

Software Requirements Specification

Version 1.7

for

Room Booking System

Prepared by

Robert McAuley	27195788	mcauley.robert@live.com
David Dubé	27525036	dav_dube@encs.concordia.ca
Harrison Andriamanantena	27435282	Harrison.a101@hotmail.com
Joshua Chan-Fee	26301959	joshuachanfee@gmail.com
Ioan Cioca	27526601	ioancioca@yahoo.ca
Ali Douch	27578253	alidouch@hotmail.com
George Kolampas	27192233	gkolampas@gmail.com
Darrel-Day Guerrero	27352409	ddayy.guerrero@hotmail.com

Instructor: Dr. Constantinides

Course: SOEN 343

Date: November 2nd 2016

Room Booking System	Version:1.7
Software Requirements Specification	Date: November 2nd 2016

Document history

Date	Version	Description	Author
October 2 nd 2016	1.1	Overall description + requirements	Robert McAuley
October 20 th 2016	1.2	Added use case model	David Dubé
October 21 st 2016	1.3	State diagrams	Ali Douch
October 25 th 2016	1.4	Domain model + contracts	Robert McAuley
October 28 th 2016	1.5	System Sequence Diagrams	Darrel-Day Guerrero
October 29 th 2016	1.6	Descriptions for system quality	David Dubé
November 2 nd 2016	1.7	Additional SSD and State Diagram + minor revisions	Ioan Cioca

Room Booking System	Version:1.7
Software Requirements Specification	Date: November 2nd 2016

Table of contents

1. Introduction	4
Purpose	4
Scope.....	4
2. Overall description	5
Product perspective.....	5
Product functions.....	5
User characteristics.....	6
Constraints	6
3. Specific requirements	7
External interfaces	7
Functionality	8
Actor goal list	8
Use case view	9
Reliability	9
Usability	9
Efficiency	10
Maintainability	10
Portability.....	10
Design constraints.....	10
4. Analysis Models	11

Room Booking System	Version:1.7
Software Requirements Specification	Date: November 2nd 2016

List of figures

Figure 1. Block Diagram.	5
Figure 2. Log in.	8
Figure 3. Calendar Monthly View.	8
Figure 4. Modify and Delete Reservations.....	8
Figure 5. Use Case Diagram.	9
Figure 6. Domain Model.	11
Figure 7. System Sequence Diagram for Logging In.....	12
Figure 8. State Diagram for Logging In.....	13
Figure 9. System Sequence Diagram for Creating a Reservation.	14
Figure 10. State Diagram for Creating a Reservation.	15

Room Booking System	Version:1.7
Software Requirements Specification	Date: November 2nd 2016

1. Introduction

This section gives an overview of the entire software requirements specification document, along with its purpose and scope.

1.1 Purpose

The purpose of the software requirements specification is to provide a description of the software system that is to be developed, which is a capstone room-booking system for ENCS students at Concordia University. The functional and non-functional requirements are described as well as the use cases indicating how users should interact with the system. This document is to be presented to customers with the aim of receiving their approval of the product.

1.2 Scope

The ENCS room-booking system is intended to provide users with an intuitive way of managing their reservations. This includes creating, modifying, and deleting reservations. Each available room has its own calendar that displays available time slots to the user.

Room Booking System	Version:1.7
Software Requirements Specification	Date: November 2nd 2016

2. Overall description

This section is intended to give an overview of the whole system. This includes the system's functions, intended stakeholders, constraints, and assumptions.

2.1 Product perspective

The system will consist of three parts: a web server, a core application, and a database server. The web server's purpose is to display the booking information to the users and allow them to create, modify, and delete reservations. The transfer of data is bidirectional from the web server to the database: a change made by the user is saved into the database, and all existing information is propagated upward from the database to the web server. Every interaction is managed by the core application, which can either reject or approve any changes made by the user.

