



**Qatar University**

**College of Engineering**

**Department of Computer Science and Engineering**

# **Senior Project Report**

***Sports community***

**Project Group Members:**

Elbaraa Elhawary (201703540)

Mahmoud Gazalh (201405637)

Mohamed Khattab (201707188)

**Supervisor:** Prof. Mohammad Saleh

**2022**

This project report is submitted to the Department of Computer Science and Engineering of Qatar University in partial fulfillment of the requirements of the Senior Project course.

## Declaration

This report has not been submitted for any other degree at this or any other University. It is solely our work except where cited in the text or the Acknowledgements page. It describes work carried out by us for the senior project. We are aware of the university policy on plagiarism and the associated penalties, and we declare that this report is the product of our work.

**Student:** Elbaraa Elhawary

**Date:** 2022/4/7

**Signature:** البراء الهوارى

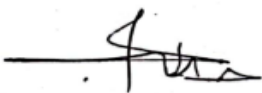
**Student:** Mahmoud Gazalh

**Date:** 2022/4/7

**Signature:** 

**Student:** Mohamed Khattab

**Date:** 2022/4/7

**Signature:** 

## Abstract

The motivation for us to create the Sports community application and plan for it is to help people keep their health and increase awareness of the importance of practicing sports. Also, we aim to generate revenue through the application for us and the stakeholder. In our project, we aim to make finding the closest training partner or teammate easier for anyone seeking to practice any sport and find the most immediate appropriate sports organizations and facilities to practice a particular sport. This app targets a vast sector of the society as we are gathering many sports services providers in one app to present and propose their services and make it easier for any person who is passionate about any sport to find a teammate who is interested in the same sport. Our app is an entrepreneurship idea that enables us to make a profit by collaborating with business owners who may find it helpful for them to subscribe to our service. It benefits us by having a commission on every reservation a user reserves with the facility owner. Our app will be developed using the most recent technologies, and we are going to provide the user with a friendly user interface that gives the best user experience. However, we may face some problems at the beginning of distributing that app among the athletes and facilities providers; we will use all the advertisement tools available to make it reach the highest number of users in the shortest time possible. The application also enhances the concept of competition by making users able to rank each other depending on their performance and how recommended is it to have a person in a team; this gives a chance to every passionate person to present their skills and has an evaluation from other users about their athletic performance, it also makes it easier for commercial clubs to find great athletes who other athletes recommend. We may also include some tutorials and advice for users about a particular sport and how to improve their performance and schedule time to practice. Our app is a great idea to increase society's awareness of the importance of sports, the role of sport in social gatherings, and the importance of technology to facilitate people's lives.

## Acknowledgment

This project is a result of dedicated work and the actual contribution of all the team members; we would like to acknowledge every person who helped complete this project. Also, we would like to recognize all computer science and Engineering department professors who helped us throughout our educational path to reach this level of excellence that enabled us to create this project, including our supervisor Dr. Mohammed Saleh.

## Learn and applied knowledge

We have learned a lot of soft and hard skills. We've learned how to plan precisely before implementing, which motivated us to use many projects management tools and team working software. Furthermore, we have boosted our programming and software engineering skills by using famous programming platforms such as android studio and the new programming language kotlin. Moreover, we have planned the project structure using the UC diagram, and we have surveyed people to collect data using google forms. In addition, we have learned different types of high-level architecture and when to use some of them.

# Table of Contents

Declaration.....	ii
Abstract.....	iii
Acknowledgment .....	iv
Learn and applied knowledge .....	iv
List of Figures .....	vi
List of tables .....	vii
1. Introduction and Motivation .....	8
.1.1 Problem statement .....	8
.1.2 Project significance .....	9
1.3. Project objectives.....	9
1.4. Market Research and Business Viability .....	9
2. Background and related work.....	11
2.1. Background .....	11
2.2. Related work .....	12
3. Requirement's analysis .....	24
3.1. Software development process .....	24
3.2. Functional requirements.....	25
3.3. Non-functional requirements .....	27
3.4. Assumptions.....	28
3.5. Ethics .....	28
4. Solution Design .....	29
4.1. Overview .....	29
4.2. Architecture Diagram.....	30
4.3. Structural model .....	31
4.4. Behavioral model .....	32
4.5. Database design .....	40
4.6. User interface design .....	41
4.7. Design patterns .....	45
4.8. Hardware/software to be used.....	45
5. Implementation .....	47
6. Testing.....	48
7. Impact of your project solution .....	53

