**SOFTWARE REQUIREMENTS SPECIFICATION**

**For**

**HOSTEL AUTOMATION SYSTEM**

***Prepared by***

## Subin K Thomas 700040 subinkthomas@mca.ajce.in

**Project Guide: Navyamol K T**

**Date:**

**DEPARTMENT OF COMPUTER APPLICATIONS, AJCE**

**CHAPTER - 1**

# INTRODUCTION

## 1.1 PURPOSE

The purpose of this document is to provide a detailed description of the ‘Hostel Automation System’. This SRS will give a complete understanding of the ‘Hostel Automation System’ to be constructed. Online Hostel Automation is a system for managing the various activities in the hostel. It is used for managing the hostel information. It manages the student and faculty information, room information, room allocation details, fee details and mess bill details, out pass management.

It is also used to generate reports of student details, fee details and mess bill details of the student. It keeps track of the number of student in the room and availability of the room. It helps organization from the manual work from which it is very difficult to find the record of the students and the mess bills of the students.

## 1.2 SCOPE

The Hostel Automation System is a web based application. The Hostel Automation is used to keep track of hostel information in the organization. It is used to improve clerical services for all the students of the hostel. The system will help to the users to check the mess bills of every student and the student’s hostel dues. The main feature of this project is easy to allocate hostel rooms for the student and also easy generation of mess bill with all accuracy.

Major functional modules:

* Registration and login
* Room Management
* Room Allocation
* Outpass management
* Student Rating
* Fee Management
* Complaints and Request

## 1.3 DEFINITIONS, ACRONYMS, AND ABBREVIATIONS

PHP-Personal Home Page

JS-Java Script

HTML-Hypertext Markup Language

SQL-Structured Query Language

## 1.4 REFERENCES

* [www.w3schools.com](http://www.w3schools.com/)
* www.stackoverflow.com

## 1.5 OVERVIEW

### EXISTING SYSTEM

The existing system Manually Managing the various aspects of hostel such as admission, outpass, room allocation, room shifting, complaints and request and vacation.

### DRAWBACKS OF EXISTING SYSTEM

• More human power

• More strength and strain of manual labour needed

• Low security.

• Data redundancy.

• Difficulty to handle.

• Difficulty to update data.

• Record keeping is difficult.

### PROPOSED SYSTEM

The proposed system is having many advantages over the existing system. The proposed system allows registration and login. The proposed system offers facilities for manage a Hostel remotely and automates many works used in managing the Hostel and we can easily give personalized important notification to each member. The all data and information are stored in digital format so accessing the information and finding the relevant information is easy. It requires less overhead and very efficient. The proposed system deals with the fee calculation include mess, room rent and laundry and allotment process efficiently. The primary users of the system are the Student, Faculty, Warden and Administrator who are provided with different interfaces based on their role.

The main features include:

* Registration and login
* Hostel Settings

• Application

• Room Allocation

• Room Shifting

• Outpass Details

• Room Vacate

• Complaints and Feedbacks

• View Notice and Details

• Fee Management

### ADVANTAGES OF PROPOSED SYSTEM

The system is very simple in design and to implement. The system requires very low system resources and it will work in almost all configurations. It has got following features:

* Less human error
* Strength and strain of manual labour can be reduced
* High security
* Data redundancy can be avoided to some extent
* Data consistency
* Easy to handle
* Easy data updating
* Easy record keeping
* Backup data can be easily generated

**CHAPTER - 2**

# OVERALL DESCRIPTION

**2.1 PRODUCT PERSPECTIVE:**

The Hostel Automation System is an independent product that automates all the tasks that are carried out in a Hostel. It maintains all the details related to the various Students and Faculty involved in the Hostel. The all data and information are stored in digital format so accessing the information and finding the relevant information is easy. The Hostel Automation System keeps all the details of the Room allocation and shifting, Outpass fee and laundry management, student rating and complaints and feedback details. This system helps to keep a track of student details, rooms allocated to them, their admission and vacation dates, out pass approved or rejected. This system automates the room allocation process and shift rooms automatically, and calculates fees for each student in the hostel. The users of the system include Students, Faculties Warden and Administrator.

**2.2 PRODUCT FUNCTIONS/FEATURES:**

The objective is to build a software system that is user friendly and accurately generates and stores data that automates all the tasks in the Hostel. The software system should be convenient for its users. The aim of proposed system is to develop a system with improved facilities.

The major functional modules are:

##  Registration and login

The Hostel Automation System allows the registration and login process for Students and Staff. The Student and Staff can directly register their details and can login. The Wardens are registered by the Admin using their official details. The username and password are send to the Warden thorough E-mail or SMS (OTP).

** Room Management**

## It handles the allocation of rooms and vacating rooms by the students.

* Room Allocation

During admission procedures according to the room vacancy and the hostel preferred each applicant is admitted to the hostel by allocating a room.

* Room Vacate

Room Vacate can be done in two ways, either individually or batch wise. To vacate an individual, the warden enters the loginid of the student or faculty. It is also allows to vacate a whole batch after completing the course.

##  Outpass Management

In this modules the student, apply the outpass by filling the details in its online form. The details like where they are going, purpose, when they are going and when they will return are asked in the form. Once the request is made, the warder can see it. The warden approves or rejects the outpass.

##  Fee Management

This module done by the Administrator. This module deals with fee processing of the student felt like,

* Monthly Fee
* Yearly fee, by using E-pay system. Warden generates pdf of the corresponding

Fees and email it to the bank.

* Hostel Fee
* Mess Fee
* Laundry Fee
* Additional Fees (Paper, Net etc.)
* Fines

In this module Fines can be charged to inmates because of their misbehaviour.

* Concessions

Concessions can be granted to inmates because to any genuine reason of inmate.

