Screen Authorization 38](#_Toc8635)

# Introduction

The Infrastructure Module aims at providing an efficient, secure, and user-friendly platform to manage Land, Building, Allotment of residences, De-allotment of residences, change request of residences. It is an integrated system that provides all the necessary tools to manage the various aspects of running operations in regard to building infrastructure. This web-based solution enables administrators and employees to easily track and manage activities related to rooms in a building, allotment of residence, de-allotment of residence and change request for residence.

Scope of project:

1. Land
2. Buildings
3. Building Room
4. Application for Residence
5. Residence Allotment
6. Application for Residence De-Allotment
7. Residence De-Allotment
8. Residence Change request

# Design Overview:

# i) Master Screens

## 1. Land

### 1.1 Overview

**The Land document will hold all the data of any land, like owner details, plot** number, land complete address, land valuation, BDA Approval, Sate of planned development, etc.

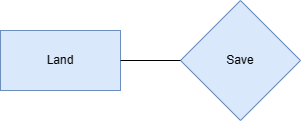
### 1.2 Description of Problem

Create a page containing all information about the land, including owner details,plot number, address, land valuation, date of planned development, land blueprint, etc.

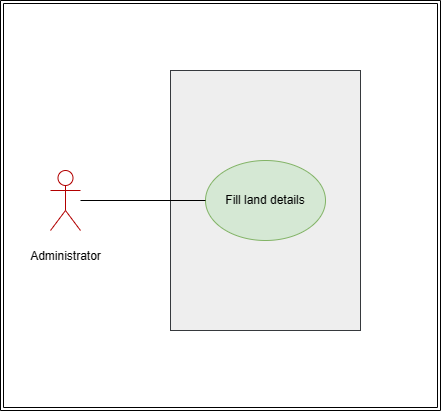
### 1.3 Dependency

Not dependent on anything other than being filled and saved by the administrator.

### 1.4 Flow Chart

-

### 1.5 Use Case Diagram



**Brief Description -**

Administrator: Able to create Land by filling out the necessary details about land.

### 1.6 Form Elaboration

|  |  |  |  |
| --- | --- | --- | --- |
| **Label (Fields)** | **Type (Fields)** | **Options (Fields)** | **Mandatory** |
| **Land Owner Details** | | | |
| Name | Data |  | Y |
| Address | Data |  | Y |
| Phone | Data |  | Y |
| Email | Data |  | Y |
| Property on lease | Select | Yes  No | Y |
| Lease | Attach |  |  |
| **Land Details** | | | |
| Plot number | Data |  | Y |
| Land Complete Address | Small Text |  | Y |
| Pin code | Data |  |  |
| Land Size | Data |  | Y |
| Land Valuation | Currency |  |  |
| Start Date | Date |  | Y |
| End Date | Date |  | Y |
| Land Use | Drop down | Academic  Residential  Both | Y |
| Date of Planned Development | Date |  |  |
| BDA Approval | Attach |  |  |
| Land Blueprint | Attach |  |  |

### **1.7 Methods**

**land.js:**

To validate that the end date is not before the start date in the UI

**land.py:**

1. def date: To validate if the start date is not after the end date, if it is then throw an error
2. def pincode: Validation for pin code and length, If less than or greater than 6 throw an error.

### 1.8 Screen Authorization

|  |  |
| --- | --- |
| **User** | **Type of access** |
| Administrator | Read/ Write/ Modify |

## Buildings

### 2.1 Overview

All the building information shall be stored here. The buildings can be residential, academic or both. The actions on this screen shall be performed by the administrator only.

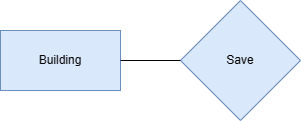
### 2.2 Description of Problem

Create a page on which the administrator can fill in and store the building details. E.g: Building name, building address, number of floors, type of building, etc.

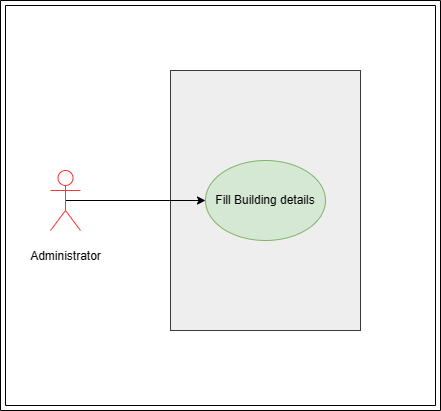
### 2.3 Dependency

Not dependent on anything other than being filled and saved by the administrator.

### 2.4 Flow Chart

****

### 2.5 Use Case Diagram

****

**Brief Description -**

Administrator: Able to create building by filling the fields and then submit.

