TED UNIVERSITY

CMPE 491

Senior Design Project I

VENATOR

Analysis Report

SECTION 1 - GROUP 6

26/11/2023

Team Members:

- **Arda Uyaroğlu**
- **Berker Tomaç**
- **Selçuk Akif Topkaya**
- **Utku Oktay**

Table of Contents

1 Introduction

This report contains a detailed analysis of the problem and addresses all relevant issues related to the project. The requirements are discussed and analyzed carefully with Professor Çapın, our advisor who represents the customer role, and this document is produced as a result of this thorough communication with the customer. This analysis report aims to provide a correct, complete, consistent and verifiable model of the system by formalizing the system specifications produced during requirements elicitation, and it examines the boundary conditions and exceptional cases in more detail.

2 Current System

Venator, with its current status, is capable of presenting a few essential objectives of the desired end-product. The current capabilities include detecting and tracking the players, goalkeepers, referees and the ball components of a game with a mean average precision of 0.89.

The confusion matrix is shown in the following figure:

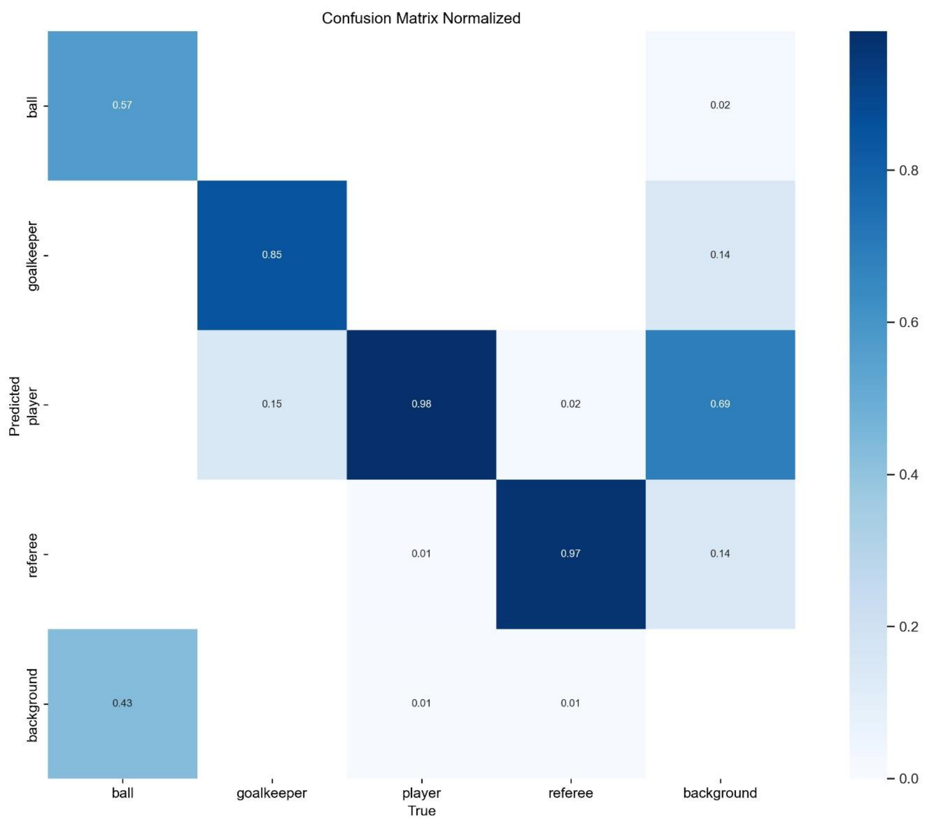


Figure 1: confusion matrix

Here is an exemplary snapshot of a single frame, showing the detect & track algorithm in work:

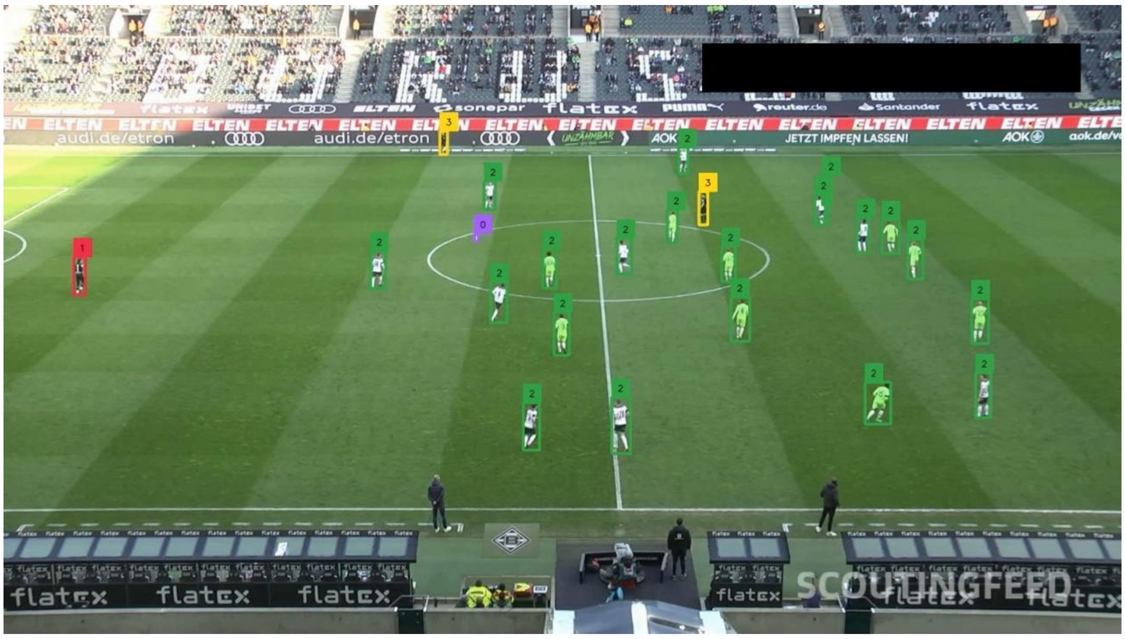


Figure 2: current system's detect & track algorithm in work

3 Proposed System

3.1. Overview

Venator detects and tracks the various components within a football match, namely the players, goalkeepers, referees and the ball, and presents some statistics about these components, computing and preparing them by applying computer vision techniques on the video recordings of the matches.

3.2. Functional Requirements

Venator has a list of absolute and optional requirements, which it includes or may choose to include (if we, the developers feels that it enhances the product or the implementation is viable within the extent of our resources and limited time frame) the functionality they describe in the end-product, respectively.

Here are the functional requirements of Venator listed as follows:

- Venator must find, localize and track the players.
- Venator must give unique IDs to each detected player.

- Venator must compute the distance coverage of each player.
- Venator must compute the average position for each player.
- Venator must compute a heat-map of each player.
- Venator must compute a predicted location of the ball.
- Venator must compute the ball possession duration of each team.
- Venator must differentiate the players of the two opposing teams of the football match.
- Venator may identify each detected player and associate ID-based data with the real player.
- Venator may compute percentage of attack in different zones.
- Venator may detect action events such as pass, goal, corner, out, throw-in, offside, etc. and count the number of action events for each team.

3.3. Nonfunctional Requirements

Below are the nonfunctional requirements of Venator, which defines its user-visible aspects such as usability, reliability, performance and supportability, etc:

- Venator should provide a user-friendly interface that allows the end-users to easily navigate through the app and access the statistics that they are interested in.
- Venator should provide a detailed tutorial and documentation on the usage of the app’s various features by the end-user.
- Venator should be scalable in the sense that it should operate independently from the number of players in a match, the stadium the match is played in, and the level of play the players demonstrate in the match.
- Venator should be developed with maintainability in mind. The modularity of the code should not make updating the app difficult.
- Venator should provide a secure environment that preserves confidentiality for the data that resides in its databases with its authentication and verification processes.
- Venator should not allow unauthorized users to use the app’s features with its encrypted access control mechanism.
- Venator should operate with acceptably high accuracy.

3.4. Pseudo Requirements

The pseudo requirements (constraints that are separated from the "quality" requirements) in which Venator is to operate are:

- Venator's algorithm should use the resources efficiently and in an environment-friendly manner in its training phase and its end-product's operating phase where the statistics are calculated.
- Venator should comply with the current laws and regulations for recording video and collecting data during sport events.
- Venator should be developed (and produce the statistics) in an unbiased way and thus, maintain fairness in its algorithm without giving any room to developers' or customers' personal biases when it comes to a players' race, gender, religion, etc.
- Venator should give strong importance in highlighting the data collection and statistical computation aspects of the app to clearly inform the end-users and the components (players, goalkeepers, referees, etc.) of the product for providing an informed and consented environment.
- Venator should compute and deliver the statistics in a transparent manner without favoring a player/customer over another player/customer. It should not provide untrue/uncalculated information to the end-users of the product.

3.5. System Models

3.5.1 Scenarios

In this section, the scenarios for how a typical user interacts with the system is discussed.

Scenario name	User Registration
Participating actor instances	User
Flow of events	1. User enters their name and surname. 2. User enters an e-mail address. 3. User enters a password. 4. User re-enters the password for confirmation. 5. User clicks on *Sign Up* button.

Scenario name	User Login
Participating actor instances	User
Flow of events	1. User enters e-mail and password. 2. User clicks on *Login* button.

Scenario name	User Login with their Google account
Participating actor instances	User
Flow of events	1. User clicks on *Log in with Google* button. 2. User selects a Google account.

Scenario name	First time user login with e-mail and password
Participating actor instances	User
Flow of events	1. User enters e-mail and password. 2. User clicks on *Login* button. 3. The tutorial is initiated.

Scenario name	First time user login with their Google account
Participating actor instances	User
Flow of events	1. User enters e-mail and password. 2. User clicks on *Login* button. 3. The tutorial is initiated.

Scenario name	User Logout
Participating actor instances	User
Flow of events	1. User clicks on *Logout* button.

Scenario name	Forgot Password
Participating actor instances	User
Flow of events	1. User clicks on *Forgot your password?* button. 2. User enters an e-mail to obtain the resetting link. 3. User enters a new password. 4. User re-enters the password for confirmation. 5. User clicks on *Reset* button.

Scenario name	Create a new game record
Participating actor instances	User
Flow of events	1. User navigates to *Games* page. 2. User enters the record's name. 3. User enters the teams. 4. User uploads the video recording of the game. 5. User clicks on *Save* button to initiate the analyzation.

Scenario name	Edit a game record
Participating actor instances	User
Flow of events	1. User navigates to *Games* page. 2. User selects the game s/he intends to edit. 3. User edits the record name or teams. 4. User clicks on *Save Changes* button.

Scenario name	Delete a game record
Participating actor instances	User
Flow of events	1. User navigates to *Games* page. 2. User selects the game s/he intends to delete. 3. User clicks on *Delete Game Record* button.

Scenario name	Viewing the past games and statistics
Participating actor instances	User
Flow of events	1. User navigates to *Games* page. 2. User selects a game s/he intends to view.

Scenario name	Viewing the account information
Participating actor instances	User
Flow of events	1. User clicks on to the user icon (on the top-right of the page).

Scenario name	Editing the account info page
Participating actor instances	User
Flow of events	1. User clicks on to the user icon (on the top-right of the page). 2. User clicks on *Account Settings* button. 3. User edits their name, surname, e-mail or password. 4. User clicks on *Proceed* button.

