*Project Report On*

**Hostel Management System**

*Submitted by*

**K Sujith Bhatt (181mt018)**

**Shreehari M (181cv145)**

**Chirag K (181me119)**

**VI SEM B.Tech (IT)**

*Under the guidance of*

**Dr. Anand Kumar M**

**Dept of IT, NITK Surathkal**

*in partial fulfillment for the award of the degree*

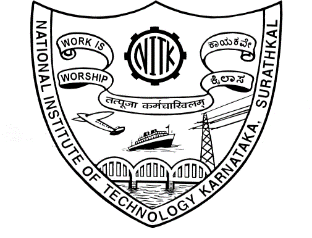
*of*

**Bachelor of Technology**

in

**Information Technology**

at



**Department of Information Technology**

**National Institute of Technology Karnataka, Surathkal**.

***February 2020***

**Department of Information Technology**

**National Institute of Technology Karnataka, Surathkal**

**Mid Semester Evaluation (February 2020)**

*Course code :* IT 399

*Course Title:* Minor Project

*Title of the Project: Hostel Management System*

*Details of Project Group*

*Name of the Student Register No. Signature with Date*

1. K Sujith Bhatt 181mt018

2. Shreehari M 181cv145

3. Chirag K 181me119

**Name of Project Guide: Dr. Anand Kumar M**

Signature of the Project Guide:

Place:

Date:

Hostel Room Allocation System

Software Design Description

**K Sujith Bhatt: 181MT018**

**Shreehari M: 181CV145**

**Chirag K: 181ME119**

**Alpha version: (2.0.1) Date: 12/6/2020**

Table of Content

### *Introduction -------------------------------------------05*

* 1. Purpose ------------------------------------------------05
  2. Objective -----------------------------------------------05
  3. Scope of the Project --------------------------------05
  4. Overview of the Project ----------------------------05

### *Data Design ------------------------------------------****06***

* 1. Conceptual Schema --------------------------------06

### *Entities and Attributes -----------------------------****07***

* 1. Administrator -------------------------------------------07
  2. Student --------------------------------------------------08
  3. Courses -------------------------------------------------09
  4. Room ----------------------------------------------------10
  5. States ---------------------------------------------------10
  6. User Registration ------------------------------------11

### *Working -----------------------------------------------****12***

### *Languages and Tools Used ---------------------****12***

### *Bugs ---------------------------------------------------****15***

### *Contributions -----------------------------------------****15***

# Introduction

The Software Design Document is a document to provide documentation which will be used to aid in software development by providing the details for how the software should be built. Within the Software Design Document is narrative and graphical documentation of the software design for the project including ER Diagrams, Conceptual Schema, and other supporting requirement information.

## Purpose

The purpose of the Software Design Document is to provide a description of the design of a system fully enough to allow for software development to proceed with an understanding of what is to be built and how it is expected to build. The Software Design Document provides information necessary to provide the description of the details for the software and system to be built.

## Objective

* To deal with Hostel Management System in an easy and an efficient manner.
* Create strong and database that allows for any connection in a secret way, to prevent any outside or inside attacks.

## Scope of the Project

* Hostel Management System is designed for Hostel (like schools, Universities).
* There will be predefined criteria for the Reservation to the hostels.
* He/She checks the attested application forms of the students obtained from the internet and verify it with the student database.
* If the students are found eligible then they are allotted to the hostel Room.