##  Student Rating

Student rating done according to the number of warnings, number of suspensions and rating provided by the warden.

##  Complaints and Request

In this module the students can login into their account and make complaints or requests. The warden looks into these complaints or requests and takes necessary steps and provides feedback to the student.

##  Laundry Management

It management all the laundry services of student.

**2.3 USER CHARACTERISTICS:**

The Hostel Automation System has 4 users.

1. **Admin Module:** The admin is the overall controller of the system. The admin can add wardens, blocks, and approve or reject students to the hostel. Admin can manage fee, foods, Events and rooms.
2. **Warden Module:** The warden can view the details of all students and faculty in the hostel and add room. Also the warden can give the Outpass to the students and faculty.
3. **Student Module:** The Student can view or edit profile himself. Also request or view outpass, generate feedback and complaints, fee payment, request room vacate and room shift.

* **Faculty Module:** The Faculty can view or edit profile himself. Also request or view outpass, generate feedback and complaints, fee payment, request room vacate and room shift. Also view room allocation.

## 2.4 OPERATING ENVIRONMENT

### Hardware Specification

Processor : Pentium IV/AMD Dual core or above

RAM : 2 GB and above

Hard disk : 500 GB and above

### Software Specification

Front End : PHP (Code igniter)

Backend : MySQL

Client on PC : Windows 7 and above.

Technologies used : JS, HTML5, AJAX, J Query

### 2.5 CONSTRAINTS: \*/

It provides the genera1 description of the constraints such as regulatory policies, audit functions, reliability requirements, and so on.

### 2.6 USER DOCUMENTATION

The ‘Hostel Automation System’ will provide a user’s manual for the administrator and manager. To assist the user in understanding the product better and to assist them in better utilization of the product and its features, we will be providing an interactive user interface (in DOC format).

### 2.7 ASSUMPTION AND DEPENDENCY

We are assuming that the users will not use our product to download copyright. Also, we expect that the users not only use the software to view resources, they will also be updating details and resources for being viewed by other users. We also assume that product users will not do malicious activities on the software.

**CHAPTER - 3**

# SPECIFIC REQUIREMENTS

## 3.1 EXTERNAL INTERFACE REQUIREMENTS

The “**Hostel Automation System**” will use the following input/output devices for a personal computer. The external interfaces are:

* Keyboard
* Mouse
* Monitor
* Printer

### 3.1.1 User Interfaces

|  |  |
| --- | --- |
| **FORMS** | **DESCRIPTION** |
| Home | Log into the site as an Administrator, Warden, Student and Faculty. |
| User Registration | Administrator, Student and Faculty can register to get access. |
| Student Management | Save Student information, view and update them. |
| Room management | Student and faculty room allocation and room shift. |
| fee Payment | Pay mess, Rent and Laundry fee by the student and faculty account. |
| Outpass Management | Student and Faculty can apply for Outpass and admin accepts or reject those outpass request. |
| Student Rating | Done by Warden according to the no.of warnings and suspensions. |
| Complaints and Request | Student can make complaints and request. |
| Laundry Management | Manage all the laundry services of student. |

### 3.1.2 Hardware Interfaces/\*

**Hostel Automation System** is a web based application.

### 3.1.3 Software Interfaces

Database Server: MySQL

Development End: PHP

Client on PC: Any Browser

# 3.2 FUNCTIONAL REQUIREMENTS

Functional requirements define the fundamental actions that system must perform. The functional requirements for the system are divided into some main categories Student management, Room Allocation and shift, Outpass Management, fee Management, Student rating, Stock Management, Laundry Management, and Complaints and feedback.

## 3.2.1 Registration

3.2.1.1 Introduction

The Student and Faculty can register into the system. There is warden registration, student registration and faculty registration. The Warden are registered by the admin. The student and faculty can directly register their details.

3.2.1.2 Inputs

In this module, for student registration, the student information like student id, Name, address, course, branch, batch, DOB, Gender, Religion , Gender ,Email, contact details, joining date, image, father name ,Guardian details etc are entered into the database. The Faculty registration include details like faculty id, faculty name, Department, Mobile.no, image etc.

3.2.1.3 Processing

After student and faculty registration, they can login to the site and can update personal information and password.

3.2.1.4 Outputs

When registration is completed, a success message is displayed in case of student and Faculty.

3.2.1.5 Error Handling

During the registration process, all the inputs are validated and if there is any error, it is displayed. If a username is already in use, it also displayed.

## 3.2.2 Room Management

3.2.2.1 Introduction

This module is for managing Room allocation, Room shift and vacating rooms by the student and faculty.

3.2.2.2 Inputs

The Room number, Room type, and Room status (present/absent), Allotment date, Vacate date are entered into the database.

3.2.2.3 Processing

The warden manages Room allotment, Room vacating and Room shift.

3.2.2.4 Outputs

The Room details can be viewed.

3.2.2.5 Error Handling

If an attendance is entered incorrectly, it can be changed.

## 3.2.3 Outpass Management

3.2.3.1 Introduction

This module is for handling the Outpass applications. The Student and faculty can apply

for Outpass.

3.2.3.2 Inputs

Outpass id, date, time purpose, and place are entered into the database.

3.2.3.3 Processing

The warden can approve or reject the outpass applications.

3.2.3.4 Outputs

The outpass status can be checked by the student and faculty.

3.3.3.5 Error Handling

If outpass for a date is already applied, and if student try to apply it again, it is not entered into the database.

## 3.2.4 Fee Payment

3.2.4.1 Introduction

This module is for fee processing of student and faculty.it concerned with hostel, mess, laundry fees.

3.2.4.2 Inputs

The Fee id, fee name and fee cost are entered into the database.

3.2.4.3 Processing

The fee is paid through online. The amount is credited to student account.

