Sportify

Product Management

Version 3.0

Idea: A web app that shows the sporting events around you, sending emails or reminders when there is an event that matches a user's interests.

Vision Statement

For sport enthusiasts who want to find sporting events and games around them, and keep up to date with their local sporting community. Sportify is a social web application that allows people to organize and find the sporting events around their area and to connect with people in their local sport community.

Unlike other social media platforms such as Facebook Events and Meetup, Sportify is tailored towards sporting events and, in addition to allowing users to find games to watch, it also allows users to post and find recreational pickup games to partake in.

The target customers of Sportify are any sport fans, including people that want to get back into the sports they once loved as kids, but don't have a team or league to formally participate in, as well as athletes looking for a more laid-back and social way to engage with their sport.

Profile Page

1. Zoe Kulphongpatana

1.1 Contact Info:

- Personal email: zckulphong@gmail.com
- KU email: zoekulphong@ku.edu
- Phone: 913-328-2465

1.2 Availability:

- MW: 12pm-8pm, Fri: 12pm-5pm, Sat-Sun: 12pm-4pm

1.3 Major: Computer Science

1.4 Year: Junior

1.5 Relevant Courses: EECS 368, EECS 388, EECS 448, EECS 510, EECS 563, EECS 645

1.6 Programming Languages: Python, C++, Swift, HTML/CSS, JavaScript, Ruby

1.7 **Hobbies:** Crocheting, Painting, Soccer, Cooking

2. Firangiz Ganbarli

2.1 Contact Info:

- Personal email: firangizganbarlii@gmail.com
- KU email: fganbarli@ku.edu
- Phone: 785-226-4968

2.2 Availability:

- MW: 9am-11am, 2pm-8pm, Fri: 12pm-5pm, Sat/Sun: 10am-6pm

2.3 Major: Computer Science

2.4 Year: Junior

2.5 Relevant Courses: EECS 368, EECS 388, EECS 448, EECS 510, EECS 168, EECS 268

2.6 Programming Languages: Python, JavaScript, C++, HTML/CSS

2.7 **Hobbies:** Photography, Filmmaking, Reading, Basketball

3. Raven Duong (Mai)

3.1 Contact Info:

- Personal email: tohka2801@gmail.com
- KU email: raven.d.28@ku.edu
- Phone: 785-393-1877

3.2 Availability:

- MF: 11am-onwards, TRW: 4pm-onwards, Weekends: anytime (online)

3.3 Major: Computer Science

3.4 Year: Junior

3.5 Relevant Courses: EECS 368, EECS 388, EECS 448, EECS 510, EECS 168, EECS 268, EECS 645, EECS 560

3.6 Programming Languages: Python, JavaScript, C++, HTML/CSS, C#, Haskell

3.7 Hobbies: Playing games, singing, baking, working out, pets and food

4. Huy Troung

4.1 Contact Info:

- Personal email: truonggiahuy2410bd@gmail.com
- KU email: huytroung@ku.edu
- Phone: 408-717-3708

4.2 Availability:

- MF: after 12pm, W: 12pm-2pm, after 4pm, F: after 12pm, Weekends: anytime

4.3 Major: Computer Science

4.4 Year: Junior

4.5 Relevant Courses: EECS 368, EECS 388, EECS 448, EECS 210, EECS 560, EECS 510, EECS 168, EECS 268

4.6 Programming Languages: Python, JavaScript, C++, HTML/CSS

4.7 Hobbies: singing, soccer, badminton, playing games, watching tennis

Roles and Responsibilities

1. Zoe Kulphongpatana:

1.1. Design Lead: Maintain the design and UI, take care of the creative aspect of the project

2. Firangiz Ganbarli:

2.1. Team Administrator: Manage the team meetings, maintain the meeting logs and make sure the team is up to date on every part of the project

3. Raven Duong:

3.1. Product manager: Understand the customer needs and identify the project requirements and features

4. Huy Troung:

4.1. Technical team leader: Assure technical accuracy, typesetting uniformity, check consistencies and manage collaboration of the technical work