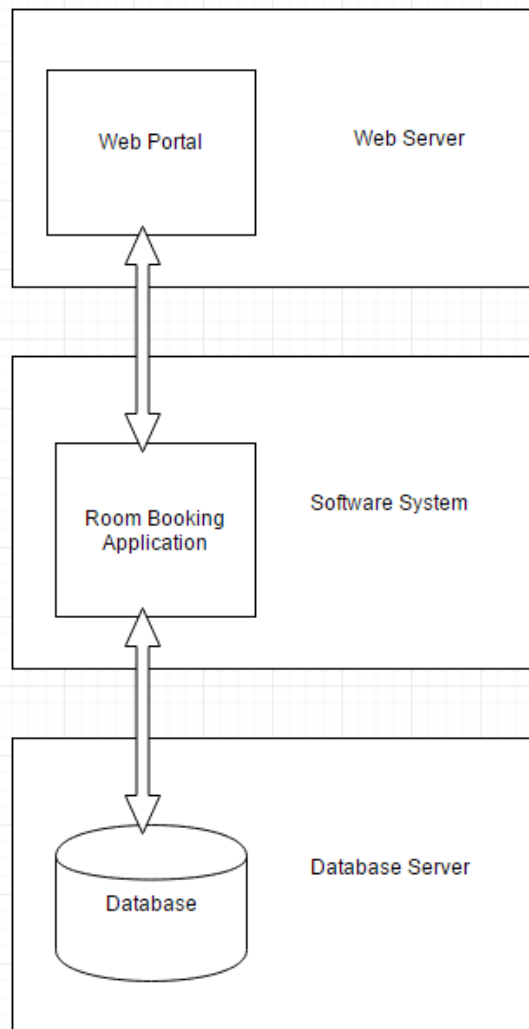


Figure 01: Block Diagram

Room Booking System	Version:1.7
Software Requirements Specification	Date: November 2nd 2016

2.2 Product functions

Through the web server, users can view, create, modify, or delete existing reservations. These actions depend on existing data to be validated. Users can only create reservations within time slots that are not occupied, users cannot modify reservations that are not theirs, and users cannot delete reservations that are not theirs. If valid input is entered, a confirmation message is displayed. If not, an error message is displayed explaining why the issue arose.

2.3 User characteristics

The room booking system is only intended to be used by ENCS students that are undergoing their capstone project. All users have the same priority level and are void of any administration privileges. No experience or technical expertise is required to operate the room booking system.

2.4 Constraints

Internet connection is required for users to interact with the web server, therefore a faulty or non-existing connection is a major constraint. The core system is constrained by the database server when several users attempt to view or modify the same data simultaneously. This constraint is not as serious when viewing information, however it can be limiting when several users attempt to create a reservation for the same time slot. In this case, one or more users would be forced to wait for the first user to finish making their reservation, thus releasing priority to the next user. Therefore, the system uses a first come, first serve implementation. There are no hardware constraints because the application does not rely on heavy CPU or memory usage. The final constraint is that the user must already be registered within the system (the user must be an ENCS student with an ENCS account).

Room Booking System	Version:1.7
Software Requirements Specification	Date: November 2nd 2016

3. Specific requirements

This section contains all requirements in detail: Functional as well as non-functional requirements (quality attributes and constraints). The quality attributes are listed per the *ISO/IEC 25010* standard that classifies software quality in a structured set of characteristics and sub-characteristics.

3.1 External interfaces

The first interface the user is presented with is a log in screen. This is where the student's ENCS information is entered to gain access to the system.

Figure 02: Log in

Once the user successfully logs in, they are presented with an overview page of all the rooms and reservations that currently exist within the system. The user may decide to simply view the information by selecting one of the rooms. This displays all the booked and available times for that specific room in a calendar format. The calendar can change between three views: daily, weekly, or monthly.

Figure 03: Calendar Monthly View

Room Booking System	Version:1.7
Software Requirements Specification	Date: November 2nd 2016

The user may choose to create a reservation by filling in the form under the calendar (Figure 03). A start date and an end date are required to submit the reservation. If an error occurs, a message is displayed to the user explaining the issue.

