National Institute of Technology Calicut

Department of Computer Science & Engineering

Software Requirements Specification

Online Hall Booking System

<version 1.0>

Team – 01

[Leader] Neha Kumari

Neethu PK

Subhodip Banerjee

Amith Prakash

<u>Table of Contents</u>

List of Figures

1. <u>Introduction</u>

1.1 **Purpose:**

The purpose of this *Software Requirement Specification*(SRS) document,is to present a detailed description of "*NITC Online Hall booking System*". It will explain the purpose, the features, the interfaces of the system, the constraints under which it must operate and how the system will react to external stimuli. The document will also explain the *functional and non-functional* aspects of the software. The intended audience for this document are the users and developers of the software.

1.2 **Scope:**

This Online Hall Booking System is designed for National Institute of Technology, Calicut. It will be designed to maximize the administration productivity by providing assisstance in automating the Hall Status and Booking process, which would otherwise have to be performed manually. *Details* of all the Halls along with the *prior reservations* will be displayed in the website for any viewers knowledge, but it allows only *authorized users* to book the Hall.The system also contains a relational database containing a list of the registered user and their password, and different lists containing fields such as the item and their respective attributes.

More specifically, this system is designed to provide a *user-friendly environment* for the Booker to book any of the Halls for his/her event. The system will facilitate to search for the appropriate hall by taking in the requirements of the Booker, *check the Status* of a Hall immediately whenever a Booker wants to know, *request for booking* of the hall and also reserving the hall for a particular event.

1.3 **Definitions, Acronyms & Abbreviations:**

Term	Definition
Attribute	It describes the various feature of the item.
Database	Collection of all the information monitored by this system.
Field	A cell within a form in a webpage.
Booker	A person who wants to book a Hall.
Department-InCharge	A person who is responsible for the booking of all the available Halls which comes under the respective Department.

Administrator	A person who has the access to add new Departments or to make changes to the system.
Software Requirements Specification	A document that completely describes all of the functions of a proposed system and the constraints under which it must operate.
Hall	Refers to Conference hall, Auditorium, Open Air Theatre, etc, in NITC.
Request Booking	Refers to the act of a Booker sending a request for reservation of a particular Hall.
Check Status	Act of a Department-InCharge, checking the pending booking requests.
Cancel Booking	Act of a Booker trying to cancel the previously booked Hall.
Account	Refers to the profile of a particular user in the system.
System	Refers to the NITC Online Hall Booking application to be developed.

1.4   **References:**

-IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.* IEEE Computer Society, 1998.
-Software Requirements Specification, Version 1.0, April 15, 2004, Web Publishing System - Joan Teamleader, Paul Adams, Bobbie Baker, Charles Charlie

1.5   **Overview of Document:**

The next section, the *Overall Description section*, of this document gives an overview of the functionality of the product. It describes the *Use Cases* which will help us determine and provide better understanding for developing the technical aspects of the document.
The third chapter, *Requirements Specification* section, of this document is written primarily for the developers and describes in technical terms the details of the functionality of the product.