### 2.6 Form Elaboration

|  |  |  |  |
| --- | --- | --- | --- |
| **Label (Fields)** | **Type (Fields)** | **Options (Fields)** | **Mandatory** |
| **Building Address** | | | |
| Land Plot number | Link | Land |  |
| Land Address | Data |  |  |
| Building Name | Data |  | Y |
| Building Address | Data |  | Y |
| Post Office | Data |  | Y |
| District | Link | Districts | Y |
| State | Data |  | Y |
| Pin Code | Data |  | Y |
| Building on Lease | Drop down | Yes  No | Y |
| Lease | Attach |  | Y |
| **Building Details** | | | |
| Building Type | Data | Academic  Non-Academic | Y |
| Total Rooms | Data |  |  |
| Total Floors | Date |  | Y |
| Facing | Date | North  South  East  West | Y |
| Campus | Data |  | Y |
| Start date | Date |  |  |
| End date | Date |  |  |

### **2.7 Methods**

**buildings.js:**

1. To set filter on the Plot number field, so as to fetch only those buildings which are between the start and end date of the land with respect to today’s date.
2. To validate end date is not before the start date in UI

**buildings.py:**

1. def date: To validate if the start date is not after the end date
2. def pincode: Validation for pin code length
3. SQL query to fetch only those buildings which are between the start and end date of the land with respect to today’s date

### 2.8 Screen Authorization

|  |  |
| --- | --- |
| **User** | **Type of access** |
| Administrator | Read/ Write/ Modify |

## Building Room

### 3.1 Overview

This screen shall be able to create rooms and store the features of every room while also assigning a room number.

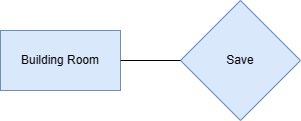
### 3.2 Description of Problem

Create a page containing all the information about the room in a building, including type of room, room number, vacancy status, and details about the features like parking and garden availability.

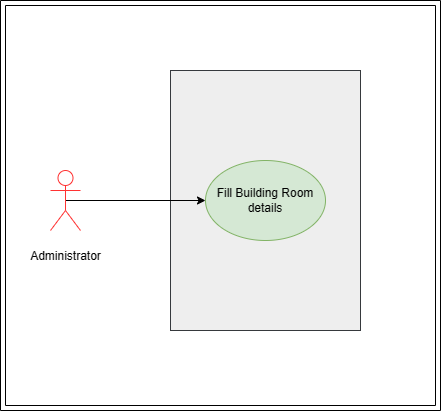
### 3.3 Dependency

Not dependent on anything other than being filled and saved by the administrator.

### 3.4 Flow Chart

****

### 3.5 Use Case Diagram

****

**Brief Description -**

Administrator: Able to create building room by filling the fields and then submit.

### 3.6 Form Elaboration

|  |  |  |  |
| --- | --- | --- | --- |
| **Label (Fields)** | **Type (Fields)** | **Options (Fields)** | **Mandatory** |
| **Building Details** | | | |
| Building Name | Link | Buildings |  |
| Building Type | Data |  |  |
| Building Address | Data |  | Y |
| Building's Land Plot number | Data |  | Y |
| Building's Land Address | Data |  | Y |
| District | Data |  | Y |
| State | Data |  | Y |
| Pin Code | Data |  | Y |
| **Basic Information** | | | |
| Room No. | Data |  | Y |
| Floor | Data | Ground  1 ….. |  |
| Type of Room | Link | Building type Room | Y |
| Type of Residence | Link | Type of Residence | Y |
| Residence Type Name | Link | Residence Type | Y |
| Vacancy status | Drop down | Vacant  Not Vacant |  |
| Allotment Status | Data |  |  |
| Employee Allotment Status | Data |  |  |
| **Residence Details** | | | |
| Start date | Date |  |  |
| End date | Date |  |  |
| Description | Data |  |  |
| Unit Area (sq. m.) | Data |  |  |
| Air Conditioning | Data |  |  |
| Residence Orientation | Data |  |  |
| Lawn/Garden Available | Data |  |  |
| Lawn/Garden Area (sq. m.) | Data |  |  |
| Roof type | Data |  |  |
| Parking Available | Data |  |  |
| Parking Type | Data |  |  |
| Parking Vehicle | Data |  |  |
| Parking Area (sq. m.) | Data |  |  |

### **3.7 Methods**

**building\_room.js:**

* + 1. To filter residence type name in the link field based on selected type of residence
    2. To filter buildings which are currently between start and end date
    3. To validate end date is not before the start date

**building\_room.py:**

1. To filter buildings which are currently between start and end date
2. def duplicate: To check for any duplicate records within a building regarding room no. or room type
3. def dateValidate: To validate if the start date is not after the end date in allotable room type

### 3.8 Screen Authorization

|  |  |
| --- | --- |
| **User** | **Type of access** |
| Administrator | Read/ Write/ Modify |

## 3a Building type Room

### 3a.1 Overview

**This screen helps define the type of the room in a building, for example if it is a classroom or residential or conference hall.**

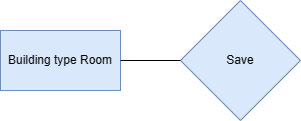
### 3a.2 Description of Problem

Create a page containing information of types of room eg. Classroom or residential or any other.

