Software Requirements Specification

Version 1.0
November, 2014

Resource Management System

Abhishek Tiwary
Ahmad S. Jauhar
Nisarg Shah
Shubham Jain
Shubham Jain
Varsheeth Talluri

Submitted in partial fulfillment
Of the requirements of
CS F213 Object Oriented Programming

Table of Contents

1.0. Introduction

1.1. Purpose

The purpose of this document is to present a detailed description of a Resource Management System. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli.

1.2. Scope of Project

This software system will be an End User oriented Resource Management System for scheduling of transport and lecture sessions in BITS Goa. This system will be designed to minimize the time required by the user to book either of the two, cabs or lecture halls, which would have taken inordinate amounts of time and effort if done manually . By minimizing the time taken for the user system will meet the user's needs while remaining easy to understand and use.

The system is also designed to allow the service providers to manage the system easily, while ensuring the the data of users is not compromised.

1.3. Glossary

Term	Definition
Database	Collection of all the information monitored by this system.
Admin	Instructor-in-charge for approving the room bookings. Cab dealer in case of cab booking, who keeps updating the available cabs.
User	Student who wants to book a cab or a room.

1.4. References

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.* IEEE Computer Society, 1998.

1.5. Overview of Document

The next chapter, the Overall Description section, of this document gives an overview of the functionality of the product. It describes the informal requirements and is used to establish a context for the technical requirements specification in the next chapter.

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.

Both sections of the document describe the same software product in its entirety, but are intended for different audiences and thus use different language.

2.0. Overall Description

.1 System Environment

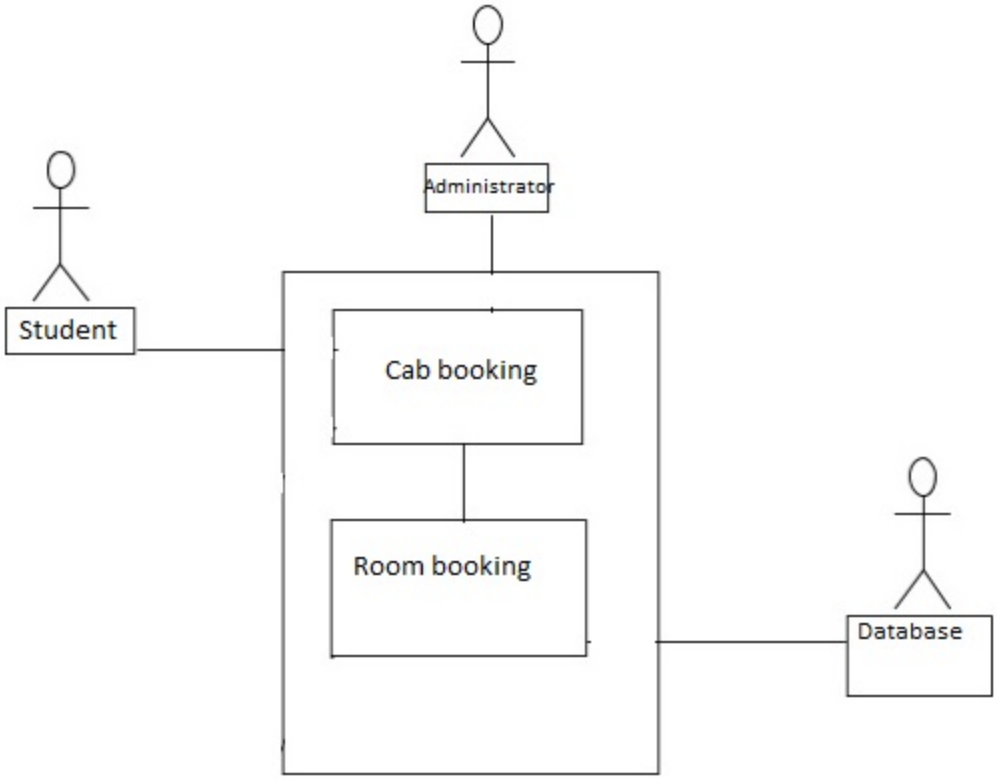


Figure 1 - System Environment

The Resource Management System has two actors and one co-operating system - student and administrator. Both the student and the administrator access the system directly. The student will have certain restrictions regarding accessing all the features of the system. The administrator will not have these restrictions.

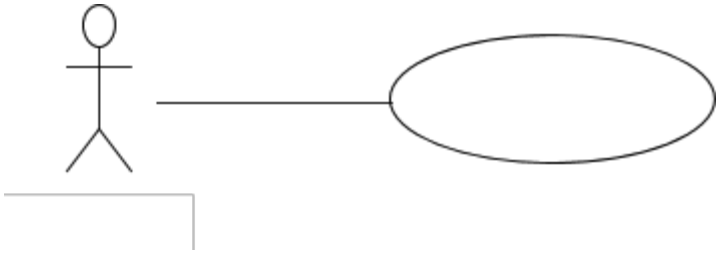
<< The division of the Resource Management System into two component parts, a room booking service and a cab booking service, is an example of using domain classes to make an explanation clearer. >>

2.2 Student Use Cases

The user has four use cases while the administrator will have two use cases and the database will have three. This section outlines each of the four use cases of student.

