Table of Contents

[1. Introduction/Overview 2](#_Toc463855729)

[1.1. Purpose 2](#_Toc463855730)

[1.2. Scope 2](#_Toc463855731)

[1.2.1. Main Objective 2](#_Toc463855732)

[1.2.2. Specific goals 2](#_Toc463855733)

[1.3. Overview of Document 3](#_Toc463855734)

[2. Users 3](#_Toc463855735)

[2.1. Who are the users? 3](#_Toc463855736)

[2.2. Use cases and use case diagrams: 3](#_Toc463855737)

[3. System 7](#_Toc463855738)

[3.1. Functional Requirements 8](#_Toc463855739)

[3.1.1. Issues 8](#_Toc463855740)

[3.1.2. Major Subsystems or Functions 9](#_Toc463855741)

[3.1.3. Major classes identified: 9](#_Toc463855742)

[3.1.4. Minor system functions: 10](#_Toc463855743)

[3.2. User Interface Specification 10](#_Toc463855744)

[3.3. Non-functional requirements 10](#_Toc463855745)

[3.3.1. Management 10](#_Toc463855746)

[3.3.2. Technical 11](#_Toc463855747)

[3.3.3 Performance 11](#_Toc463855748)

[3.4. System Evolution/Maintenance 11](#_Toc463855749)

[4. Other Deliverables 11](#_Toc463855750)

[5. Glossary 11](#_Toc463855751)

[6. References 12](#_Toc463855752)

# 

# 1. Introduction/Overview

Online ticketing system deals with reservation of tickets different events. In this system user will be able to reserve tickets for an event through online. This will make their work easy to attend an event. They don’t have to go to the counter to reserve a seat. They can do it online through this system. They can pick up their tickets in the counter at the venue of the event.

# 1.1. Purpose

The purpose of this document is to specify the requirements of the online ticketing system and to inform the customer about the development of the intended systems such as hardware and software requirements, major users, both major and minor functions, constraints, and intended user interface.

# 1.2. Scope

# 1.2.1. Main Objective

The main purpose of this project is to help the user to search for tickets for different events and reserve the ticket for that event. This is a web based application and we are using some languages like C#, .net, HTML, CSS and Java Scripts for the front end and SQL server for the backend as well. In this system, there are three types of users: registered, guest and admin users. Registered users and guest users can reserve the ticket and cancel the ticket. The admin is also considered as a type of user since he is using this system to perform some tasks. Admin users manage the reservation system. This system will not handle payment. So the user has to collect the ticket at the counter by paying the required money. The Guests and registered users cannot choose a particular seat.

# 1.2.2. Specific goals

The specific goal is to develop an online reservation system to help users in reserving tickets through online. It also helps them in checking the availability of tickets for an event.

# 1.3. Overview of Document

The following document is arranged as follows. Section 2 describes the users of the system and shows Unified Modeling Language use case diagrams. Section 3 specifies the target and development environment and it also discusses about functional requirements, user interface specification and non-functional requirements of the system. Functional requirements deal with issues, major functions, major classes and minor system functions. Whereas, non-functional requirements include constraints such as management, technical, and performance of the system. User interface specification shows a mockup screen of the application. Other deliverables required for this project are mentioned in Section 4. The Glossary of this document is present in section 5 and the references are listed under Section 6.

# 2. Users

# 2.1. Who are the users?

The user of this application is anyone who wants to reserve tickets.

# 2.2. Use cases and use case diagrams:

The Figure 1 illustrates the use case diagram where the activities of the system are divided into use cases. The figure 1 also shows the different actors involved in the system along with their responsibilities and activities performed towards the system.