Scenario name	Viewing the home page
Participating actor instances	User
Flow of events	1. User navigates to *Home* page.

3.5.2 Use Case Model

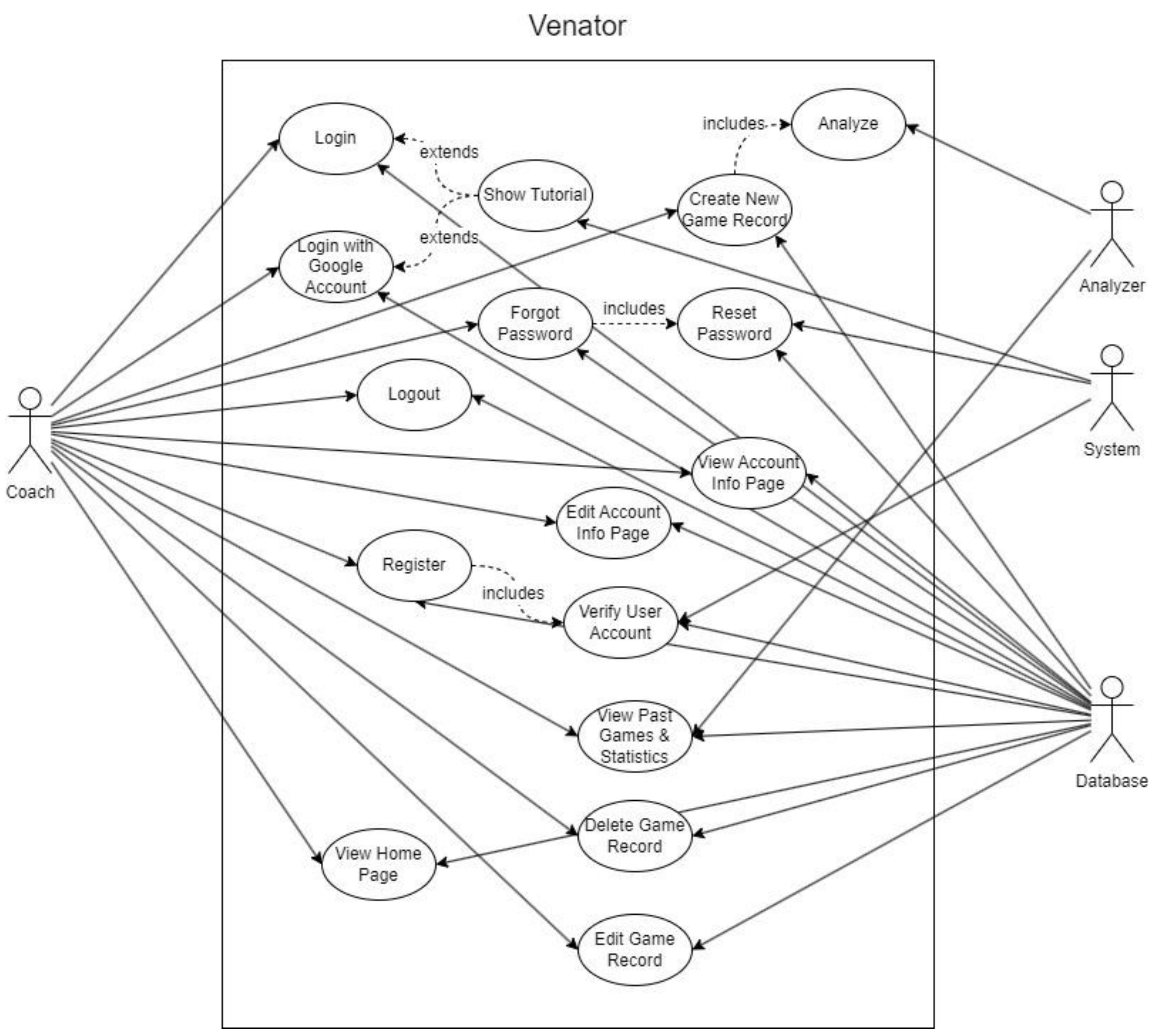


Figure 3: Use Case Model

3.5.3 Object and Class Model

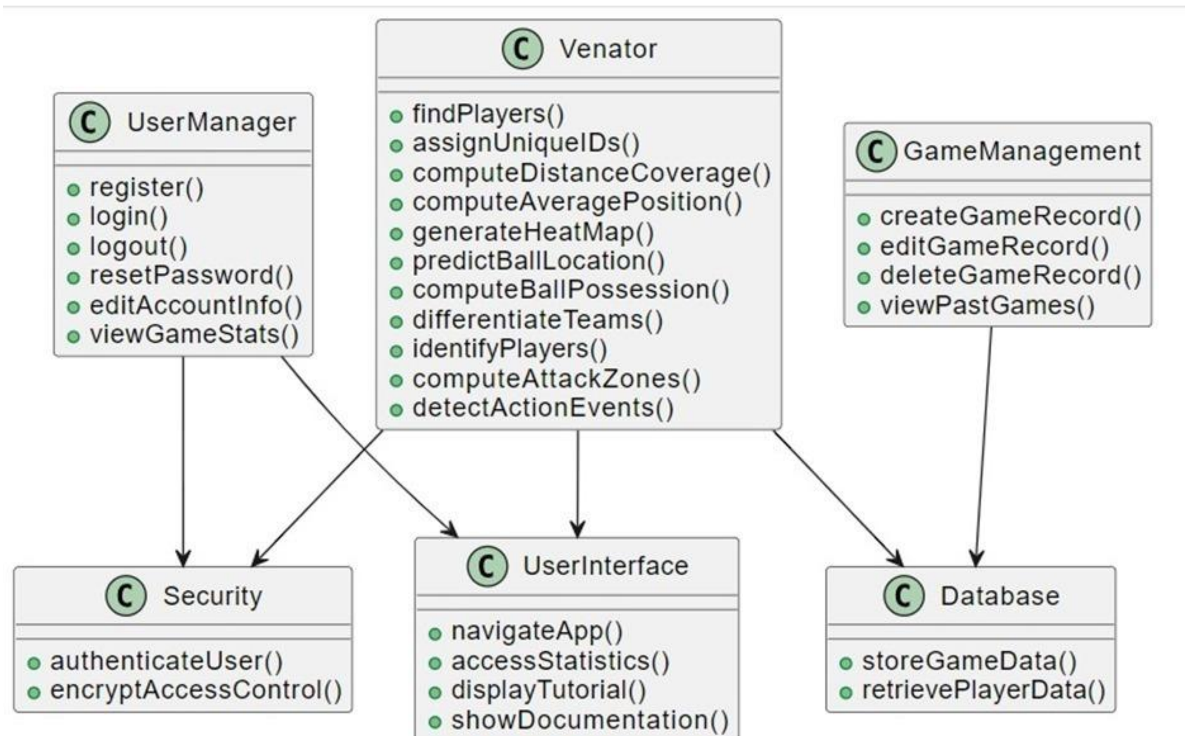


Figure 4: Object and Class Model

3.5.4 Dynamic Models

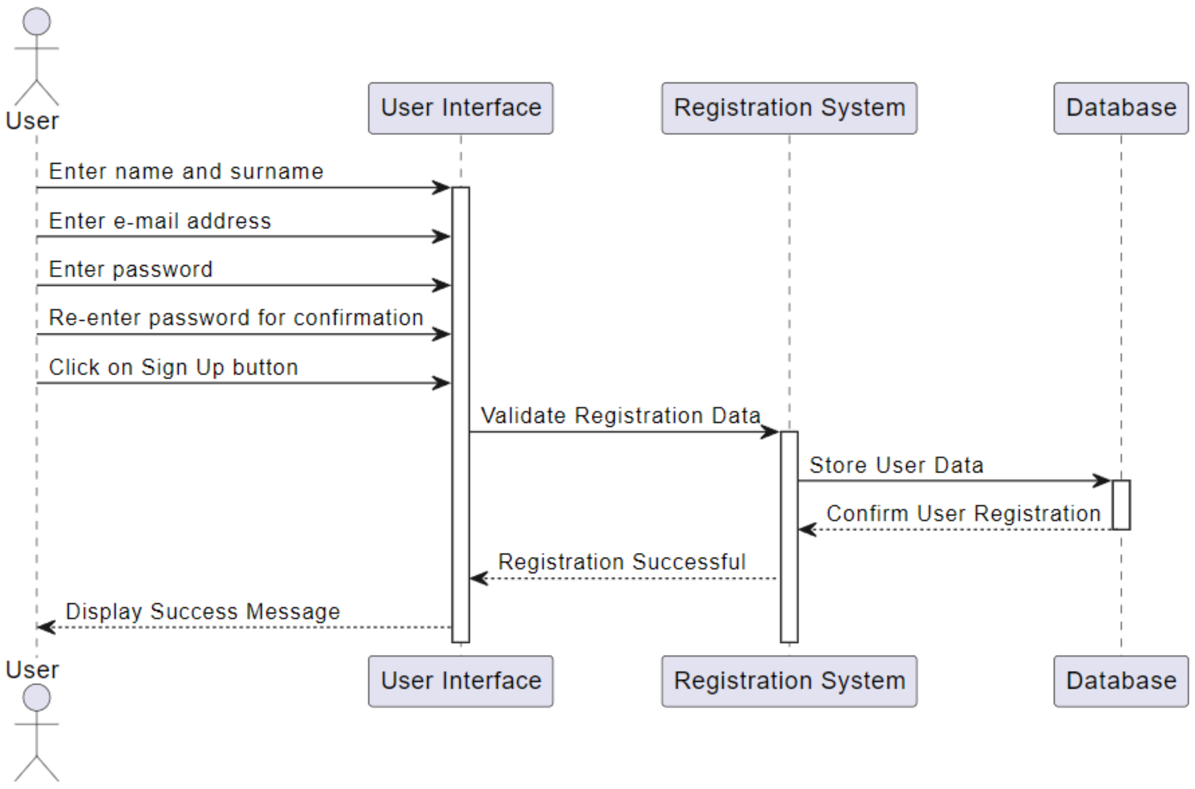


Figure 5: Sequence diagram for registration