2. <u>Overall Description</u>

2.1 **System Environment**:
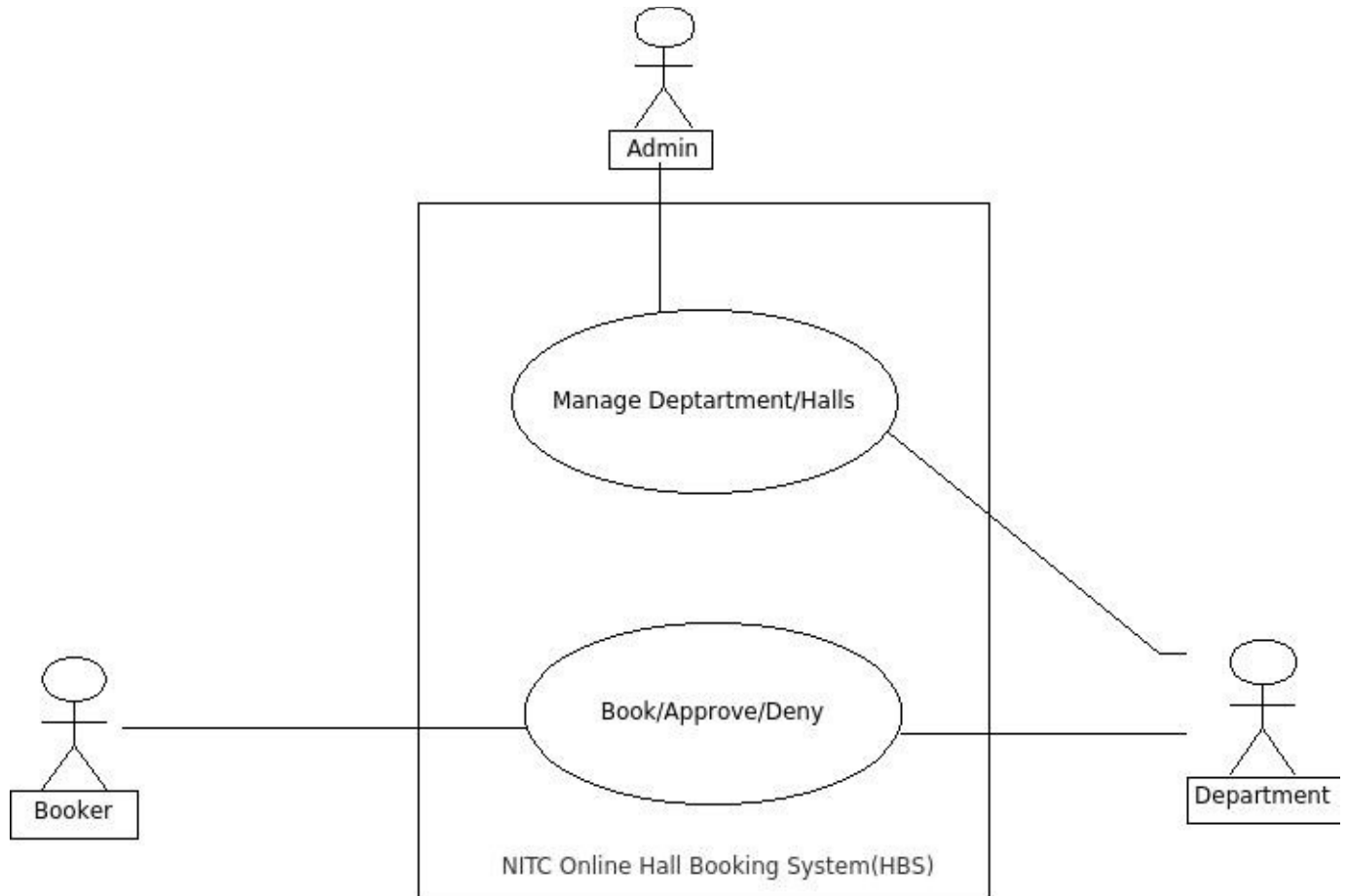
Figure 1 - System Environment

The NITC Online Hall Booking System has 3 types of users, namely *Booker, Administrator and Department-InCharge*. There are 5 kinds of Bookers namely Student(CR)/SAC representative, Faculty, HoD, Dean & Director/Registrar. Each hall comes under the supervision of a Department. Departments-InCharge is mainly responsible for the approval of a booking and maintaining student accounts. Each kind of Booker is given different *priorities*. Administrator looks after the Department-InCharges' accounts and making changes in the system.

2.2 **Functional Requirements Specification**:

This section outlines the use cases for each of the active users separately. The *Booker*, the and *Department-InCharge* have are the main actors in this system.