Meeting Log

Date	Time	Description	Author
09/11/2022	1h	Decided the team's name, came up with the project idea, decided on the roles and responsibilities	Firangiz Ganbarli
09/15/2022	45min	Worked on the vision statement of our product, and discussed the details of our target audience	Firangiz Ganbarli
09/29/2022	1h	Came up with the requirements specifications, use cases and the supplementary specifications	Firangiz Ganbarli
02/10/2022	45min	Updated the requirements dividing up the work among team members	Firangiz Ganbarli
06/10/2022	30min	Proofread the requirements, created a GitHub repository for the project	Firangiz Ganbarli

Sportify
Software Requirements Specifications

Version 2.0

Revision History

Date	Version	Description	Author
02/10/2022	<1.0>	First complete draft of Software Requirements Specifications	Firangiz Ganbarli
08/10/2022	<2.0>	Proofreading, additional tweaks	Firangiz Ganbarli

Table of Contents

Software Requirements Specifications

1. Introduction

1.1 Purpose

The purpose of this document is to describe the external behavior of the application, the nonfunctional requirement specifications, design constraints and other factors necessary for a complete description of the requirements for a meetup based social media app for sport enthusiasts.

The intended audience of this document includes the developers of this app, as well as the technical assessment personnel of the organization.

1.2 Scope

The software system to be produced is called Sportify which is a social media platform for sport meetups, hosting and organizing activities that are geared towards sport enthusiasts.

Sportify will allow users who are interested in finding sport events around them search for their specific interests, sign up and subscribe to different event and sport categories. The web app will also allow people looking to host sporting events to create an event with the necessary information to publicize and get people interested in their event, and they will have the ability to delete or update their event information.

Sportify can be used by the public to sign up or create a sport specific meetup. The objective is to be a social platform for people to organize and find sporting events around their surrounding area and allow people to connect with their local sporting community. The users can be both participants or attendees of such sport meetups, or they can also partake in organizing and act as a host as well.

1.3 Overview

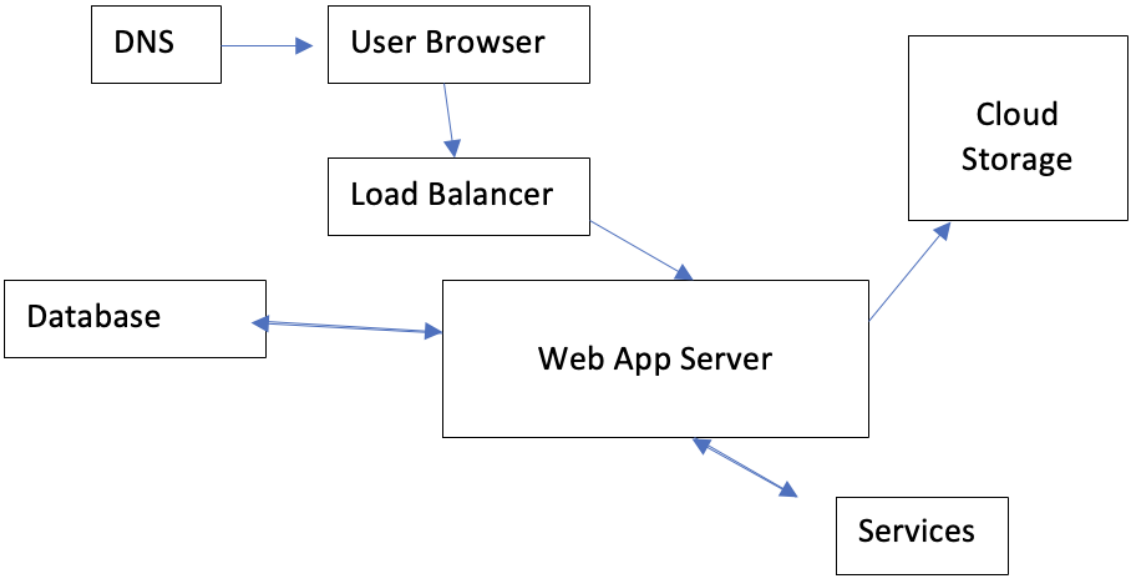
The rest of this document contains an overall description of the Sportify software system (Section 2), and the specific requirements for the system (Section 3).