3.2.4.4 Outputs

The fee details can be viewed by Faculty and student.

3.2.4.5 Error Handling

If student enters wrong bank details, cvv.no then appropriate error message given to the student.

## 3.2.5 Student Rating

3.2.5.1 Introduction

This module is for warden to rate each student in the hostel

3.2.5.2 Inputs

Student name, room number, student rating are entered into the database.

3.2.5.3 Processing

Warden can reduce or increase student rating according to behavior of the student.

3.2.5.4 Outputs

A success message is displayed and directed to home page.

## 3.2.6 Laundry Management

3.2.6.1 Introduction

This module is used to manage laundries and complaints associated with the laundry.

3.2.2.2 Inputs

The Item details, room no, student id are entered into the database.

3.2.2.3 Processing

The Student can check status laundry and register complaints.

3.2.2.4 Outputs

The Status of laundry is displayed.

## 3.2.6 Complaints and Feedback Management

3.2.6.1 Introduction

This module allows to students can login into their account and make complaints or requests.

3.2.2.2 Inputs

Complaint id, feedback id, date, description, Roon.no are entered into the database.

3.2.2.3 Processing

The warden can approve or reject the outpass applications.

3.2.2.4 Outputs

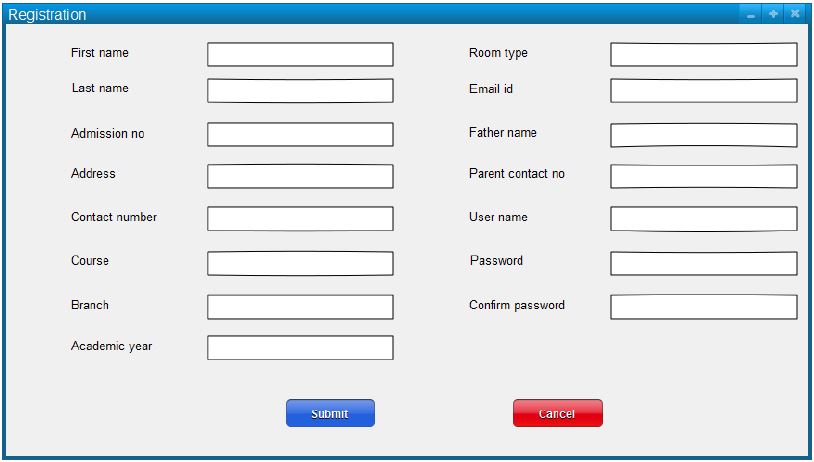
The outpass status can be checked by the student and faculty.