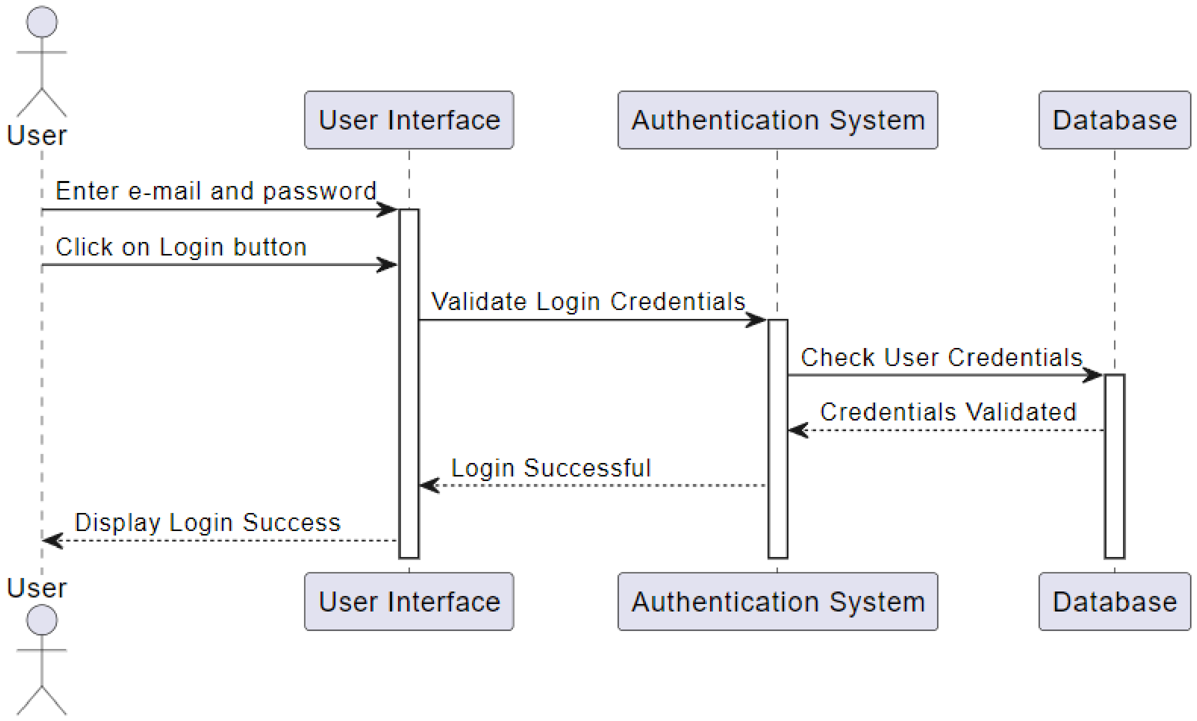


Figure 6: Sequence diagram for logging in with e-mail and password

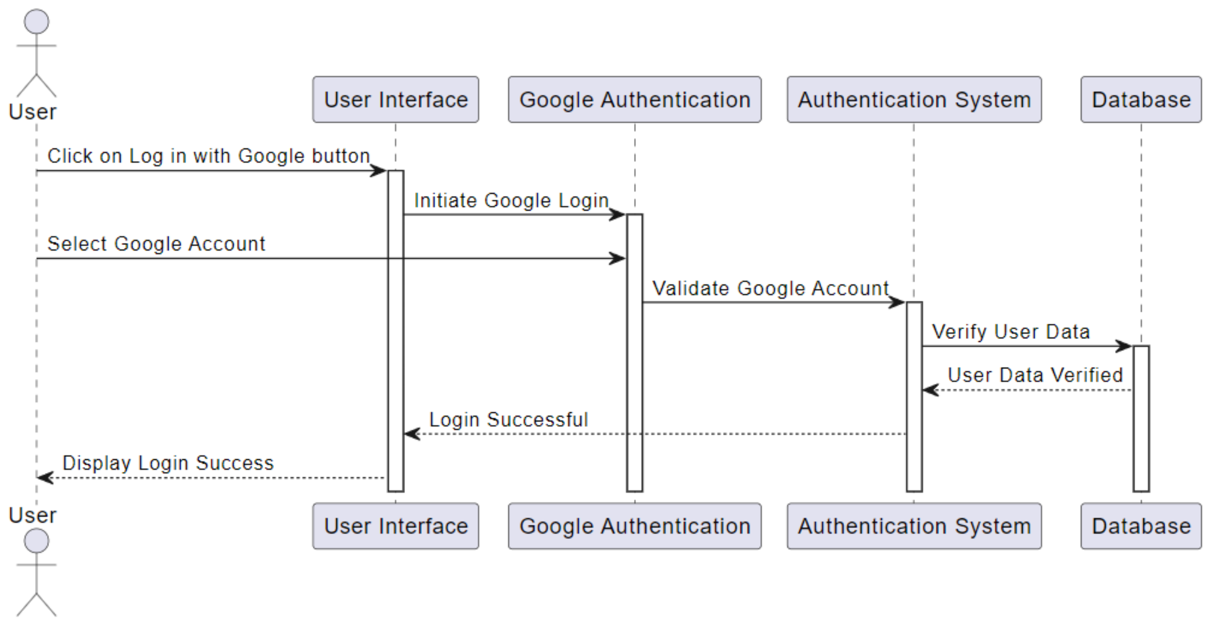


Figure 7: Sequence diagram for logging in with Google account

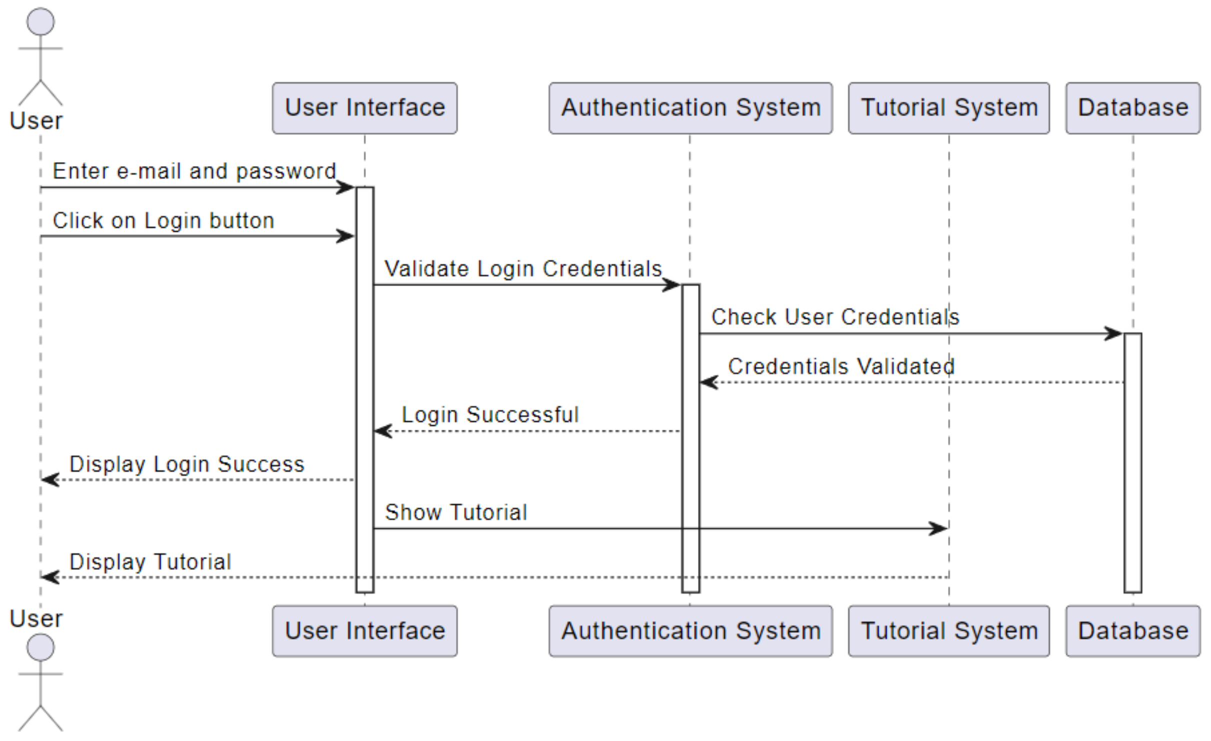


Figure 8: Sequence diagram for first-time logging in with e-mail and password

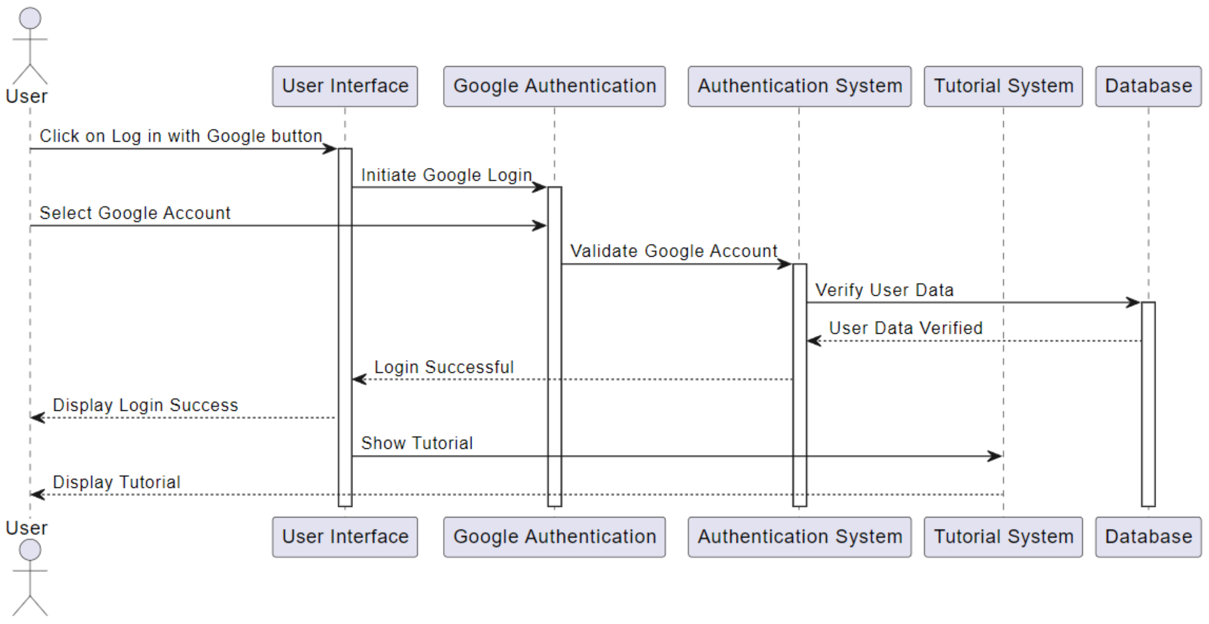


Figure 9: Sequence diagram for first-time logging in with Google account