8. Conclusion.....	53
9. Future work.....	54
10. Student reflections.....	54
References .....	55
Appendix A – Project plan.....	56
A.1. Project milestones .....	56
A.2. Project timeline .....	59
A.3. Anticipated risks .....	61
Appendices B.....	62
Appendices C.....	70

## List of Figures

Figure 1. Model-view view- model .....	11
Figure 4. UI الرياضة للجميع .....	12
Figure 5. Malaeb UI.....	13
Figure 6. Meet Up UI.....	14
Figure 7. QSFA UI .....	15
Figure 8. Sporty UI .....	16
Figure 9. Find Me Football UI.....	17
Figure 10. Sport Easy UI .....	18
Figure 11. Padel Qatar UI.....	19
Figure 12. WhatsApp UI .....	20
Figure 13. Agile development process.....	24
Figure 14. use case diagram.....	25
Figure 15. MVVM .....	30
Figure 16. Class Diagram.....	31
Figure 17. Activity Diagram.....	32
Figure 18. Sequence Diagram of addFriend.....	33
Figure 19. Sequence Diagram of addVenue.....	33
Figure 20. Sequence Diagram of Sign Up.....	34
Figure 21. Sequence Diagram of addTeams .....	34
Figure 22. Sequence Diagram of notifications.....	35
Figure 23. Sequence Diagram of ShowFriends .....	36
Figure 24. Sequence Diagram of Sign in .....	37
Figure 25. Sequence Diagram of ShowTeams.....	37
Figure 26. Sequence Diagram of ShowVenueDetails.....	38
Figure 27. Sequence Diagram of ShowProfile.....	38
Figure 28. Package Diagram.....	39
Figure 29. Database Entity Relation Diagram .....	40

Figure 30. Sign In and Sign-Up Interface .....	41
Figure 31. Normal User Interface(Home, Drawer, edit profile) .....	41
Figure 32. Normal User Interface(ShowFriends, addFriend, ShowRequestes).....	42
Figure 33. Normal User Interface(showVenueDetails, bookVenue).....	42
Figure 34. Normal User Interface(ShowEvent, ShowTeams, CreateTeam) .....	43
Figure 35. Normal User Interface(CreateEvent, ShowEventDetals) .....	43
Figure 36. Venue Owner User Interface(ShowVenues, addVenue, addImage).....	44
Figure 37. Firebase .....	44
Figure 38. Design Patterns Diagram.....	45
Figure 39. Unit Test .....	49
Figure 40. Activity Diagram .....	52

## List of tables

Table 1. Comparison Between Apps .....	21
Table 2. Use cases summary .....	26
Table 3. Non-functional requirements.....	27
Table 4. Ethics .....	28
Table 5. Anticipated risks .....	61
Table 6. Login .....	62
Table 7. Create account .....	62
Table 8. Create new Team .....	63
Table 9. Book Venue .....	63
Table 10. Show Venues .....	63
Table 11. Add Venues .....	64
Table 12. Complete Payment.....	65
Table 13. Show Friends .....	65
Table 14. Show Profile .....	66
Table 15. Push Notification .....	66
Table 16. Show suggested friends .....	67
Table 17. Show related events.....	67
Table 18. Show teams .....	68
Table 19. View nearest.....	68
Table 20. Make event.....	69
Table 21. Show venue details .....	69

## 1. Introduction and Motivation

Many people have some difficulties finding people with the same interest, especially in sports; it isn't easy to find people who like your favorite sport. So, our project offers a solution to this problem: gather those interested in the same mark in one place. This is very important for those interested in sports, as this project will remove a significant obstacle from their way.

