# 

**Software Requirements Specifications**

**Finder Matchmaking Web Application**

# \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Version 2.0 approved**

**Authors**

Trần Thái Dương

Đỗ Đức Cường

**Members**

Trần Thái Dương

Đỗ Đức Cường

Đỗ Đình Dũng

Nguyễn Thành Huy

Nguyễn Hữu Phước

**Table of content**

[\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 0](#_heading=)

[1. Introduction 1](#_heading=)

[1.1 Purpose 1](#_heading=)

[1.2 Scope 1](#_heading=)

[1.3 References 1](#_heading=)

[1.4 Overview 1](#_heading=)

[2. Overall description 1](#_heading=)

[2.1 Product perspective 1](#_heading=)

[2.1.2 User interfaces 2](#_heading=h.m0e1s7wewfnw)

[2.2 Product functions 2](#_heading=)

[**3. External Interface Requirements 3**](#_heading=h.fpcq934z7kxp)

[3.1 User Interfaces: 3](#_heading=h.ba0grxi1kylc)

[3.1.1 End Users: 3](#_heading=h.drhzfwroza0y)

[3.1.2 Administrator: 3](#_heading=h.m7lildw31f2o)

[4. System features 4](#_heading=)

[4.1 Functional requirements 4](#_heading=)

[4.1.1 Login 4](#_heading=)

[4.1.2 Change password 5](#_heading=)

[4.1.3 Register 6](#_heading=)

[4.1.4 Update user information 7](#_heading=)

[4.1.5 Searching for event 8](#_heading=)

[4.1.6 Viewing event detail 9](#_heading=)

[4.1.7 Viewing other user’s description 10](#_heading=)

[4.1.8 Joining events 10](#_heading=)

[4.1.9 Hosting events 11](#_heading=)

[4.1.10 Events management 12](#_heading=)

[**5. Other Non-functional Requirements 13**](#_heading=h.vw830i93dxs0)

[5.1 Performance requirements 13](#_heading=h.yombydm3t8fr)

[5.2 Design Constraints 14](#_heading=h.utjz6xux9k5x)

[5.3 Logical database requirements 14](#_heading=h.g3uscbtxs3oq)

[5.3.1 User collection 14](#_heading=h.4nx2qxli2poq)

[5.4.2 Profile collection 14](#_heading=h.9adaib93s39p)

[5.4.3 Event collection 14](#_heading=h.83gvjn16skfb)

[5.5 Software system attributes 15](#_heading=h.vvvhb1d8urae)

[5.5.1 Reliability 15](#_heading=h.xc82wwz965hh)

[5.5.2 Availability 15](#_heading=h.c7u7wc2887cx)

[5.5.3 Security 15](#_heading=h.t2osgjh0610v)

[5.5.3.1 Secured communication 15](#_heading=h.vwin2517w539)

[5.5.3.2 Controlled access rights 16](#_heading=h.lqe3refv3xq2)

[5.5.3.3 Logging user activities 16](#_heading=h.alubuc7sc58n)

**Revision History**

| **Name** | **Date** | **Reason for Changes** | **Version** |
| --- | --- | --- | --- |
| Assignment 1 | 28/3/2024 |  | 1.0 |
| Assignment 2 | 5/5/2024 | Update new format of SRS-Document | 2.0 |

# 1. Introduction

## 1.1 Purpose

The purpose of this documentation is to present a detailed description of the Matchmaking Website. It will elaborate the purpose and features of the system, interfaces, what the system will do, how the system will react to external stimuli, and the constraints under which it operates. This document is intended for the stakeholders and the developers.

## 1.2 Scope

This software system, Matchmaking Website, will be used by the public as a medium for socialization. The software will be designed to let users search and join small group activities based on their preference.

More specifically, the system is designed to let users find a group of people to do an activity with via the Internet. The system allows users to either host or join an activity session. Users who host activity sessions can set up a schedule with location and activity. Meanchile, users who seek to join an activity session can search for activities or have the activities recommended to them based on location and matching schedules.