The user may modify or delete existing reservations. This can be done by consulting a list of current reservations made by that user and by either clicking the modify button or the delete button. A confirmation message is sent to ensure the user is aware of the changes before proceeding.

Current Reservations				
1	Room: ____ DD/MM/YYYY	Start: ____ End: ____	Modify	Delete
2	Room: ____ DD/MM/YYYY	Start: ____ End: ____	Modify	Delete
3	Room: ____ DD/MM/YYYY	Start: ____ End: ____	Modify	Delete
4	Room: ____ DD/MM/YYYY	Start: ____ End: ____	Modify	Delete

Figure 04: Modify and Delete Reservations

3.2 Functionality

Functional requirements capture the intended behaviour of the system. This section contains the *Actor Goal List* and the *Use Case view*.

3.2.1 Actor goal list

Actor	Goal
User	Log in
User	Log out
User	Create reservation
User	Modify reservation
User	Delete reservation
User	View all rooms
User	View all reservations
User	View own reservations

Room Booking System	Version:1.7
Software Requirements Specification	Date: November 2nd 2016

3.2.2 Use case view

The use case model is shown in Figure 05.

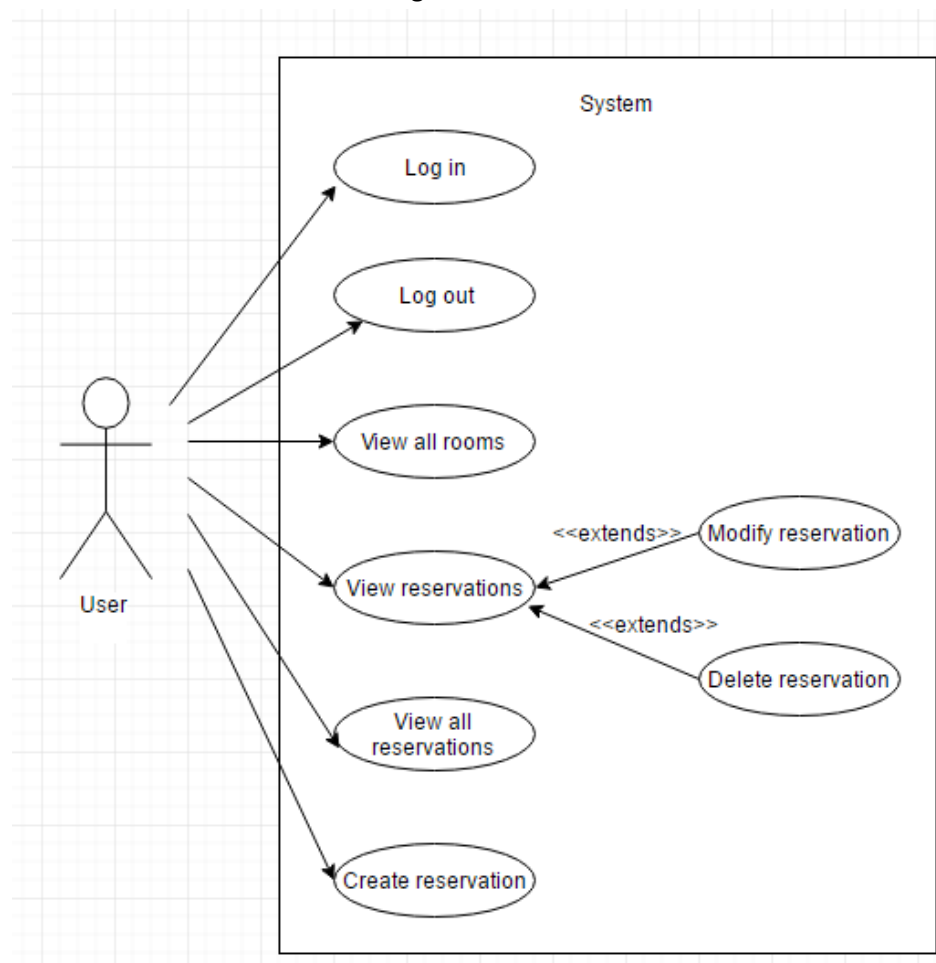


Figure 05: Use case diagram

3.3 Reliability

The system provides excellent reliability in regards to fulfilling the user goals. This includes logging in and viewing, creating, modifying, and deleting reservations.