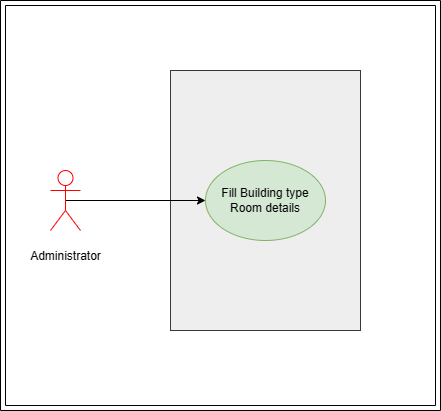
### 3a.3 Dependency

Not dependent on anything other than being filled and saved by the administrator.

### 3a.4 Flow Chart

****

### 3a.5 Use Case Diagram

****

**Brief Description -**

Administrator: Able to create building type room by filling the fields and then submitting.

### 3a.6 Form Elaboration

|  |  |  |  |
| --- | --- | --- | --- |
| **Label (Fields)** | **Type (Fields)** | **Options (Fields)** | **Mandatory** |
| Type of Room | Data |  | Y |
| Allotment Status | Drop down | Allottable  Non Allottable | Y |

### 3a.7 Screen Authorization

|  |  |
| --- | --- |
| **User** | **Type of access** |
| Administrator | Read/ Write/ Modify |

## 3b Type of Residence

### 3b.1Overview

**This screen helps in the classification of residence, whether it shall be a quarter or a duplex or even a villa.**

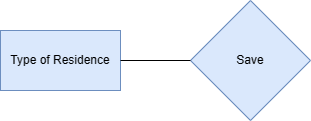
### 3b.2 Description of Problem

To create a page in which the administrator can store details regarding the type of residences available. E.g: Quarter or Duplex or Villa.

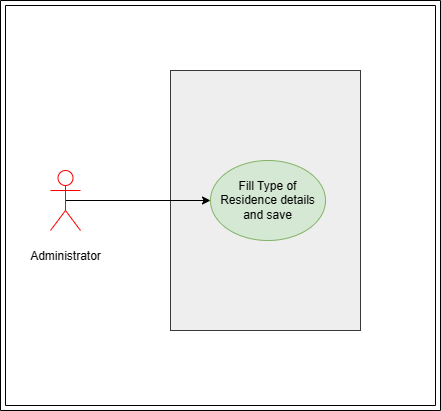
### 3b.3 Dependency

Not dependent on anything other than being filled and saved by the administrator.

### 3b.4 Flow Chart

****

### 3b.5 Use Case Diagram

****

**Brief Description -**

Administrator: Able to create type of residence by filling the fields and then submitting.

### 3b.6 Form Elaboration

|  |  |  |  |
| --- | --- | --- | --- |
| **Label (Fields)** | **Type (Fields)** | **Options (Fields)** | **Mandatory** |
| Type of Residence | Data |  | Y |

### 3b.7 Screen Authorization

|  |  |
| --- | --- |
| **User** | **Type of access** |
| Administrator | Read/ Write/ Modify |

## 3c Residence Type

### 3c.1Overview

**This screen helps in further classification of residences in a building room based on types of residence. Example: 2BHK Quarter or 4BHK Villa and so on.**

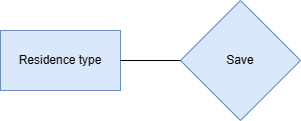
### 3c.2 Description of Problem

Create a page which can hold detailed information about the different types of residences available to the applicant. E.g:  **2BHK Quarter 4BHK Villa, etc.**

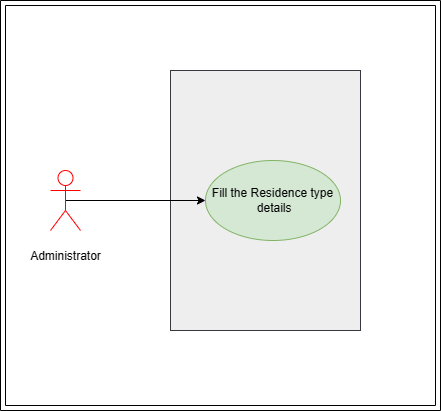
### 3c.3 Dependency

Not dependent on anything other than being filled and saved by the administrator.

### 3c.4 Flow Chart

****

### 3c.5 Use Case Diagram

****

**Brief Description -**

Administrator: Able to create Residence type by filling up the fields and then submitting.