**3.3 INPUT FORMS**

**Login Form**



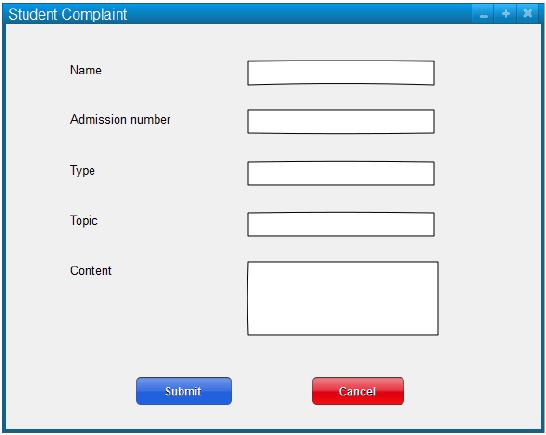
**Student Registration**



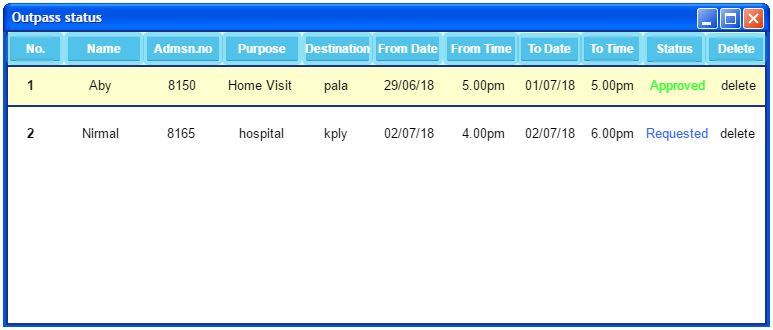
**Outpass Request**



**Student Complaint**



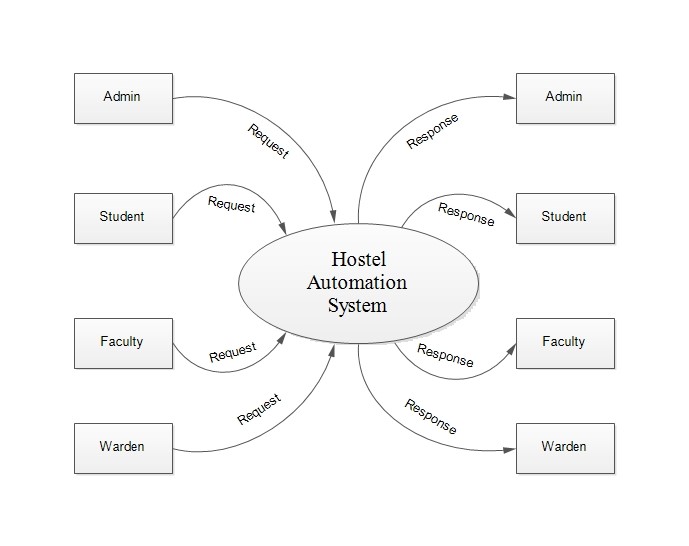
## 3.4 OUTPUT FORMS

**Outpass Status**

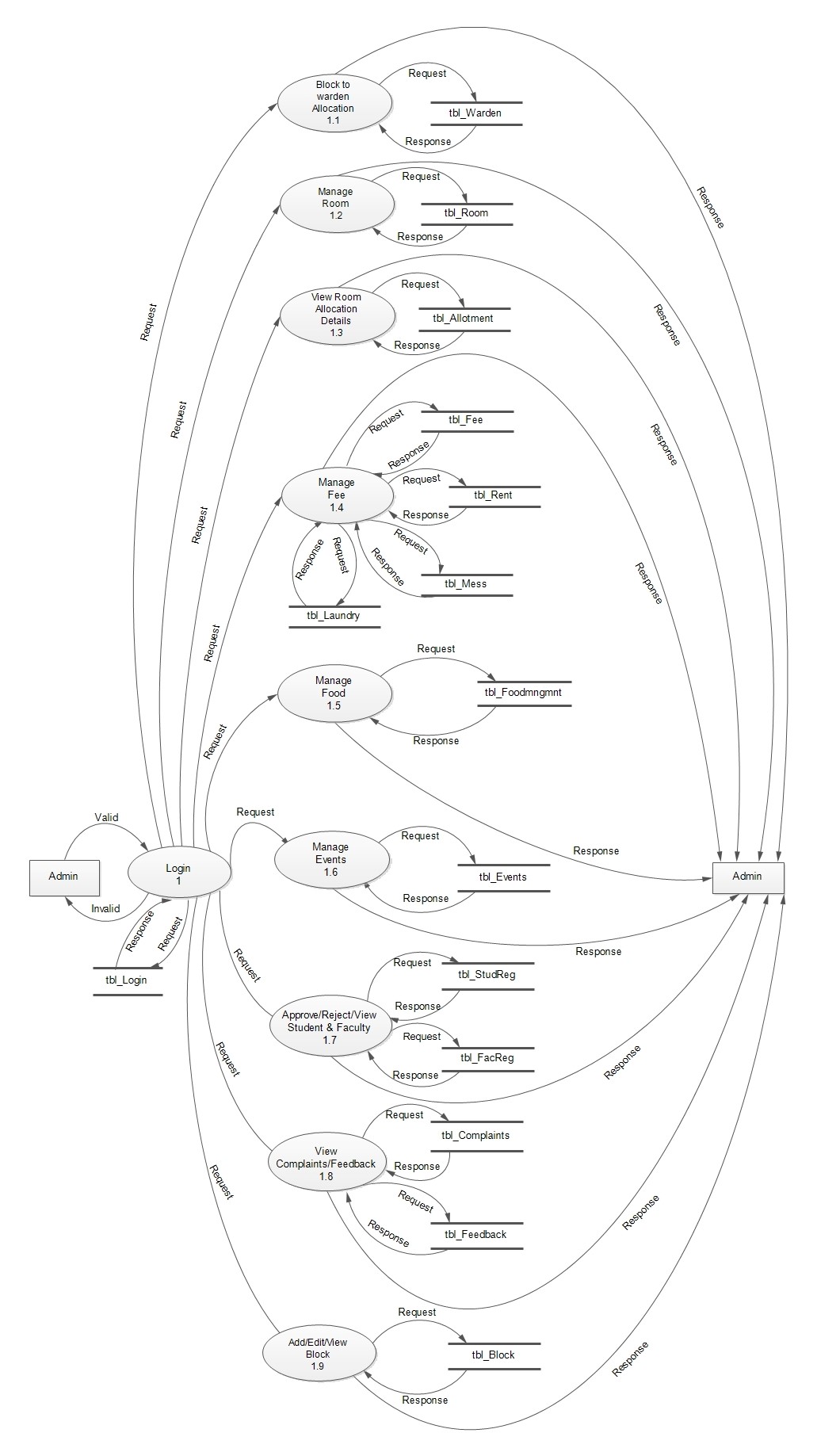
**3.5 Other Documents**

**3.5.1 DATA FLOW DIAGRAM**

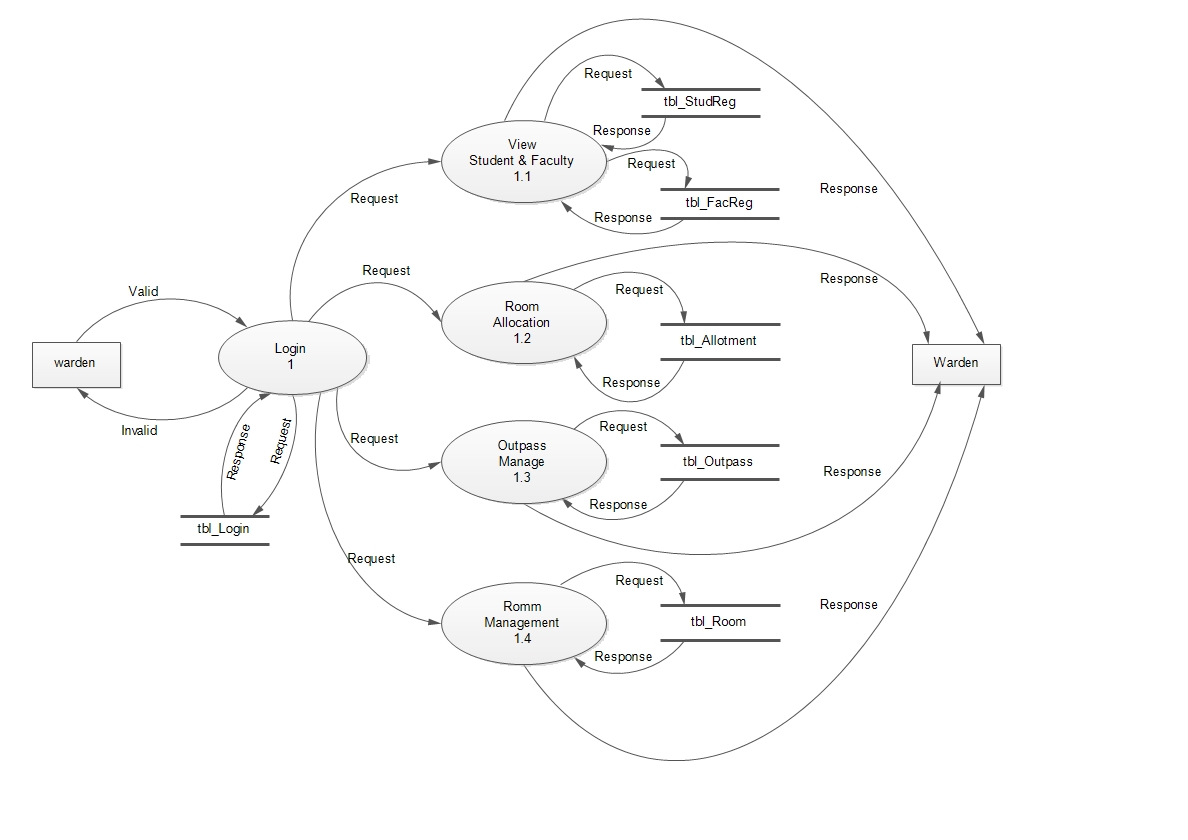
**ZEROTH LEVEL DFD FOR HOSTEL AUTOMATION SYSTEM**