2.2.1 **Department-InCharge Use Case**:

The Department-InCharge has the following set of Use Cases, Diagram**:**
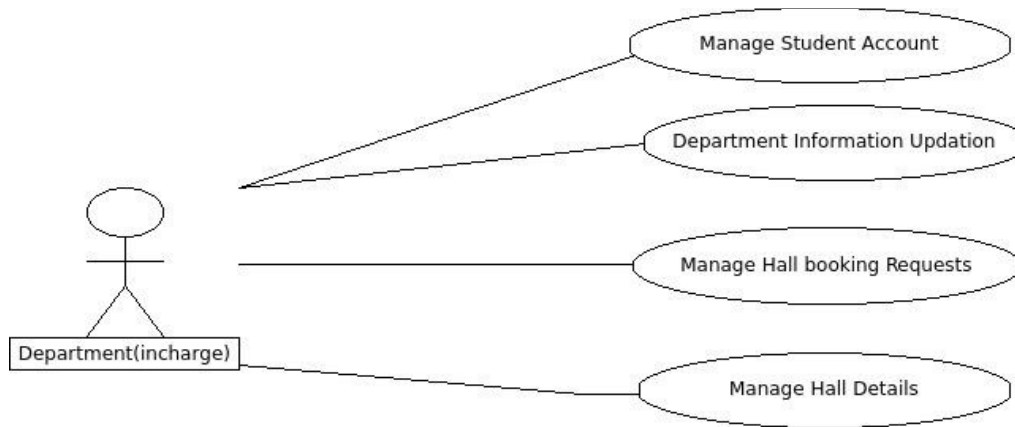
Figure 2- Department Use Cases

Manage Student Account Use Cases:
a)Use Case: **Update Student details**
Diagram:

Brief Description:
  The Department-InCharge can *update* the information regarding the current Student representative/Faculty or *add new* Student representative/Faculty.

Initial Setp-by-Step Description:
  Before this use case can be initiated, the Department-InCharge has already accessed his profile page in the HBS.
• The Department-InCharge selects to *Add/Update* student representative/Faculty.
• The System presents its choice of adding.
• The Department-InCharge chooses to add or update.
○ Student/Faculty *Name*
○ Student/Faculty *Contact Details*
○ Student/Faculty *ID no*
• If the Department-InCharge is *updating* an Student represntative/Faculty, the system presents a list of Student Representative types(Student Class Representative, SAC Representative, Student Association Presidents) or Faculty to choose from.
• The Department-InCharge *fills in the information and submits* the form.
• The System *validates* the information and redirects the Department-InCharge to his profile page.

Department Information Updation Use cases
b)Use Case: **Edit Department Information**
Diagram:

Department(incharge)

Brief Description:
  The Department-InCharge *updates* the Existing department details.

Initial Setp-by-Step Description:
  Before this use case can be initiated, the Department-InCharge has already accessed his profile page in the HBS.
• The Department-InCharge selects to *Add/Update*

Department Information Details.
- The System presents its choice of adding or updating.
- Department-InCharge chooses to add or update information.
  - *Person In Charge details* Updation
  - *Contact information* Updation
  - *Department Location/Address* updation
  - *Number of Halls Manged* Updation
- The Department-InCharge *fills in the information and submits* the form.
- The System *validates* the information and redirects the Department-InCharge to his profile page.

Manage Hall Details Use Case:
c)Use Case: **Update Hall Details**
Diagram:

Department(incharge)

Brief Description:
The Department-InCharge *updates* the Existing Hall details.

Initial Setp-by-Step Description:
Before this use case can be initiated, the Department-InCharge has already accessed his profile page in the HBS.
- The Department-InCharge selects to *Add/Update* Hall Information Details.
- The System presents its choice of adding or updating.
- Department-InCharge chooses to add or update information.
  - *No. of Seats* Updation
  - *Description of Hall* Updation
  - Availability of:
    - *Podium*
    - *Projector*
    - *Sound System*
    - *AC*(Air Conditioner) / *Non AC*
- The Department-InCharge *fills in the information and submits* the form.
- The System *validates* the information and redirects the Department-InCharge to his profile page.

Manage Hall Booking Requests Use Cases:
d)Use Case: **Check Status of Booking requests**
Diagram:

Brief Description:
The Department-InCharge *checks the status* of pending Booking requests.

Initial Setp-by-Step Description:
Before this use case can be initiated, the Department-InCharge has already accessed his profile page in the HBS.
• The Department-InCharge selects to *Check-Status* button.
• The System presents the *list of pending requests*.
• Department-InCharge *selects* a particular request.

e)Use Case: **Approve / Decline a booking request**
Diagram:

Brief Description:
The Department-InCharge *Approve or Declines* a Booking request.