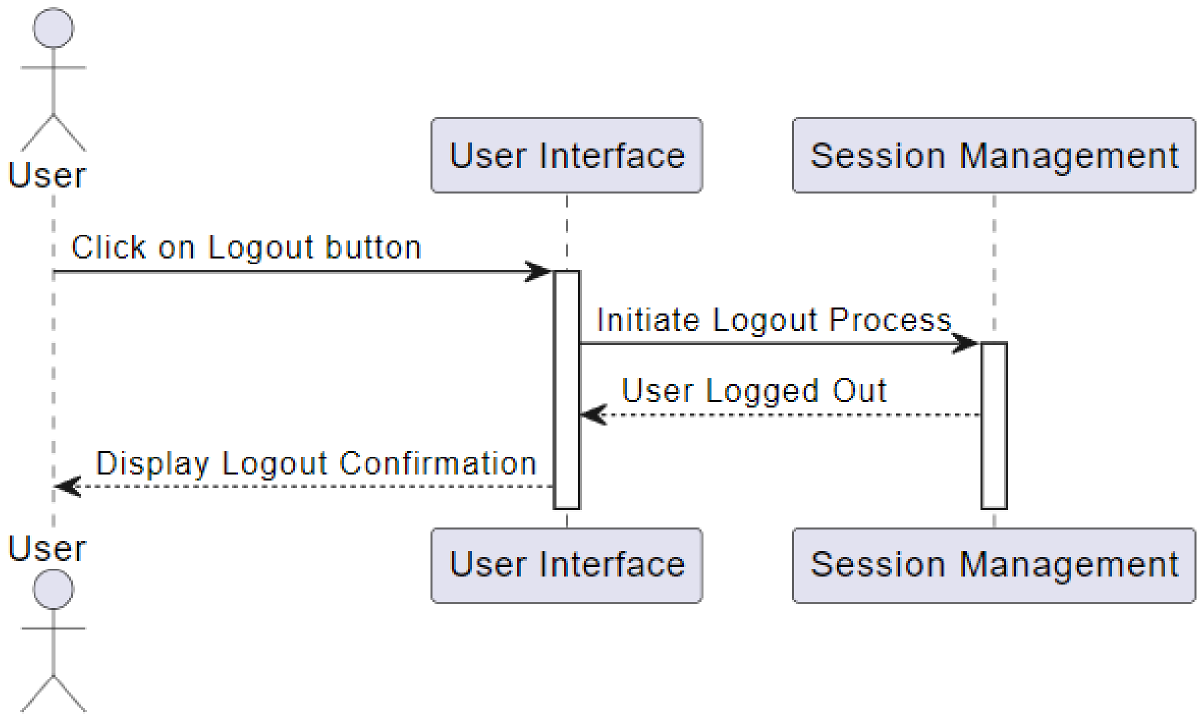


Figure 10: Sequence diagram for logging out

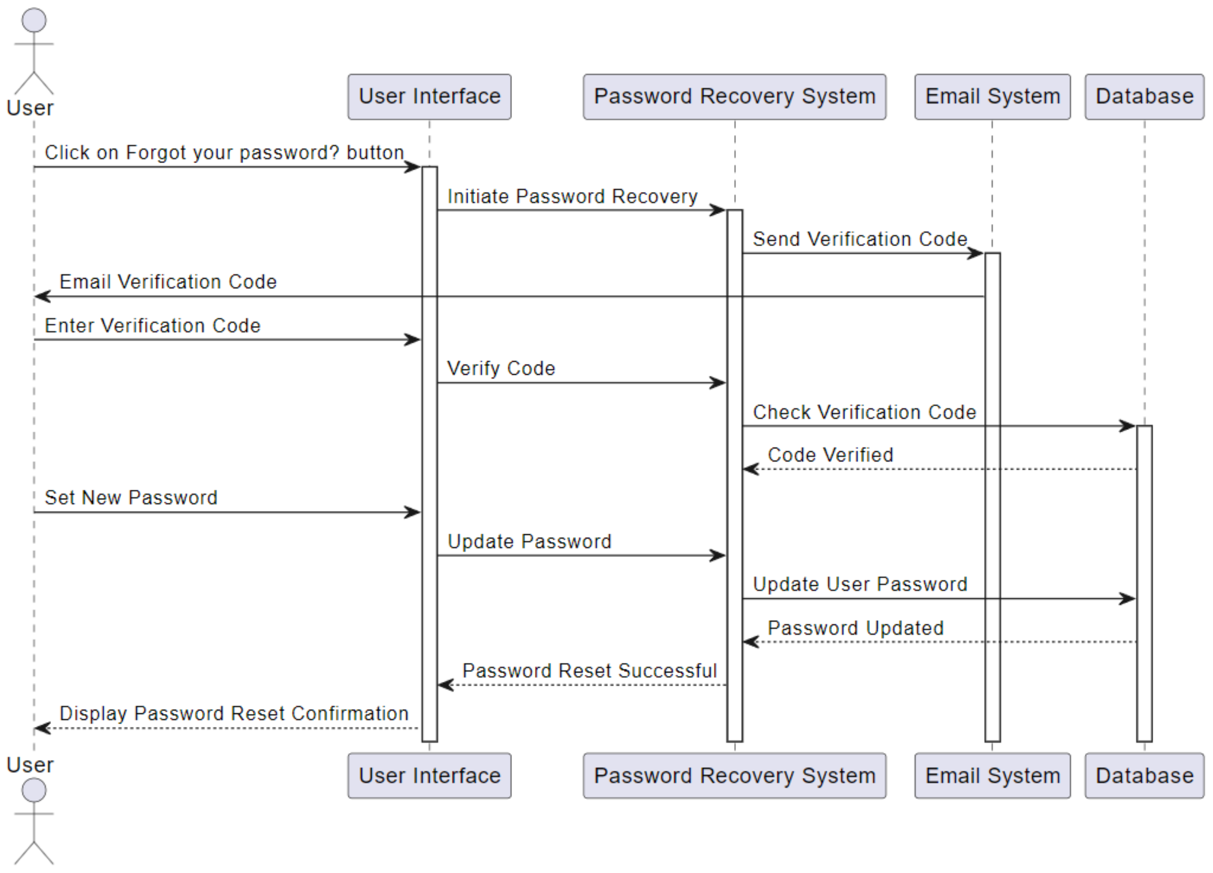


Figure 11: Sequence diagram for forgetting password

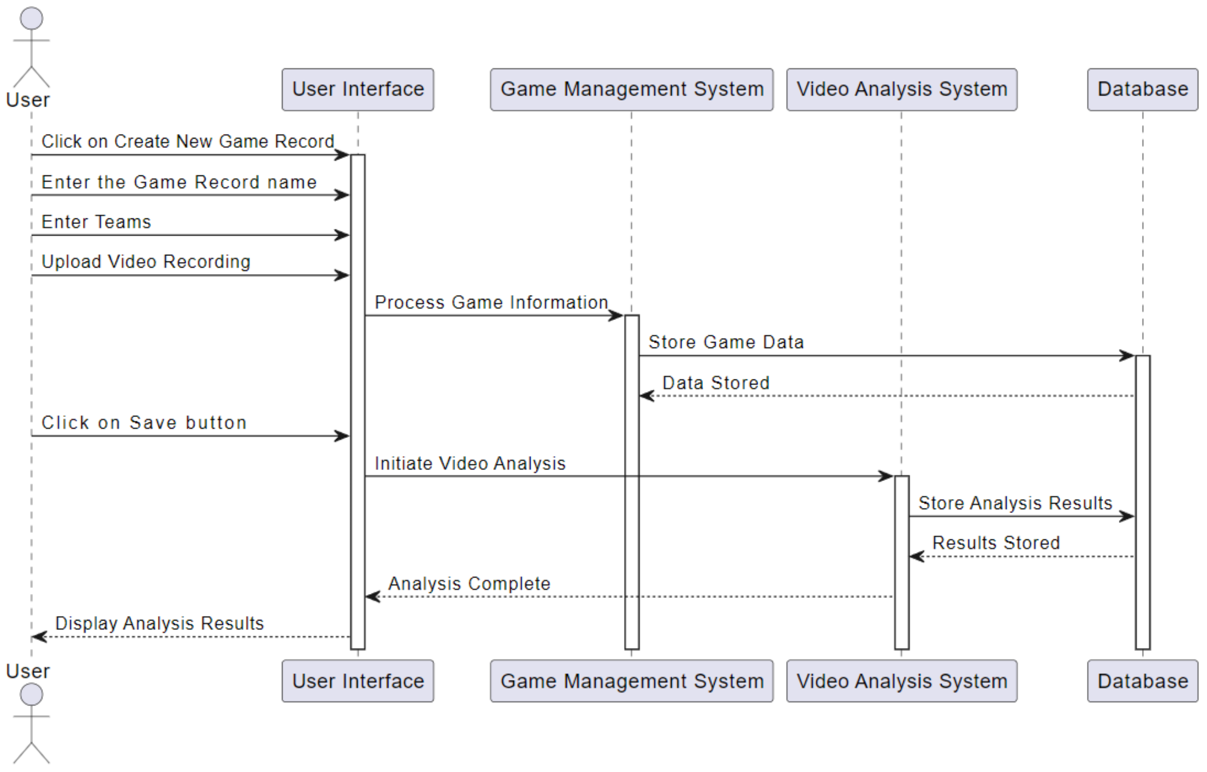


Figure 12: Sequence diagram for adding a new game record

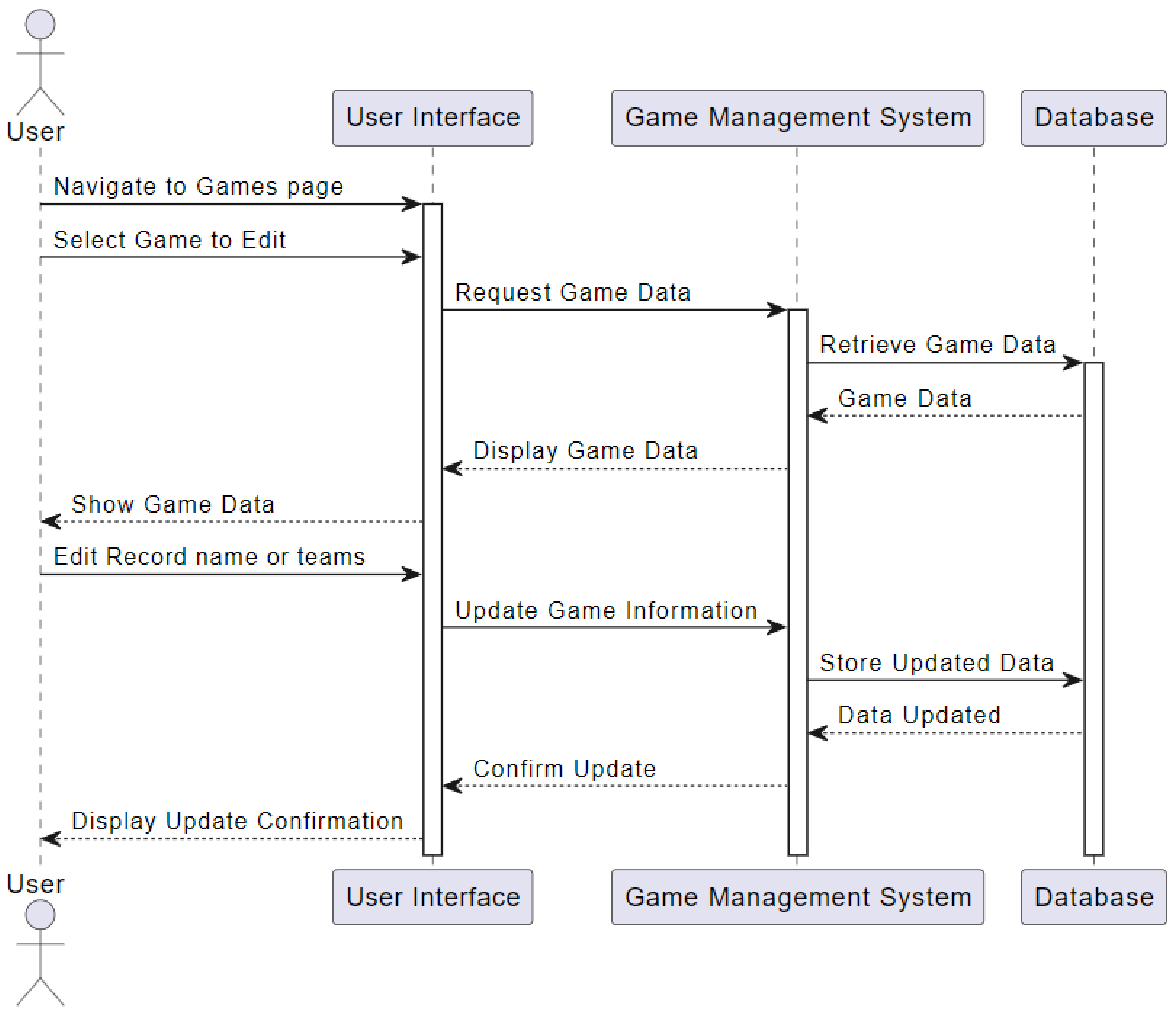


Figure 13: Sequence diagram for editing a game record

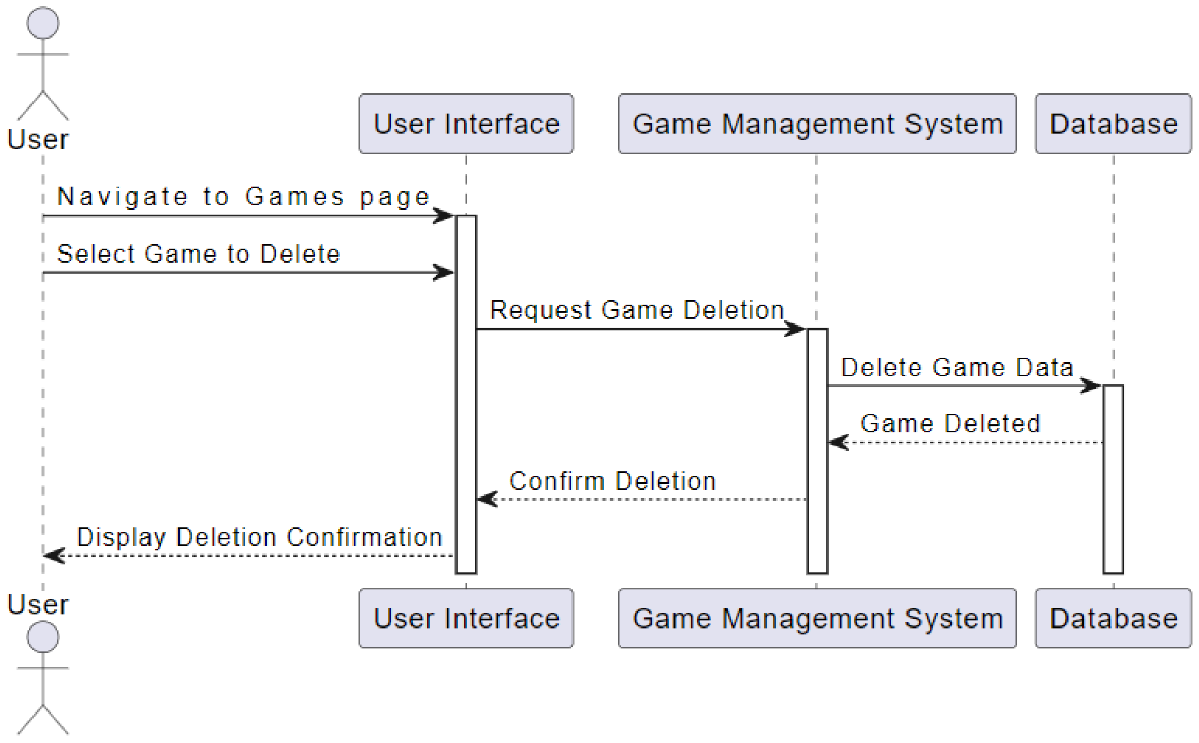


