# 1.0. Introduction

## 1.1. Summary

The software JOGU is a mobile platform where football amateur players can share experiences. The users of this application can create football games, called events, and invite other users to join and create a game. This app permit to share live goals, attendance and results of each event. Each player can also check past games, goals and statistics.

## 1.2. Goals

This software is a platform of share experience of the amateur football players. The user can create events. To each event, other users can be invited and attend. After the users confirm that they will attend the event, the teams are made. All the participants of the event can share stats of the event. The creator of the event is set as an administrator of that event and can overwrite any stat publish by the attendees. The administrator can attribute administrator’s permissions to other users on the event.

Each user can connect to application through login and access their own stats.

This application is targeting mobile devices (Android and IOS). To publish stats the application needs access to internet but also will permit to consult the stats offline.

## 1.3. Goals And/or diagram

## 1.4. Glossary

|  |  |
| --- | --- |
| **Term** | **Definition** |
| User | Any amateur football player that use the application to create events, attend to events or check stats. |
| Event | Is a meeting created by the users which users can attend with the information about the time and the place where the amateur football game will be taking place. |
| Administrator | In case of incorrect stats, the administrator is the user with the permission to alter the stats publish by the users. |
| Super Administrator | Person of developing team charged to intervene in case of failure. |
| Stats | Published information that can be goals, attendance or final score of the event |
| Team | At event, the participants are divided in to groups. |
| Invitation | An invitation send by the creator of the event. The users can accept this invitation |
| Statistics | Information that the user can access about the events, goals, results and attendance of past events. |

## 1.5. References

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

## 1.6. Stakeholders

The relevant stakeholder of this application are the product owner, the group of developers and the people that will use the app.