Initial Setp-by-Step Description:
Before this use case can be initiated, the Department-InCharge has already accessed his profile page in the HBS.
• The Department-InCharge *selects Pending Request*.
• The System presents two options: *Approve* Request & *Reject* Request.
• Department-InCharge *selects* a particular Option.
• The system sends a *Confirmation or Declination Mail* to the respective Booker.

- The System Sends a copy of the mail in the Department-Incahrge Mail.
- The system *updates* the Database.
- The System redirects the Department-Incharge to the Status Page.

2.2.2 **Administrator Use Case**:
The Administrator has the following set of Use Cases,
Diagram:
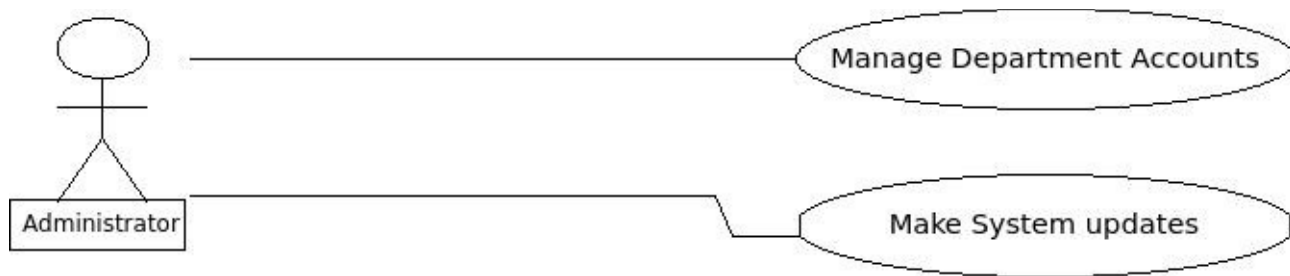
Figure-3 : Use Case Administrator

Manage Department Accounts Use Cases:
f)Use Case: **Add Department**
Diagram:

Brief Description:
The Administrator *adds a new* Department to the Already Existing lists of Department.

Initial Setp-by-Step Description:
Before this use case can be initiated, the Administrator has already accessed his profile page in the HBS.
- The Administrator Selects *Add Department*.
- The System Presents a *choice for addition* of Department.
- The Administrator fills in the required details

○ *Name* of the department
○ *Person in charge* of the department
○ *No. of Halls* managed by the department.
○ *Contact details* of the department.
• The System *validates* the information and creates a new profile for that department.
• The Administrator is redirected to his profile page.

g)Use Case: **Manage Department account**
Diagram:

Brief Description:
The Administrator *manages* the existing Department details.

Initial Setp-by-Step Description:
Before this use case can be initiated, the Administrator has already accessed his profile page in the HBS.
• The Administrator Selects *Edit Department*.
• The System Presents a *choice for updation* of Department details.
• The Administrator makes changes in the required feilds
○ *Name* of the department
○ *Person in charge* of the department
○ *No. of Halls* managed by the department.
○ *Contact details* of the department.
• The System *validates* the information and updates the profile for that department.
• The Administrator is redirected to his profile page.

Make System Updates Use Cases:
h)Use Case: **Manage System Details**
Diagram:

Brief Description:
> The Administrator *updates* the details of/in the system.

Initial Setp-by-Step Description:
> Before this use case can be initiated, the Administrator has already accessed his profile page in the HBS.
- The Administrator Selects *UpdateDetails*.
- The System Presents a *choice for updation* of Details.
- The Administrator *fills in the required details and submits*.
- The System *validates* the information and updates the system.
- The Administrator is redirected to his profile page.