2. Overall Description

2.1 Product perspective

In most of the social media platforms to find events around you, there is no clarity regarding who can sign up or attend the meetups. Additionally, the existing meetup platforms do not have a specific category or a set audience, therefore, all the information available can become daunting. For sports fans who might not have the chance to engage with their sport a lot in their daily life, making a commitment to attending an event from Meetup or Facebook Events may be too much, however, organizing or finding quick recreational games in their local community are more feasible and accessible. Sportify lets people engage with their favorite sports in any way they can, be it keeping up to date with the sport games going around them, or the small meetups in their local area.

2.1.1 System Interfaces



Sportify is a web app, so we need the web app server. DNS (Domain Name System) helps search the domain name and IP address, so the server receives the request that the user sent. Load Balancer directs the requests to one of the servers and sends an answer to the user. Web App server processes the user's request and will send a JSON file back to a browser, and it will refer to the back-end infrastructure in the database. Database will aid in organizing, adding, searching, and manipulating all the computations, especially with the user account information. And the services are required for when the web application reaches a specific level, so that servers can communicate with the services. Our web app will be stored in the cloud.

2.1.2 User Interfaces

The web app should have a user interface that is available through all the browser and will be responsive regardless of the device being used. Database and Server does not have a user interface, but will be located in the same host to communicate.

2.1.3 Hardware Interfaces

All components of this web app must be able to execute on any browser and on a personal computer.

2.1.4 Software Interfaces

We will be using SQL for our database, with Google Cloud Platform as our cloud storage platform. Out of all the DNS service providers, GoDaddy is the optimal tool for DNS.

2.1.5 Communication Interfaces

DNS communicates with the user's browser for the load balancer and that is connected to the web app server. Our database and overall services communicate with the web app server.

2.1.6 Memory Constraints

The server and database must be able to operate within 256MB.

2.1.7 Operations

The operation for logging in, creating an account, finding, and creating events must be easy and intuitive for all users. Intuitive UI/UX will enable users of any age and any background, with minimal knowledge of computers and the Internet, use the platform without needing additional help.

2.2 Product functions

The two main functions of Sportify social media platform are to allow users to create events that others can sign up for, and to allow any user to search for events around them and sign up.

Users of Sportify can have access to both the search of events and creation of them. For creating events, the organizer of the event clicks on the 'Create Event' page. This page asks for some required fields such as the event name, location, date and time, and the sport category it is in. The host's first name and profile bio will be shown to users as extra information. After all the required fields have been filled out, the organizer can click on the create event button to publicize the event. They can modify the information and delete it as needed.

For users who want to be up to date with the events around them, the home page will list a few events that are going on around them. Search page will allow them to search for a specific sport they would like to partake in, and they can filter and sort the results as wanted. If an event catches their eye, they can click on it to be redirected to Event Details page which will include all the information that the organizer publicized. In this page, they can click on sign up button to RSVP for the event.

In addition to these features, the app will need account management page, in which users can log out, create an account, modify their information, reset their password and more.

2.3 User characteristics

The users do not need extensive knowledge about web technology but should have some familiarity with websites and social media platforms. Otherwise, the app should be user intuitive.

2.4 Constraints

The system needs to enforce user authentication security and guarantee privacy of user data.

2.5 Assumptions and dependencies

No specific assumptions or dependencies.