### 3c.6 Form Elaboration

|  |  |  |  |
| --- | --- | --- | --- |
| **Label (Fields)** | **Type (Fields)** | **Options (Fields)** | **Mandatory** |
| Type of Residence | Link | Type of Residence | Y |
| Residence Type name | Data |  | Y |
| Description | Data |  |  |
| Unit Area (sq. m.) | Data |  |  |
| Air Conditioning | Drop down | Yes  No | Y |
| Parking Available | Drop down | Yes  No | Y |
| Parking Type | Drop down | On Road  Ramp Parking  Lot Parking  Garage  Shed |  |
| Parking Vehicle | Drop down | Two Wheeler  Four Wheeler |  |
| Parking Area (sq. m.) | Data |  |  |
| Residence Orientation | Drop down | Multi storey  One Unit Up, One Down  Side-By-Side | Y |
| Roof type | Drop down | Flat  Gable  Hip  Gambrel  Mansard |  |
| Lawn/Garden Available | Drop down | Yes  No | Y |
| Lawn/Garden Area (sq. m.) | Data |  |  |
| Eligibility | Table | Residence Eligibility |  |

### 3c.7 Screen Authorization

|  |  |
| --- | --- |
| **User** | **Type of access** |
| Administrator | Read/ Write/Modify |

## 3d Residence Eligibility

### 3d.1Overview

Residence Eligibility is a child table which shall store the pay grade cutoff of the employee for any specific residence to be alloted.

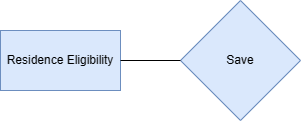
### 3d.2 Description of Problem

Create a table which will store residence eligibility in terms of employee pay grade.

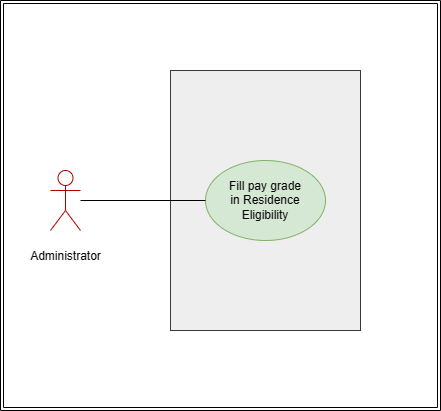
### 3d.3 Dependency

Not dependent on anything other than being filled and saved by the administrator.

### 3d.4 Flow Chart

****

### 3d.5 Use Case Diagram

****

**Brief Description -**

Administrator: Able to fill the pay grade in residence eligibility.

### 3d.6 Form Elaboration

|  |  |  |  |
| --- | --- | --- | --- |
| **Label (Fields)** | **Type (Fields)** | **Options (Fields)** | **Mandatory** |
| Pay grade | Data |  | Y |

### 3d.7 Screen Authorization

|  |  |
| --- | --- |
| **User** | **Type of access** |
| Administrator | Read/ Write/ Modify |

# ii) Transactional Screens

## Application for Residence

### 1.1 Overview

The Application for residence shall hold the application details and the request made by the applicant(employee) for residence allotment.

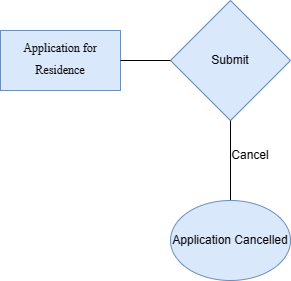
### 1.2 Description of Problem

Create a submittable page in which the applicant can submit their request for residence and give out application number with date of application and other necessary details like application status.

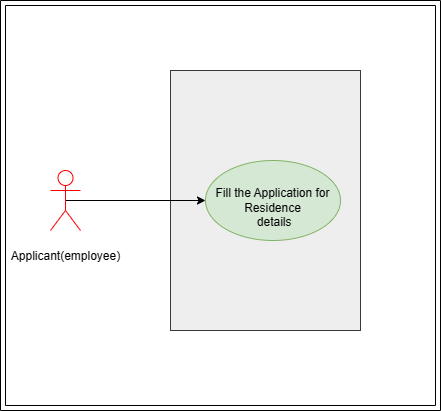
### 1.3 Dependency

Not dependent on anything other than being filled and saved by the applicant(employee).

### 1.4 Flow Chart

****

### 1.5 Use Case Diagram

****

**Brief Description -**

Applicant: Able to fill the Application for residence form and then submit.

