The International Football Association Portal

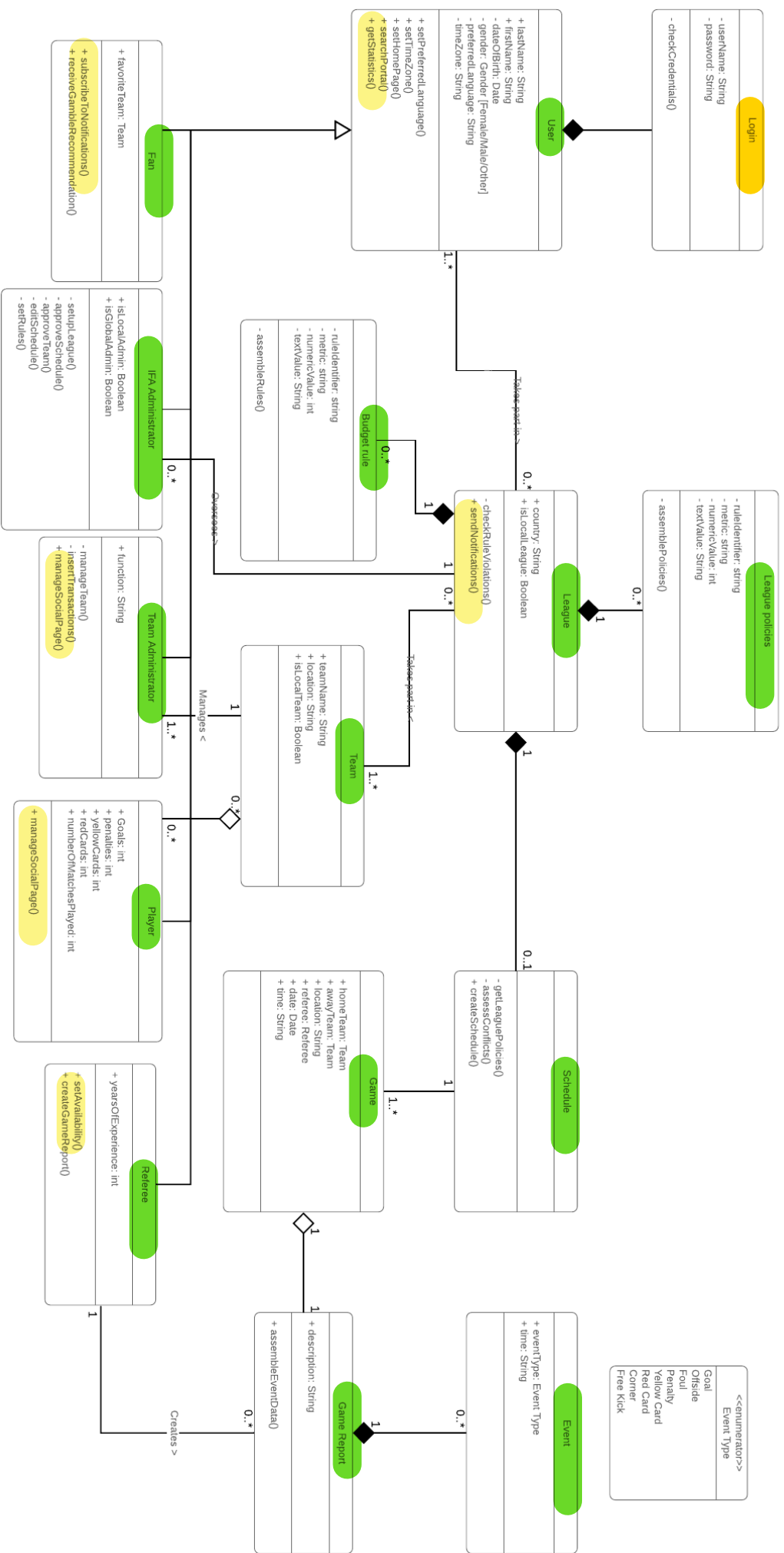## 1. Conceptual Model: UML Class Diagram

The UML Class Diagram for the IFA Portal is visualized on the next page. It shows the domain entities and relationships between these entities for the IFA Portal, including the attributes, functions, and cardinalities. The derivation process for this model is as follows:

All elements of the UML Diagram are derived from the Use Cases that are shown in Appendix A. For each use case, I would simply run through all the elements of the Use Case and imagine what classes would be necessary for the system to work. Obviously, any nouns in the name of the Use Cases or any object that would be created in a Use Case would refer to an entity that needs to be included in the Class Diagram. An important element to look at was the Primary Actors of the Use Cases. Since these would all be users with different roles, it was clear that there would be inheritance for the different types of users. Because all of these User subtypes have some kind of interaction with other entities, they are required to be modeled as their own class, instead of just having a '*Type*' attribute on the *User* entity.