## 1.3 References

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

## 1.4 Overview

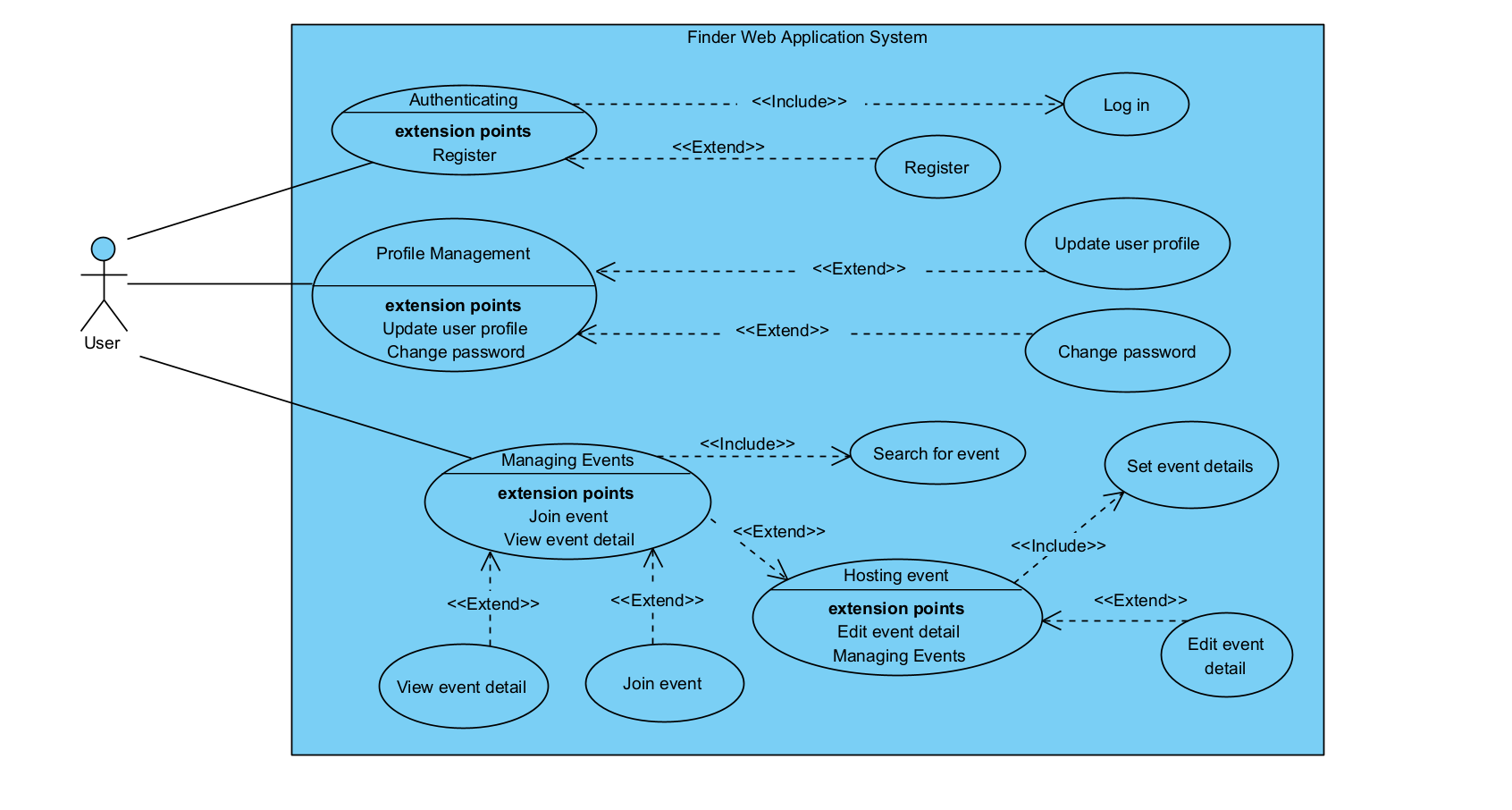
The first section will be an overview of the software, what it needs to do, and the glossary to explain terms used in this document. The section section will give an overview of how the system works and its basic functionalities. Finally the document will go into specific requirements of the software, which include its use cases, constraints, database logics, and other system attributes.

# 2. Overall description

## 2.1 Product perspective

2.1 System environment

The actor that will be interacting with this system is the User. The following diagram will show how the User interacts with the system.



### 2.1.2 User interfaces

The user will access the main functionality of the website using a horizontal navigation bar at the top of the web page. The navigation bar will be the same for all web pages of the software. All components of the user interface should conform to a design system. The user interface should be easy enough for an average new user to know how to use after 15 minutes on the website.