3. Specific Requirements

3.1 Account Management

3.1.1 The user should be able to create an account

3.1.1.1 For sign-up, they would need to provide their username, email address, a password, confirmation of both of

those, and the city that they live in, and a short bio

3.1.2 The user should be able to log in to their existing account

3.1.2.1 The user can retrieve their forgotten password with a verification email.

3.1.3 The user can log out of their account

3.1.4 The user can modify their account information after login

3.2 Homepage

3.2.1 Homepage displays a list of events near the user

3.2.1.1 The list can be sorted by distance and date

3.2.1.2 The list can be filtered by sport category, distance to the user, or date

3.2.2 The user should be redirected to Event Details page on-click of the event title

3.3 Event Details

3.3.1 Event details page should display the event name, location, date and time, and category

3.3.2 The user should be able to sign up for the event by a click of a button on the event details page

3.3.3 The user can view host's bio and first name

3.4 Search Results

3.4.1 Search Results page should display the list of events that match the user's search query

3.4.1.1 This list can be sorted by distance and date

3.4.1.2 This list can be filtered by sport category, distance, and date.

3.4.2 The user should be redirected to Event Details page on-click of the event title

3.5 Event Creation

3.5.1 Event creation requires the following information: event name, category, date and time, location

3.5.1.1 A cover photo for the event

3.5.1.2 A field called "Description" for hosts to provide more info about the event

3.5.2 The event organizer can click on 'Create Event' button to share the event information publicly

3.5.3 The event organizer can see the first name and bio of the people who signed up

3.5.4 The event organizer can delete and update the information of the event they have created

4. Classification of Functional Requirements

Functionality	Type
3.1.1 The user should be able to create an account	**Essential**
3.1.1.2 For sign-up, they would need to provide their username, email address, a password, confirmation of both of those, and the city that they live in, and a short bio	**Essential**
3.1.2 The user should be able to log in to their existing account	**Essential**
3.1.2.1 The user can retrieve their forgotten password with a verification email	**Desirable**
3.1.3 The user can log out of their account	**Essential**
3.1.4 The user can modify their account information after login	**Essential**
3.2.1 Homepage displays a list of events near the user	**Essential**
3.2.1.1 The list can be sorted by distance and date	**Desirable**
3.2.1.2 The list can be filtered by sport category, distance to the user, or date	**Desirable**
3.2.2 The user should be redirected to Event Details page on-click of the event title	**Essential**
3.3.1 Event details page should display the event name, location, date and time, category	**Essential**
3.3.2 The user should be able to sign up for the event by a click of a button on the event details page	**Essential**
3.3.3 The user can view host's bio and first name	**Desirable**
3.4.1 Search Results page should display the list of events that match the user's search query	**Essential**
3.4.1.1 This list can be sorted by distance and date	**Desirable**
3.4.1.2 This list can be filtered by sport category, distance, and date	**Desirable**
3.4.2 The user should be redirected to Event Details page on-click of the event title	**Essential**
3.5.1 Event creation requires the following information: event name, category, date and time, location	**Essential**
3.5.1.1 Optional cover photo for the event	**Optional**
3.5.1.2 A field called "Description" for hosts to provide more info about the event	**Optional**

3.5.2 The event organizer can click on 'Create Event' button to share the event information publicly	**Essential**
3.5.3 The event organizer can see the first name and bio of the people who signed up	**Desirable**
3.5.4 The event organizer can delete and update the information of the event they created	**Essential**

Sportify
Supplementary Specifications

Version 2.0

Revision History

Date	Version	Description	Author
02/10/2022	1.0	Added all supplementary specifications	Zoe Kulphongpatana
08/10/2022	2.0	Tweaks, proofreading	Zoe Kulphongpatana

Table of Contents

Supplementary Specifications

1. Introduction

1.1 Purpose

Supplementary specifications detail all requirements that are not previously defined in the use case model. These such requirements may include legal standards, quality aspects, reliability, supportability, and execution criteria of the system.

1.2 Scope

This document details the supplementary specifications of Sportify.

1.3 Definitions, Acronyms, and Abbreviations

None

1.4 References

UPEDU: http://www.upedu.org/

1.5 Overview

This document contains the supplementary specifications as they relate to a variety of metrics, such as usability, reliability, performance, supportability, security, and other metrics.

2. Assumptions and Dependencies

This application will be created using everyday student laptops, IDEs, and programming languages available to the general public. This application will be used and ran on any web browsers and using internet that is easily accessible.

3. Usability

It is assumed that target users of this web application are familiar with using web browsers and the internet. This application is designed to be intuitive and user-friendly, and users should not require specific training to use this application.

3.1 The user will be able to use this application using a web browser of their choosing

The user will be able to choose a web browser that they are comfortable and familiar with to run this web application. The application will function similarly for all users regardless of the web browser being used.

4. Reliability

4.1 This application will be available 99.5% of the time

This application will be available at nearly all times of the day, and will only be made unavailable when maintenance is being performed

4.2 This application will have a MTBF of 1 month

This application have a mean time between failure of 1 month.