## Overview of Project

Hostel Room Allocation System is a web application which aims at computerization of the current procedure of allocating hostel rooms

# Data Design

## Conceptual Schema

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | User\_name | email | Password | Reg\_date | Update\_date |

**Admin**

**Course**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Coarse\_code | Coarse\_sn | Coarse\_fn | posingDate |

**Registration**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID | roomNo | seater | feespm | foodstatus | stayfrom | duration | course | regno | firatName |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| middelname | lastName | gender | Contact no | emailId | eygcontact | gurdianName | Gurdian\_rel | Gurd\_contact | correAdress |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| correCity | correState | correPin | pmtAddress | pmtCity | pmtSate | pmtPin | postingDate | updationDate |

**Rooms**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | seater | RoomNo | fees | PostingDate |

**States**

|  |  |
| --- | --- |
| ID | State |

**User Registration**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID | regNo | FirstName | Middlename | lastName | gender | contactNo | email | Password | regDate |

|  |  |
| --- | --- |
| updationDate | passUdate |

# Entities and Attributes

This section of the document explains the entities used in the project, their attributes and how they will work together. Basically, this is intended to make the design more easy and understandable for everyone.

### **Entities**

1. Administrator
2. Student registration
3. Coarses
4. Room
5. States
6. Userregrisrations

## Administrator

Every hostel has an administrator and is represented using the ‘administrator’ entity. Administrator entity takes part in following relationships.

1. **Administrator**​ manages Hostel.

#### Attributes

|  |  |  |
| --- | --- | --- |
| **Name** | **Data Type** | **Type** |
| ID | integer | Primary Key attribute |
| username | string | Non\_key attribute |
| email | varchar | Non\_key attribute |
| Password | varchar | Non\_key attribute |
| Reg\_date | timesamp | Non\_key attribute |
| Updation\_date | date | Non\_key attribute |

## Student

Every hostel has students and they are represented by the ‘student’ entity.

Student entity participates in the following relationships.

1. Hostel has ​**Students**​.
2. **Student** ​has personal detailes.
3. **Students**​ stay at room

#### Attributes

|  |  |  |
| --- | --- | --- |
| **Name** | **Data Type** | **Type** |
| Student\_ID | integer | Primary Key attribute |
| Room\_id | integer | Foreign Key attribute |
| seater | integer | Non\_key attribute |
| feespm | integer | Non\_key attribute |
| stayfrom | date | Non\_key attribute |
| duration | integer | Non\_key attribute |
| course | varchar | Foreign Key attribute |
| regno | integer | Non\_key attribute |
| Fname | string | Non\_key attribute |
| Lname | string | Non\_key attribute |
| gender | varchar | Non\_key attribute |
| Mob\_No | bigint | Non\_key attribute |
| emailid | varchar | Non\_key attribute |
| egycontactno | Bigint | Non\_key attribute |
| Guardian name | varchar | Non\_key attribute |
| Guardian\_rel | varchar | Non\_key attribute |
| Guardian\_cont\_no | bigint | Non\_key attribute |
| correCity | varchar | Non\_key attribute |
| correAddress | varchar | Non\_key attribute |
| correState | varchar | Non\_key attribute |
| correPincode | integer | Non\_key attribute |
| pmtAddress | integer | Non\_key attribute |
| pmtState | varchar | Non\_key attribute |
| pmtPincode | ineger | Non\_key attribute |
| postingdate | timestamp | Foreign key attribute |
| updationDate | date | Non\_key attribute |

3.3 Courses

This entity is added so that students in same courses get allotted to same rooms.

**Attributes**

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Type** |
| ID | integer | Primary key attribute |
| Course\_code | varchar | Non\_key attribute |
| Course\_sn | Varchar | Non\_key attribute |
| Course\_fn | varchar | Non\_key attribute |
| Posting\_date | timestamp | Foreign key attribute |

## 3.4 Room

Every Hostel has rooms and they are represented using ‘room’ entity. Room entity participates in the following relationships.

1. Hostel has ​**Rooms**​.
2. Student stays at ​**room**​.

#### Attributes

|  |  |  |
| --- | --- | --- |
| **Name** | **Data Type** | **Type** |
| ID | integer | primary Key attribute |
| seater | integer | Foreign key attribute |
| Room\_no | integer | Foreign key attribute |
| fees | integer | Non\_key attribute |
| postingdate | timestamp | Foreign key attribute |

3.5 States

This entity is created to allot rooms for people from same states.