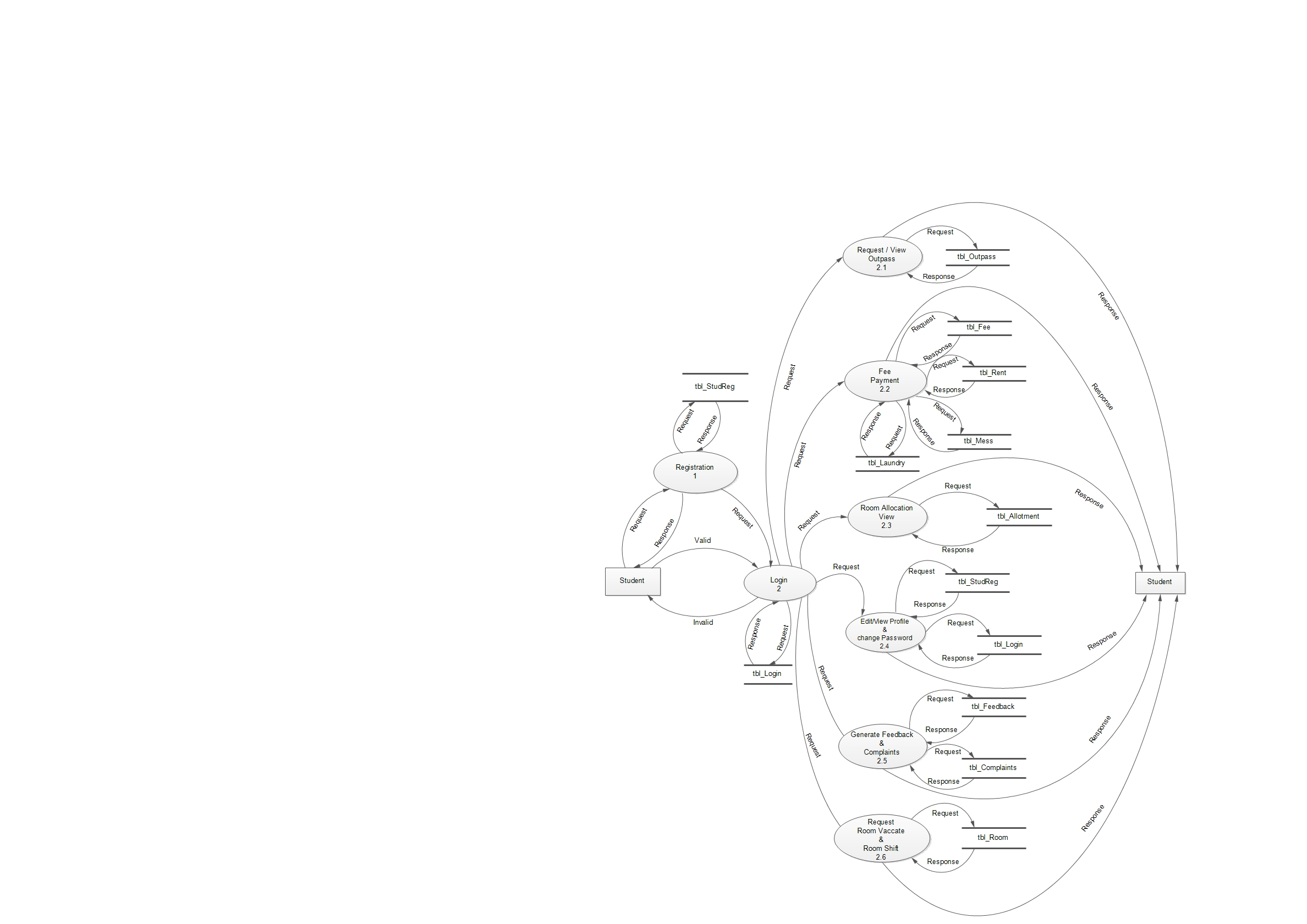
**FIRST LEVEL DFD FOR HOSTEL AUTOMATION SYSTEM – ADMIN**