### 1.6 Form Elaboration

|  |  |  |  |
| --- | --- | --- | --- |
| **Label (Fields)** | **Type (Fields)** | **Options (Fields)** | **Mandatory** |
| Date of Application | Link |  |  |
| Application Number | Data |  |  |
| Employee Details | Data |  |  |
| Employee ID | Data | Employee | Y |
| Employee Name | Data |  | Y |
| Designation | Data |  | Y |
| Department | Data |  | Y |
| **Allotment Application Details** | | | |
| Type of residence requested | Link | Type of Residence |  |
| Type of residence name requested | Link | Residence Type |  |
| Application Attachments | Attach |  |  |
| Application Status | Data |  |  |
| Current Application Status | Data |  |  |

### **1.7 Methods**

**application\_for\_residence.js:**

To filter residence type name in the link field based on selected type of residence

**application\_for\_residence.py:**

def duplicate: To validate for any duplicate application record for residence application

def applicationNumberField: To get the doc series name in a field

def applicationStatus: To set value for Application status in Application for Residence

def currentApplicationStatus: To set value for Application status in Application for Residence

def cancelRejected: To set value for Application status as “Cancelled by Applicant” in Application for Residence screen on cancel.

### 1.8 Screen Authorization

|  |  |
| --- | --- |
| **User** | **Type of access** |
| Administrator | Read |
| Applicant(employee) | Read/ Write/ Cancel |

## 2. Residence Allotment

### 2.1 Overview

The Residence Allotment document shall assist the administrator in allotment of residence on the basis of the applications sent by the applicant(employees).

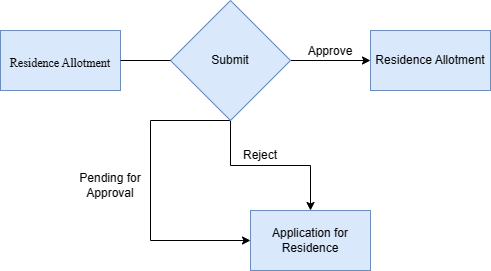
### 2.2 Description of Problem

Create a page **to allot the vacant and not allotted residences to the applicants and also update the application status with respect to the approval state.**

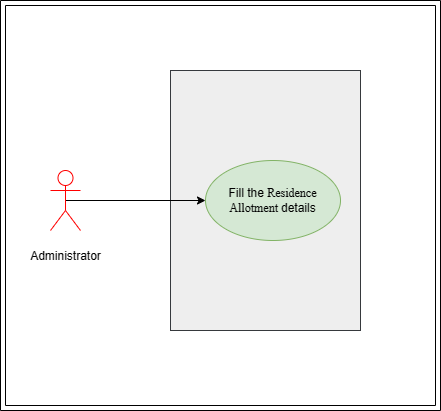
### 2.3 Dependency

Dependency on the “Application for Residence” screen, once the mentioned screen is filled and submitted, the administrator can further take actions on the Residence Allotment screen.

### 2.4 Flow Chart

****

### 2.5 Use Case Diagram

****

**Brief Description -**

Administrator: Able to fill the Residence Allotment form and then submit.

### 2.6 Form Elaboration

|  |  |  |  |
| --- | --- | --- | --- |
| **Label (Fields)** | **Type (Fields)** | **Options (Fields)** | **Mandatory** |
| Residence Allotment Number | Link |  |  |
| Application Number | Data |  |  |
| Last Update Date | Data |  |  |
| Employee Info | Data | Employee | Y |
| Employee Name | Data |  | Y |
| Employee ID | Data |  | Y |
| Employee Designation | Data |  | Y |
| Employee Department | Data |  | Y |
| **Residence Request** | | | |
| Type of residence requested | Data |  |  |
| Type of residence name requested | Data |  |  |
| Allotment Details | Section Break |  |  |
| Approval status | Drop down | Approved  Pending for Approval  Rejected | Y |
| Start Date | Date |  |  |
| End Date | Date |  |  |
| Building | Link | Buildings |  |
| Residence Type | Link | Type of Residence |  |
| Residence Type Name | Link | Residence Type |  |
| Residence Serial Number | Link | Building Room |  |
| Residence Number | Data |  |  |
| Employee Allotment status | Data |  |  |
| Vacancy Status | Data |  |  |
| Current Employee Allotment status | Data |  |  |
| Current Vacancy Status | Data |  |  |
| **Residence Details** | | | |
| Floor | Data |  |  |
| Building Address | Data |  |  |
| Building's Land Address | Data |  |  |
| Description | Data |  |  |
| Unit Area (sq. m.) | Data |  |  |
| Residence Orientation | Data |  |  |
| Lawn/Garden Available | Data |  |  |
| Lawn/Garden Area (sq. m.) | Data |  |  |
| Roof type | Data |  |  |
| Parking Available | Data |  |  |
| Parking Type | Data |  |  |
| Parking Vehicle | Data |  |  |
| Parking Area (sq. m.) | Data |  |  |

### **2.7 Methods**

**residence\_allotment.js:**

1. To filter applications in the link field based on the application status
2. To filter residence type name in the link field based on selected residence type
3. To filter buildings in the link field based on building type selected
4. To filter residence serial number in the link field based on selected building and residence type name and also which are allottable & vacant
5. To validate end date is not before the start date