2.2.1 Use case: Book a room

Diagram:



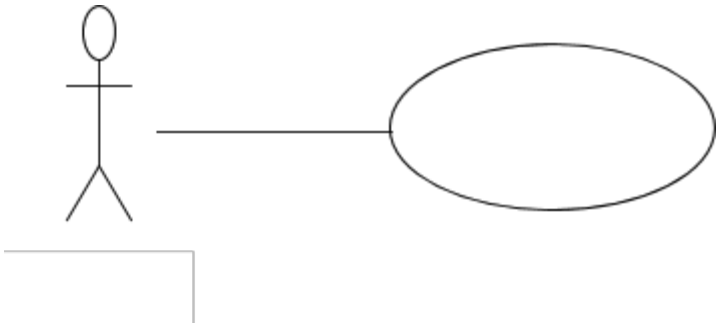
Brief Description
The user access the Room sharing system, enters his details and chooses one of the available rooms. A notification is the sent to the instructor to approve it.

Initial Step-By-Step Description
Before this use case can be initiated, the student has already authenticated to the website using his LDAP credentials.

1. The student enters the detials about the preferred room, which includes number of people, whether he needs the projector or not, etc.
2. The student chooses a room from a list of rooms offered by the system after quering the database.
3. Once chosen, a notification is sent to the instructor for approval.

2.2.2 Use case: Book a cab

Diagram:



Brief Description

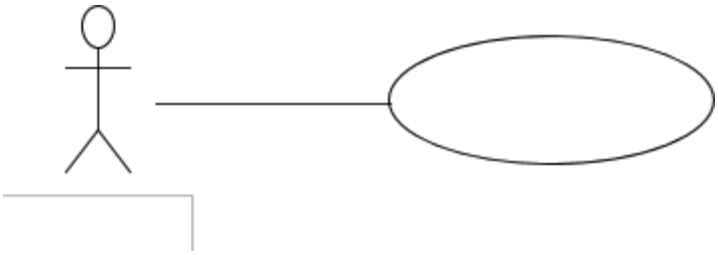
The user enters his credentials, accesses his profile and chooses one of the rooms or cabs he has already booked previously. He can then cancel any number of those bookings.

Initial Step-By-Step Description

1. User fills in a form to enter relevant details.
2. The user then decides which cab to book.
3. The user may choose to share an already booked cab.

2.2.3 Use Case: Cancel

Diagram:



Brief Description

The user enters his credentials, accesses his profile and chooses one of the rooms or cabs he has already booked previously. He can then cancel any number of those bookings.

Initial Step-By-Step Description

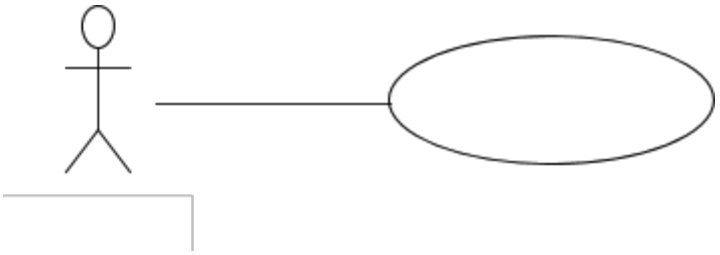
Before this use case can be initiated, the Author has already authenticated to the website using his LDAP credentials.

1. The student chooses the booking he wishes to cancel.
2. The system then sends the details of the entry to the database where the data of the booking has been saved.
3. If a match is found in the database, the respective booking is cancelled.

4. The system then displays a confirmation about the cancellation or displays that the booking reference is invalid.

2.2.4 Use Case: Check status

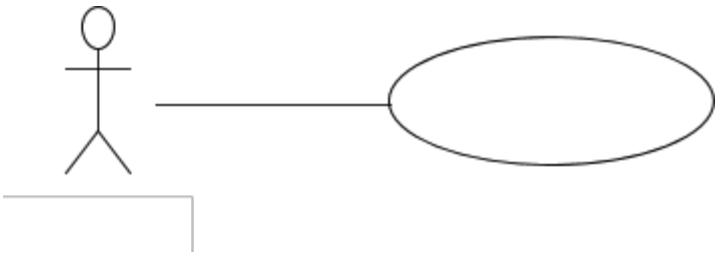
Diagram:



Brief Description
The user enters his credentials, accesses his profile and check the status of his bookings.
Initial Step-By-Step Description
Before this use case can be initiated, the Author has already authenticated to the website using his LDAP credentials.

1. The student accesses his profile.
2. The student then chooses to see the status of his bookings.
3. The system then consults with the database and finds out about the bookings done by the particular ID and displays them.