****

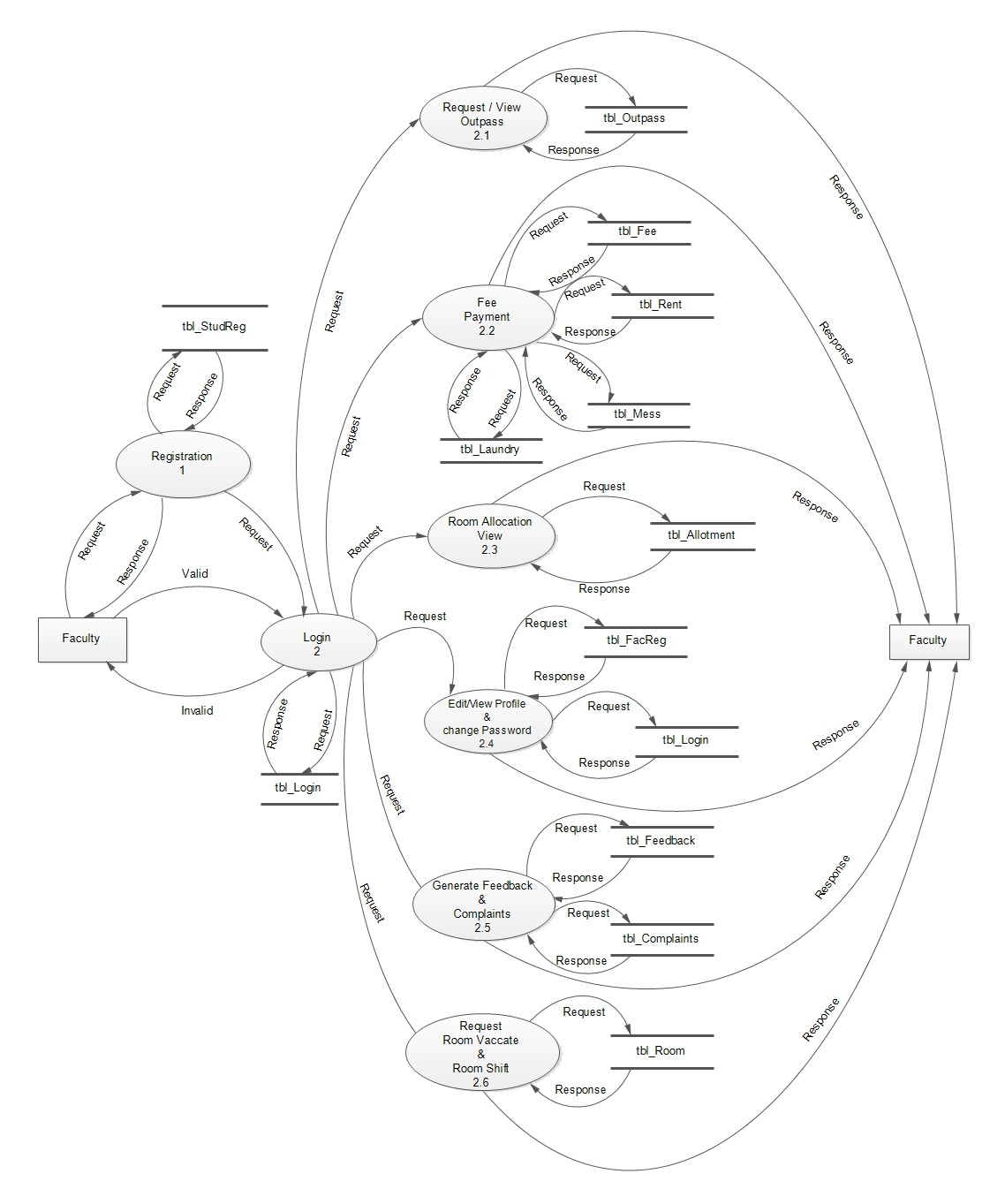
**FIRST LEVEL DFD FOR HOSTEL AUTOMATION SYSTEM – WARDEN**

****

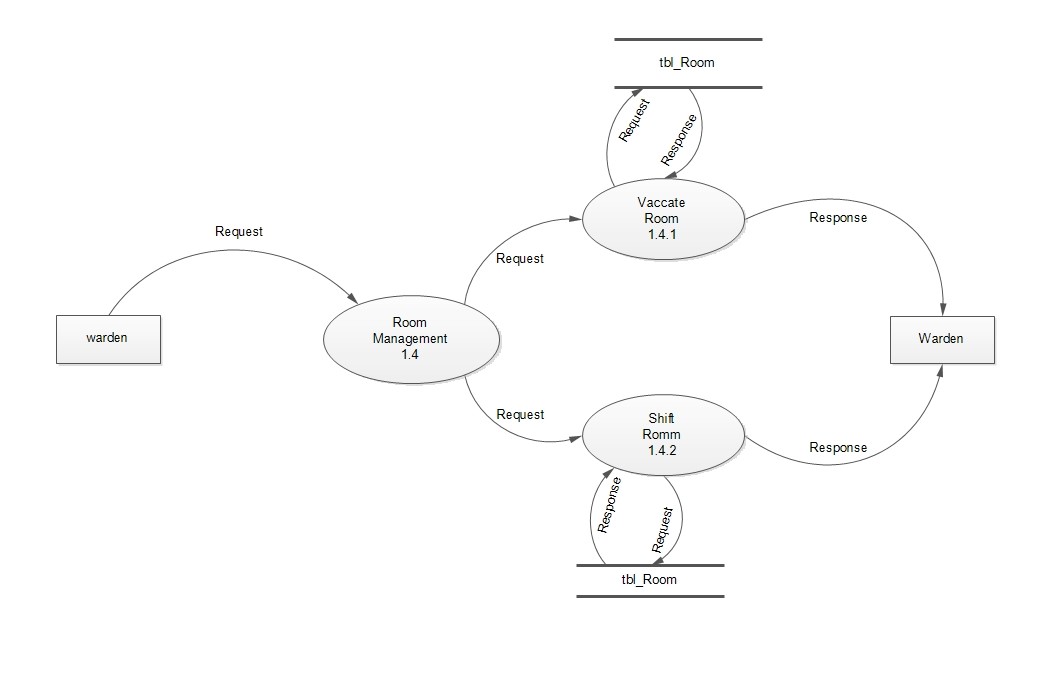
**FIRST LEVEL DFD FOR HOSTEL AUTOMATION SYSTEM – STUDENT**

