**Hostel Management System**

For Partial Fulfillment of Requirement of  5th semester database

management system laboratory of BE in

**Computer Science and Engineering**

**The National Institute Of Engineering**

(Autonomous under VTU)

Mananthvadi road, mysore-570008



**Submitted by:**

**PRAKSHITHA M N**

**MEDHAVINI RAO**

**POOJA JAYANT**

**POOJA R**

**Under guidance of:**

**Mrs. Pallavi Y** **Mrs. Nandini P**

Asst .Professor, Asst .Professor,

Department of CSE, Department of CSE,

NIE, Mysore NIE, Mysore

**Contents:**

* Abstract
* Introduction of project
* Literature survey
* Existing and purpose system
* Design and ER diagram
* Test cases
* Software and hardware requirements
* Results-snapshot with display about snapshot
* Future enhancement with conclusion

**ANDINI**

**Abstract:**

Hostel management system is a software application to provide student accommodation to the hostel more efficiently . As the name specifies “HOSTEL MANAGEMENT SYSTEM” is software developed for managing various activities in the hostel. For the past few years the number of educational institutions is increasing rapidly. Thereby the number of hostels is also increasing for the accommodation of the students studying in this institution. And hence there is a lot of strain on the person who are running the hostel and software’s are not usually used in this context. This particular project deals with the problems on managing a hostel and avoids the problems which occur when carried manually.

Identification of the drawbacks of the existing system leads to the development of computerized hostel management system that will be compatible to the existing system with the system which is more users friendly and more GUI oriented. We can improve the efficiency of the system, thus overcome the drawbacks of the existing hostel management system. 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. This project also keeps details of the hostelites and employees. It is headed by warden who is also the administrator. This document is intended to minimize human work and make hostel allocation is an easier job for hostel authorities by implementing computerized database.

**Introduction of project**:

The hostel management is a system for managing the various activities in hostel. It can manage the hostel information. It manages the student information, room information, warden information, visitors details, outing details, room allocation details, fees details, mess employee details of the hostel. It can also keeps track of the number of the student in the room and availability of the room. The main purpose of this project Hostel Management System is to manage the details of hostel, rooms, allotees, rent. It manages all the information about payments, rents.

The aim of **Hostel Management System**project is to maintain all activities in managing a hostel. The project is developed by python(front end), SQL as back end server data-base. The number of hostels is increased rapidly due to this maintaining the hostel became very difficult. So to avoid all the troubles in maintaining the hostel this management system was developed. The software is completely GUI interface with simple instructions.

**Literature survey**:

Hostel management system development using python program has lots of codes, Using Internet in gathering information partially contributed to the success of this project. Due to the fact that Python is an open source program, development of hostel management system was not too difficult.  However, thanks to the cyber world (Internet) that makes it possible to study and make comparison in needs of some code function.

Among other, sqlite3 with python tutorial, python GUI with Tkinter helped us lot. It provided me with different codes which I used in the development of this program .Using of textbooks and journal on the net was also a great source of information and assistance in realizing the goal of this project. For instance, "Think Python”by Allen B. Downey and “learning python” by Mark lutz and David ascher gave a good layout of project design.

**Existing and purpose system:**

* The analysis is made when the manager begins studying the program flow. The data about various views are collected. To draw the data-flow system, modeling Experience is required for developing desired model. This project includes checking out of rooms, allocating rooms, calculating fees bill, vacating rooms, employee details, warden details.
* The Administrator can:

1. He can allot students to the hostel.

2. He can vacate the students form the hostel.

3. Control the status of the fee payment.

4. Edit the details of the students and modify the

student records.

The existing system is a manually maintained system. All the hostel records are to be maintained for the details of each student. Fee details, Room allocation, attendance etc. All these details are entered and retrieved manually.

**Purpose system:**

* The proposed system is the computerized version of the existing system.
* Provides easy and quick access over the data
* Authorization schemes
* Python and sqlite has been chosen for this project

**Table design and ER diagram**:

The following are the tables involved in proposed system

**Login:**

|  |  |
| --- | --- |
| Fieldname | **Datatype** |
| wardenId | varchar |
| Password | varchar |

**Student:**

|  |  |
| --- | --- |
| s\_id | varchar |
| s\_name | varchar |
| mob\_no | Integer |
| g\_name | varchar |
| DOB | Date |
| Sem | Integer |
| Branch | varchar |
| Age | Integer |
| Area | varchar |
| City | varchar |
| State | varchar |
| Room no | Integer |

**Fees:**

|  |  |
| --- | --- |
| S\_id | varchar |
| Room no | Integer |
| Block id | varchar |
| Rent | Integer |
| Mess expense | Integer |
| Duedate | Date |
| Paid date | Date |
| Status | Varchar |

**Room details:**

|  |  |
| --- | --- |
| Room\_no | Integer |
| block\_id | Integer |
| room\_capacity | Integer |
| Occupied | Integer |

**Block details:**

|  |  |
| --- | --- |
| Block id | Integer |
| Warden id | Integer |
| Warden name | varchar |
| No of rooms | Integer |
| Total capacity | Integer |
| Room capacity | Integer |