3.4 Usability

The system has a low learning curve and the user interface is very intuitive. Therefore, the user can quickly achieve their desired objectives.

Room Booking System	Version:1.7
Software Requirements Specification	Date: November 2nd 2016

3.5 Efficiency

The system does not require much hardware to be operational. One server is needed for hosting and each operation can be completed within a few seconds.

3.6 Maintainability

The system's structure allows for it to be very adaptable when new requirements are to be implemented. The user interface is detached from the operational logic. Therefore, any changes made to the user interface would not affect the rest of the system.

3.7 Portability

The system can be operated in a multitude of environments. It can be run on laptops and desktops with a variety of operating systems (Windows, Linux, and macOS).

3.8 Design constraints

There are no design constraints that are imposed by the requirements specification. The designers are free to choose the languages, processes, architecture, and tools used.

Room Booking System	Version:1.7
Software Requirements Specification	Date: November 2nd 2016

4. Analysis Models

Below is the domain model for the room booking system (Figure 06).

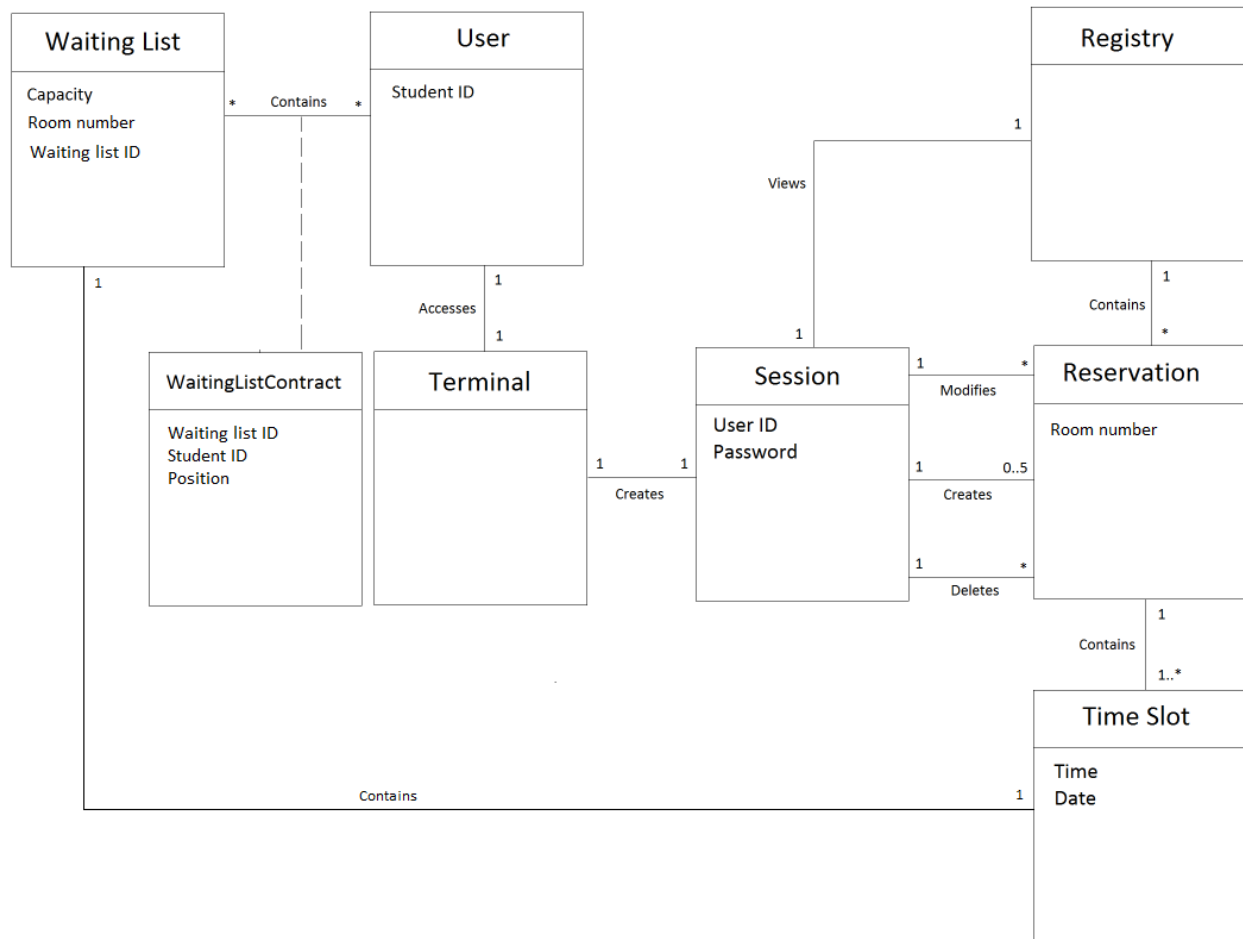


Figure 06: Domain model

Domain Class	Narrative
User	The user accesses a terminal and can be placed on several waiting lists . A user contains a student ID.
Terminal	The terminal is used by a user to create a session .
Session	A session is created with a user ID and a password . Through a session, a user may view the registry or create, modify, or delete a reservation(s) .
Registry	The registry contains a list of all room reservations and their information (date and time).
Reservation	A reservation is created, modified, and deleted by a user through a session . It contains the room and time slot information.

Room Booking System	Version:1.7
Software Requirements Specification	Date: November 2nd 2016

Time slot	Each time slot contains the date and time of the reservation.
Waiting list	A waiting list is created when a user books a reservation for a time that is already taken. It contains a capacity, room, and time slot . One user can simultaneously appear on several waiting lists, which is why a waiting list contract is needed to encapsulate this many-to-many relationship.
Waiting list contract	A waiting list contract links one user to one waiting list . It contains the IDs of both the user and the waiting list as well as the position the user appears on that waiting list.

Two critical use cases are defined in this section: **logging in** and **creating a reservation**. Four diagrams are given for each of them below:

- 1) The system sequence diagram
- 2) System operations
- 3) Operation contracts
- 4) State diagram