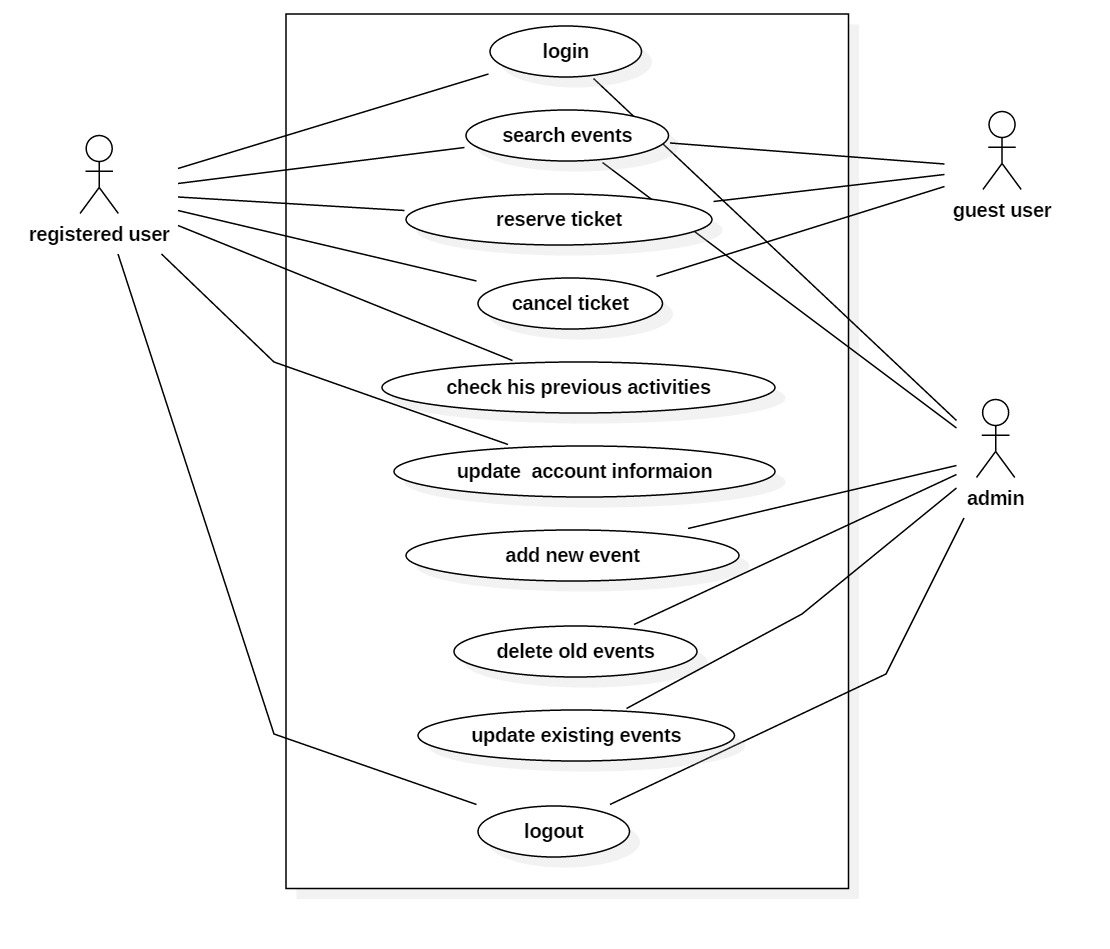
****

Figure 1: Use case diagram

The Figure 2 illustrates the class diagram of the online ticketing system. The diagram represents the classes involved in the system along with their operations that are performed.