**Mess employee:**

|  |  |
| --- | --- |
| Emp\_id | varchar |
| Emp\_name | varchar |
| Mobile no | Integer |
| Salary | Float |

**Warden :**

|  |  |
| --- | --- |
| Warden\_id | varchar |
| Warden\_name | varchar |
| Mobile\_no | Integer |
| Salary | Float |

**Mess:**

|  |  |
| --- | --- |
| Emp\_id | varchar |
| Emp\_name | varchar |
| Timings | Integer |

**Outings:**

|  |  |
| --- | --- |
| S\_id | varchar |
| Date | Date |
| In\_time | Time |
| Out\_time | Timr |

**Guardian:**

|  |  |
| --- | --- |
| Guardian\_name | varchar |
| S\_id | varchar |
| G\_mob\_no | Integer |
| G\_email\_id | varchar |

**Visitors:**

|  |  |
| --- | --- |
| S\_id | varchar |
| V\_name | varchar |
| Date | Date |
| Relationship | varchar |
| In\_time | Time |
| Out\_time | Time |

**ER diagram:**

# C:\Users\Home\Downloads\erdplus-diagram (5).png

**Test cases:**

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. Testing presents an interesting of a system using various test data. Preparation of the test data plays a vital role in the system testing. After preparation the test data, the system under study is tested those test data. Errors were found and corrected by using the following testing steps and corrections are recorded for future references. Thus, series of testing is performed on the system before it is already for implementation.

**System Testing:**

Software testing is a critical element of software quality assurance and represents the ultimate review of specifications, design and coding. The testing phase involves the testing of system using various test data; Preparation of test data plays a vital role in the system testing. After preparation the test data, the system under study is tested.

Those test data, errors were found and corrected by following testing steps and corrections are recorded for future references. Thus a series testing is performed on the system before it is ready for implementation.

The various types of testing on the system are:

* Unit testing
* Integrated testing
* Validation testing
* Output testing
* User acceptance testing

**Unit testing**

Unit testing focuses on verification effort on the smallest unit of software design module. Using the unit test plans. Prepared in the design phase of the system as a guide, important control paths are tested to uncover errors within the boundary of the modules. The interfaces of each of the modules under consideration are also tested. Boundary conditions were checked. All independent paths were exercised to ensure that all statements in the module are executed at least once and all error-handling paths were tested. Each unit was thoroughly tested to check if it might fall in any possible situation. This testing was carried out during the programming itself. At the end of this testing phase, each unit was found to be working satisfactorily, as regarded to the expected out from the module.

**Integration Testing**

Data can be across an interface one module can have an adverse effect on another’s sub function, when combined may not produce the desired major function; global data structures can present problems. Integration testing is a symmetric technique for constructing tests to uncover errors associated with the interface. All modules are combined in this testing step. Then the entire program was tested as a whole.

**Validation Testing**

At the culmination of integration testing, software is completely assembled as a package. Interfacing errors have been uncovered and corrected and final series of software test-validation testing begins. Validation testing can be defined in many ways, but a simple definition is that validation succeeds when the software functions in manner that is reasonably expected by the consumer. Software validation is achieved through a series of black box tests that demonstrate conformity with requirement. After validation test has been conducted, one of two conditions exists.

* The function or performance characteristics confirm to specification that are accepted.
* A validation from specification is uncovered and a deficiency created.

Deviation or errors discovered at this step in this project is corrected prior to completion of the project with the help of user by negotiating to establish a method for resolving deficiencies. Thus the proposed system under consideration has been tested by using validation testing and found to be working satisfactorily.

**Output Testing**

After performing the validation testing, the next step is output testing of the proposed system, since a system is useful if it does not produce the required output in the specific format required by them tests the output generator displayed on the system under consideration. Here the output is considered in two ways: - one is onscreen and the other is printed format. The output format on the screen is found to be correct as the format was designed in the system design phase according to the user needs. As far as hardcopies are considered it goes in terms with the user requirement. Hence output testing does not result any correction in the system.

**User Acceptance Testing**

User acceptance of the system is a key factor for success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with prospective system and user at the time of developing and making changes whenever required.