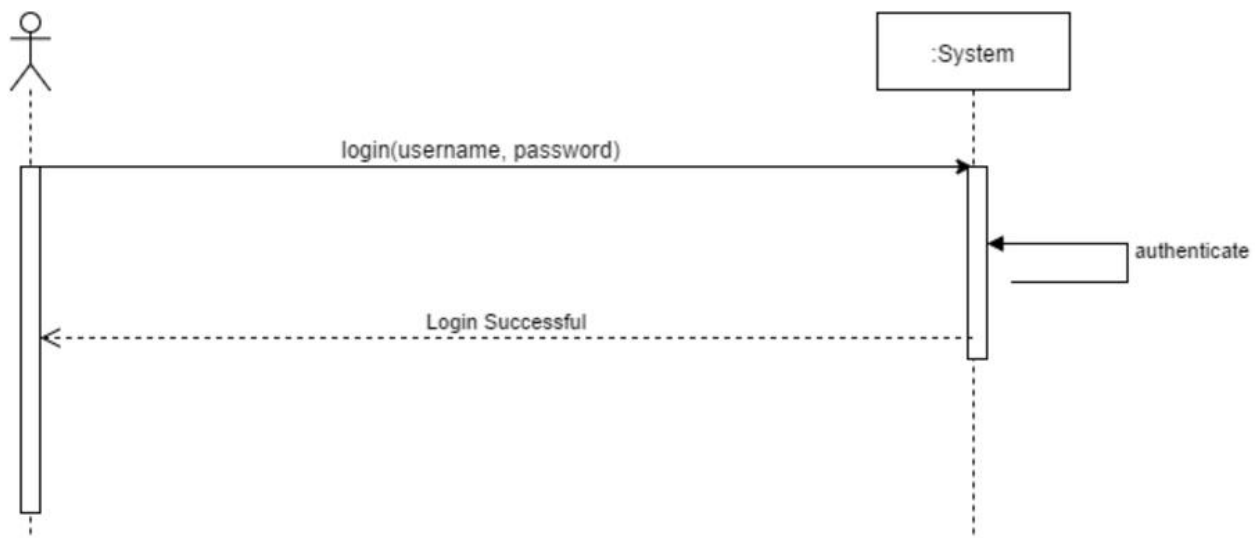


Figure 07: System sequence diagram for logging in

The user sends a message to the system with a username and password. Once the system authenticates the request, it replies with a confirmation message.

System operations for logging in

Room Booking System	Version:1.7
Software Requirements Specification	Date: November 2nd 2016

1) login(username, password)

Operation Contract CO1	login
Operation	login(username, password)
Cross Reference	Use case login
Preconditions	None
Postconditions	1) A session is instantiated 2) The session is associated with terminal 3) Session attributes are initialized

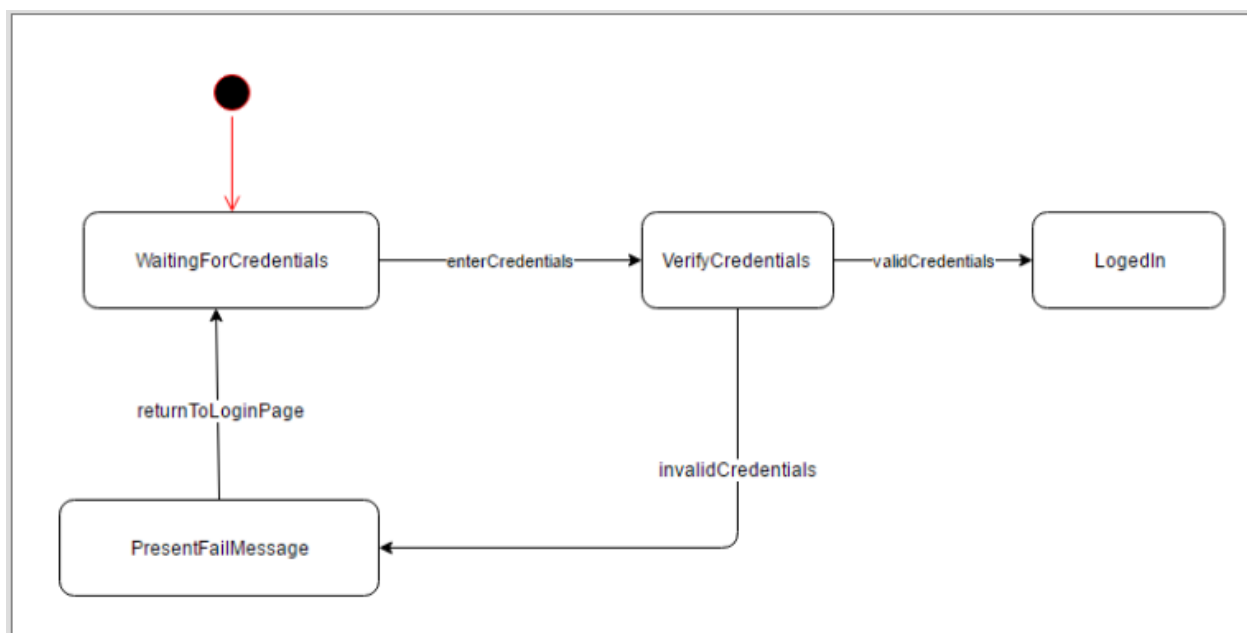


Figure 08: State diagram for logging in

There are 4 states in the logging in state diagram: waiting for credentials, verifying credentials, presentation of a failure message, and logged in. Initially, the system is in the waiting state. Once credentials are entered, it goes to the verifying state and evaluates the credentials. If they are incorrect, an error message is displayed and it goes back to waiting. If they are correct, it transitions into the logged in state which is the intended final state.