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **type** |
| ID | integer | Primary key attribute |
| State | varchar | Foreign key attribute |

3.6 User registration

This entity is created for having the data of the user, so as to check during userlogin

|  |  |  |
| --- | --- | --- |
| **Name** | **Data Type** | **type** |
| ID | integer | Primary attribute key |
| regno | varchar | Non\_key attribute |
| firstName | varchar | Non\_key attribute |
| lastName | varchar | Non\_key attribute |
| gender | varcahr | Non\_key attribute |
| ContactNo | bigint | Non\_key attribute |
| email | varchar | Non\_key attribute |
| password | varchar | Non\_key attribute |
| regDate | timestamp | Non\_key attribute |
| updationDate | varchar | Non\_key attribute |
| passUdate | varchar | Non\_key attribute |

4. Working

The working of the model is described in the presentation file.

<https://drive.google.com/folderview?id=1pWzGPTkJCn8CCysPXfEi_TPMk2UFyXbq>

5. Languages and Tools

**A.Languages**

1. HTML
2. CSS
3. php
4. Mysql
5. Javascript

5.A.1 HTML

**HTML** is the universal markup language for the Web. **HTML** lets you format text, add graphics, create links, input forms, frames and tables, etc., and save it all in a text file that any browser can read and display.

5.A.2 CSS

Cascading Style Sheets (**CSS**) is the standard language for styling structured documents, such as HTML. However, **CSS** lacks most of the traditional programming constructs, including variables and functions, which enable code reuse and structured programming.

5.A.3 php

**PHP** is a serverside scripting **language**. that is used to develop Static websites or Dynamic websites or Web applications. **PHP** stands for Hypertext Pre-processor, that earlier stood for Personal Home Pages. **PHP** scripts can only be interpreted on a server that has **PHP** installed.

5.A.4 Mysql

**MySQL** is a freely available open source Relational Database Management System (RDBMS) that uses Structured Query Language (SQL). SQL is the most popular language for adding, accessing and managing content in a database. It is most noted for its quick processing, proven reliability, ease and flexibility of use.

5.A.5 Javascript

JS, is a programming language that conforms to the ECMAScript specification. JavaScript is high-level, often just-in-time compiled, and multi-paradigm. It has curly-bracket syntax, dynamic typing, prototype-based object-orientation, and first-class functions.

**B. Tools**

1. Xampp
2. Chrome
3. Sublime text
4. MySQL workbench
5. bootstrap

5.B.1 Xampp

**XAMPP** is web server software package with it's core as apache along with maria DB and interpreters of PHP, Perl, etc pre-configured. In an **abstract** point of view, **XAMPP** IS Apache \*for Windows\*. It is a single software package which installs WAMP stack in just few simple clicks.

5.B.3 Sublime text

**Sublime Text** is a shareware cross-platform source code editor with a Python application programming interface (API). It natively supports many programming languages and markup languages, and functions can be added by users with plugins, typically community-built and maintained under free-software licenses.

5.B.4 MySQL workbench

**MySQL Workbench** is a unified visual tool for database architects, developers, and DBAs. **MySQL Workbench** provides data modeling, SQL development, and comprehensive administration tools for server configuration, user administration, backup, and much more.

5.B.5 Bootstrap

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.

6. Bugs:

1. During hostel booking, once the room is selected, the seater of the room displayed can be changed manually.
2. During hostel booking, the total amount of fees isn’t displayed automatically.
3. The total amount of fees calculated shown after the booking has a logical mistake in the equation.

7. Contributions:

1.Chirag K:

a. Research into the topic, the tools that can be used, the languages required.

b. Deciding on a basic structure for the webpage

c. Coding the user login page

d. Searching for the bugs

2. K Sujith Bhatt:

a. Coding the app body and the rest of the interface.

b. Searching for bugs

c. Debugging the code

d. Report preparation

1. Shreehari M:
2. Creating the conceptual schema
3. Debugging the code
4. Improving the format of the webpage
5. Report preparation