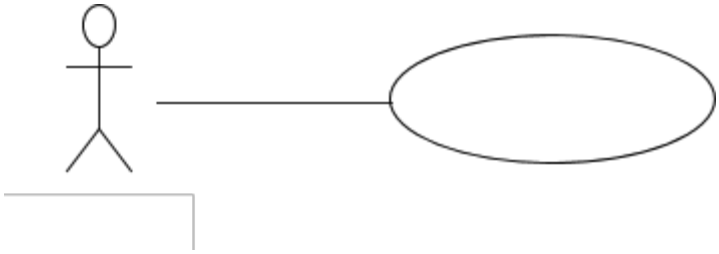
2.2.5 Use Case: Login



Brief Description
The user enters his ID and password.

2.3. Administrator Use Cases

Diagram:



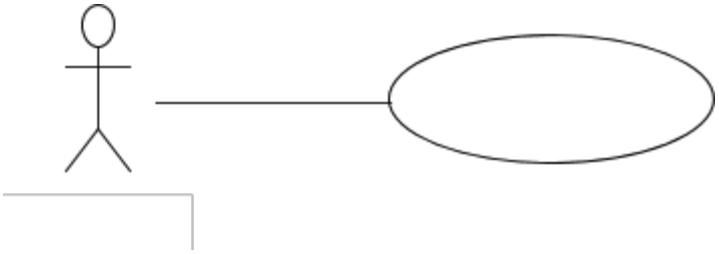
Brief Description
The administrator reviews the room/cab bookings.

Initial Step-By-Step Description

1. The administrator is provided with an interface such that he can view pending room booking requests (if he is an instructor) or check status of booked cabs (if he is a cab dealer).
2. The admin approves or rejects the room based on the purpose.
3. Has access to the database and maintains it.

2.4 Database Use Cases

Diagram:



Brief Description
The database maintains all the bookings.

Initial Step-By-Step Description

1. Whenever a new booking is created, it is stored in the corresponding table in the database depending upon whether it is a room booking or a cab booking.

2.5 User Characteristics

The user is the one who selects to book a cab or a room. The user logs in with his ID and password, which the system checks for authenticity.

The user is expected to be aware of the classrooms in BITS-Goa which are to be booked.

The user is expected to be knowing how to use java console.

3.0. Requirements Specification

3.1 External Interface Requirements

The only link to an external system is the link to the Student Welfare Division Database to verify the membership of a student or instructor. The SWD Database fields of interest to the Resource Management Systems are student's name and ID.

3.2 Detailed Non-Functional Requirements

3.2.1 Logical Structure of the Data

The logical structure of the data to be stored in the internal Article Manager database is given below.

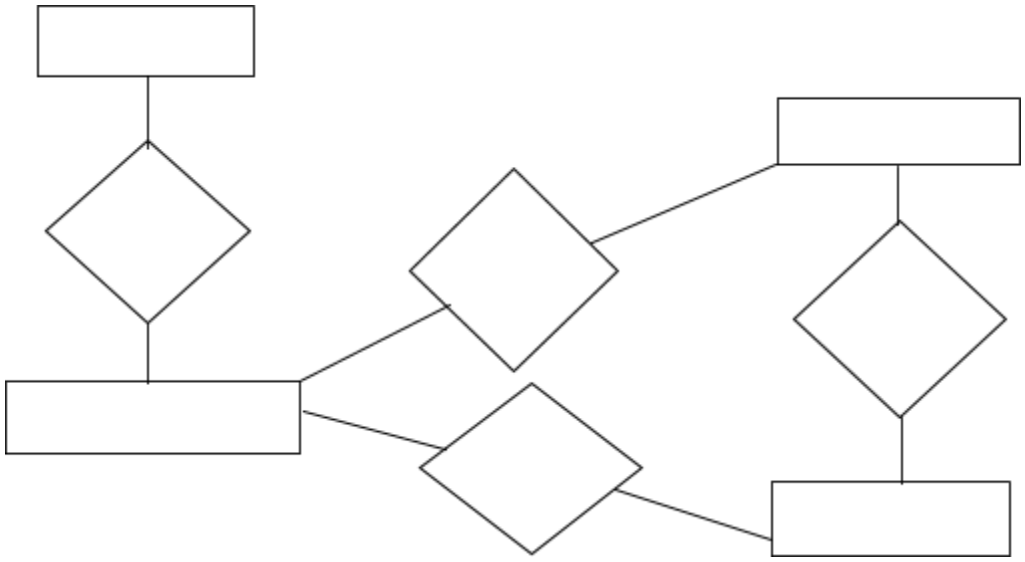


Figure 4 - Logical Structure of the Article Manager Data

The data descriptions of each of these data entities is as follows:

User Data Entity

Data Item	Type	Description	Comment
Name	Text	Name of principle author	
ID	Text	Login ID	

Room Data Entity

Data Item	Type	Description	Comment
Room	Room	Stores all details of available rooms	
History	Text	Comments on past bookings	

Cab Data Entity

Data Item	Type	Description	Comment
Cab	Cab	Stores all details of available cabs	
History	Text	Comments on past bookings	

Instructor Data Entity

Data Item	Type	Description	Comment
Name	Text	Name of Article	
ID	Text	Author entity	Name of principle author

3.2.2 Security

The database will have its own security. Only the admin will be able to look into the data by providing valid ID and password.Only the person who has booked a particular cab/room can cancel it.