2.2.3 **Booker Use Case**:
> The Booker has the following set of Use Cases,
> Diagram:
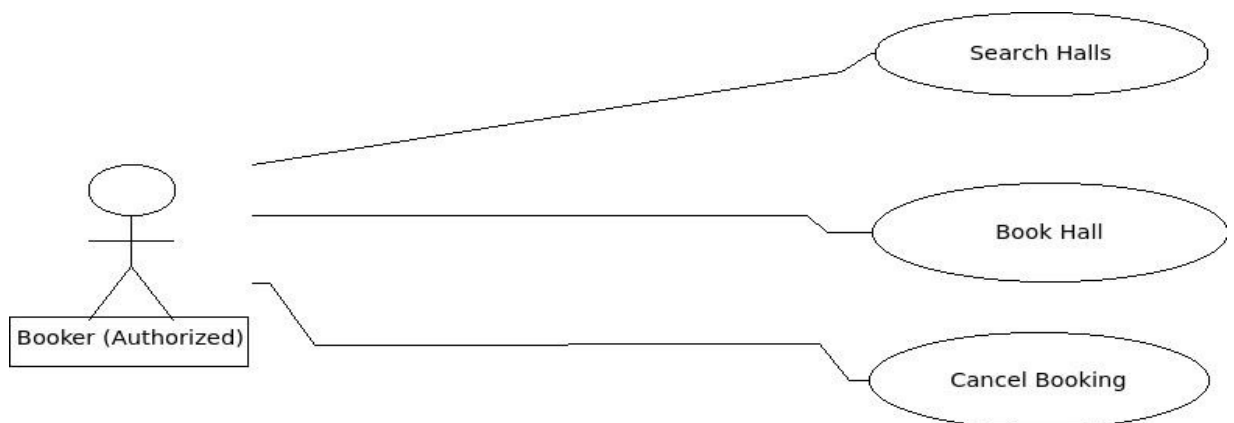
Figure 4: Use Case Booker

Search Hall Use Cases:
i)Use Case: **Requirement Specific Hall Search**
Diagram:

Booker (Authorized) — Requirement Specific Hall Search

Brief Description:
 The Booker *searches* for the hall according to appropriate requirements.

Initial Setp-by-Step Description:
 Before this use case can be initiated, the Booker has already accessed his profile page in the HBS.
• The Booker Selects to *SearchHall*.
• The System Presents a choice of requirements for the search.
○ *Seating Capacity*
○ *Time*
○ *Date*
○ Addtional features
▪ *Projector availability*
▪ *Sound System availability*
▪ *Podium availability*
▪ *AC / Non-AC*
• The Booker *fills in the requirments* and performs *search*.
• The System *computes* the appropriate result and *displays* the list to he Booker.

Book Hall Use Cases:
j)Use Case: **Fill event details**
Diagram:

Booker (Authorized) — Fill Event Details

Brief Description:
The Booker fills in the *details of the event* to be organized, for which the Hall is to be booked.

Initial Setp-by-Step Description:
Before this use case can be initiated, the Administrator has already accessed his profile page in the HBS.

- The Booker *selects* the appropriate hall for the event.
- The System Presents *form* for filling up the event details.
- The Booker fills in the required details.
  ○ *Event Name*
  ○ *Event Type*
  ○ *Duration of the event*
  ○ *Alternate Contact details*
  ○ *Additional requirements*
- The System *validates* the information and updates the system.
- The Booker is redirected to confirmation page.

k)Use Case: **Request for booking**
Diagram:

Booker (Authorized) — Request For Booking

Brief Description:
The Booker confirms the event details and *sends the request* to the Department concerned.

Initial Setp-by-Step Description:
Before this use case can be initiated, the Booker has already accessed his profile page in the HBS.

- The Booker *confirms* the filled in event details.
- The System Presents the choice for *Submission or Cancellation* of booking request.
- The System sends the *mail* to the concerned department, a *pending booking request* to the Department-InCharge profile and also send a *mail* to the Booker regarding the booking status which includes unique *booking_id* along with other booking details.
- The Booker is redirected to profile page.

Cancel Booking Use Cases:
m)Use Case: **Cancel Booking**
Diagram:

Booker (Authorized)

Cancel Booking

Brief Description:
The Booker cancels the Hall reservation.

Initial Setp-by-Step Description:
Before this use case can be initiated, the Booker has already accessed his profile page in the HBS.
- The Booker selects to *Cancel* Hall reservation.
- The System Presents the list of *reservations* made by the Booker.
- The Booker *chooses* the appropriate reservation for *cancellation*.
- The System sends a *Cancellation mail* to the Department-InCharge and also a copy of the *mail* is sent to the Booker.
- The System updates the datbases.
- The Booker is redirected to profile page.