****

**FIRST LEVEL DFD FOR HOSTEL AUTOMTION SYSTEM – FACULTY**

****

**SECOND LEVEL DFD FOR HOSTEL AUTOMATION SYSTEM – WARDEN**

****

**1.5.2 USE CASE DIAGRAM**

**3.5.3 SEQUENCE DIAGRAM**

**3.5.4 TABLE DESIGN**

**Table 1**: tbl\_StudReg

Primary key : StudRegId

Foreign key : LoginId,StateId,AddId,CourseId,BranchId,DistrictId

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| StudRegId | Integer | 2 | Student Registration Id |
| LoginId | Integer | 2 | Login Id references from tbl\_Login |
| AddId | Integer | 2 | References from tbl\_Address |
| StateId | Integer | 2 | References from tbl\_State |
| DistrictId | Integer | 2 | References from tbl\_District |
| CourseId | Integer | 2 | References from tbl\_Course |
| BranchId | Integer | 2 | References from tbl\_Branch |
| RoomTypeId | Integer | 2 | References from tbl\_RoomType |
| FName | Varchar | 50 | First name of the Student |
| LName | Varchar | 50 | Last name of the Student |
| From | Varchar | 10 | Start of course |
| To | Varchar | 10 | End of course |
| Batch | Varchar | 20 | Academic Duration |
| ContactNo | Integer | 10 | Mobile number of the Student |
| Image | Varchar | 20 | Photo |
| FatherName | Varchar | 50 | Father Name |
| Occup | Varchar | 20 | Father Occupation |
| MobNo | Integer | 10 | Father Mobile Number |
| GName | Varchar | 50 | Guardian Name |
| Gmob | Integer | 10 | Guardian Mobile Number |
| Religion | Varchar | 20 | Religion |
| Dob | date |  | Date of Birth |
| Gender | boolean | 1 | Gender |
| Email | Varchar | 30 | Email Id |
| Cast | Varchar | 20 | Cast |
| StudRegStatus | Integer | 2 | Valid/Invalid |
| WardenRating | Int | 4 | Warden Rating |

**Table 2**: tbl\_FacReg

Primary key : FacRegId

Foreign key : LoginId

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| FacRegId | Int | 2 | Faculty Registration Id |
| LoginId | Int | 2 | Reference from tbl\_Login |
| AddId | Int | 2 | Reference from tbl\_Address |
| desig | Varchar | 20 | Designation |
| FName | Varchar | 50 | First name of the faculty |
| LName | Varchar | 50 | Last name of the faculty |
| Dept | Varchar | 20 | Depertment |
| Mobno | Int | 10 | Mobile number |
| Image | Varchar | 20 | Photo |
| FacRegStatus | Int | 2 | Valid/Invalid |

**Table 3**: tbl\_Login

Primary key : LoginId

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| LoginId | Int | 2 | Login Id |
| username | Varchar | 50 | Username |
| Password | Varchar | 50 | Password |
| userType | Int | 2 | User Type |
| LogStatus | Int | 1 | Exit/Not |

**Table 4**: tbl\_Room

Primary key : Roomno

Foreign key : BlockId,RoomTypeId

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| Roomno | Int | 2 | Room Number |
| BlockId | Int | 2 | Reference from tbl\_Block |
| RoomTypeId | Int | 2 | Reference from tbl\_RoomType |
| capacity | Int | 2 | Capacity of Room |
| Nooffans | Int | 2 | Number of Fans in Room |
| RoomStatus | Int | 2 | 0-> No bed space available  1->One bed space available  2->Two bed space available |

**Table 5**: tbl\_RoomType

Primary key : RoomTypeId

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| RoomTypeId | Int | 2 | Room Type Id |
| RoomType | Varchar | 20 | Type of Room |

**Table 6**: tbl\_Allotment

Primary key : AllotmentId

Foreign key : LoginId,Roomno

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| AllotmentId | Int | 2 | Allotment Id |
| LoginId | Int | 2 | Reference from tbl\_Login |
| Roomno | Int | 2 | Reference from tbl\_Room |
| AllotmentDate | Date |  | Room Allotment date |
| Vacate | Date |  | Room vacate date |

**Table 7**: tbl\_Warden

Primary key : WardenId

Foreign key : LoginId

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| WardenId | Int | 2 | Warden Id |
| LoginId | Int | 2 | Reference from tbl\_Login |
| Name | Varchar | 50 | Warden Name |
| Phno | Int | 10 | Phone number of warden |
| Email | Varchar | 30 | Email Id of the warden |
| Housename | Varchar | 30 | House name |
| Place | Varchar | 30 | Place of warden |
| Pinno | Int | 6 | Pin number |
| WardenStatus | Int | 1 | Status |

**Table 8**: tbl\_Outpass

Primary key : OutpassId

Foreign key : LoginId

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| OutpassId | Int | 2 | Outpass Id |
| LoginId | Int | 2 | Reference from tbl\_Login |
| FromDate | Date |  | Issue Date |
| ToDate | Date |  | End Date |
| FromTime | Time |  | Issue Time |
| ToTime | Time |  | End Time |
| Place | Varchar | 20 | Place |
| Purpose | Varchar | 30 | Purpose for Outpass |
| OutpassStatus | Int | 1 | Status |