# 2.0. Overall Description

## 2.1 System Environment

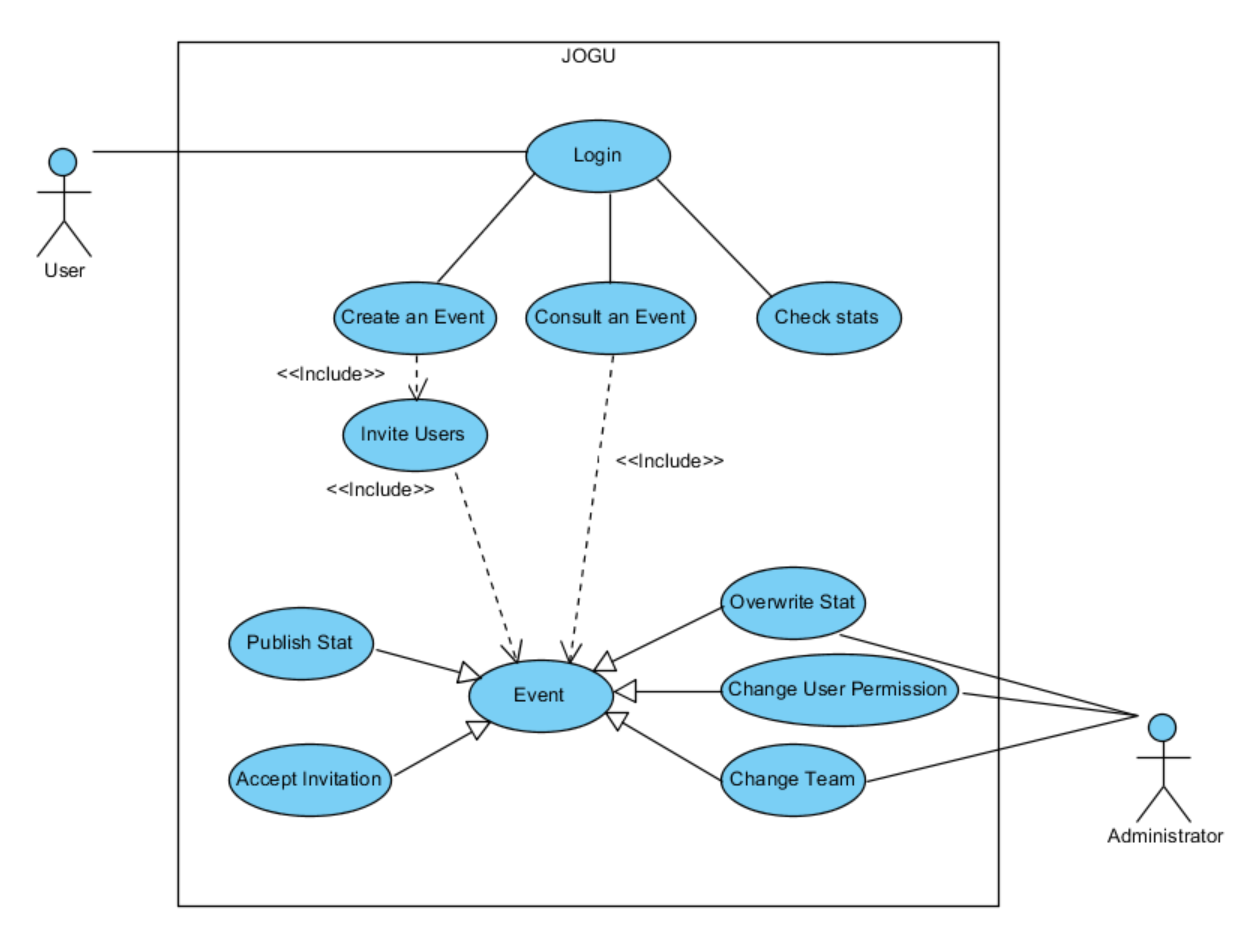


Figure - System Environment

## 2.2 Actors and roles

The JOGU has two active actors.

The User and the Administrator. The Administrator is a user with more permissions. Any user can only access the applications through login. The login can be by email and password or using the social session of Google or Facebook.

The user is expected to create and consult events, check results and publish stats.

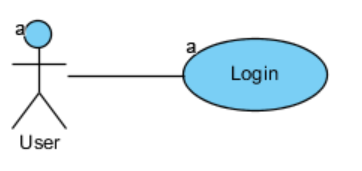
The administrator is expected to change any incorrect stat of an event. By default, user that creates an event is automatically assign as its administrator. The administrator can grant permissions to other users in the event.

## 2.3 Functional Requirements Specification

### 2.3.1 Login use case

#### Use case: The user can login in system to access the events and stats

**Diagram:**



**Brief Description**

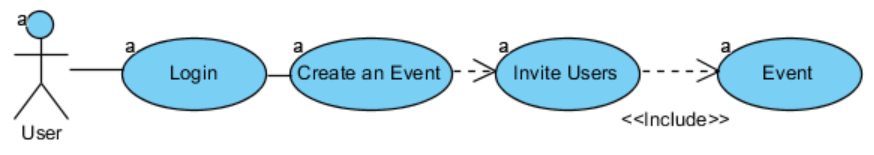
The user first interaction with the application is the login with the system. This login can be made by Email and Password or Google account or Facebook account. In case the user is not registered, then an email is sent to confirm his identity.

|  |  |
| --- | --- |
| **Use Case Name** | The user can login in system to access the events and stats |
| **XRef** | Section 2.3.1 |
| **Trigger** | The user opens application |
| **Precondition** | The user has not yet logon on application |
| **Basic Path** | 1. The user opens the application 2. The system displays login by email, Google or Facebook. 3. The user has to choose the login type and enter the credentials. 4. The system will check if the credentials are correct 5. The main window of the app is shown |
| **Alternative Paths** | In step 2., if user already logon on app go to 5.  In step 4., if is the first login of the user, the system will send a verification email to user.   * 1. System hangs until confirmation of email and proceeds to 5. |
| **Postcondition** | The user credentials are registered in system. |
| **Exception Paths** | In case of no internet, the system message user that cannot proceed without internet |
| **Other** |  |

### 2.3.2 Create an event use case

#### Use case: The user can create events to invite users

**Diagram:**



**Brief Description**

The user creates an event with date and location and invite other users to attend the event.

|  |  |
| --- | --- |
| **Use Case Name** | The user can create events to invite users |
| **XRef** | Section 2.3.2 |
| **Trigger** | The user selects the option “Create an event” |
| **Precondition** | The user has already logon on system |
| **Basic Path** | 1. The system ask user to date and confirmation of the event 2. The system verifies information and shows the invite window 3. The user selects the users to invite (from past events) or inputs the other users email. 4. The system notifies the invited users |
| **Alternative Paths** |  |
| **Postcondition** | The user credentials are registered in system. |
| **Exception Paths** | In case of no internet, the system message user that cannot proceed without internet |
| **Other** |  |

### 2.3.3 Consult an event use case

### 2.3.4 Accepts invitation use case

### 2.3.5 Publish stat use case

### 2.3.6 Overwrite stat use case

### 2.3.7 Change user permission use case

### 2.3.8 Change team use case

### 2.3.9 Check stats use case

## 2.4 Non-Functional Requirements

Balla asdn kasld kajsdl kajsç dlkjsdfkjasdf

Asd flaksjd fçalksjd f

A sdflkjasdflkjasdf

## 2.5 Scenarios

### 2.5.1 User creates an event Scenario

### 2.5.2 User consult and publish Scenario

### 2.5.3 User check stats Scenario