**Software and hardware requirements**:

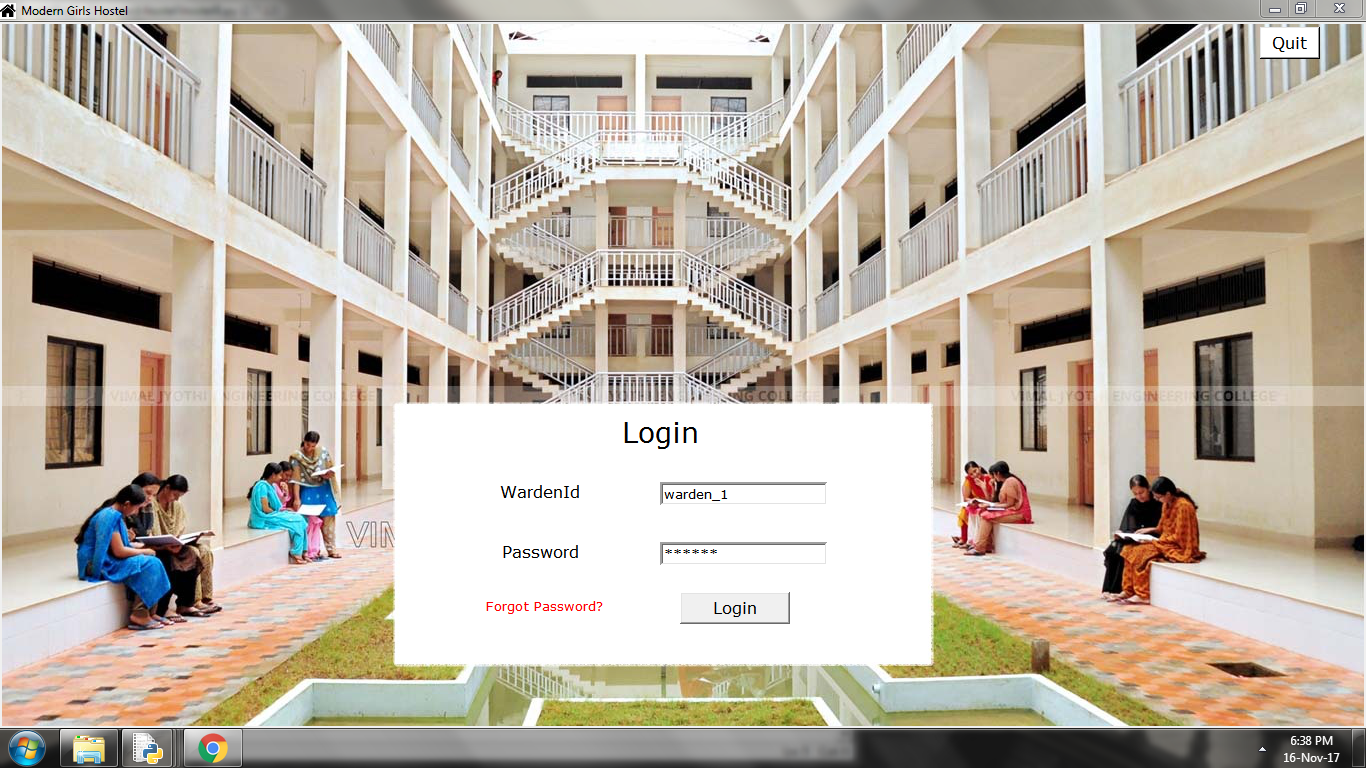
* Operating system – Windows XP or higher
* Python 2.x/3.x (front-end)
* SQL server - MySql (back-end)

**Hardware requirements**:

* RAM – 2GB or more
* Processor – Intel pentium 4 or higher
* Hard disk – 40GB or higher
* External devices – Not required