### 1.1. Problem statement

Nowadays, many sports are played in groups, and forming a team is necessary to play a particular sport. Also, practicing sports in groups makes it more fun and motivational. However, sports enthusiasts' problems obstruct them from playing and practicing their favorite sports. Therefore, we aim by Sports community application to solve these problems. Firstly, sports enthusiasts, athletes, and even the general population find it difficult to practice their sport when their sports companions are busy or traveling.

Secondly, sports enthusiasts suffer to find appropriate places prepared to play a particular sport and all related information such as prices and available times.

Thirdly, many sports facilities exist but lack a unified platform to advertise themselves or do not advertise (like playgrounds in schools).

These facilities eventually could shut down due to the added maintenance expenses or the cost of running them. We also realized that the MENA region lacks applications that directly address the problems above; we came across attempts to rectify some of the problems, but these attempts either lack outreach or are inactive. There are also some technical and non-technical challenges to applying this solution. Firstly, one of the technical challenges in analyzing users' data is to match them without any latency.

Also, for the system to be scalable, it will require more resources, hardware, and network components, affecting other aspects of the project, such as the financial resources.

Secondly, the non-technical challenges will be required to convince the sport places owners to use our reservation system to gain profits and provide service to business owners and users.

Another challenge is choosing an algorithm that will give the distance between the user and the other users interested in the same activity and between users and the appropriate sports places.

Moreover, people may fear that they play sports with strangers and meet with them continually without any previous idea about them. These problems can be solved by adding a rating functionality about the user behavior by other users.

As for the issue of the considerable distance between users, it can be overcome by comparing and choosing the best algorithm which will provide the location of the online users and recommend them to the other online closet users who are interested in the same sport.



## 1.2. Project significance

The Sports Community Project is considered one of the most critical projects that can benefit the community. It is concerned with the people interested in sports, a large group in society. This group faces some problems, including the problem of finding people with the same interest in sports. Because there is no such functionality that facilitates gathering them, practicing their favorite sports become more complex. Also, it has a significant advantage for the sports owners' places, as it will create new revenue streams and will enhance the old revenue streams as the app will provide a chance to communicate with clients and to provide offers and advertise their events.

The sports community project will achieve great success in Qatar and the region due to the high percentage of youth interested in sports. And this project corresponds with the Qatar National Vision 2030 because sports are the essential pillars of Qatar Vision 2030 [1].

We are excited about this project because it shows that sport is essential to the individual and society. Because this project benefits us, we are football fans, which will make it easier for us to find people interested in football.

It will help us to gain profit from the stakeholders and advertisement. Also, it will help us to increase our programming experience and knowledge by implanting a real-world project; it also will give us the chance to work as a team and make the world a better place.

## 1.3. Project objectives

The project aims to allow sports enthusiasts to create and join the local sports community and to encourage them to practice their favorite sport. Also, to provide a platform that connects venues owner and sports enthusiasts.

The solution can be achieved through these objectives:

1. Design and implement an android application that can show all sports venues and their related information in Qatar.
2. Use a reliable database for data storage such as Firebase.
3. Take feedback from users to ensure that the application is usable and reliable.
4. Add features in the application which enable the users to create or join teams and events.
5. Create a booking system in the application which connects the venues' owners with the sports enthusiasts.

## 1.4. Market Research and Business Viability

### 1. Describe the market need and the market size.

The target group of this project is the sports amateurs in the community. The number of sports institutions is 275 sports institutions [12], and the number of sports practitioners is 26,527 per month[12]. The athletes are divided into less than 18, 18 to 25, 26 to 50, and more than 50 [12].

In this project, we need to reach 50% of the sports institutions, which will be a way to go 188000 sports amateurs [12], equivalent to 50% of the sports amateurs in the country.

**2. Identify the target customers and their demographic.**

- Sport amateurs over the age of 16 who live in Doha need to practice sports with others and practice their favorite sports weekly.
- Sport amateurs looking for groups to join in Doha and suffer from a lack of relations with those around them.
- Owners of sports facilities in big cities in Qatar want to extend their relationship with their customers and build channels with them.
- Owners of free space lands who want to earn money from it by hourly renting using our platform.