2.3 User Characteristics:

The user is expected to be *Internet literate*. The main screen of the Website will have options to *login* as a authorized user and to *view the reservations* of Halls for a specific date and time, which have been confirmed. Once logged in, the user must be capable of *entering required fields* to search for the Hall and to perform *basic tasks*. The user should be able to *select the required option*, make *necessary decisions* and be able to fill in any details if required.

2.4 Assumption and dependencies:
- There is a requirement of Internet.
- The Hall can be booked only by Authorized users.
- Viewers are only permitted to view the upcoming events.
- Authorized users provide genuine information for booking details.

3 <u>Requirements Specification</u>

3.1 External Interface Requirement:

        Once the confirmation mail is recieved by the Booker, a Hard-Copy of the mail which includes the unique booking_id and event / booking details has to submitted to the Seargent Officer of NITC for him to keep a manual record of the events for which the hall has been booked, in his register, as there is no system connectivity in his office.

3.2 Functional Requirements:

The logical structure of the data is displayed in section 3.3.1

3.2.1 Update Student/Faculty Details:

Use case name	Update Student/Faculty Details
XRef	Section 2.2.1 a)
Trigger	Department-InCharge clicks on *UpdateStudent.*
Precondition	The user(Department-InCharge) is a *authorized* user. (login, password)
Basic Path	-The user *selects a Student/Faculty* of that particular department whose details have to be changed / updated. -All the available details of that Student/Faculty will be displayed. -Student/Faculty Name, Student/Faculty Contact Details, Student/Faculty ID number ,etc. -UserName Password -The user then edits or adds the required fields and submits.
Alternative Path	None.
Post Condition	After the submission, he system Validates the information and the entry in the database will be updated, a Success message will be displayed. The user will be directed to his/her profile page.
Exception Path	If user abandons the operation to update before pressing "Submit", then entry will not be updated in the database.
Other	The Student/Faculty List to be selected from, will be displayed only after Department-InCharge selects to Update the student.

3.2.2 Edit Department Information:

Use case name	Edit Department Information
XRef	Section 2.2.1 b)
Trigger	The Department-InCharge clicks on *EditDepartmentInformation.*

Precondition	The user(Department-InCharge) is a *authorized* user. (login, password)

Basic Path	-The user selects to Update Department details.
	-All available details of the Department is displayed.
	-*Person In Charge details* Updation
	  Contact information Updation
	  Department Location/Address updation
	  Number of Halls Manged Updation
	-User makes changes in the required fields and submits.

Alternative Path	None.
Post Condition	After the submission, the system Validates the information and the entry in the database will be updated, a Success message will be displayed. The user will be directed to his/her profile page.
Exception Path	If user abandons the operation to update before pressing "Submit", then entry will not be updated in the database.
Other	The Department details to be editted, will be displayed only after Department-InCharge selects to Update Department.

3.2.3 Manage Hall Details:

Use case name	Update Hall Details
Xref	Section 2.2.1 c)
Trigger	Department-InCharge clicks on *EditHallDetails.*
Precondition	The user(Department-InCharge) is a *authorized* user. (login, password)
Basic Path	-The user selects to Update Hall details. -Incase the user selects *UpdateHall*, all available details of the selected Hall is displayed. -User makes changes in the required fields and submits.

Alternative Path	-Incase the user selects *AddNewHall*, a form with all the details to be filled about the Hall will be displayed.
	-*No. of Seats*
	 Description of Hall
	 Availability of:
	     • *Podium*
	     • *Projector*
	     • *Sound System*
	     • *AC*(Air Conditioner) / *Non AC*
	-User fills in all the details and submits.

Post Condition	After the submission, the system Validates the information and the entry in the database will be updated, a Success message will be displayed.

	The user will be directed to his/her profile page.
Exception Path	If user abandons the operation to update before pressing "Submit", then entry will not be updated in the database.
Other	The Hall details to be editted, will be displayed only after Department-InCharge selects to Update Hall.