4.3 This application will have an MTTR of 1 hour

This application will have a mean time to repair of 1 hour.

4.4 This application will be 99.5% accurate

This application will almost completely accurate, and all errors will be addressed and corrected as they are made known.

4.5 This application will have a maximum bug rate of 1 per 1,000 lines of code

This application will be have no more than 1 but rate per every 1,000 lines of code, and all bugs will be addressed and corrected as they are made known.

5. Performance

The application will be able to perform in a manner that is comparable to other similar web applications.

5.1 The application will perform at a high speed

This application will run at a speed comparable with other similar web applications without any noticeable delay or lag time.

5.2 The application will be scalable

This application will perform similarly as more users access the web application and add their data to its database.

6. Supportability

This application will not need to be downloaded or installed and will be able to run and perform similarly on any web browser.

7. Security

The data of users of this web application will be private and will not be available for access by others using the web application.

8. Online User Documentation and Help System Requirements

The application be intuitive and user-friendly and will thus not require extensive training, documentation, help systems, or help about notices. All functionalities of the web application will be clearly stated and written in concise language.

9. Interfaces

9.1 User Interfaces

User interfaces will be intuitive, simple, and responsive to the web page environment. Users will be able to navigate the application using a mouse, trackpad, or keyboard.

9.2 Hardware Interfaces

There will be no applicable hardware interfaces.

9.3 Software Interfaces

This software application will be self-contained and will not substantially interact with any software beyond the scope of the application itself.

9.4 Communications Interfaces

This web application will communicate with networks such as the internet in order to share data information regarding events between users of the app.

Sportify
Use-Case Specifications

Version 2.0

Revision History

Date	Version	Description	Author
02/10/2022	1.0	Added all use case specifications	Zoe Kulphongpatana
08/10/2022	2.0	Tweaked and formatted	Firangiz Ganbarli

Table of Contents

Use-Case Specifications

1. Use-Case Model

1.1 Introduction

Users of the Sportify web application will be able to view, interact with, and create events. A user accesses the application by providing their username and password, which are previously defined when the user creates an account. Users will access the application using any web browser with access to the internet. Users will have options to view details about events, sort events, search events, sign up for events, as well as create, edit, and delete their own events. All data will be stored in a database.

1.2 General Actors Descriptions

1.2.1 Customer

Interacts with this web application by reading and signing up for events in the database. Has a personal username and password to access the system.

1.2.2 Host

Interact with this web application by creating, reading, editing, and deleting events from the database. Has a personal username and password to access the system.

1.3 Use-Case Model Hierarchy

1.3.1 Mngmt - Customer

- **Description**
  This package contains all management functionalities that a customer can exert.
- **Use Cases**
  - Login
  - Logout
  - Create an account
  - View an event
  - Sign up for an event
  - Search for events
- **Actors**
  - Customer – interacts with the web application by reading and signing up for events
- **Relationships**
  - None
- **Packages Owned**
  - None

1.3.1.1 Packages Diagram

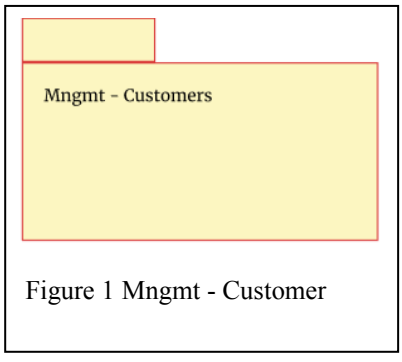


Figure 1 Mngmt - Customer

1.3.2 Mngmt - Host

- **Description**
  This package contains all management functionalities that a host can exert.
- **Use Cases**
  - Login
  - Logout
  - Create an account
  - Create an event
  - Edit an event
  - Delete an event
  - View an event
  - Sign up for an event
  - Search for events
- **Actors**
  - Host – interacts with the web application by creating, editing, and deleting events
- **Relationships**
  - None
- **Packages Owned**
  - None