Figure 14: Sequence diagram for deleting a game record

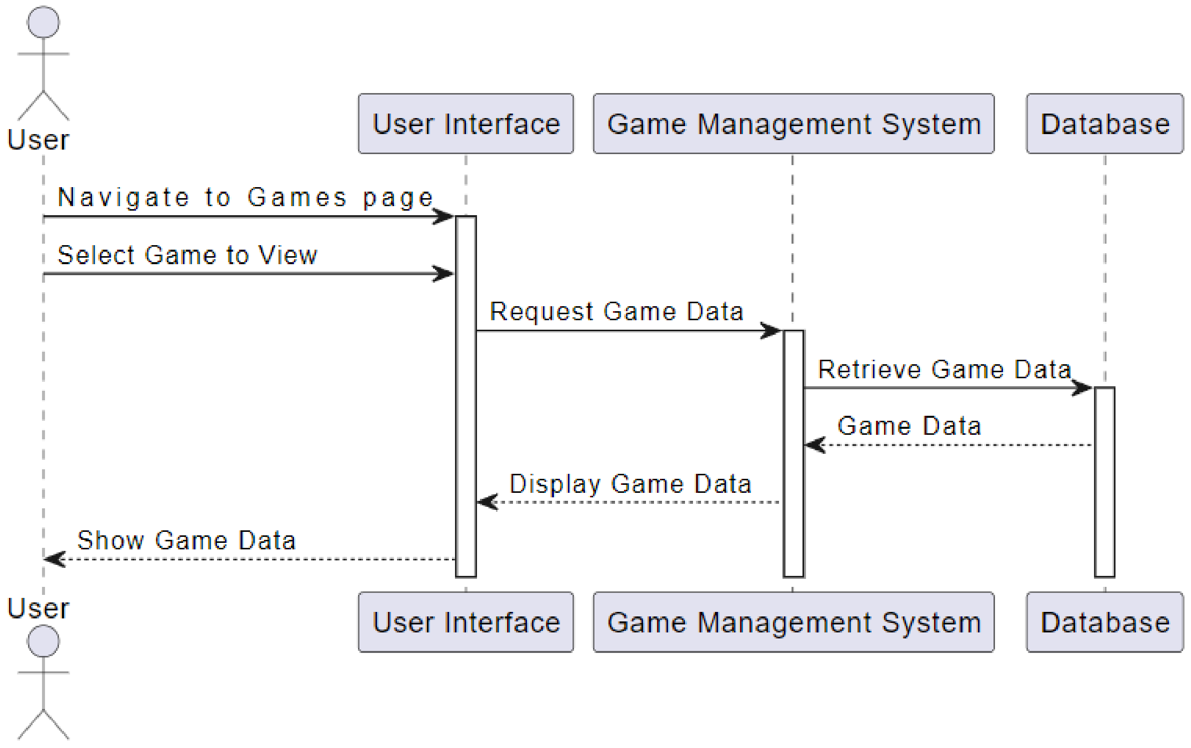


Figure 15: Viewing a past game record

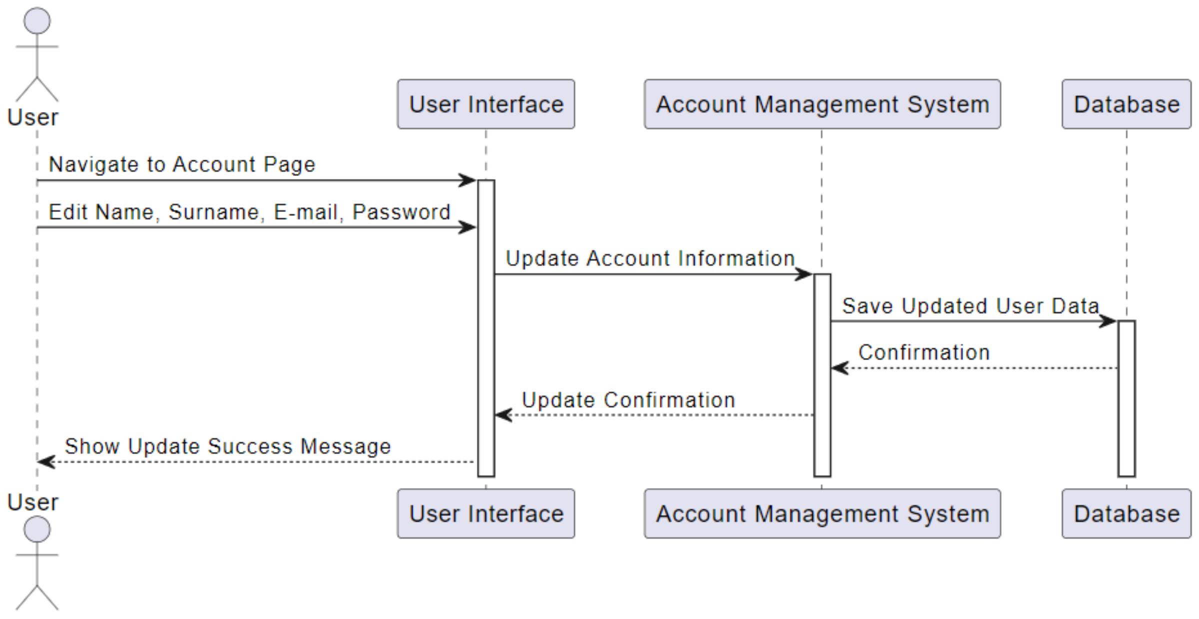

User
User Interface
Account Management System
Database
Navigate to Account Page
Edit Name, Surname, E-mail, Password
Update Account Information
Save Updated User Data
Confirmation
Update Confirmation
Show Update Success Message
User
User Interface
Account Management System
Database