## 2.2 Product Functions

The Matchmaking Website software will have the following functionalities:

1. Authentication: Users can login/register into their account.

2. Changing password: Users can change their password.

3. Account management: Users can modify their personal and publicly displayed information.

4. Searching for events: Users can look up for schedules hosted by other users that suits their preference using filters.

5. Joining events: Users can choose to join events that are hosted by other people.

6. Hosting events: Users can create new events and set the event attributes such as date, time, location, and type of event to their liking to let other people join in.

7. Managing events: Users can leave an event if they joined an event as guests, or they can cancel events if they are hosts.

# 

# 3. External Interface Requirements

## 3.1 User Interfaces:

### 3.1.1 End Users:

* No specific knowledge or skills are required from the end user.
* End user should have basic ideas about computer operations.

### 3.1.2 Administrator:

* Administrators must be capable of managing members’s rights.
* If the network connection does not work properly then our system should not work as intended.
* Also, it is assumed that the product is installed properly at the web server.
* This system will not take care of any virus problem, which might occur either on the Client or the server system. Avoiding the use of pirated software and ensuring that floppies and other removable media are scanned for viruses before use could minimize the possibility of viral infection.
* Recovery of data after a system crash will be possible only if backups are taken at regular intervals.

# 4. System features

## 4.1 Functional requirements

### 4.1.1 Login

| Use case number | 01 | Use case name | Login |
| --- | --- | --- | --- |
| Description | User logs in to use the software’s functions. | | |
| Pre-condition | User already has an account.  Clicking on the website's login button.  Clicking on an event when not logged in. | | |
| Post-condition | Users successfully log into their account. | | |
| Basic path | No. | Performed by | Action |
| 1 | User | Click on the login button or an event card. |
| 2 | System | Display login function’s user interface. |
| 3 | User | Users fill in their username/email address and password. |
| 4 | User | Request login. |
| 5 | System | Check if all the fields are filled with the correct type of data. (Table 3.1.1.1) |
| 6 | System | Check if the filled user credential is correct. |
| 7 | System | Display corresponding contents to the user if logged in successfully. |
| Alternative path | No. | Performed by | Action |
| 5a | System | System informs the user if the information data type is incorrect or a field is missing information. |
| 6a | System | System informs the user if the user credential filled in the fields is wrong. |

*Table 4.1.1.1*

| No. | Data field | Description | Required | Condition | Example |
| --- | --- | --- | --- | --- | --- |
| 1 | Email | Username/Email field input | Yes | Correct email format | abc@gmail.com |
| 2 | Username | Username/email field input | Yes | Correct allowed characters | User1234 |

### 4.1.2 Change password

| Use case number | 02 | Usecase name |  |
| --- | --- | --- | --- |
| Description | User can change password | | |
| Pre-condition | User has to be logged in first. | | |
| Post-condition | User’s password changed | | |
| Basic path | No. | Performed by | Action |
| 1 | User | Clicks on username on navigation bar. |
| 2 | System | Transition to a web page that display user’s information. |
| 3 | User | Click on the Change *password* button. |
| 5 | System | Displays fields for changing password. |
| 6 | User | Fill the old password field. |
| 7 | User | Fill in the new password field. |
| 8 | User | Submit password. |
| 9 | System | Inform the user that the password has been updated. |
| Alternative path | No. | Performed by | Action |
|  | 8a | System | If the old password is wrong, inform the user of the wrong password error. |
|  | 8b | System | If any character that is not allowed put in the field, the system informs the user character error. |

*Table 4.1.2*

| No. | Data field | Description | Required | Condition | Example |
| --- | --- | --- | --- | --- | --- |
| 1 | Old password | Old password field input | Yes | Matches old password | AdfRLidfsdE |
| 2 | New password | New password field input | Yes | Valid characters | AFSDFDSF |
| 3 | Confirm new password | New password field input | Yes | Matches data in New password field | AFSDFDSF |

### 4.1.3 Register

| Use case number | 03 | Usecase name | Creating an account |
| --- | --- | --- | --- |
| Description | Let users create a new account to access the application’s services. | | |
| Pre-condition | User is not logged in. | | |
| Post-condition | User created a new account and logged into it. | | |
| Basic path | No. | Performed by | Action |
| 1 | User | User clicks on any event card or the sign in button in the navigation. |
| 2 | System | Transition to register web page. |
| 3 | User | Fill in the required information fields. (Table 3.1.3.1) |
| 4 | User | Submit the information. |
| 5 | System | Check if all the required information are filled correctly. |
| 6 | System | Create a new user account. |
| Alternative path | No. | Performed by | Action |
| 6a | System | If a required information is missing, inform the user of the error. |
| 6b | System | If the data type of the information is incorrect, inform the user of the error. |
|  | 6c | System | Inform the user of a mismatched password if the password field does not match with the main password field. |