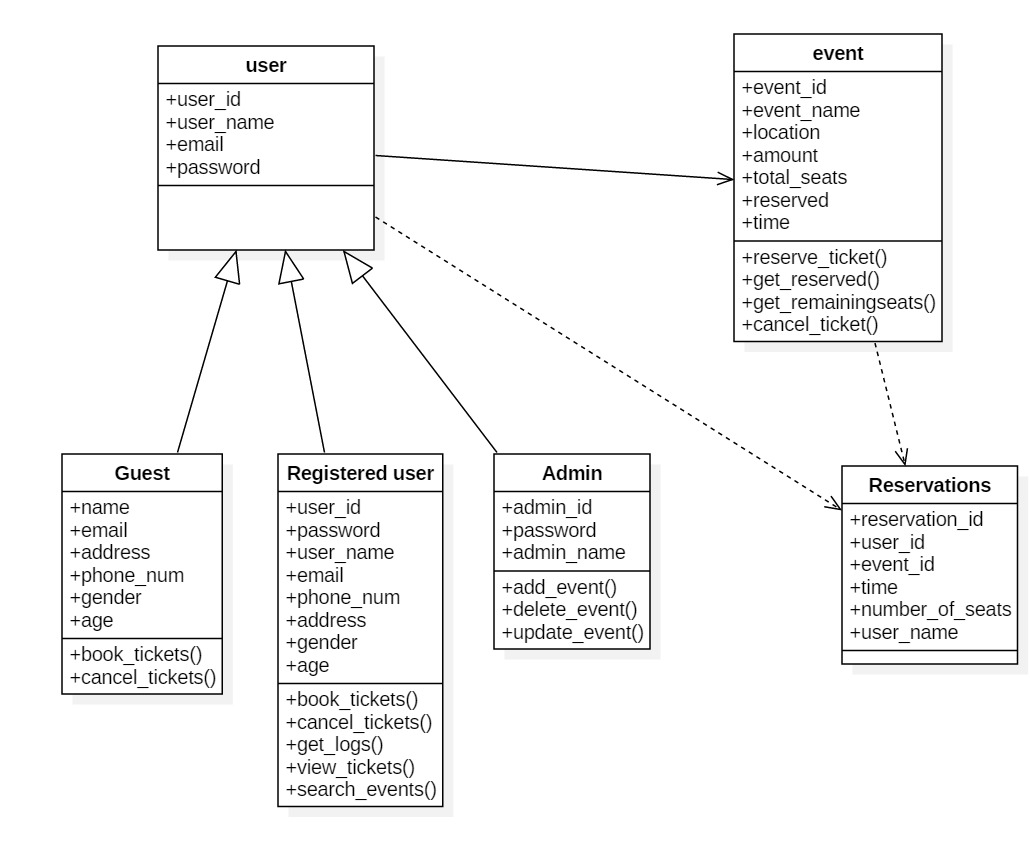
****

Figure 2: Class Diagram

The Figures 3 and 4 illustrates the sequence diagram of the online ticketing system. The diagram represents the shows how objects operate with one another and in what order.