3.2.4 Check Status of Booking requests

Use case name	Check Status of Booking requests
Xref	Section 2.2.1 d)
Trigger	Department-InCharge clicks on *CheckStatus*.
Precondition	The user(Department-InCharge) is a *authorized* user. (login, password)
Basic Path	-A list of pending booking request is shown to the User. -User selects a particular request.
Alternative Path	None.
Post Condition	After the selection, the User gets two options : Accept / Decline
Exception Path	Incase of no pending requests, a message is displayed and the user is redirected to his profile page.
Other	Accept / Decline options will only be displayed if there is a pending request.

3.2.5 Accept / Decline booking request

Use case name	Accept / Decline booking request
XRef	Section 2.2.1 e)
Trigger	Department-InCharge selects a *pending request*.
Precondition	User selected *CheckStatus*.
Basic Path	-The user selects *Approve* request. - A confirmation mail is sent to the booker with all the booking details. -A copy of the mail is sent to the Department-InCharge. -A message is displayed.
Alternative Path	-The User selects Decline request. -A Rejection mail is sent to the Booker. -A message is displayed.
Post condition	-The system Validates the information and Database is updated and the User is redirected to his Profile page.
Exception Path	None.
Other	Accept / Decline options will only be displayed if there is a pending request.

3.2.6 Add Department

Use case name	Add Department
Xref	Section 2.2.2 f)
Trigger	Administrator click on *AddDepartment.*
Precondition	Administrator has logged in. (Authorized user)

Basic Path	-List of details to be filled up in order to add a new department is displayed.
	-Name of the Departmental
	  Person in charge
	  Number of Halls managed by the Departmental
	  Contact Details of the Departmental
	-After filling the details, the User submits the form.

Post condition	-The system Validates the information and creates a new profile for the department by updating the database. -A success message is displayed. -The User is redirected to his Profile page.
Exception Path	None.
Other	A new department is only created when it doesn't exist in the database.

3.2.7 Manage Department Account

Use case name	Manage Department Account
Xref	Section 2.2.2 g)
Trigger	Administrator click on *EditDepartmentAccountInformation.*
Precondition	Administrator has logged in. (Authorized user)
Basic Path	-A choice for updation of Department Account Information is displayed. -The Administrator makes changes in the required fields. -UserName Password -Once the changes are made, the Administrator submits the form.
Post condition	-The system Validates the information and updates the database. -A success message is displayed. -The User is redirected to his Profile page.
Exception Path	None.
Other	None.

3.2.8   Manage System Details

Use case name	Manage System Details
Xref	Section 2.2.2 h)
Trigger	Administrator click on *ChangePassword*
Precondition	Administrator has logged in. (Authorized user)
Basic Path	-The Administrator has to enter his current password. -Administrator then enters his new Password and Confirms the

	Password.
Post condition	-The system Validates the information and updates the database. -A success message is displayed. -The User is redirected to his Login page.
Exception Path	None.
Other	None.

3.2.9   Requirement Specific Hall Search

Use case name	Requirement Specific Hall Search
Xref	Section 2.2.3 i)
Trigger	The user(Booker) clicks on *SearchHall.*
Precondition	Booker has logged in. (Authorized user)

Basic Path	-The choice of requirement for the search is filled in by the user.
	-Seating Capacity
	  Date and Time of event
	  Additional features
	  • 　　　Projector availability
	  • 　　　Sound System availability
	  • 　　　Podium availability
	  • 　　　AC or Non AC
	-The User clicks on SearchHall.

Post condition	-The system Validates the information and updates the database. -A success message is displayed. -The User is redirected to his Profile page.
Exception Path	-If the User doesn't fill in any requirements needed to search, then all the Halls will be listed.
Other	None.

3.2.10   Fill Event Details

Use case name	Fill Event Details
Xref	Section 2.2.3 j)
Trigger	The User(Booker) selects appropriate Hall for the event.
Precondition	The User has logged in. (Authorized user)

Basic Path	-When the User wants to request for booking of a Hall, he/she will have to fill in the event details.
	-Event Type
	  Event Name
	  Duration of the Event
	  Alternate Contact details
	  Addtional requirements.
	-The user submits the form.