Then, for each of these Users, the Main Success Scenario of the Use Cases for these users was analyzed to identify any relationship with other entities. Whenever a certain user type would e.g. create a report, this would indicate that there is a relationship between this report and the user type. After doing this for every user, I felt confident that I had all the essential entities for the system.

Similarly, any interaction or functionality that these entities have would give you a direction for the attributes and functions of the entities. For each entity, I ran through the Main Success Scenarios and the Special Requirements to identify any important attribute or function that is necessary to display in the model. Next to that, some attributes were added without reference to the Use Cases, but simply by thinking of what would be necessary for a particular entity.

The cardinalities of relationships between the entities and association types were also derived from mostly my own thought process after all entities, relationships, attributes, and functions were identified.

Login
- userName: String
- password: String
- checkCredentials()

User
+ lastName: String
+ firstName: String
- dateOfBirth: Date
- gender: Gender [Female/Male/Other]
- preferredLanguage: String
- timeZone: String
+ setPreferredLanguage()
+ setTimeZone()
+ setHomePage()
+ searchPortal()
+ getStatistics()

Fan
+ favoriteTeam: Team
+ subscribeToNotifications()
+ receiveGambleRecommendation()

IFA Administrator
+ isLocalAdmin: Boolean
+ isGlobalAdmin: Boolean
- setupLeague()
- approveSchedule()
- approveTeam()
- editSchedule()
- setRules()

Budget rule
- ruleIdentifier: string
- metric: string
- numericValue: int
- textValue: String
- assembleRules()

League
+ country: String
+ isLocalLeague: Boolean
- checkRuleViolations()
+ sendNotifications()

League policies
- ruleIdentifier: string
- metric: string
- numericValue: int
- textValue: String
- assemblePolicies()

Team Administrator
+ function: String
- manageTeam()
- insertTransactions()
+ manageSocialPage()

Team
+ teamName: String
+ location: String
+ isLocalTeam: Boolean

Player
+ Goals: int
+ penalties: int
+ yellowCards: int
+ redCards: int
+ numberOfMatchesPlayed: int
+ manageSocialPage()

Schedule
+ getLeaguePolicies()
- assessConflicts()
+ createSchedule()

Game
+ homeTeam: Team
+ awayTeam: Team
+ location: String
+ referee: Referee
+ date: Date
+ time: String

Referee
+ yearsOfExperience: int
+ setAvailability()
+ createGameReport()

Event
+ eventType: Event Type
+ time: String

Game Report
+ description: String
+ assembleEventData()

<<enumerator>>
Event Type
Goal
Offside
Foul
Penalty
Yellow Card
Red Card
Corner
Free Kick

Takes part in >

Manages <

Oversees <

Creates >

1..*   0..*   1   0..*   1   0,1   1   1..*   1..*   0..*   0..*   1

Use Case UC1: Log into application

Scope: IFA Portal
Level: Subfunction (Precondition for all other Use Cases)
Primary Actor: Any user
Preconditions: User has an account with appropriate role(s)
Main Success Scenario:
1. User enters his/her user name
2. User enters his/her password
3. System checks if the given credentials are valid
4. System creates a new session with appropriate roles to the user
5. User is redirected to the home page
Alternate flows:
3a. User provides invalid credentials
       1. System redirects user to Login page
       2. System informs user that he/she has entered the wrong credentials
Special Requirements:
- Users can either log into a web-based portal or a mobile application
- At any time, 50.000 people should be able to access the system
- Home page for fans should be a feed for notifications and information on their subscribed events/teams/players
- Home page for IFA administration should be a dashboard for checking on leagues / teams and budget auditing
- Home page for players and referees should be an overview of their next matches and information
- Home page for teams should be a dashboard of transactions and overview of scheduled matches

Use Case UC2: Indicate Availability

Scope: IFA Portal
Level: User goal
Primary Actor: Referee
Preconditions: Local League has been created in the system and games have not yet been scheduled
Main Success Scenario:
1. System notifies referee that his availability has to be entered
2. Referee opens his personal schedule
3. Referee indicates the days and times of availability
4. Referee indicates his preferences for times and locations
5. System stores referee's availability and preferences
Special Requirements:
- Local IFA adminstrations can determine how often a referee requires to be available to ensure that there's always a referee for a match

Use Case UC3: Determine budget rules and policies

Scope: IFA Portal
Level: User goal
Primary Actor: IFA Administration member
Preconditions: IFA Administrators have been identified
Main Success Scenario:

1. System notifies IFA administration members that budget rules and league policies are required
2. IFA administration member opens the rule and policy builder
3. IFA administration member defines the budget rules and its boundaries
4. IFA administration member defines the league's policies
5. IFA administration member indicates when to receive notifications on violations
6. System saves the rules and policies for the locally defined league

Alternative flows:

1a. Budget rules and league policies are already in place

       1. IFA administration member can edit rules if the season has not yet started

Special requirements:

- Local League IFA administrations determine policies and rules for their own national league
- Global League IFA administrations determine policies and rules for international leagues
- Global IFA administration can oversee activities over all leagues
- A league can define it's own policies on scheduling, positioning of teams based on its achievements
- System will provide a 'building-block' interface for determining the rules, based on metrics and values.