**residence\_allotment.py:**

1. def dateValidate: To validate if the start date is not after the end date
2. def duplicateResidenceAllot: To validate every employee is alloted only one quarter
3. def allotmentNumberField: To get the doc series name in a field
4. def residenceAllotmentStatus: To change employee allotment status in "Residence Allotment"
5. def buildingRoomStatus: To change employee allotment status and vacancy status in "Building Room"
6. def currentResidenceApplicationStatus: To set value of current Application status based on the approval selection
7. def currentResidenceAllotmentStatus: To set value of current employee allotment status and current vacancy status in "Residence allotment"
8. def residenceUpdate: To set value of allotment details in "Residence Allotted" child table in "Employee" screen
9. def allottmentstatusCancel: To update the value of Allotment status and vacancy status field in Residence Allotment screen
10. def allottmentCancelled: To update the value of current application status in Application for Residence screen on cancellation of allotment
11. def allottmentCancelledRoom: To update the value of Allotment status and vacancy status fields in the Building Room screen on Cancel

### 2.8 Screen Authorization

|  |  |
| --- | --- |
| **User** | **Type of access** |
| Administrator | Read/Write/Modify |

## 3. Employee: Residence Alloted(Child Table)

### 3.1 Overview

Residence Eligibility is a child table which shall store the essential residence allotment details in employee screen.

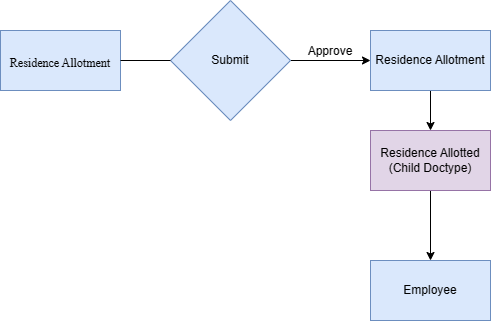
### 3.2 Description of Problem

Create a page which shall store the residence allotment details in a table under the Accommodation tab in Employee screen.

### 3.3 Dependency

Dependency on the “Residence Allotment” screen for the data it will carry.

### 3.4 Flow Chart

****

### 3.5 Form Elaboration

|  |  |  |  |
| --- | --- | --- | --- |
| **Label (Fields)** | **Type (Fields)** | **Options (Fields)** | **Mandatory** |
| Residence Allotment Number | Data |  |  |
| Application Number | Data |  |  |
| Residence Type | Data |  |  |
| Residence Type Name | Data |  |  |
| Residence Number | Data |  |  |
| Floor | Data |  |  |
| Building Address | Data |  |  |
| Unit Area (sq. m.) | Data |  |  |
| Parking Available | Data |  |  |
| Parking Type | Data |  |  |
| Parking Area (sq. m.) | Data |  |  |
| Parking Vehicle | Data |  |  |
| Current Employee Allotment status | Data |  |  |

### 3.6 Screen Authorization

|  |  |
| --- | --- |
| **User** | **Type of access** |
| Administrator | Read |
| Applicant(employee) | Read |

## 4. Application for Residence De-Allotment

### 4.1 Overview

This document will hold the request for Residence De-Allotment like The de-allotment number, date of application, alloted residence, etc.

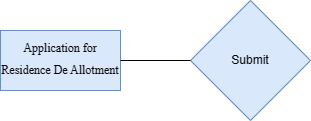
### 4.2 Description of Problem

Create a page to assist the applicant in requesting De-Allotment of their residence.

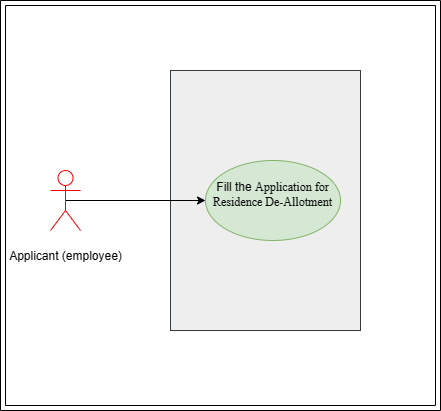
### 4.3 Dependency

Not dependent on anything other than being filled and saved by the applicant(employee).

### 4.4 Flow Chart

****

### 4.5 Use Case Diagram

****

**Brief Description -**

Applicant: Able to fill the Application for residence de-allotment form and then submit.