*Table 4.1.3.1*

| No. | Data field | Description | Required | Condition | Example |
| --- | --- | --- | --- | --- | --- |
| 1 | Email | User email input field | Yes | Valid email format | abc@vnu.edu.vn |
| 2 | Username | User username input field | Yes | Unique username, valid and allowed characters only. | Usern1235 |
| 3 | Password | User password input field | Yes | Any valid characters | Fd$ljhf@-r51 |
| 4 | Confirm password | User password input field | Yes | Matches | Fd$ljhf@-r51 |
| 5 | Gender | Drop down box to pick gender | Yes | Pick male, female, or other | Male |
| 6 | Date of birth | Date input text field | Yes | Valid date | 01/04/2004 |
| 7 | Country | Drop down box to pick country | Yes | Pick Vietnam, United States, Japan,… | Vietnam |

### 4.1.4 Update user information

| Use case number | 04 | Usecase name | Update user information |
| --- | --- | --- | --- |
| Description | Users can change their profile’s public and private information. | | |
| Pre-condition | Users must be logged in first. | | |
| Post-condition | User information is changed. | | |
| Basic path | No. | Performed by | Action |
| 1 | User | User clicks on their username on the navigation bar. |
| 2 | System | Transition to a web page that displays user’s information. |
| 3 | User | User clicks on the edit profile button. |
| 4 | System | Displayed texts turn into an editable form. |
| 5 | User | Modify the contents of the form. |
| 6 | User | Save changes by clicking on save button. |
| 7 | System | Check if all the fields are filled or filled with the correct data type. |
| 8 | System | Update user’s information. |
| Alternative path | No. | Performed by | Action |
| 8a | System | Inform the user about the error if the required fields are left blank or the fields are filled with the wrong data type. |

*Table 4.1.4.1*

| No. | Data field | Description | Required | Condition | Example |
| --- | --- | --- | --- | --- | --- |
| 1 | Country | Drop down box to pick country | Yes | Pick Vietnam, United States, Japan,… | Vietnam |
| 2 | Date of birth | Date input text field | Yes | Valid date | 01/04/2004 |
| 3 | Gender | Drop down box to pick gender | Yes | Pick male, female, or other | Male |
| 4 | User description | Input text field | No | Valid characters, maximum length of 250 words | My name is Eughesgs, I’m looking for… |
| 6 | User contact detail | Input text field | No | Valid characters | Facebook: facebook.com/user123  My phone: 09031321321 |
| 5 | User profile picture | Box to select image | No | Valid image format: png, jpg.  In the range of minimum and maximum size. | The 30 Cutest Dog Breeds - Most Adorable Dogs and Puppies |

### 4.1.5 Searching for event

| Use case number | 05 | Usecase name | Search for event |
| --- | --- | --- | --- |
| Description | Users can use a set of filters to search for events that they desire. | | |
| Pre-condition | User has to be in a Join *a match* by clicking on the *Join a match* button in the top navigation bar of the web page. (User do not have to be logged in first) | | |
| Post-condition | Display search results that match the user’s filters. | | |
| Basic path | No. | Performed by | Action |
| 1 | System | System recommends and displays events in which the user is not the host or a participant by default. |
| 2 | User | Users select one of the many filters (Table 3.1.5.1). |
| 3 | User | Users modify the filter that they have chosen. |
| 4 | User | Users repeat path No.2 and No.3 until they are satisfied. |
| 5 | User | Users request for a result that matches their filter by clicking the *Find* button. |
| 6 | System | Display results that have attributes that match the filter values given by the users. |
| Alternative path | No. | Performed by | Action |
| 6a | System | If there is no result that matches the user’s request, the system informs the user that there is no matching event. |