Room Booking System	Version:1.7
Software Requirements Specification	Date: November 2nd 2016

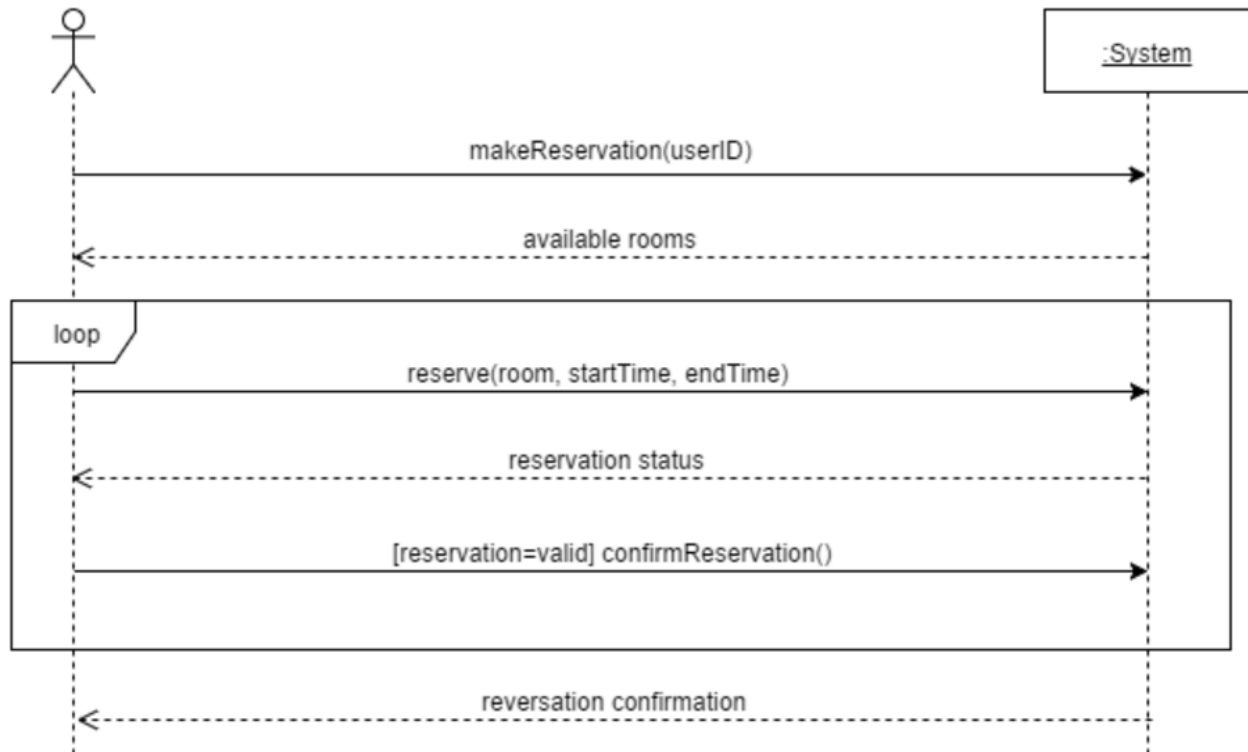


Figure 09: System sequence diagram for creating a reservation

The user sends a message to the system with their user ID and the system replies with a list of available rooms to choose from. From there, the user may send a message to reserve a room with a start and end time, and then confirm this reservation. This can be done as many times as the user wishes. Once finished, the system returns a confirmation message.