Figure 16: Sequence diagram for editing the account information

3.5.5 User Interface - Navigational Paths and Screen Mock-ups

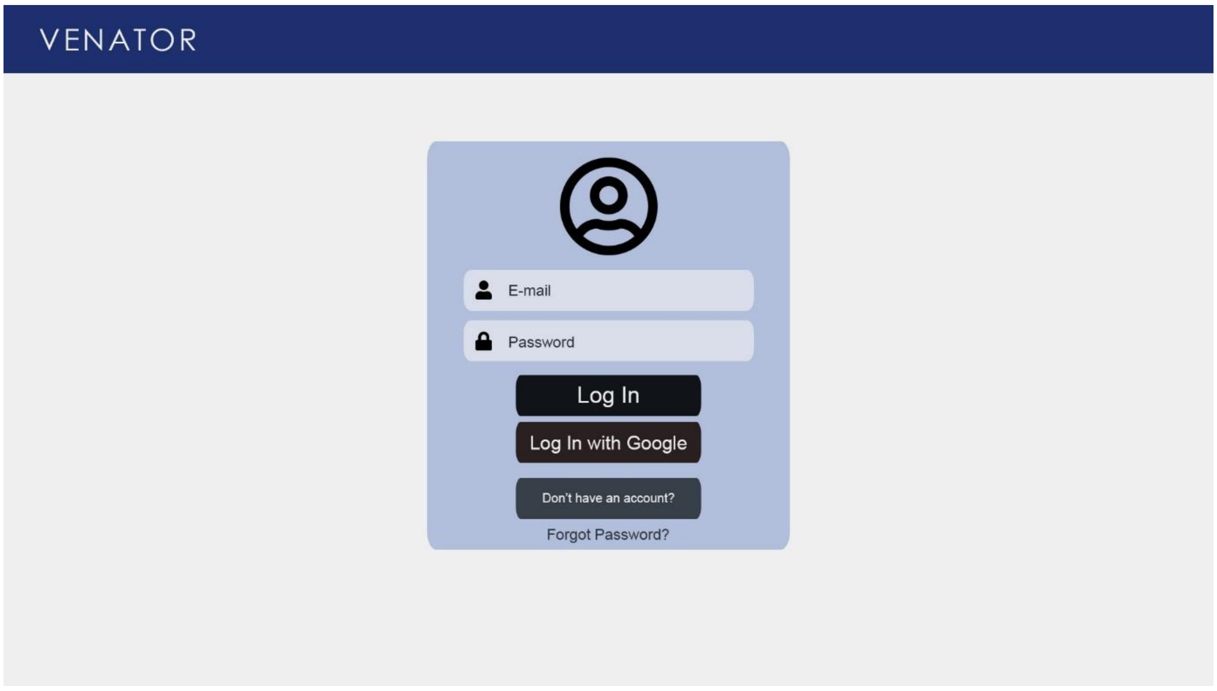


Figure 17: LogIn

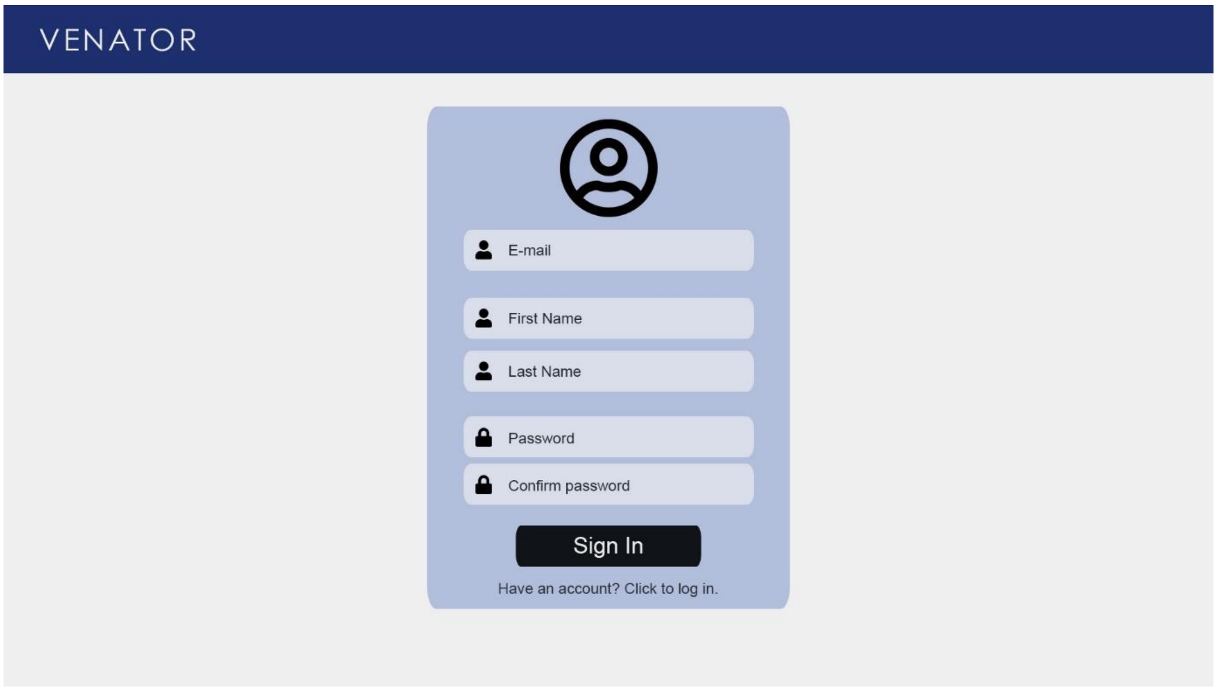


Figure 18: SignIn

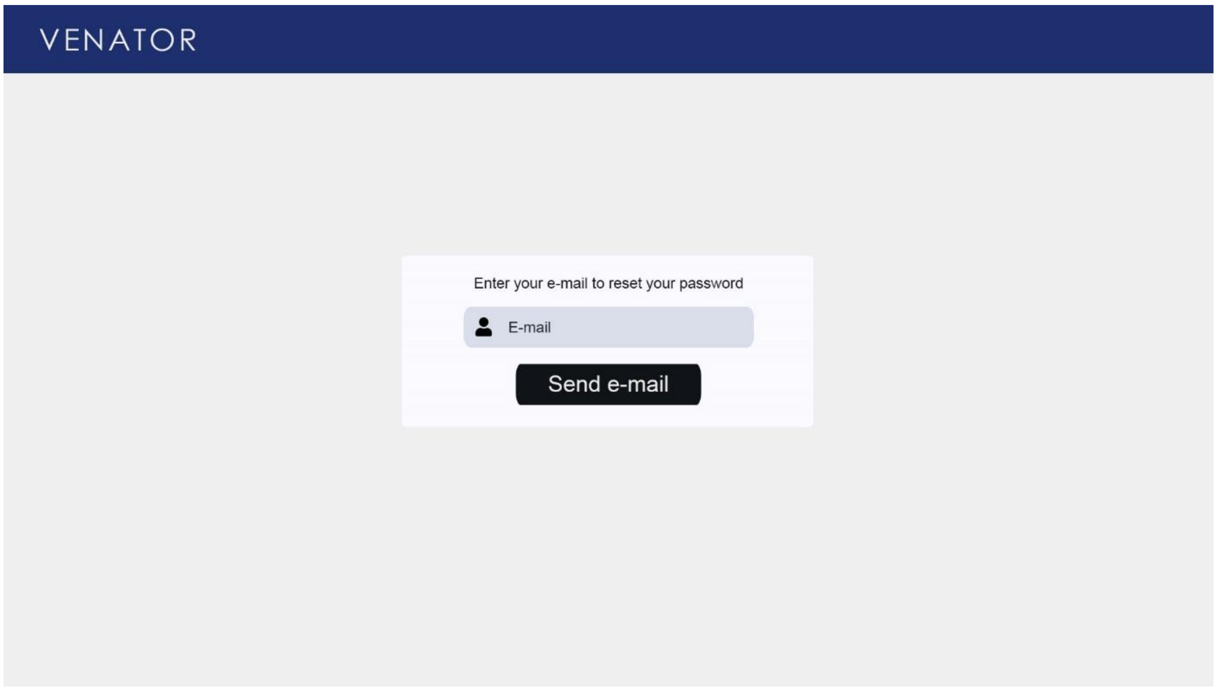


Figure 19: Forgot Password

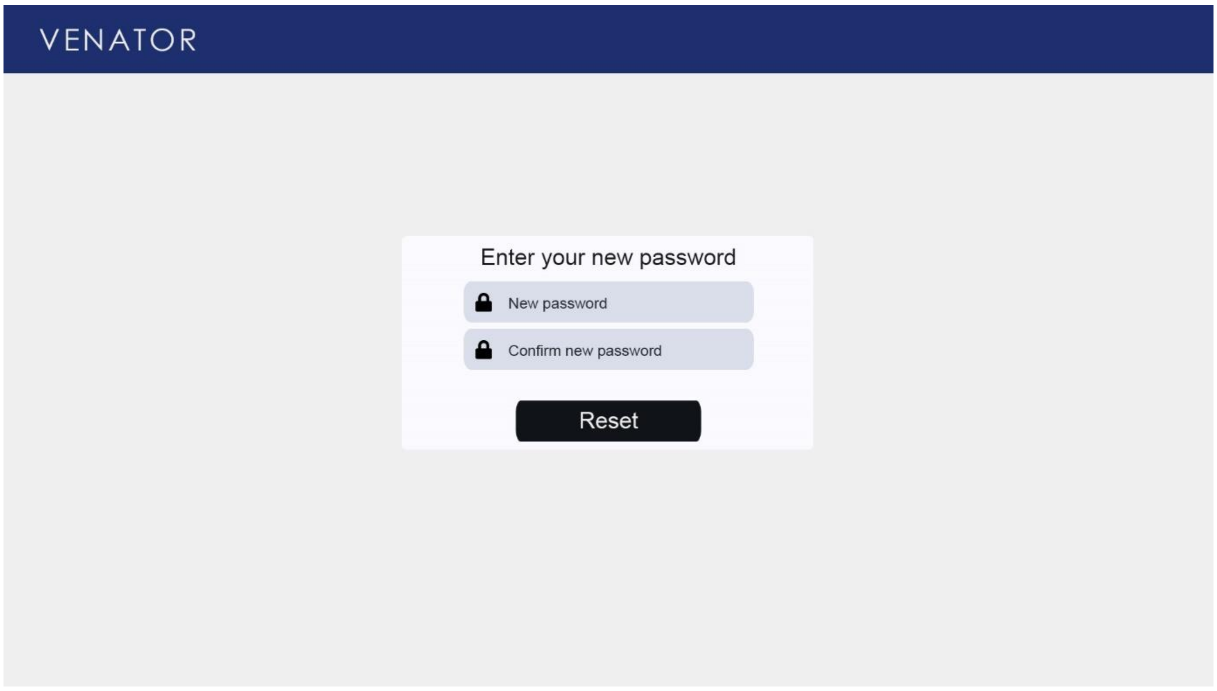
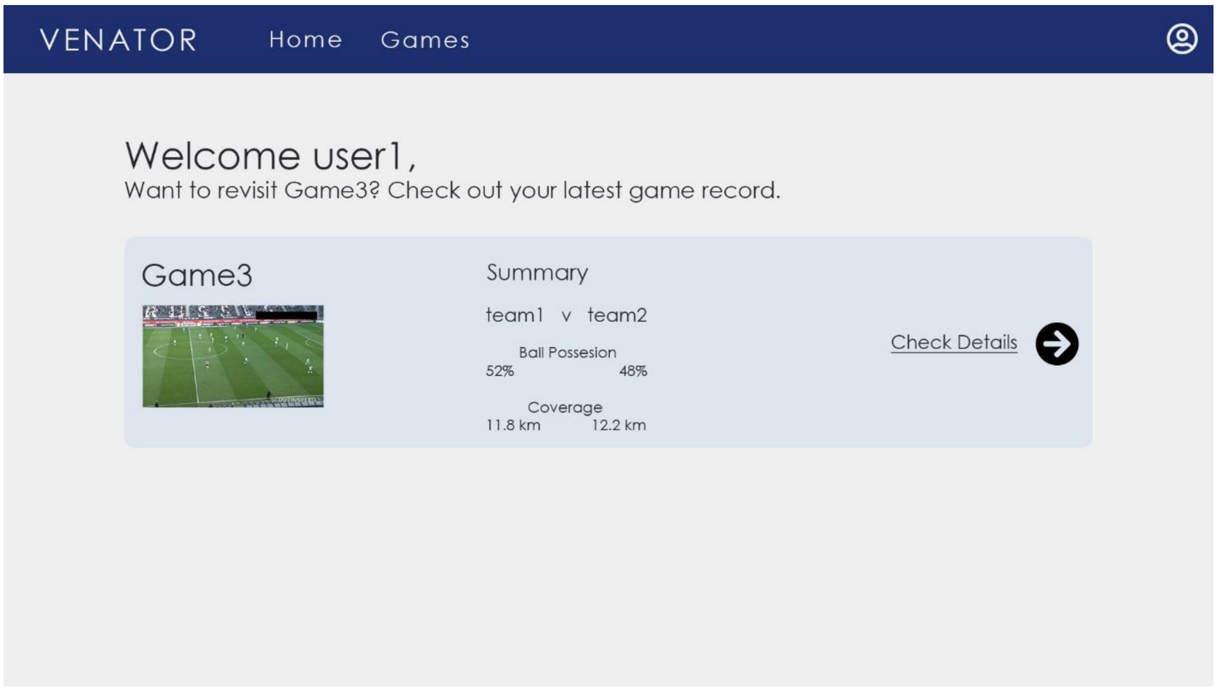


Figure 20: Reset Password


VENATOR Home Games

Welcome user1,
Want to revisit Game3? Check out your latest game record.