**Table 9**: tbl\_Block

Primary key : BlockId

Foreign key : WardenId

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| BlockId | Int | 2 | Block Id |
| WardenId | Int | 2 | Reference from tbl\_Warden |
| BlockName | Varchar | 20 | Name of the block |
| StartRoom | Int | 2 | Starting Room |
| EndRoom | Int | 2 | Ending Room |

**Table 10**: tbl\_FoodMngmnt

Primary key : FoodId

Foreign Key : ItemId

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| FoodId | Int | 2 | Food Id |
| ItemId | Int | 2 | Reference from tbl\_Items |
| Time | Time |  | Food Time |
| Date | Date |  | Date |
| Day | Varchar | 10 | Day |
| Status | Int | 1 | Status |

**Table 11**: tbl\_Items

Primary key : ItemId

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| ItemId | Int | 2 | Item Id |
| ItemName | Varchar | 20 | Name of the food item |

**Table 12**: tbl\_Fee

Primary key : FeeId

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| FeeId | Int | 2 | Fee Id |
| FeeName | Varchar | 20 | Fee Name |
| FeeCost | Int | 10 | Fee Cost |

**Table 13**: tbl\_Rent

Primary key : RentId

Foreign key : LoginId

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| RentId | Int | 2 | Rent Id |
| LoginId | Int | 2 | Reference from tbl\_Login |
| From | Date |  | From Date |
| To | Date |  | To Date |
| Cost | Int | 10 | Room Rent |
| Pdate | Date |  | Payment Date |

**Table 14**: tbl\_Mess

Primary Key : MessId

Foreign Key : LoginId

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| MessId | Int | 2 | Mess Id |
| LoginId | Int | 2 | Reference from tbl\_Login |
| From | Date |  | From Date |
| To | Date |  | To Date |
| Cost | Int | 10 | Mess Fee Cost |
| Pdate | Date |  | Payment Date |

**Table 15**: tbl\_Laundry

Primary Key : LaundryId

Foreign Key : LoginId

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| LaundryId | Int | 2 | Laundry Id |
| LoginId | Int | 2 | Reference from tbl\_Login |
| From | Date |  | From Date |
| To | Date |  | To Date |
| Cost | Int | 10 | Laundry Cost |
| Pdate | Date |  | Payment Date |

**Table 16**: tbl\_Complaints

Primary Key : ComplaintId

Foreign Key : Roomno , LoginId

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| ComplaintId | Int | 2 | Complaint Id |
| LoginId | Int | 2 | Reference from tbl\_Login |
| Roomno | Int | 2 | Reference from tbl\_Room |
| date | Date |  | Complaint Date |
| ComplaintDes | Varchar | 50 | Description of Complaint |
| Status | Int | 1 | Status |

**Table 17**: tbl\_Feedback

Primary Key : FeedbackId

Foreign Key : Roomno , LoginId

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| FeedbackId | Int | 2 | Feedback Id |
| LoginId | Int | 2 | Reference from tbl\_Login |
| Roomno | Int | 2 | Reference from tbl\_Room |
| Feedbackdes | Varchar | 50 | Feedback Description |
| Status | Int | 1 | Status |

**Table 18**: tbl\_Events

Primary Key : EventId

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| EventId | Int | 2 | Event Id |
| Eventtime | Time |  | Event Time |
| EventDes | Varchar | 30 | Event Description |
| FromDate | Date |  | Event Start Date |
| ToDate | Date |  | Event End Date |

**Table 19**: tbl\_EventItems

Primary key : ItemId

Foreign key : EventId

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| ItemId | Int | 2 | Event Item Id |
| EventId | Int | 2 | Reference from tbl\_Events |
| Itemname | Varchar | 30 | Event Item Name |
| ItemFirstprize | Varchar | 15 | First Prize |
| ItemSecondprize | Varchar | 15 | Second Prize |
| ItemThirdprize | Varchar | 15 | Third Prize |
| ItemTime | Time |  | Event Time |
| ItemDate | Date |  | Event Date |

**Table 20**: tbl\_EventReg

Primary key : EventRegId

Foreign key : LoginId , EventId , ItemId

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| EventRegId | Int | 2 | Event Registration Id |
| LoginId | Int | 2 | Reference from tbl\_Login |
| EventId | Int | 2 | Reference from tbl\_Events |
| ItemId | Int | 2 | Reference from tbl\_EventItems |
| Date | Date |  | Date of event registration |

**Table 21**: tbl\_State

Primary key : StateId

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| StateId | Int | 2 | State Id |
| StateName | Varchar | 20 | Name of State |
| Status | Int | 1 | Valid/Not |

**Table 22**: tbl\_District

Primary key : DistrictId

Foreign key : StateId

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| DistrictId | Int | 2 | District Id |
| StateId | Int | 2 | Reference from tbl\_State |
| District | Varchar | 20 | District Name |
| Status | Int | 1 | Valid/Not |

**Table 23**: tbl\_Address

Primary key : AddId

Foreign key : DistrictId

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| AddId | Int | 2 | Address Id |
| DistrictId | Int | 2 | Reference from tbl\_District |
| HouseName | Varchar | 30 | Name of House |
| Place | Varchar | 30 | Place to live |
| Pinno | Int | 6 | Pin Number |
| Status | Int | 1 | Valid/Not |

**Table 24**: tbl\_Course

Primary key : CourseId

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| CourseId | Int | 2 | Course Id |
| CourseName | Varchar | 20 | Name of the Course |
| Status | Int | 1 | Valid/Not |

**Table 25**: tbl\_Branch

Primary key : BranchId

Foreign key : CourseId

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Field Type | Size | Description |
| BranchId | Int | 2 | Branch Id |
| CourseId | Int | 2 | Reference from tbl\_Course |
| Branch | Varchar | 20 | Branch Name |
| Status | Int | 1 | Valid/Not |