1.3.2.1 Packages Diagram

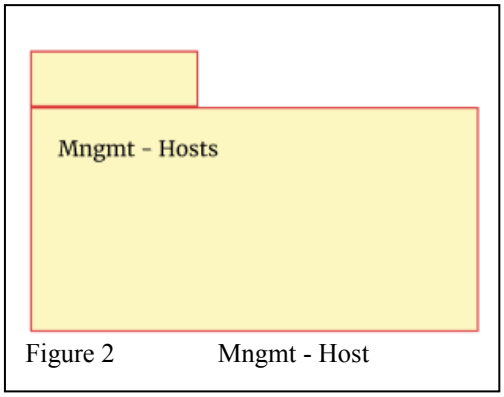


Figure 2 Mngmt - Host

1.4 Diagrams of the Use-Case Model

Diagrams of the use case model are presented in each section of the use cases below.

2. Login

2.1 Brief Description

Allowing all user types to access the web application if a valid username and password is entered.

2.2 Flow of Events

2.2.1 Basic Flow

The user will provide their username and password as strings to the login interface which will begin the login process. These entries will be validated via a database query and the results will be returned to the user.

2.2.2 Alternative Flows

None

2.3 Special Requirements

None

2.4 Preconditions

2.4.1 User profile is predefined

The user must have a predefined username and password.

2.5 Postconditions

2.5.1 User is logged in

The user is now logged in to their account and able to interact with their data accordingly.

2.6 Extension Points

2.6.1 Validate Query – include

The application validates the entered username and password.

2.7 Relationships

None

2.8 Use-Case Diagrams

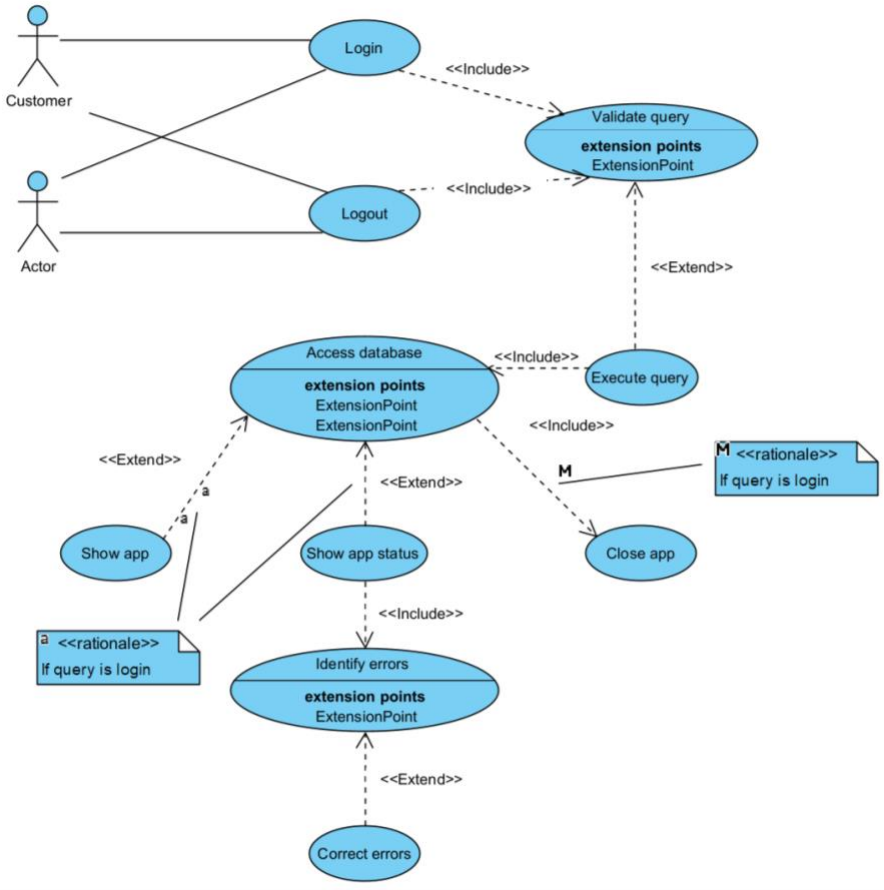


Figure 3 Login/Logout

2.9 Other Diagrams

None

3. Logout

3.1 Brief Description

Allowing all user types to logout of the application

3.2 Flow of Events

3.2.1 Basic Flow

The user will select to logout of the web application and they will be redirected to a page to reenter their username and password in order to log in.