Post condition	-The system Validates the information and updates the database.

	-A success message is displayed. -The User is redirected to his Profile page.
Exception Path	None.
Other	None.

3.2.11   Request for Booking

Use case name	Request for Booking
Xref	Section 2.2.3 k)
Trigger	The User(Booker) confirms the filled in Event details.
Precondition	The user has logged in. (Authorized user)
Basic Path	-The User clicks on *SubmitBookingRequest.* -A PendingBookingRequest mail is sent to the concerned Department. -Also a copy of the mail is sent to the Booker, containing unique booking_id and other details.
Alternate Path	-The User clicks on *Cancel.* -User is redirected to his Profile page.
Post condition	-The system Validates the information and updates the database. -A success message is displayed. -The User is redirected to his Profile page.
Exception Path	None.
Other	None.

3.2.12   Booking Cancellation

Use case name	Booking Cancellation
Xref	Section 2.2.3 m)
Trigger	The User(Booker) clicks on *CancelReservation.*
Precondition	The user has logged in. (Authorized user)
Basic Path	-List of Reservations made by the Booker is displayed. -He chooses appropriate reservation for Cancellation. -A cancellation mail is sent to the Department-InCharge. -A copy of the mail is sent to the Booker.
Post condition	-The system updates the database. -A success message is displayed. -The User is redirected to his Profile page.
Exception Path	None

3.3   Non Functional Requirement

- The mails which will be sent by the system doesn't guarentee instant delivery, but depends on the underlying mailing agent and also on the traffic.
- The Department-InCharge can make changes to the details

of the Department, Students and Faculites.
- The Administrator can make changes to the details of the department, authorized users and the system.
- The detailed description about the event will be uploaded by the booker responsible for Booking the Hall.

3.4 Performance Requirements

- If a Booker applies for booking a Hall for a particular event, then the slot of that hall for that particular time, is blocked for the next 24hours or the confirmation by the department, whichever is earliest.
- If the booking is confirmed then mails will be sent to Department-InCharge and Booker with unique booking_id.

3.5 System Interfaces

This section provides a detailed description of all inputs into and outputs from the system. It also gives a description of the hardware, software and communication interfaces and provides basic prototypes of the user interface.

3.5.1 User Interfaces:

Authorized User: user can enter the Login details in the space provided in the Home page. Then the system will redirect them to their respective profile pages, where they can performe their designated tasks, like updating details, sending booking requests, searching for halls, etc.

Normal Viewer: user can go through the event calendar which will be displayed in the home page. Viewer will only be permitted the privilege of viewing the home page. If he/she wants to book a hall, then he has to be a authorized user.

3.5.2 Hardware Interfaces:

The web portal on which the HBS will be accessed from, doesn't have any designated hardware, thus it does not have any direct hardware interfaces.

3.6 Design constraints

The system will be developed using HTML5, CSS3, Java Script, PHP and a MYSQL database.

3.7 Software System Attribute

3.7.1 Reliability

Once a particular slot has been booked, no other

booking will be taken for that particular slot i.e. It will be reserved. But if the user with a higher privilege tries to book the hall for the same slot, then based on minimum time span the booking will be overwritten or ignored. The sytem provides reliable and up-to-date information about the free slots, reservation, halls, events, etc to the Users depending on the genuinity of the information.

3.7.2 Availability

The system will be available all times (24x7), but the approval of any booking requests depends on the availibility of the Department-InCharge.

3.7.3 Security

- The entire Security of the system depends on the Authenticity of the authorization provided by the Departments and the Administrator while creating user accounts who have privileges for booking / approving / cancellation of the Hall.
- A unique booking_id will be provided to the Booker, for each successful booking that is done, such that the booking_id is generated using a secured Hashing function.

3.7.4 Maintainability

System can be maintained by the Administrator easily, as he can make required changes to the system. Concerned Departments can maintain the details regarding the department and also the authorized students and faculty who are allowed to make booking on behalf of the department.

3.7.5 Portability

The entire system can be implemented in either a Linux based server(Ex:Andromeda) or a Windows based Server. While the system can be used on any machine irrespective of the underlined operating system.