**Result-Snapshot with display about snapshot:**

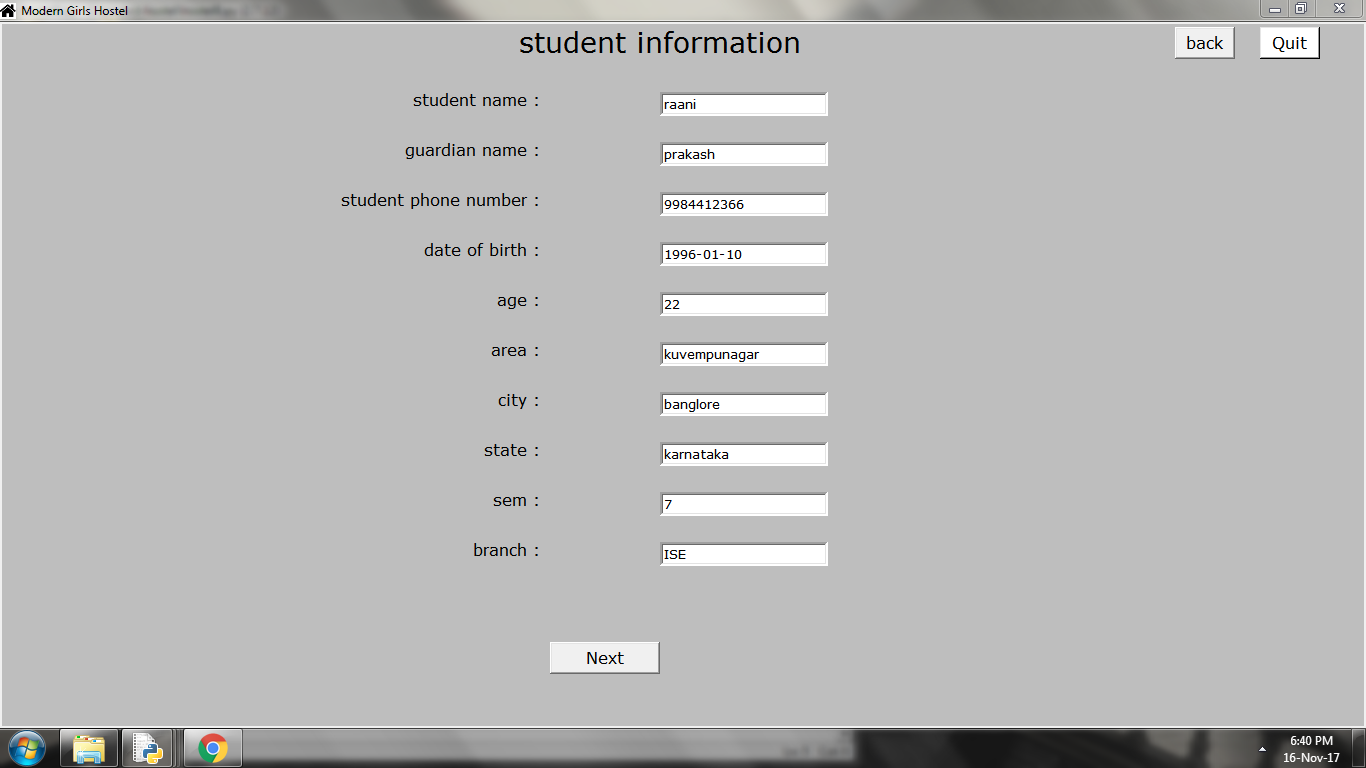
**Login page:**