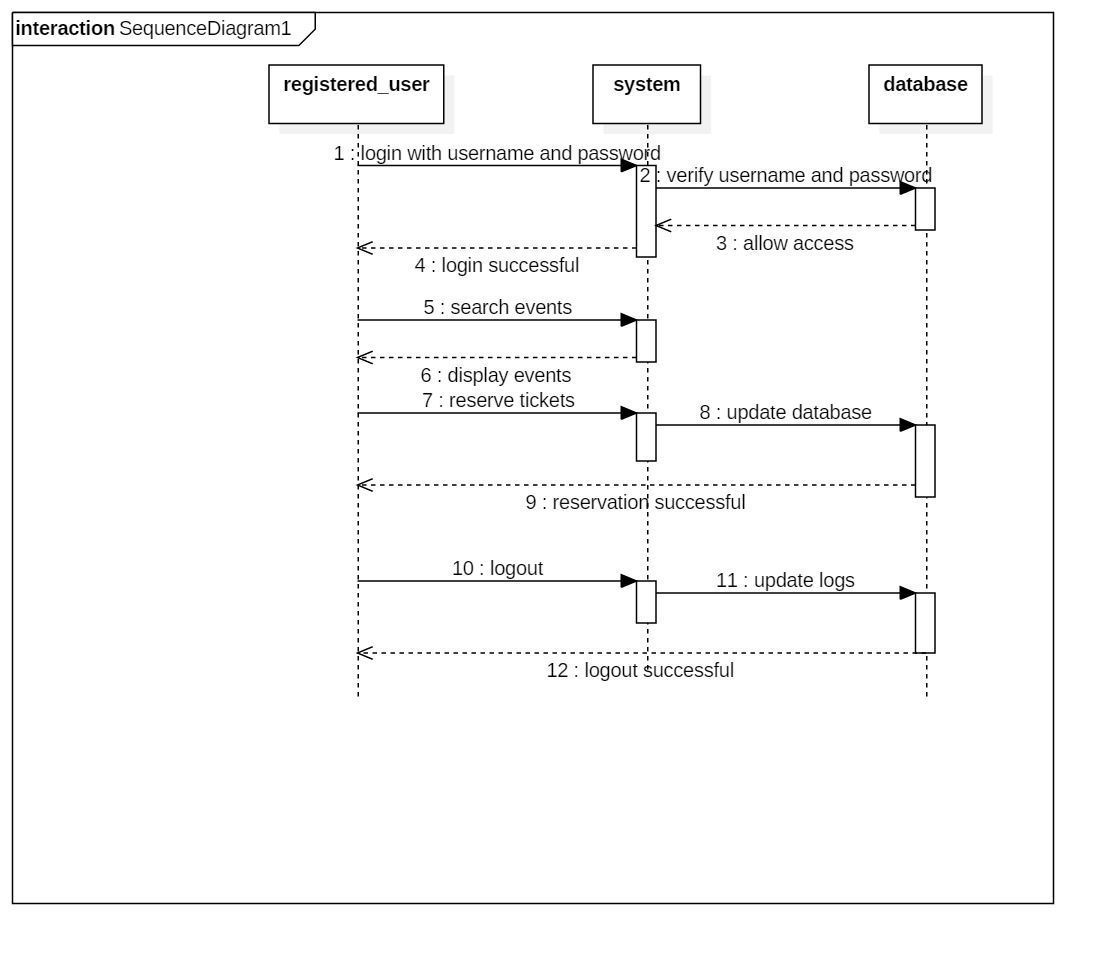
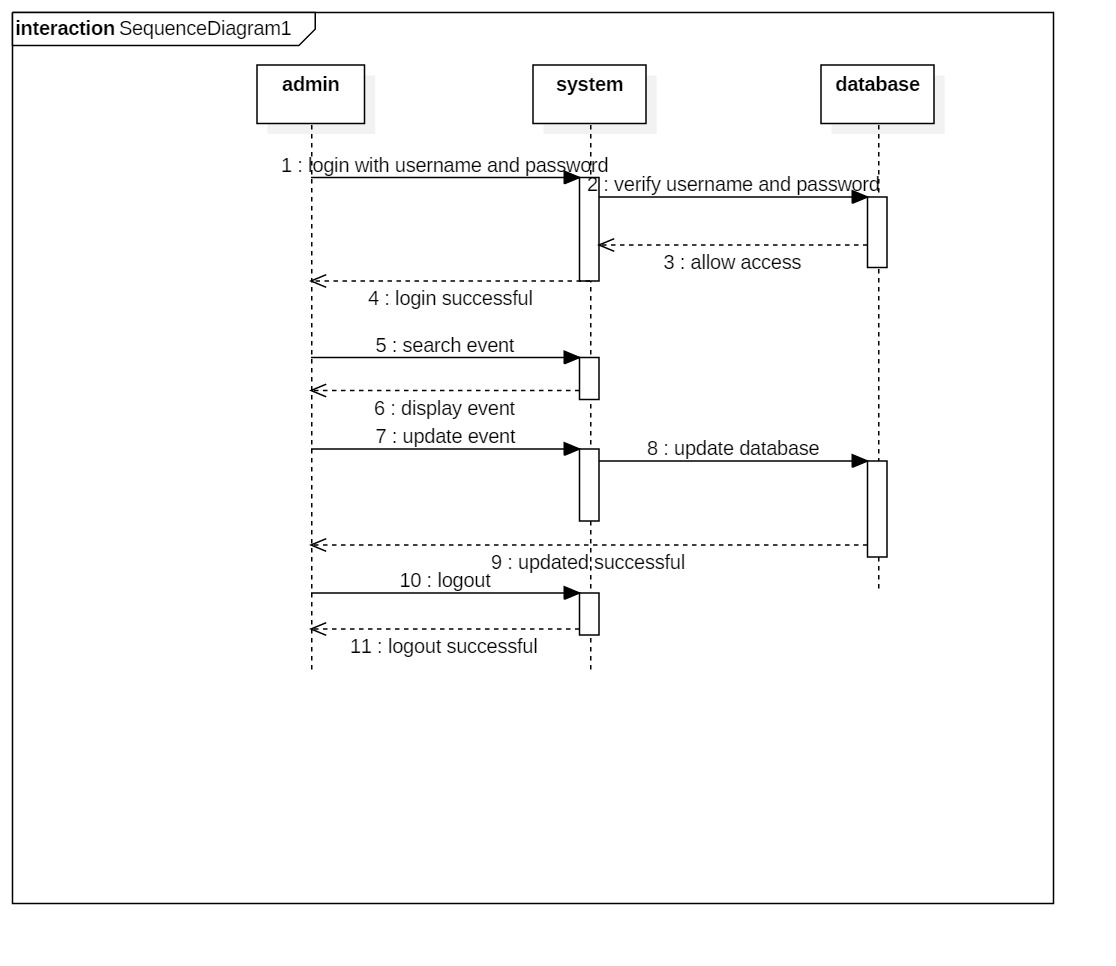