Game3

Summary

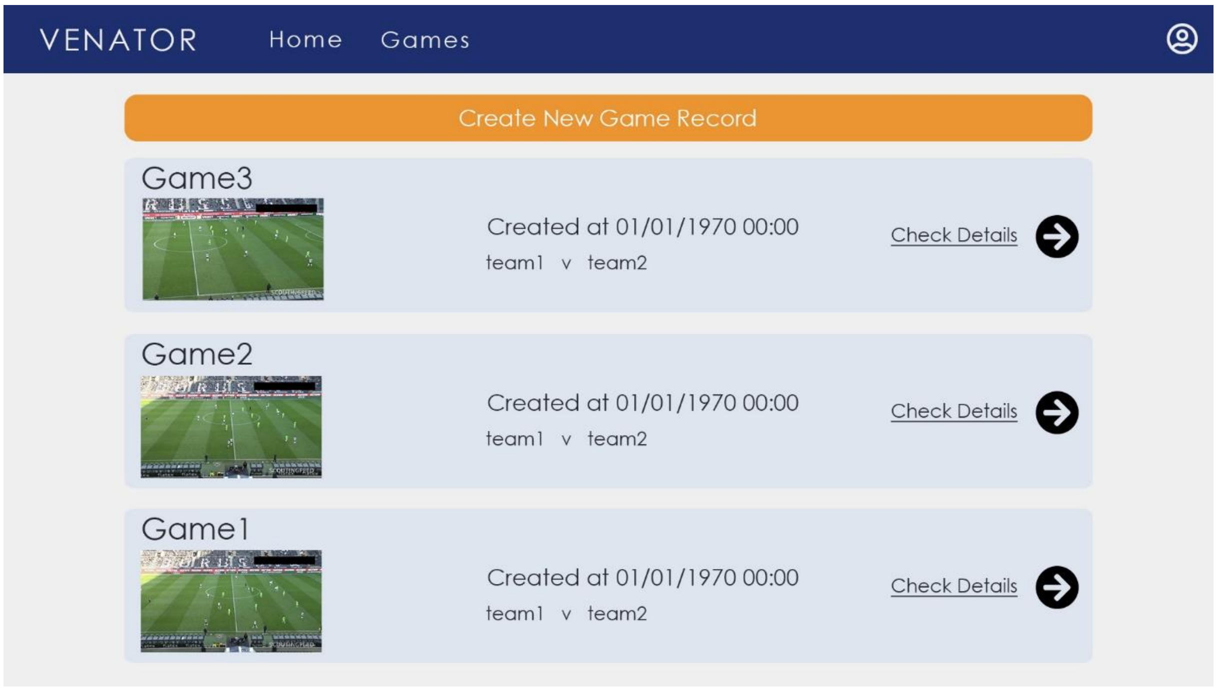
team1 v team2

Ball Possesion
52% 48%

Coverage
11.8 km 12.2 km

Check Details

Figure 21: Home Page


VENATOR Home Games

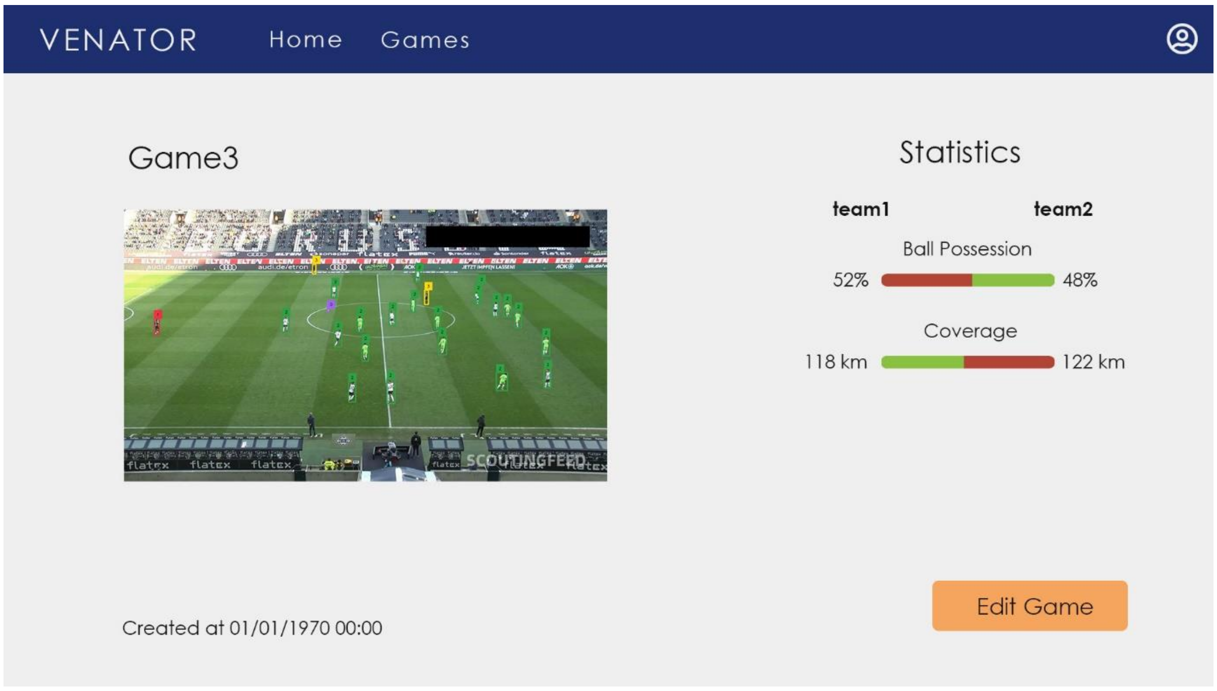
Create New Game Record

Game3
Created at 01/01/1970 00:00
team1 v team2
Check Details

Game2
Created at 01/01/1970 00:00
team1 v team2
Check Details

Game1
Created at 01/01/1970 00:00
team1 v team2
Check Details

Figure 22: Games Page



Game3

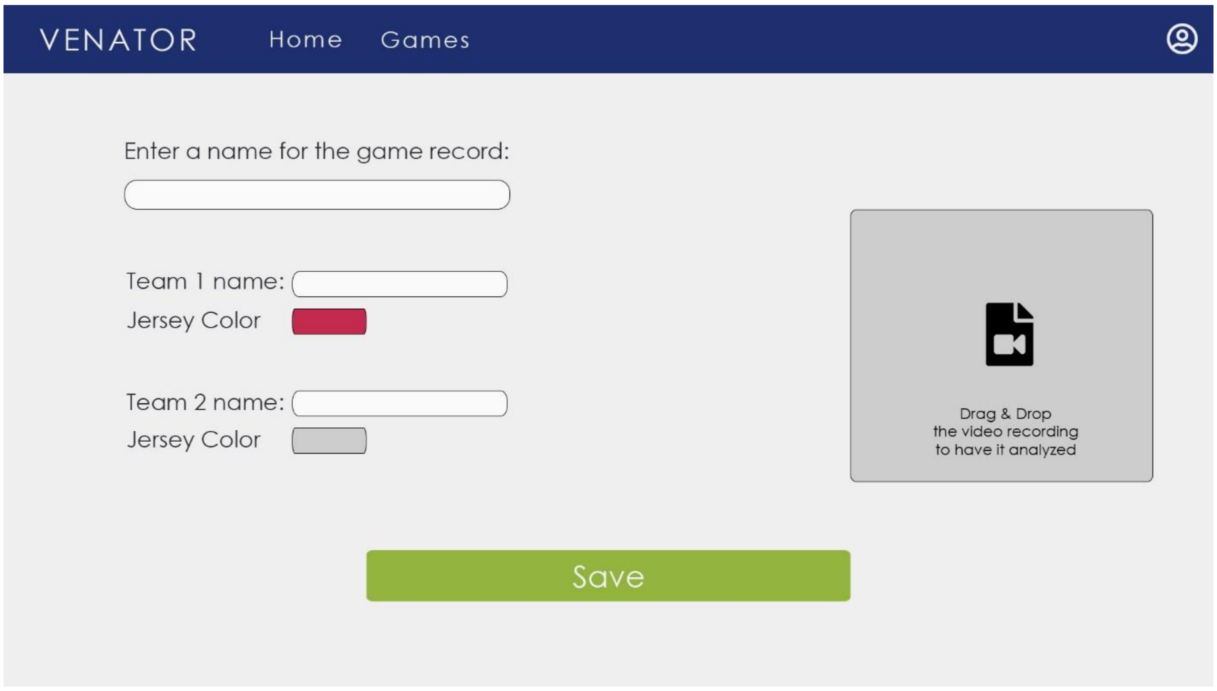
[redacted]

Statistics

team1		team2
	Ball Possession	
52%		48%
	Coverage	
118 km		122 km

Created at 01/01/1970 00:00

Figure 23: Viewing a past game & its statistics



Enter a name for the game record:

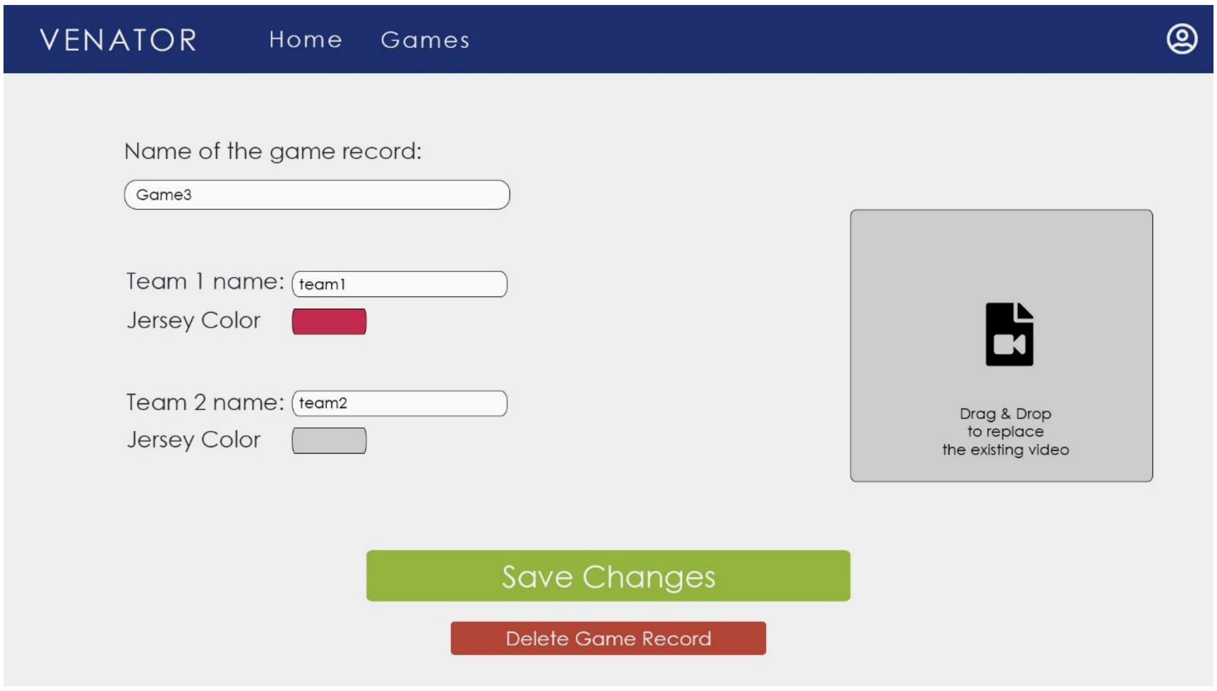
Team 1 name:

Jersey Color 🟥

Team 2 name:

Jersey Color ⬜

Figure 24: Creating a new game record


VENATOR Home Games

Name of the game record: Game3

Team 1 name: team1
Jersey Color 🟥

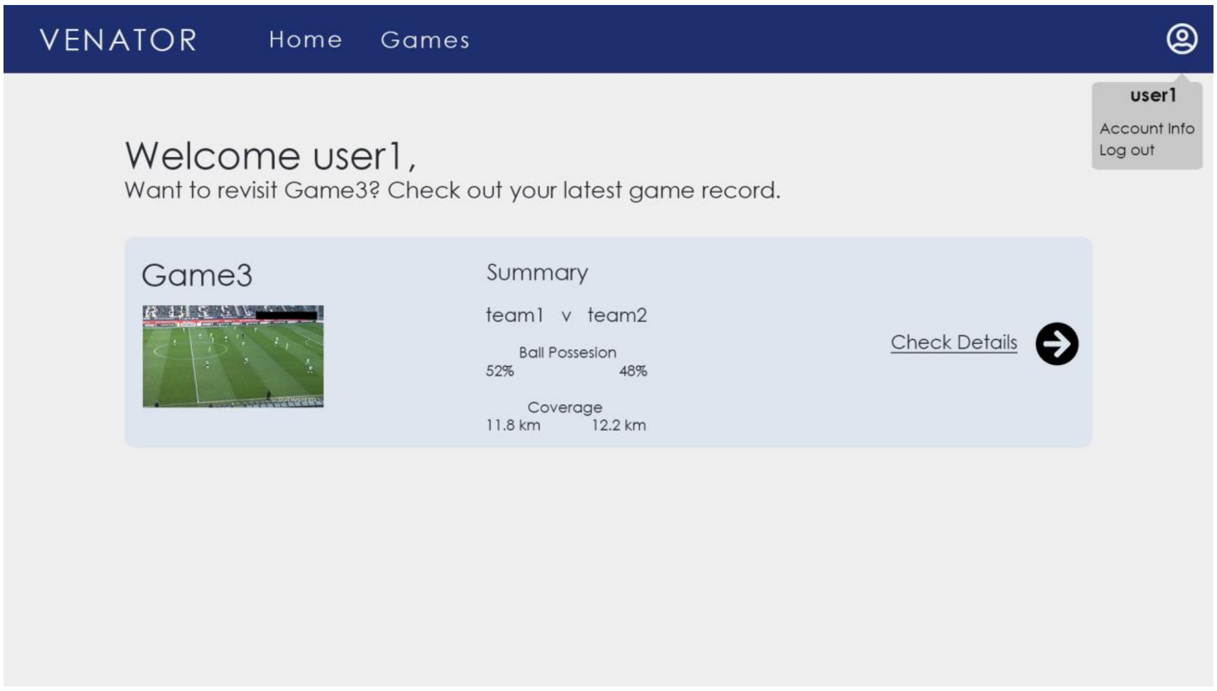
Team 2 name: team2
Jersey Color ⬜

Drag & Drop
to replace
the existing video

Save Changes

Delete Game Record

Figure 25: Editing a game record


VENATOR Home Games

user1
Account Info
Log out

Welcome user1,
Want to revisit Game3? Check out your latest game record.

Game3

Summary

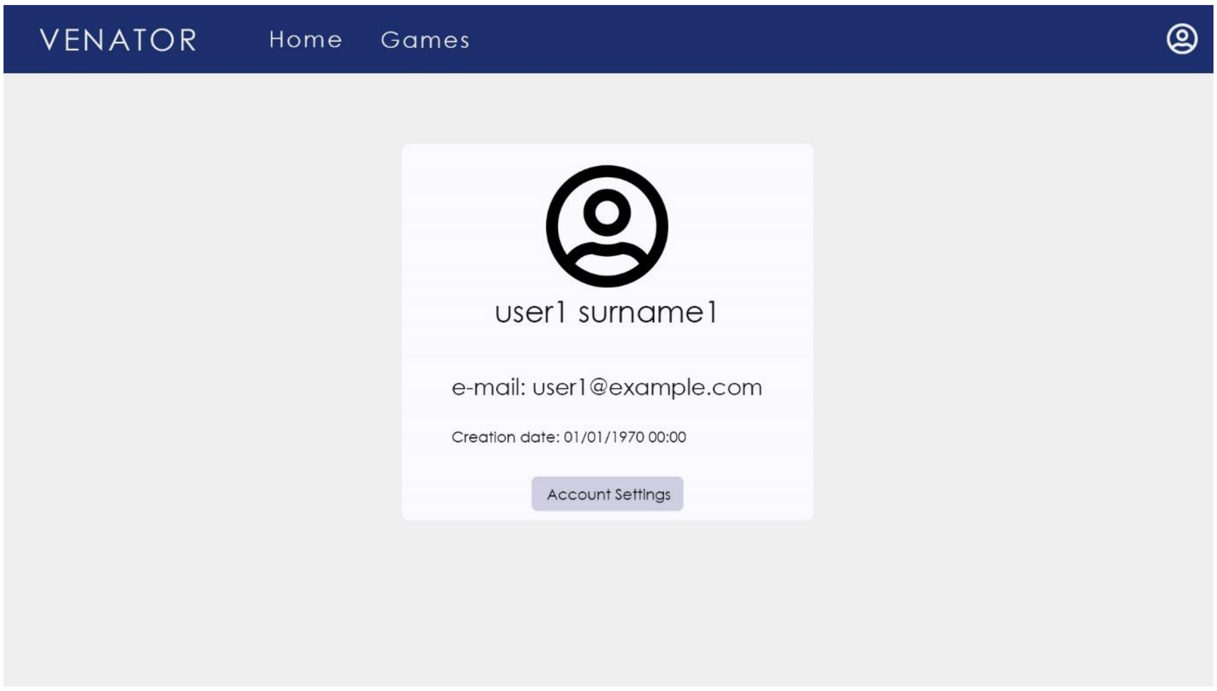
team1 v team2

Ball Possesion
52% 48%

Coverage
11.8 km 12.2 km

Check Details

Figure 26: Navigating to Account Information Page

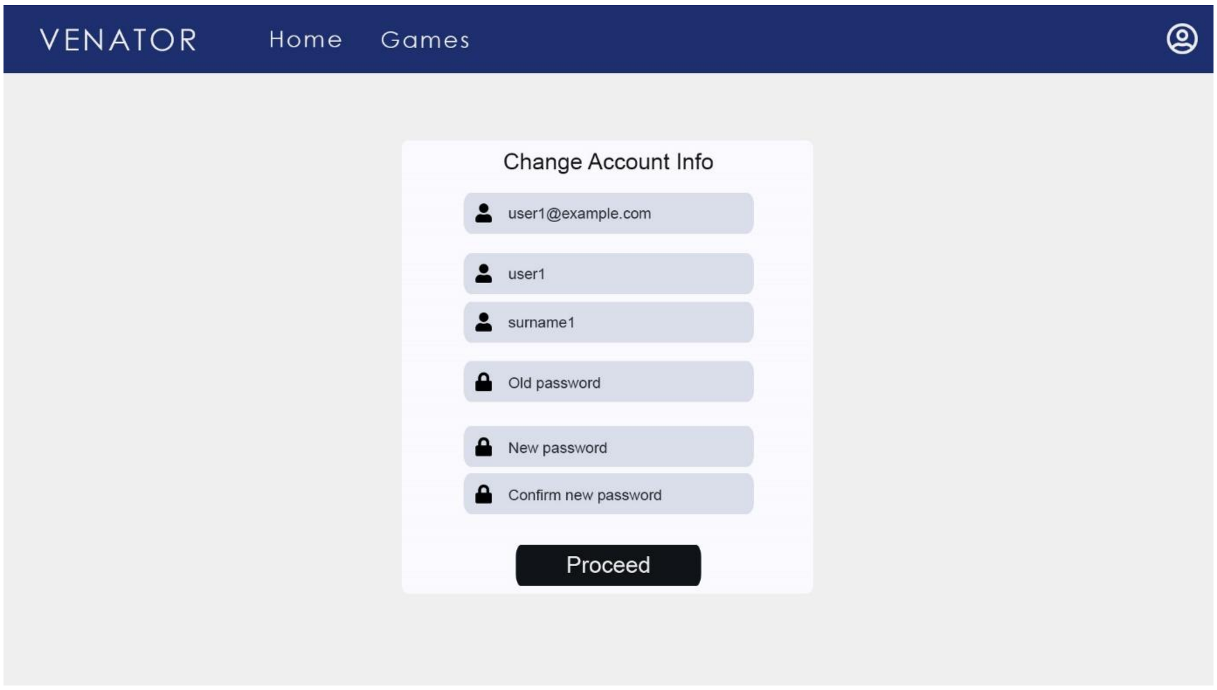


user1 surname1

e-mail: user1@example.com

Creation date: 01/01/1970 00:00

Figure 27: Viewing the Account Information Page



Change Account Info

user1@example.com

user1

surname1

Old password

New password

Confirm new password

Figure 28: Editing the Account Information Page

4 Glossary

- Detection: Identifying and locating objects within an image frame or a video recording.
- Tracking: Finding an object's location on the footage or in real time.
- Mean Average Precision: Measures the performance of an object detection model with the help of precision values.
- Confusion Matrix: Visualizes the assessment of performance values and/or accuracies in a table.
- Computer Vision Techniques: Interprets visual data from an image or a video recording with the help of different processing methods and algorithms.
- Encrypted Access Control Mechanism: Provides a secure environment for storing and retrieving data, by implementing encrypted authentication and verification processes.

5 References

1. *ACM Code of Ethics and Professional Conduct*. (n.d.). Association for Computing Machinery. https://www.acm.org/code-of-ethics
2. *IEEE Code of Ethics*. (n.d.-b). Institute of Electrical and Electronics Engineers. https://www.ieee.org/about/corporate/governance/p7-8.html
3. *Key words for use in RFCs to Indicate Requirement Levels*. IETF. (n.d.). https://www.ietf.org/rfc/rfc2119.txt
4. Myers, B. (n.d.). *Requirements Specification*. Requirements specification. https://www.cs.fsu.edu/~myers/cop3331/notes/requirements.html