### 4.6 Form Elaboration

|  |  |  |  |
| --- | --- | --- | --- |
| **Label (Fields)** | **Type (Fields)** | **Options (Fields)** | **Mandatory** |
| Residence De-Allotment Number | Data |  |  |
| Residence Allotment Number | Link | Residence Allotment | Y |
| Application Number | Data |  |  |
| Last Update Date | Date | Employee |  |
| **Allotment Details** | | | |
| Residence Serial Number | Data |  |  |
| Residence Number | Data |  |  |
| Employee Name | Data |  |  |
| Employee ID | Data |  |  |
| Building Name | Data |  |  |
| Residence Type | Data |  |  |
| Residence Type Name | Data |  |  |
| **De-Allotment Application Details** | | | |
| Reason for De-Allotment | Data |  | Y |
| Application Status | Data |  |  |
| Current Application Status | Data |  |  |

### **4.7 Methods**

**application\_for\_residence\_de\_allottment.js:**

To filter applications in the link field based on employee allotment status

**application\_for\_residence\_de\_allottment.py:**

* + 1. def duplicate: To check for any duplicate record in "applied" applications for de-allotment
    2. def fieldName: To get the doc series name in a field
    3. def applicationStatus: To set value for Application status to applied after application
    4. def currentApplicationStatus: # To set value for Current Application status to applied after application

### 4.8 Screen Authorization

|  |  |
| --- | --- |
| **User** | **Type of access** |
| Applicant(employee) | Read/ Write |

## 5. Residence De-Allotment

### 5.1 Overview

The Residence De-Allotment document will hold all the De-allotment details which will be updated on the Employee screen too via the “Residence De Allot” child table.

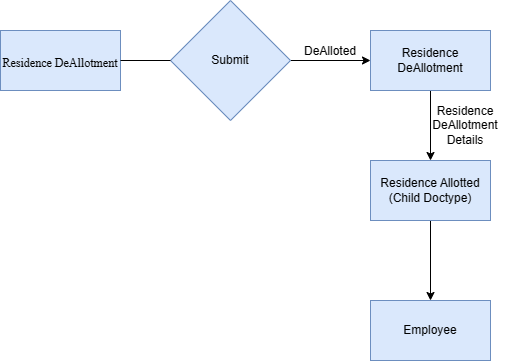
### 5.2 Description of Problem

Create a page containing all the data of the de-alloted residence like- Residence De-Allotment Number, De-Allotment Date, Reason for De-Allotment, etc.

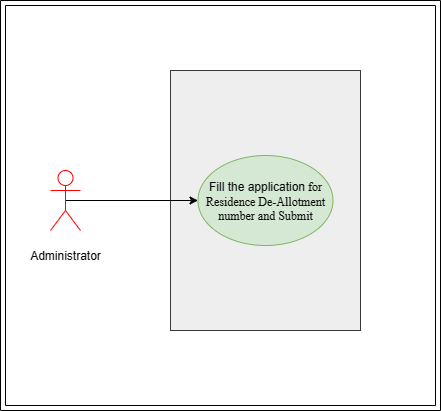
### 5.3 Dependency

Dependency on the “Application for Residence De allotment” screen, once the mentioned screen is filled and submitted, the administrator can further take actions on the Residence De Allotment screen.

### 5.4 Flow Chart

****

### 5.5 Use Case Diagram

****

**Brief Description -**

Administrator: Able to fill the Residence De-Allotment form and then submit.

### 5.6 Form Elaboration

|  |  |  |  |
| --- | --- | --- | --- |
| **Label (Fields)** | **Type (Fields)** | **Options (Fields)** | **Mandatory** |
| Residence De-Allotment Number | Link | Application for Residence De-Allotment | Y |
| Residence Allotment Number | Data |  |  |
| Application Number | Data |  |  |
| De-Allotment Date | Date |  |  |
| **De-Allotment Details** | | | |
| Residence Serial Number | Data |  |  |
| Residence Number | Data |  |  |
| Employee Name | Data |  |  |
| Employee ID | Data |  |  |
| Building Name | Data |  |  |
| Residence Type | Data |  |  |
| Residence Type Name | Data |  |  |
| Reason for De-Allotment | Data |  |  |

### **5.7 Methods**

**residence\_de\_allottment.js:**

1. To filter applications in the link field based on the application status
2. To filter Alloted residence serial number field based on the vacancy and Allotmnet status

**residence\_de\_allottment.py:**

1. def deallotmentNumberField: To get the doc series name in a field
2. def residenceAllotmentStatus: To set value of current employee allotment status and current vacancy status in "Residence allotment"
3. def currentApplicationStatus: To set value of current application status in "Application for Residence De-Allottment"
4. def buildingRoomStatus: To change employee allotment status and vacancy status in "Building Room"
5. def residenceApplicationStatus: To set value of current application status to De-Alloted in "Application for Residence"
6. def residenceUpdate: To set value of de-allotment details in "Residence Allotted" child table in "Employee" doctype

### 5.8 Screen Authorization

|  |  |
| --- | --- |
| **User** | **Type of access** |
| Administrator | Read/Write/Modify |

## Residence Change request (Workflow)

### 6.1 Overview

This is a workflow which can be accessed by both the applicant as well as the administrator. The applicant can request change of residence while the administrator can approve/reject/pending.