Use Case UC4: Creation of league schedule

Scope: IFA Portal
Level: User goal
Primary Actor: IFA Administration member
Preconditions: League policies have been defined. Teams are in the system.
Main Success Scenario:

1. IFA Administration member starts the automatic scheduling at the beginning of a season
2. System suggests a certain schedule considering all preconditions
3. IFA administration member approves the schedule
4. System will schedule the matches

Alternative flows:

2a. System is unable to come up with a suggestion

       1. System will assess the conflicts

       2. System will notify relevant users on required changes for resolving these conflicts

       3. Relevant system users will resolve conflicts

       4. IFA administration member restarts automatic scheduling

3a. IFA Administration member disapproves the suggested schedule

       1. IFA Administration member will manually change the suggested schedule

       2. Another IFA Administration member confirms the change

Special Requirements:

- The relevant teams, players, and referees should receive an update on the created schedule and whenever there are changes
- The IFA administration will be able to manually cancel / make changes to a match

Use Case UC5: Writing game report

Scope: IFA Portal
Level: User goal
Primary Actor: Referee
Preconditions: Games are scheduled.
Main Success Scenario:
1. Referees indicate who will be responsible for inserting data during match
2. Responsible referee will start the game-report during the match
3. In case of an event, the referee indicates what event occurred
4. System will record the selected event with timestamp.
5. Referee will indicate game has been finished when the game is over.
6. Referee will write down additional information.
7. Referee will save the game report.
8. System saves the game report and produces statistics based on the recorded events.

Special Requirements:
- Mobile application is required for the event registration
- Mobile application should provide a clear UI in the mobile application for the event registration
- Referees should receive priority access to the system during the matches they write game reports
- Reporting will be made possible offline and data of the match could potentially be sent after the match whenever there is a stable internet connection
- The list of events should at least include: Red/yellow card, offside, goal, injury, penalty, foul, corner, free-kick.

Use Case UC6: Setting up a team

Scope: IFA Portal
Level: User goal
Primary Actor: Team administrator
Preconditions: League has been set up and rules have been established.
Main Success Scenario:
1. Team administrator initiates creation of new team entry
2. Team administrator enters general information on team and team staff
3. Team administrator enters information of the team's players
4. Team administrator confirms and saves the entered data.
5. IFA administration member approves the team's data and credentials
6. System will create team social network page
7. System will create individual player's social network pages
8. Team information has been saved and published.

Alternative Flows:
5a. IFA disapproves team's data
       1. IFA administration member provides feedback on conflicts
       2. Team administrator resolves data conflicts
       3. IFA administration member check and approves new data

Special Requirements:
- System should offer ways for teams to input their expenses and incomes
- Players should be able to manage their own social network page
- Teams can indicate which persons are allowed to manage the team's data in the system
- Each transaction made by teams will be reported within the system to allow more transparency and budget auditing

Use Case UC7: Subscribing to information

Scope: IFA Portal
Level: User goal
Primary Actor: Fan
Preconditions: Team and player data is available.
Main Success Scenario:

1. Fan queries particular team, player, or match
2. Fan follows the page of this team/player of follow the upcoming match
3. Fan chooses to which events he/she wants to subscribe to when they follow something
4. System saves the fan's preferences.
5. System will provide notifications based on changes and events in teams/players and matches, based on the events a fan has subscribed to.
6. Fan receives this notification and can respond to events by liking or commenting on them.

Special Requirements:

- Fans should be able to query all available data on teams, players and matches in the system.
- Fans should be able to retrieve statistics on the teams, players, and matches in the system.
- Fans should be able to retrieve a gambling recommendation for a match based on system statistics

Use Case UC8: Change application settings

Scope: IFA Portal
Level: Subfunction
Primary Actor: Any user
Main Success Scenario:

1. User goes to his settings menu
2. User indicates preferred language
3. User indicates his/her time-zone
4. User indicates preferred currency indication
5. System saves user's preferences.
6. System adjusts UI based on user preferences.

Special Requirements:

- The complete interface should switch left/right – right/left, based on the writing direction of the language that has been set
- The time-zone should be taken into account whenever notifications that include a time indication are sent to users.