Figure 3: Sequence Diagram

Figure 4: Sequence Diagram

# 3. System

The system describes here concern the development environment will be as follows:

**Hardware** will include a personal (desktop or notebook) computer with the following specifications:

* a Pentium 4 microprocessor running at 2.0 GHz or above
* a minimum of 128 megabytes of system RAM
* a hard drive with at least 2 gigabytes of free space
* an XGA screen display 14” or larger
* a graphics card with 16 megabytes or more of RAM

**Software** will include:

* Windows 7
* Microsoft Visual Studio 2015
* SQL Server
* Microsoft Project

The customer’s environment will be as follows:

**Hardware** will include a personal (desktop or notebook) computer with the following specifications:

* a Pentium 4 microprocessor running at 2.0 GHz or above
* a minimum of 128 megabytes of system RAM
* a hard drive with at least 2 gigabytes of free space
* an XGA screen display 14” or larger
* a graphics card with 16 megabytes or more of RAM

**Software** utilized will include:

* Windows 2000 or above
* a web browser.

# 3.1. Functional Requirements

# 3.1.1. Issues

The proposed software has many requirements:

Basically there are two types of users. They are admin and user, in this user is divided Everyone has to log on to the system to use the system. There should be some login criteria for the system. When the user login, if he is a registered user he should be able to search the events for tickets and he should be able to reserve the tickets. Even the guest user should have the capability of searching the events and reserve the tickets. Both of the users should be able to cancel their tickets. Admin should login to the system to add new events, delete the old ones and edit the events. Registered users should be able to get their logs when they login.

# 3.1.2. Major Subsystems or Functions

While developing the software, Waterfall process model is used. The functionalities of the software are described as follows:

**Users:**

Users have some login criterion. Users have to register to have an account. They can be able to search events. They can be able to reserve tickets and cancel tickets. Registered users can get their log information.

**Admin:**

Admin have their credentials to login into system. They can search events, delete old events, add new events, edit existing events.

# 3.1.3. Major classes identified:

The major classes identified in the user are:

Registered users

Guest

Admin

Events

# 3.1.4. Minor system functions:

The system should enable the user to change his account details and his reservation details. Admin should be able to change the data of the event and should be able to notify any change in the events to the user.

# 3.2. User Interface Specification

The application will have a graphical user interface. We will have some page on our system like

Login screen: which is used by the users to login into system.

Registration Form: where the guest user can register to have an account.

Search events page: after logging in users can search for events using the event names or events venue or event date etc.

Reserve ticket page: after selecting the event user will be forwarded to this page to make a reservation.

Cancel ticket page: this page is for cancelling a reserved ticket by the user.

Add events page: this page is only accessible to the admin to add new events.

Edit events page: this page is only accessible to the admin to edit events.

Delete old events page: this page is only accessible to the admin to delete old events.

These are the major interfaces in our system. Depending on the requirements we will add some more.

# 3.3. Non-functional requirements

# 3.3.1. Management

Due to this being a student-developed project, there will be no cost incurred on our part. Project must be developed during the Fall semester and completed by December 2016. Periodic reviews and testing of the listed functions should be conducted to ensure that the project meets the requirements.

# 3.3.2. Technical

This system must run on web browser. The development environment is on Windows and it is to be written in Microsoft Visual Studio. To make this system highly reliable, testing will be performed several times during the development of the project. Testing will include unit testing, integrated testing, interface testing, and other related testing. The prototype should also be user-friendly, so as to impress those to whom it is demonstrated.

# 3.3.3 Performance

# 3.4. System Evolution/Maintenance

This application should be developed in a way as to make it easier for future developers to maintain or enhance the application.

# 4. Other Deliverables

At the end of the project (December 2016), a project plan, requirements specification document, test plan, test report, prototype, final report and user manual outline will be delivered to the customer.

# 5. Glossary

HTML- Hypertext Markup Language.

CSS- Cascading Style Sheets.

SQL- Structured Query Language.

RTTPS- Real-time Transaction Protocol System.

# 6. References

[1]. Stringfellow, Dr.Catherine.

[2]. Sommerville I., Software Engineering 9th edition, Delhi, Pearson Education, Ltd, 2011.

[3]. Pressman Roger S., Software Engineering – A Practitioner’s Approach 7th edition, New York, McGraw-Hill, 2010.

[4]. Fowler M.,“UML Distilled –third edition”.