**3. Describe the competing products, and how does yours differ from that offered by competitors?**

- One app that provides similar services is meet up, where users can set a meeting date and place and a description of the meeting session activities. Our app is created specifically for athletes. It facilitates the searching and the filtering of the people and places where they can practice their activity.
- Also, there is an app issued by the ministry of sport that can be used to reserve and pay for the football yards reservation. Our app is different in that the users create accounts and get a recommendation by the people interested in the same also; every user has a ranking and rating by other teammates.

**4. Decide the price that can be charged for your product.**

- Commission from the stakeholders 15% per payment.
- Advertisements from sports companies.

**5. Describe the plan of what is required to bring your product to market? And make it competitive?**

- 1) We must market research to validate our assumption.
- 2) Develop our business model canvas.
- 3) Prototype (minimal viable product).
- 4) Launch our product on the app play store.
- 5) Make an advert campaign using social media and other advertising tools.
- 6) Take feedback from the users and launch updates continually.

## 2. Background and related work

### 2.1. Background

Our project is an entrepreneurship idea that aims to provide services that facilitate people's sports life when finding a teammate or sports facilities. It is more like a social app for people interested in sports. We aim to solve the problem of people who suffer to find places or team members. The core goal of the App is to help the people who would like to practice a particular sport to find training partners or teams who share an interest in the same sport and to find the appropriate places that are prepared with the equipment needed to practice a particular sport. Our App will contain some functionalities, such as searching for a proper sports facility that matches the users' sports interests. Moreover, we will provide the application with algorithms that enable the users to find the best team members to join their teams according to a ranking system that evaluates the user's performance in challenging spaces. This will enhance the spirit of competition between users because every user will do his best to achieve the highest version that will enable him to have a higher rank. One advantage of this process is that sports clubs can find athletes based on their ranking, so they can easily register new players in their teams so people will work harder to join famous sports clubs. Furthermore, we will automate the sports facilities reservation systems, where people can reserve places in their facilities online. We will deal with business owners who subscribe to our service by having a 15% commission on every reservation in their facility. Therefore, we will gain our profits besides the online advertisements that people may like to advertise their products. However, we faced some problems; for instance, there may be no appropriate places for the users to practice their sport. Another problem is implementing algorithms that match the user's interest according to the information he provides about himself to find the most appropriate teammate or training partner. Moreover, we faced some problems with the sports facilities owners who may not agree to install such a technology in their facilities. In addition, we should consider the system's soft constraints, such as timing constraints, and make sure there will be no scheduling conflicts or conflicts of interest between users.

#### **The Model-view view- model architecture design.**

The MVVM architecture design is one of the common design patterns used in implementing mobile applications. Since the mobile applications mainly consist of a user and interface and a functionality the needs programming logic to be implemented the MVVM architecture can be the perfect choice since it separates between the application UI design and the application programming logic. and there is a class that connect the UI with the updated data coming from any sort of data base temporary or permanent storage. So, there is a complete separation between the data source, the functions logic and the user UI which serves the purpose of maintainability as shown in Figure 1.

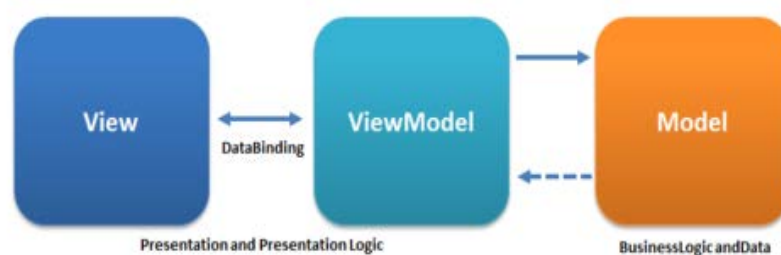


Figure 1. Model-view view- model

## 2.2. Related work

We have found similar apps to our app, but they do not exactly provide the same functionalities. Therefore, we have grasped the core ideas and modified them to serve sports amateurs. We analyzed similar arguments based on the number of downloads, features provided, Ui design, and more critical factors. We have organized them in Table 1. Below, there is a brief description of each similar idea.