3.2.2 Alternative Flows

None

3.3 Special Requirements

None

3.4 Preconditions

3.4.1 User profile is logged in

The user must be logged into the application.

3.5 Postconditions

3.5.1 User is logged out

The user will no longer be able to access their account until they reenter their username and password.

3.6 Extension Points

None

3.7 Relationships

None

3.8 Use-Case Diagrams

Refer to Figure 3.

3.9 Other Diagrams

None

4. Create an account

4.1 Brief Description

Users of all types will be able to create an account to access the web application.

4.2 Flow of Events

4.2.1 Basic Flow

The user will provide a username and password to the create account interface and their data will be added to the database accordingly.

4.2.2 Alternative Flows

None

4.3 Special Requirements

None

4.4 Preconditions

None

4.5 Postconditions

4.5.1 Account is created

The user's account is now created and can be accessed by the user via the login interface.

4.6 Extension Points

None.

4.7 Relationships

None

4.8 Use-Case Diagrams

Refer to Figure 3.

4.9 Other Diagrams

None

5. Create an event

5.1 Brief Description

Hosts will be able to create an event with specifications related to the event name, location, date and time, and the category of the event.

5.2 Flow of Events

5.2.1 Basic Flow

The user will provide an event name, location, date and time, and the category of the event they with to create. This event will be added to the database.

5.2.2 Alternative Flows

None

5.3 Special Requirements

None

5.4 Preconditions

5.4.1 User is logged in

The user must have already logged in with their username and password.

5.5 Postconditions

5.5.1 Event is created

The event will be added to the database and will now appear on the home pages of other users and will be searchable.

5.6 Extension Points

None.

5.7 Relationships

None

5.8 Use-Case Diagrams

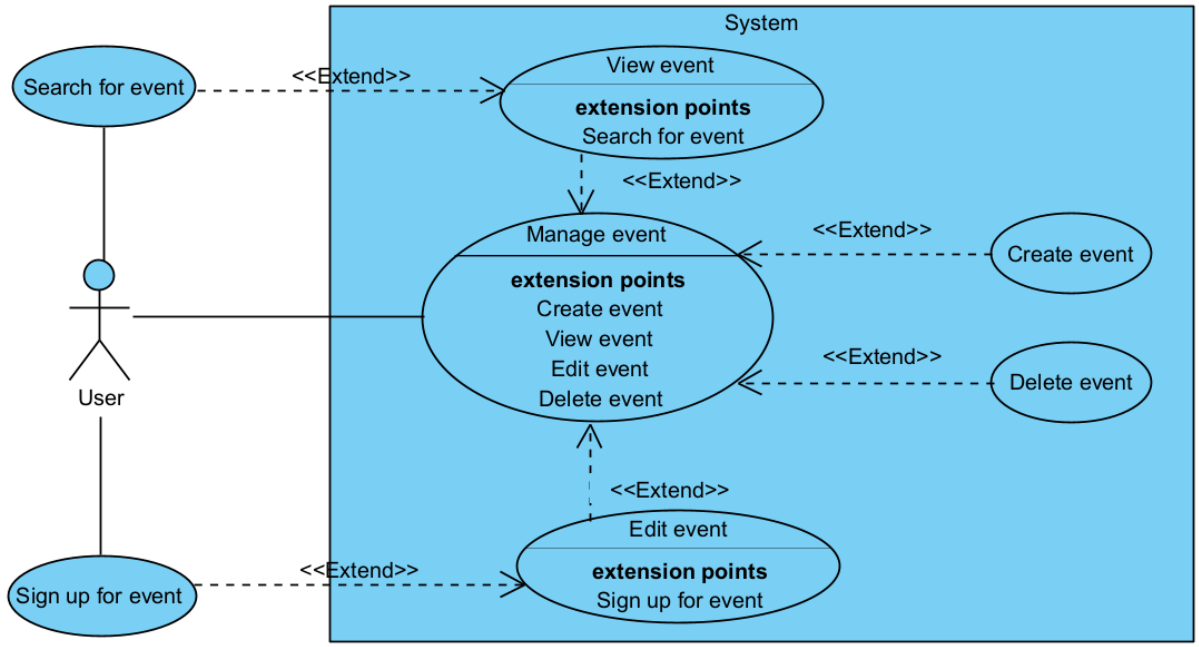


Figure 4 View event/Create event/Delete event/Edit event/Search for event/Sign up for event

5.9 Other Diagrams

None

6. Edit an event

6.1 Brief Description

Hosts will be able to edit the details of events, including the name of the event, location, date and time, and category of the event, that they have created.