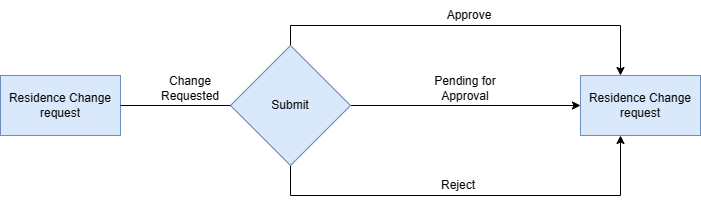
### 6.2 Description of Problem

Create a workflow containing the applicant’s residence change request and the administrators action on the request.

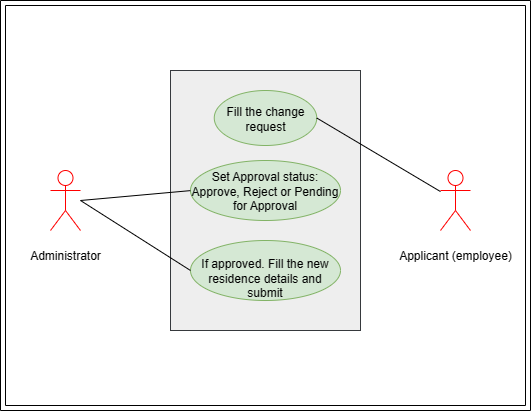
### 6.3 Dependency

No dependency

### 6.4 Flow Chart

**Residence DeAllotment flow**

### 6.5 Use Case Diagram

****

**Brief Description -**

Administrator: Able to approve/reject/pending the residence change request and change the request as per availability of residence.

Applicant: Able to fill the change request form and submit.

### 6.6 Form Elaboration

|  |  |  |  |
| --- | --- | --- | --- |
| **Label (Fields)** | **Type (Fields)** | **Options (Fields)** | **Mandatory** |
| Residence Change Request Number | Data |  |  |
| Residence Allotment Number |  |  | Y |
| Application Number | Data |  |  |
| Change Request Date | Data |  |  |
| **Employee Details** | | | |
| Employee Name | Data |  |  |
| Employee ID | Data |  |  |
| Employee Designation | Data |  |  |
| Employee Department | Data |  |  |
| **Alloted Residence Details** | | | |
| Alloted Residence Serial Number | Data |  |  |
| Alloted Residence Number | Data |  |  |
| Alloted Residence Building | Data |  |  |
| Alloted Residence Type | Data |  |  |
| Alloted Residence Type name | Data |  |  |
| **Residence Change Request** | | | |
| Residence Type requested | Data |  | Y |
| Residence Type name Requested | Data |  | Y |
| Residence Change Request Reason | Data |  | Y |
| **Changed Residence Details** | | | |
| Residence Building | Link | Buildings |  |
| Residence Type | Link | Type of Residence |  |
| Residence Type Name | Link | Residence Type |  |
| Residence Serial Number | Link | Building Room |  |
| Residence Number | Data |  |  |
| Request Status | Data |  |  |
| **Residence Full Details** | | | |
| Floor | Data |  |  |
| Building Address | Data |  |  |
| Building's Land Address | Data |  |  |
| Description | Data |  |  |
| Unit Area (sq. m.) | Data |  |  |
| Residence Orientation | Data |  |  |
| Lawn/Garden Available | Data |  |  |
| Lawn/Garden Area (sq. m.) | Data |  |  |
| Roof type | Data |  |  |
| Parking Available | Data |  |  |
| Parking Type | Data |  |  |
| Parking Vehicle | Data |  |  |
| Parking Area (sq. m.) | Data |  |  |

### **6.7 Methods**

**residence\_change\_request.js:**

1. To filter employees who have Current Employee Allotment status as "Alloted"
2. To filter residence serial number in the link field based on selected Residence Type name
3. To filter residence type name in the link field based on selected "type of residence requested"
4. To filter residence type name in the link field based on selected "type of residence"

**residence\_change\_request.py:**

1. def changeRoomStatus: To change vacancy status and employee allotment status of alloted residence in “Residence Allotment” screen
2. def buildingStatusChange: To change allotment and vacancy status of alloted residence in building room screen
3. def changeNewRoomStatus: To change vacancy status and employee allotment status of newly alloted residence after approval of change of residence request
4. def residenceUpdate: To update residence details in “Residence Alloted” child table in Employee screen
5. def changeRequestNumberField: To set the value of doc series in “Residence change request number” field

### 6.8 Screen Authorization

|  |  |
| --- | --- |
| **User** | **Type of access** |
| Administrator | Read/Write/Modify |
| Applicant(Employee) | Read/Write |