### 1. الرياضة للجميع

In this application, you will learn about participation in the community, team and individual sports, and physical activities available for all fitness levels. Community sports group leaders can create and manage sports groups; These groups can be shared, and everyone is invited to participate in the activities and enjoy the experience of physical activity with people who have similar interests. The app offers helpful search and filter options and the best tips and information about healthy living. There is a "Support Request" feature to see how the Sports communities can provide support in the best possible way to achieve the goal of making sports ideas and events or physical activities available to everyone. The rewards and loyalty program motivates and increases levels of physical activity delivered through an enjoyable user experience. The app provides notifications about your interests and events [2].

#### - App UI.



Figure 4. الرياضة للجميع UI

## 2. Malaeb

Malaeb application makes it easy for you to browse the stadiums in your area and easily find the proper stadium for you, with more than 350 stadiums in the Gulf. View stadiums near your site and check available times. The facilities available at each stadium can also be viewed. You can join matches that happen in your area; you can add your game to the page and have the community join you [3].

### - App UI.

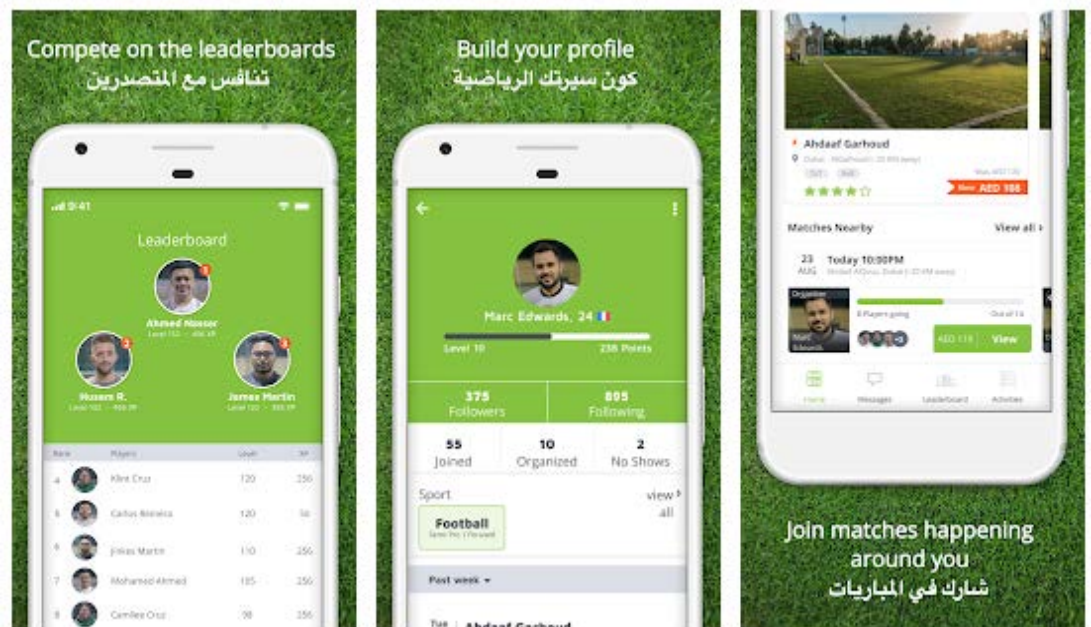


Figure 5. Malaeb UI

### 3. Meet Up

Meetup offers over 330,000 groups based on your interests. Explore events by category, search by keyword, or see what's popular in your area. Sports events you are interested in can be saved and reviewed again later. You can have discussions and use direct messages to keep in touch with the people you meet. Your events can be hosted online and in person by creating a group around your favorite topic. Easily customize and schedule events on the go and manage your group from anywhere. The app provides a way to communicate questions or suggestions via (androiddev@meetup.com). It also uses the exact location of the device (GPS and network-based) [4].