6.2 Flow of Events

6.2.1 Basic Flow

The user will access an event that they have created and will be able to edit the specifications of the event, including the name of the event, location, time and date, and category via an interface. Entries will then be updated in the database.

6.2.2 Alternative Flows

None

6.3 Special Requirements

None

6.4 Preconditions

6.4.1 User is logged in

The user must have already logged in with their username and password.

6.5 Postconditions

6.5.1 Event is edited

The specifications of the event will be updated in the database according to the user's changes.

6.6 Extension Points

None.

6.7 Relationships

None

6.8 Use-Case Diagrams

Refer to Figure 4.

6.9 Other Diagrams

None

7. Delete an event

7.1 Brief Description

Hosts will be able to delete events that they have created from the database.

7.2 Flow of Events

7.2.1 Basic Flow

The user will access an event that they have created and will be able delete the event. The event will be removed from the database and will no longer be displayed to any users and will no longer appear in search queries made by any users.

7.2.2 Alternative Flows

None

7.3 Special Requirements

None

7.4 Preconditions

7.4.1 User is logged in

The user must have already logged in with their username and password.

7.5 Postconditions

7.5.1 Event is deleted

The event will be deleted from the database and will no longer be accessible by any users.

7.6 Extension Points

None.

7.7 Relationships

None

7.8 Use-Case Diagrams

Refer to Figure 4.

7.9 Other Diagrams

None

8. View an event

8.1 Brief Description

All users will be able to view the details of events, including the name, location, date and time, and category of events that are in the database.

8.2 Flow of Events

8.2.1 Basic Flow

The user select an event from their home page to view, they will then be directed to a page which shows the details, including the event name, location, date and time, and category of the event.

8.2.2 Alternative Flows

None

8.3 Special Requirements

None

8.4 Preconditions

8.4.1 User is logged in

The user must have already logged in with their username and password.

8.5 Postconditions

None

8.6 Extension Points

None.

8.7 Relationships

None

8.8 Use-Case Diagrams

Refer to Figure 4.

8.9 Other Diagrams

None

9. Signup for an event

9.1 Brief Description

Users will be able to signup to events that have been created by other hosts. The user will then be added to the guest list of the event in the database and will receive the information of the host of that specific event.

9.2 Flow of Events

9.2.1 Basic Flow

The user will select an event from the home page or from search results. They will select the “Sign Up for Event” button, which will add the user to the guest list of the event in the database. The user will then receive the information of the host who has created the event.

9.2.2 Alternative Flows

None

9.3 Special Requirements

None

9.4 Preconditions

9.4.1 User is logged in

The user must have already logged in with their username and password.

9.5 Postconditions

9.5.1 User is signed up for event

The user will be added to the guest list of the event.

9.6 Extension Points

None.

9.7 Relationships

None

9.8 Use-Case Diagrams

Refer to Figure 4.

9.9 Other Diagrams

None

10. Search for events

10.1 Brief Description

All users will be able to perform search queries to find events by their details, including the name of the event, location, date and time, and the category of the event.

10.2 Flow of Events

10.2.1 Basic Flow

The user will select the search bar within the web application and will type their search query into the search bar. The results of the query will be displayed to the user.

10.2.2 Alternative Flows

None

10.3 Special Requirements

None

10.4 Preconditions

10.4.1 User is logged in

The user must have already logged in with their username and password.

10.5 Postconditions

None

10.6 Extension Points

None.

10.7 Relationships

None

10.8 Use-Case Diagrams

Refer to Figure 4.

10.9 Other Diagrams

None