System operations for creating a reservation	
1)	makeReservation(userID)
2)	reserve(room, startTime, endTime)
3)	confirmReservation()

Operation Contract CO1	makeReservation
Operation	makeReservation(userID)
Cross Reference	Use case make reservation
Preconditions	1) A session is underway
Postconditions	1) An association between the session and the registry is formed

Room Booking System	Version:1.7
Software Requirements Specification	Date: November 2nd 2016

Operation Contract CO2	reserve
Operation	reserve(room, startTime, endTime)
Cross Reference	Use case make reservation
Preconditions	1) A session is underway
Postconditions	1) A reservation is instantiated 2) An association between session and registry is established 3) Reservation variables instantiated 4) Association between session and reservation is established

Operation Contract CO3	confirmReservation
Operation	confirmReservation()
Cross Reference	Use case make reservation
Preconditions	1) A session is underway 2) The selected reservation is valid
Postconditions	1) Association between registry and reservation is established (the reservation is now stored)

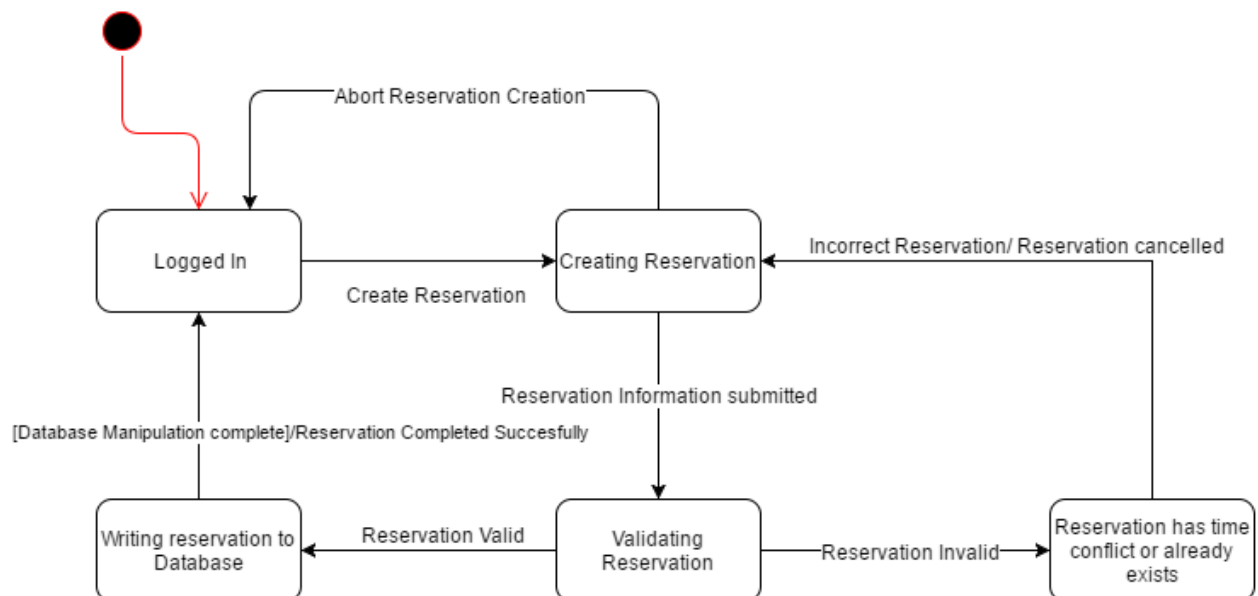


Figure 10: State diagram for creating a reservation

Room Booking System	Version:1.7
Software Requirements Specification	Date: November 2nd 2016

There are 5 states in the create reservation state diagram: logged in, create reservation, validate reservation, reservation time conflict, and write reservation to database. Initially, the system is in the logged in state. Once the appropriate information is entered, the system goes to the create reservation state. The user can either choose to abort or submit the new reservation information. At this point, the system begins validating the input. If there is a time conflict, the system transitions to the time conflict state and prompts the user for different information. If the information is valid, the system writes the new reservation to the database and returns to the logged in state where it remains on standby.