*Table 4.1.5.1*

| No. | Data field | Description | Required | Condition | Example |
| --- | --- | --- | --- | --- | --- |
| 1 | Event name | Text field input | No | Valid characters | Hang out |
| 2 | Location | Text field input/Pick from map | No | Valid characters | Cau Giay, Hanoi, Vietnam |
| 3 | User capacity | Number of maximum player per event, text field. | No | Numbers only | 5 |
| 4 | Tags | Search and select tags | No | Search and select from a box | Sport, Study, … |
| 5 | Date | Selection box/Text field | No | Can only fill valid dates, cannot pick a date from the past. | 28/03/2024 |
| 6 | Time | Selection box/Text field | No | Valid time, cannot pick a time in the past | 10pm |

### 4.1.6 Viewing event detail

| Use case number | 06 | Usecase name | Viewing event detail |
| --- | --- | --- | --- |
| Description | Users view extended information about an event. | | |
| Pre-condition | User is in the *Find a match* web page (Use case 05).  User is in the *My events* web page (Use case 10).  User is in the *Host event* web page (Use case 09). | | |
| Post-condition | Event detailed information is displayed for the user to see. | | |
| Basic path | No. | Performed by | Action |
| 1 | User | Users click on event cards that display brief information about the event. (Table 3.1.5.1) |
| 2 | System | Open a webpage to display full event information. |
| 3 | User | Users can click on the Backbutton to exit viewing event details. |

### 4.1.7 Viewing other user’s description

| Use case number | 07 | Usecase name | View user description |
| --- | --- | --- | --- |
| Description | Let the current view other user’s description (Table 3.1.4.1) participating an event | | |
| Pre-condition | User must be viewing event detail (Use case 06) | | |
| Post-condition | System shows a user’s public information. | | |
| Basic path | No. | Performed by | Action |
| 1 | User | Inside the event detail, click on a username in the list of participants or the username of the host. |
| 2 | System | Show an overlay that displays a user’s public information. |
| 3 | User | After having finished viewing, user can close the tab by pressing the X icon. |

### 4.1.8 Joining events

| Use case number | 08 | Usecase name | Joining event |
| --- | --- | --- | --- |
| Description | Users join an event, adding the event into their list of events. | | |
| Pre-condition | Users must already be logged in (Use case 01).  The event they are joining must not already be in their list of events.  User is in *Find a match* tab (Use case 05). | | |
| Post-condition | System adds the joined event into the user's list of events. | | |
| Basic path | No. | Performed by | Action |
| 1 | User | Users click on an event to join its details (Use case 06). |
| 2 | System | System displays the event’s detail, event detail shows the join button. |
| 3 | User | Users click on the join button. |
| 4 | System | Add the event into the user's list of events and inform the user that they have joined the event. |
| Alternative path | No. | Performed by | Action |
| 1a | User | Users click directly on the *Join* button on the event card without having to view its details. |
| 2b | System | If the user is the host of the event or already joined, *the Join* button is not displayed. |
| 4c | System | If the event has reached its maximum participant then inform the user that the event is full. |

### 4.1.9 Hosting events

| Use case number | 09 | Usecase name | Hosting event |
| --- | --- | --- | --- |
| Description | Users can create new events to let other users search and join. | | |
| Pre-condition | User must be logged in (Use case 01) | | |
| Post-condition | Users create a new event and add it to their list of events. | | |
| Basic path | No. | Performed by | Action |
| 1 | User | Users click on the Host *event* button on the top navigation bar to go to the hosting event web page. |
| 2 | System | Display host new event web page which include a form to fill in event’s information and display a list of joined events next to the form. |
| 3 | User | Fill in the required information (Table 3.1.9.1) to create an event. |
| 4 | User | Submit the event created. |
| 5 | System | Check for validity of the event created. |
| 6 | System | Create the event, add it to the user's event list and display it publicly in the *Find a match* web page for other users to join. |
| Alternative path | No. | Performed by | Action |
|  | 5a | System | If the event created overlaps another event’s schedule then the system would not create the event then inform the error to the user. |

*Table 4.1.9.1*

| No. | Data field | Description | Required | Condition | Example |
| --- | --- | --- | --- | --- | --- |
| 1 | Event name | Name for event, text field | Yes | Valid characters | Running |
| 2 | Date | Select from calendar, fill in text field, or drop down | Yes | Valid date and characters, cannot pick dates from the past. | 28/07/2024 |
| 3 | Time | Select from drop down, select start time and end time | Yes | Valid time, cannot pick time from the past | 10pm-11pm |
| 4 | Location | Text field input or select from map | Yes | Valid location | Hanoi, Vietnam |
| 5 | Description | Text field input | No | Limit to 250 words | This event is going to be… |
| 6 | Tag | Type of event, select from a box of events or made up tag | No | If tag is empty, the event is set to default tag *Other* | Sport, Team,… |