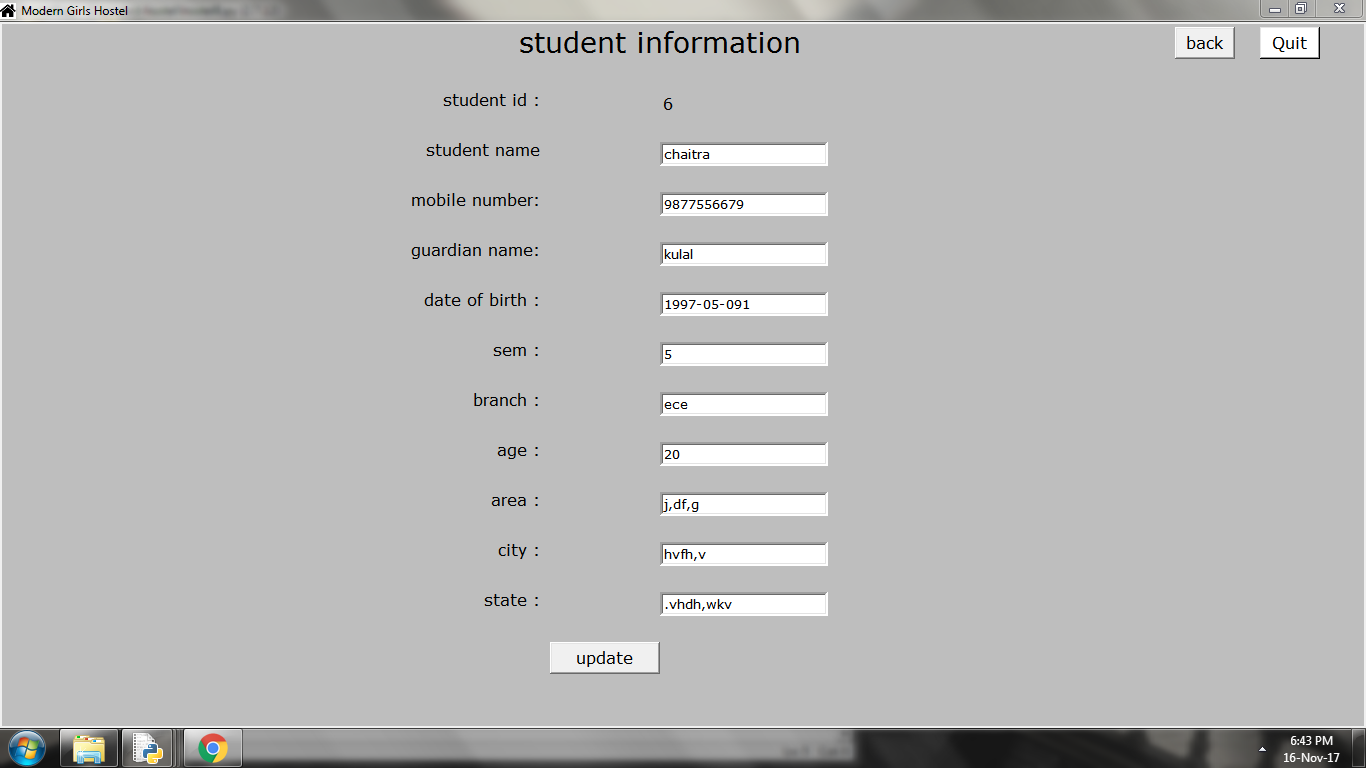
**Main page:**



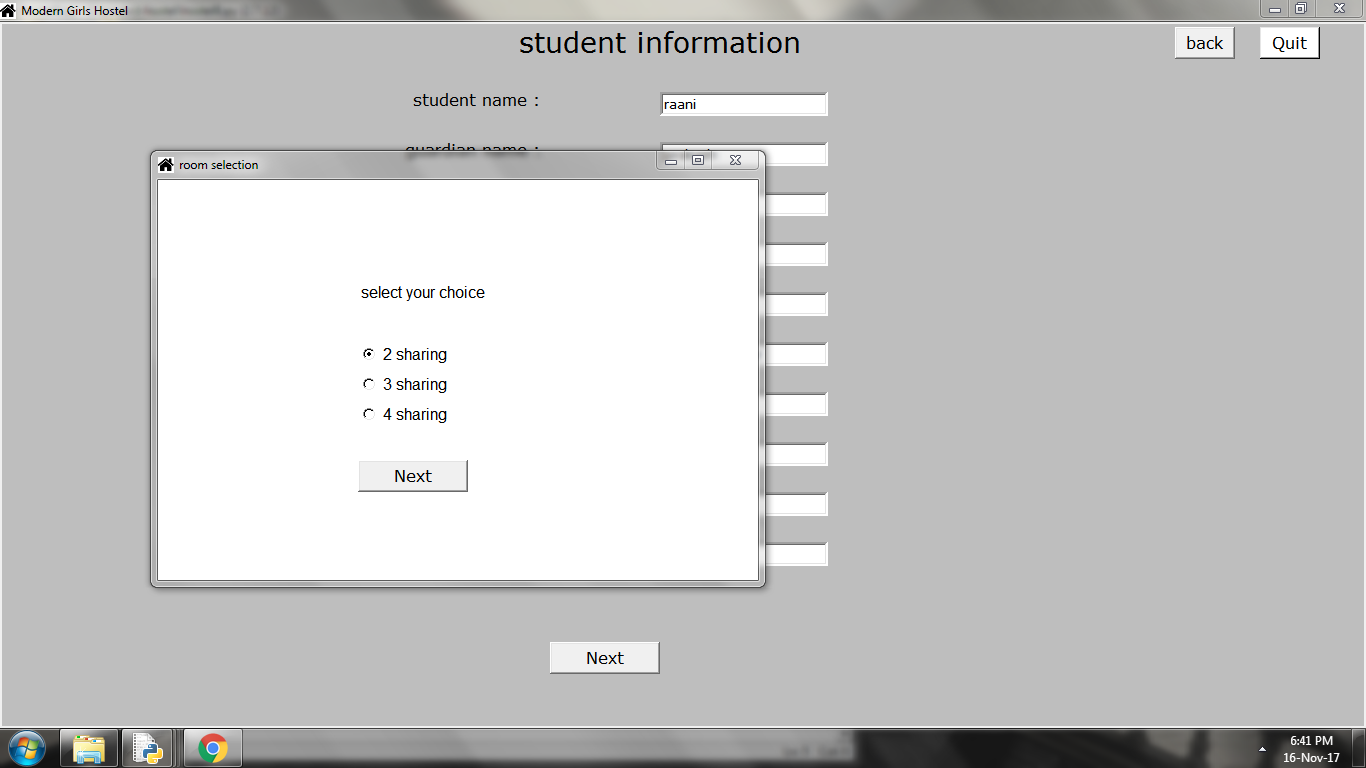
**Student information**:



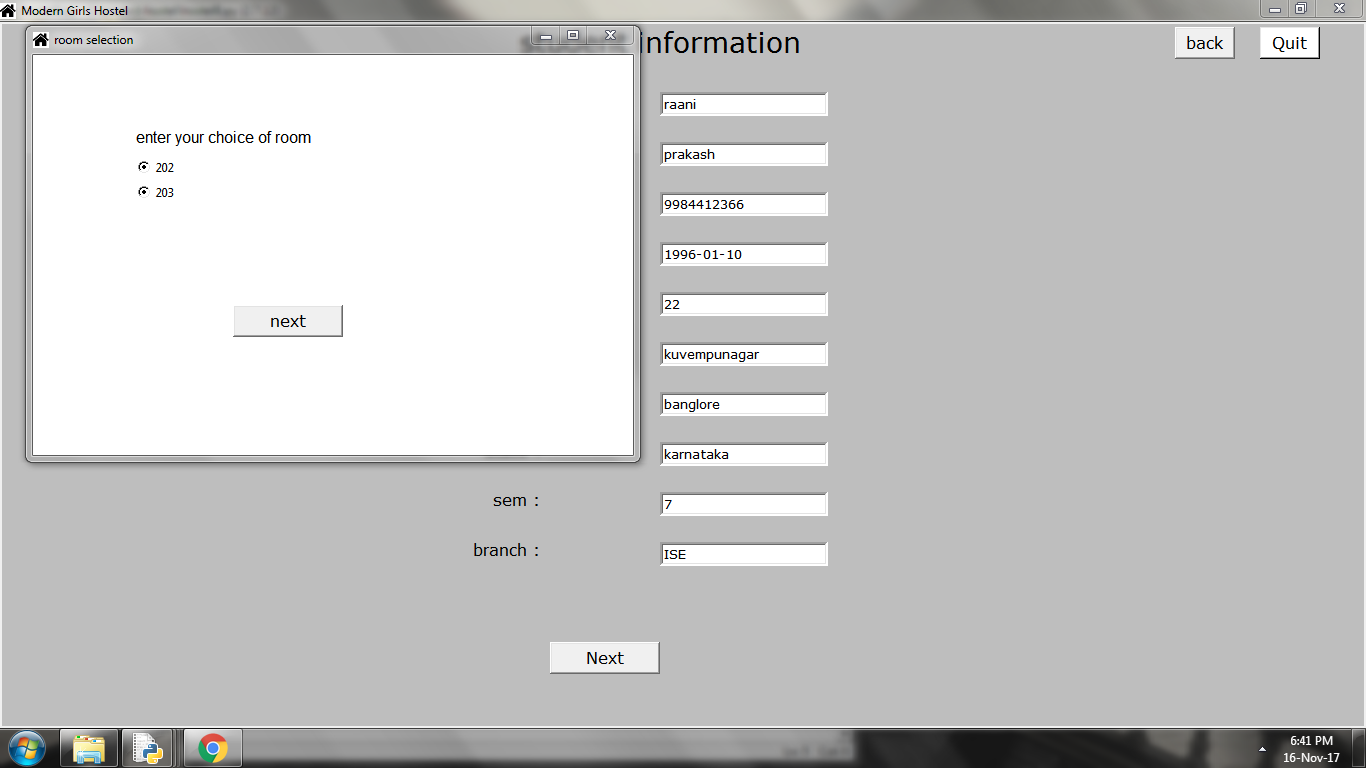
**Updation of student information:**



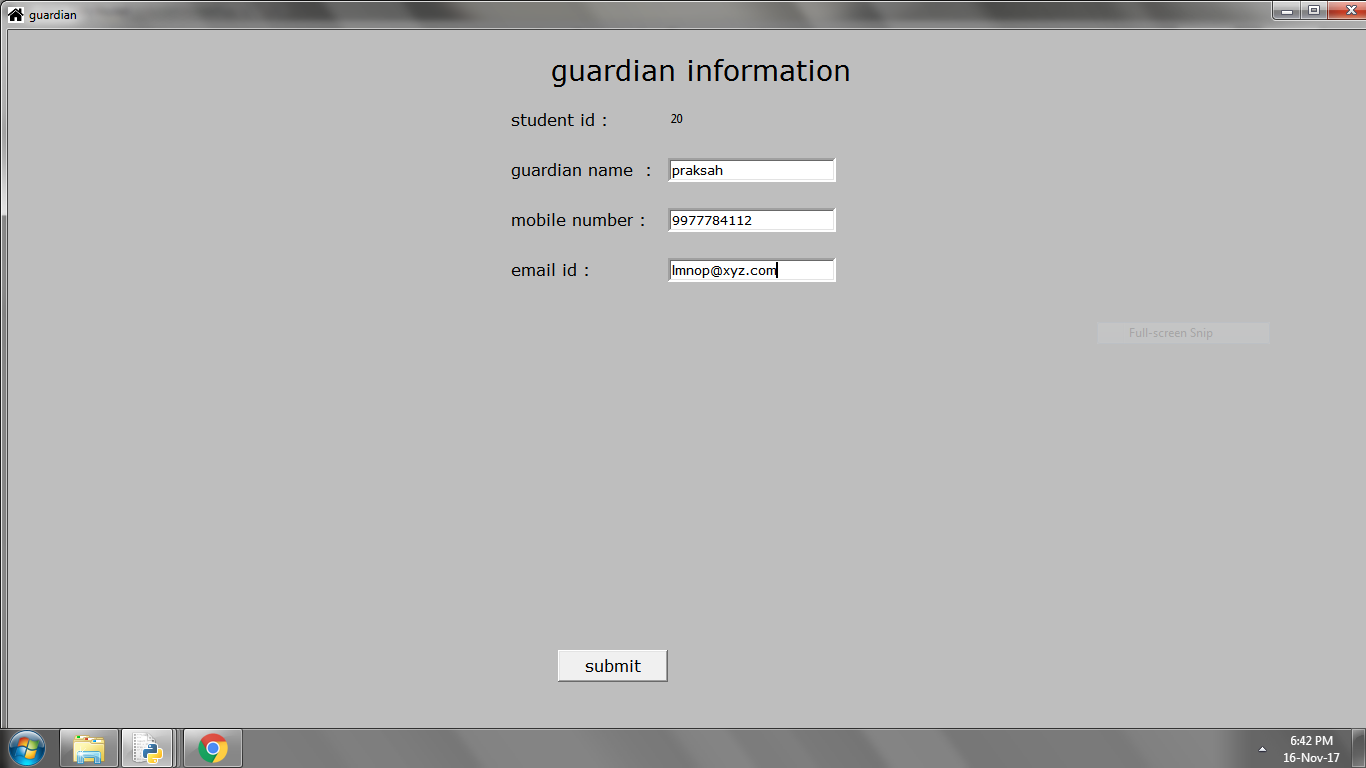
**Room details:**



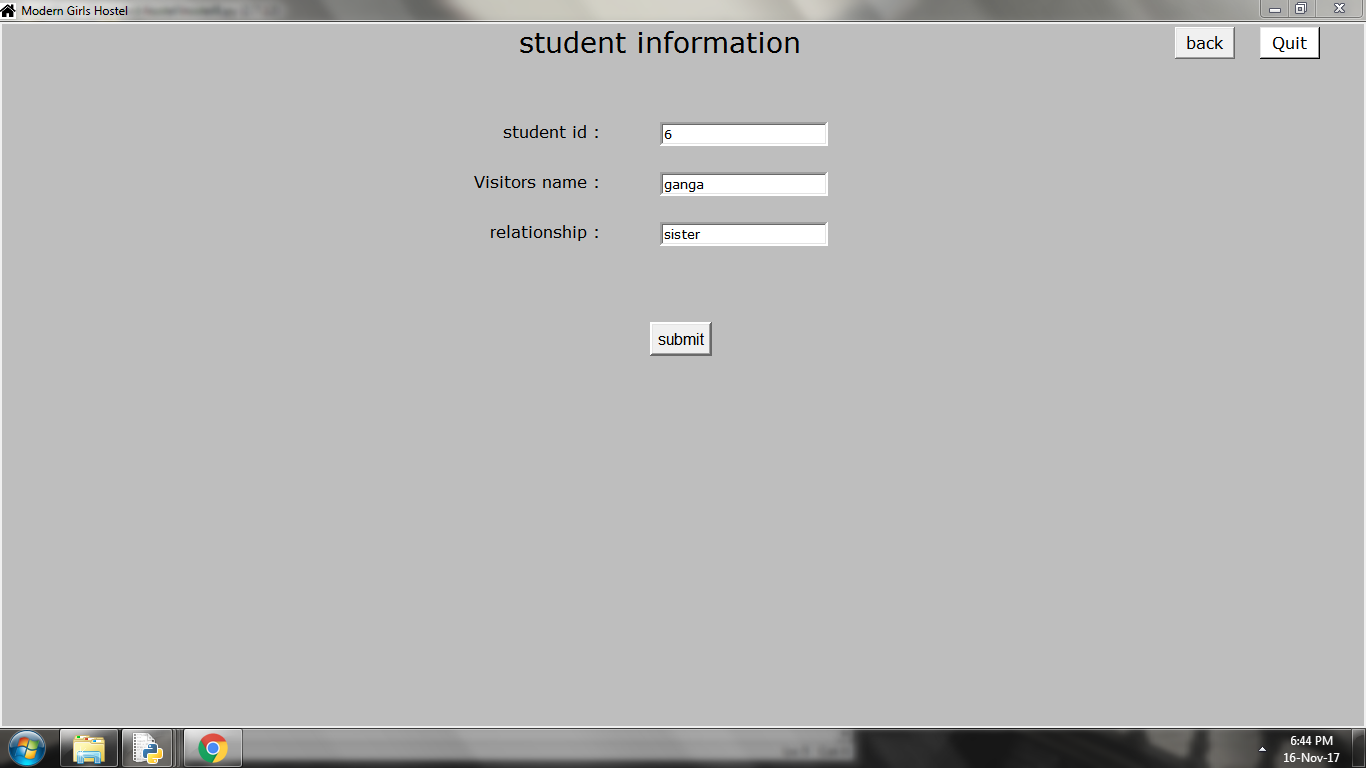
**Choice of room:**



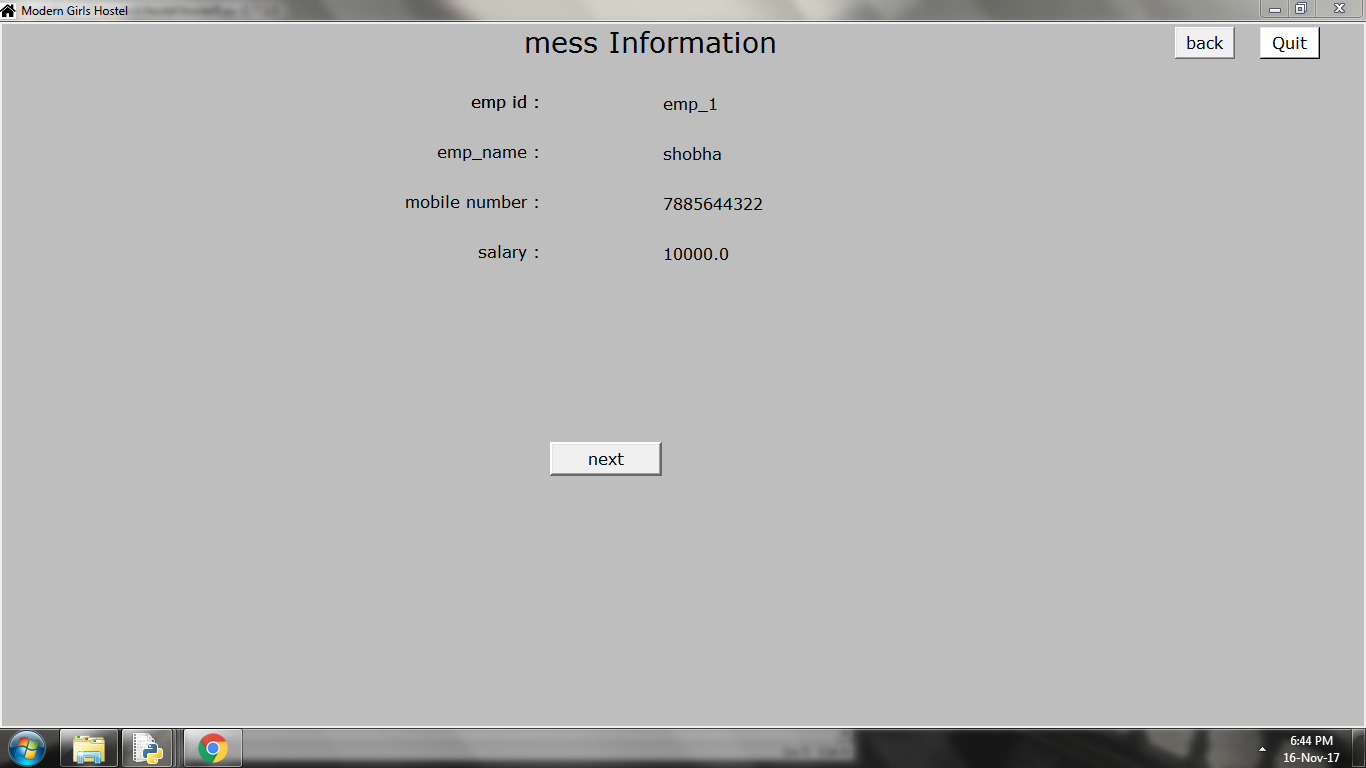
**Guardian information:**



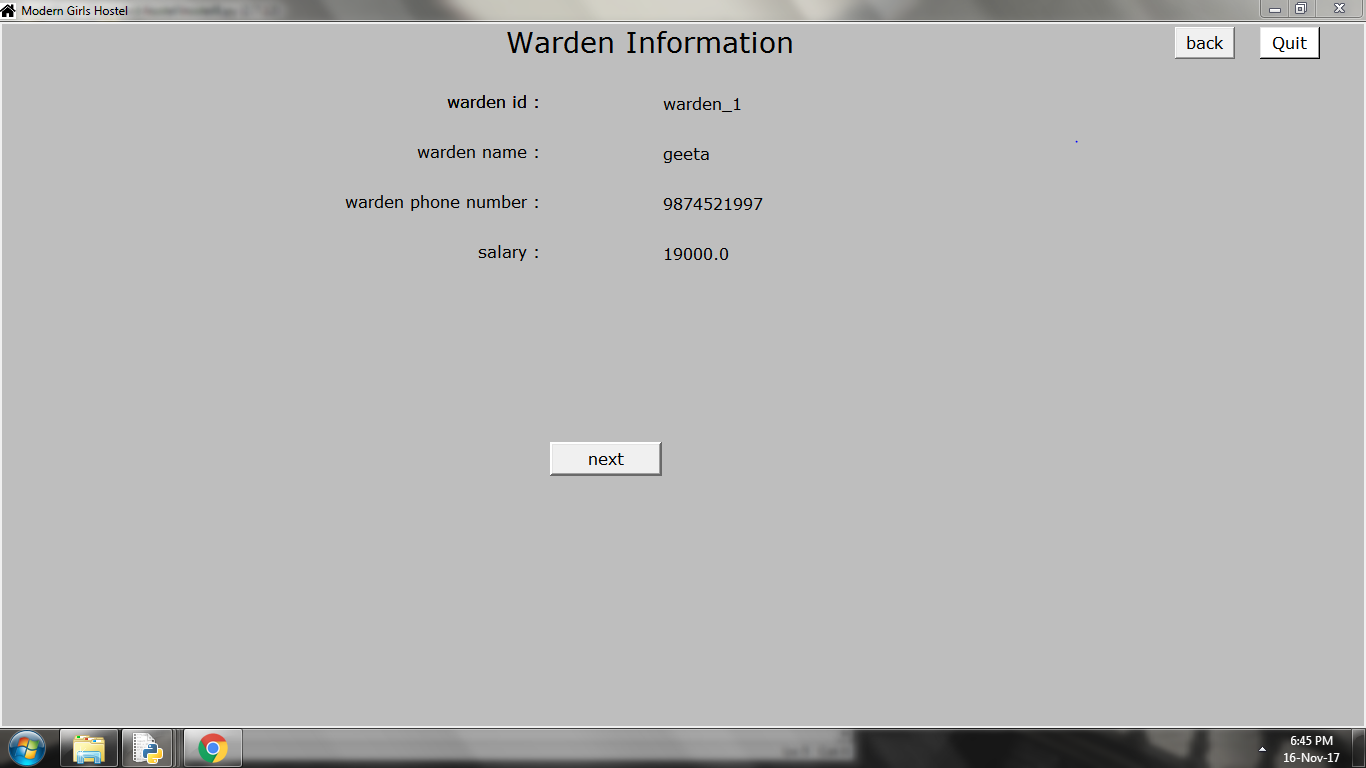
**Visitors details:**



**Mess information:**



**Warden information:**



**Future enhancement with conclusion**:

* We can also provide accommodation for staffs.
* We can also maintain attendance record.
* We can implement text message communication instead of email communication.

**Conclusion:**

* Identification of the drawback of the existing system leads to the designing of computerized system that will be compatible to the existing system which is more user friendly and more GUI oriented.
* To conclude the description about the project , developed using PYTHON(Tkinter) and SQL SERVER is based on the requirement specification of the user and the analysis of the existing system , with flexibility for future enhancement