### 4.1.10 Events management

| Use case number | 10 | Usecase name | Managing events |
| --- | --- | --- | --- |
| Description | Users can see all the events they have joined in the future and past. Users can cancel events if they are the host.  Users can leave the event if they are a participant. | | |
| Pre-condition | User must already logged in (Use case 01) | | |
| Post-condition | User’s list of events modified or viewed. | | |
| Basic path | No. | Performed by | Action |
| 1 | User | Users go to *My events* web page by clicking on the *My events* button on the top navigation bar. |
| 2 | System | Display a list of event cards where the user is a participant or a host. Events that have happened or canceled will be marked with lower opacity. |
| 3 | User | User click on an event to view it’s detail (Use case 06) |
| 4 | System | System displays event details, if the user is a participant then the Leave button is displayed. |
| 5 | User | Users click on the Leavebutton. |
| 6 | System | If the user is a participant and leaves the event, the system will remove the event from that user’s list of events. |
| Alternative path | No. | Performed by | Action |
| 5a | System | System displays event details, if the user is a host then the Cancelbutton is displayed. |
| 6a | User | Users click on the Cancelbutton. |
| 7a | System | If the user is a host and cancels the event, the system will mark that event as canceled in the host and participant’s event list. |
| 4b | System | If the event has already happened or canceled, the system will display a delete button. |
| 5b | User | User clicks on the *Delete* button. |
| 6b | System | System removes the event permanently from the user's event list and database. |

# 5. Other Non-functional Requirements

## 5.1 Performance requirements

The software should be able to handle around 50 concurrent users maximum. The maximum response time for all operations made by the user should be 2 seconds.

## 5.2 Design Constraints

The software is built using Web technology, therefore, in order to use it, the users need to be connected to the Internet. Users also have to use Google Chrome, Mozilla Firefox, Safari, or any other web browser that supports Javascript applications.

## 5.3 Logical database requirements

The system shall store user account information in a MongoDB NoSQL database due to its flexibility and scalability.

### 5.3.1 User collection

| **Fields** | **Datatype** | **Description** | **Example** |
| --- | --- | --- | --- |
| \_id | String | Store user unique id | 51231kjh3123khf |
| username | String | Unique, publicly displayed | user1235 |
| email | String | Private data | asdf@gmail.com |
| event | Array | An array of event objects |  |
| password | String | Private data | DFSDffef#423432 |
| profile | profile (Section 3.4.2) | User publicly displayed information |  |
| dob | Date | Date of birth | 08/03/2023 |

### 5.4.2 Profile collection

| **Fields** | **Datatype** | **Description** | **Example** |
| --- | --- | --- | --- |
| description | String | Store description about user | Hello this is, my name is,... |
| contact | String | Store contact details for other users to see |  |

### 5.4.3 Event collection

| **Fields** | **Datatype** | **Description** | **Example** |
| --- | --- | --- | --- |
| \_id | String | Event id | asdf34sldkfjs |
| name | String | Store event name | Walking |
| capacity | Number | Maximum number of users | 5 |
| current\_members | Number | Number of participants in the event | 3 |
| participants | Array | Store a list of participants, an array of users |  |
| host | User | Store host object |  |
| date | Date | Store event date | 03/03/1932 |
| time | Time | store starting time and end time | 22:00:00 |
| location | String | Store location of the event | Hanoi, Vietnam |
| tags | Array | Event tags | sport, study, group |
| description | String | Event description | This event is meant to,... |

## 5.5 Software system attributes

### 5.5.1 Reliability

The software should display the events created as the user intended. The software must always detect overlapped events, events that have happened, all the events that a user joins must be valid.

### 5.5.2 Availability

Systems need to record user data with checkpoints to restore data in case the server needs to recover.

### 5.5.3 Security

#### 5.5.3.1 Secured communication

* Communication between client and server is protected by encryption protocols like SSL/TLS

#### 5.5.3.2 Controlled access rights

* Only users with certain access rights can perform certain operations like modifying, deleting, adding data.
* System must have a function to manage access rights based on the role of the user.

#### 5.5.3.3 Logging user activities

* Log and keep records of all important activities from the users when using the system.