#### - App UI.

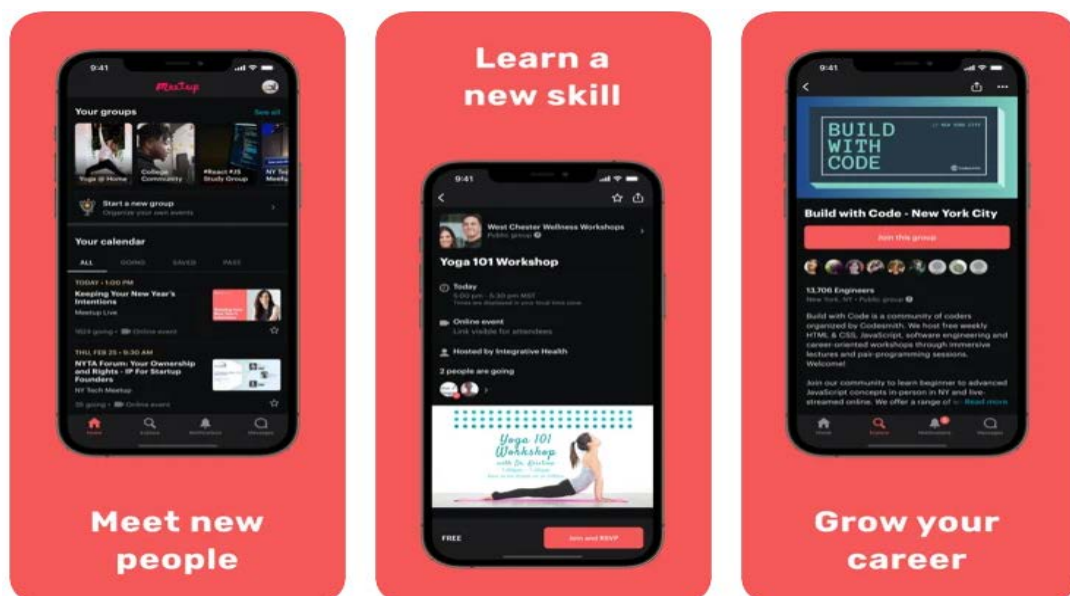


Figure 6. Meet Up UI



#### 4. QSFA

QSFA is an application in Qatar to find the best sites to exercise. The app provides an easy, interactive way to find directions and locations where you can exercise. The app is linked with the Fitbit App to display your daily activities, such as the number of steps and distance traveled [5].

##### - App UI.



Figure 7. QSFA UI

## 5. Sporty

This app helps you participate in sports events close to you, organize your sport courses enrollments, paying solutions, and it can work as a system for booking sports places. Furthermore, it is a ticketing solution for events, including QR codes to scan tickets and products [6].

### - App UI.

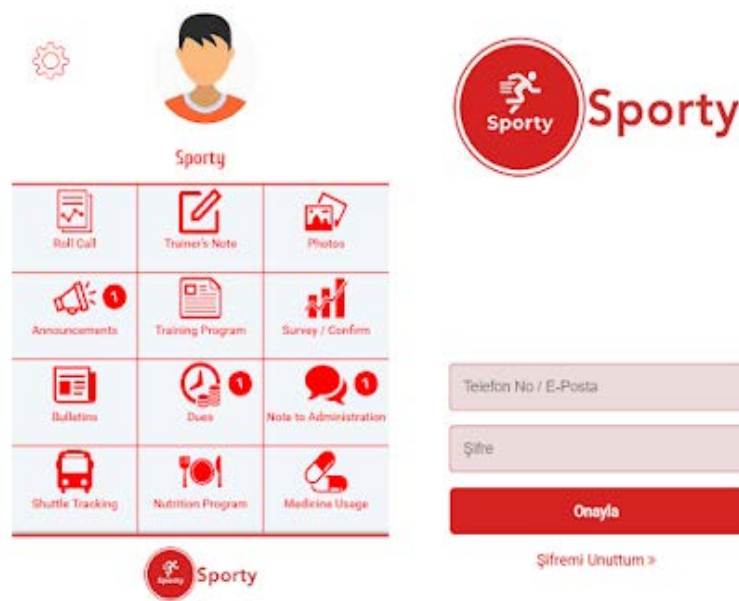


Figure 8. Sporty UI



## 6. Find Me Football

This app helps the user join football matches, play with new people, find information about football pitches, and add their football pitches if they own one. [7].

### - App UI.

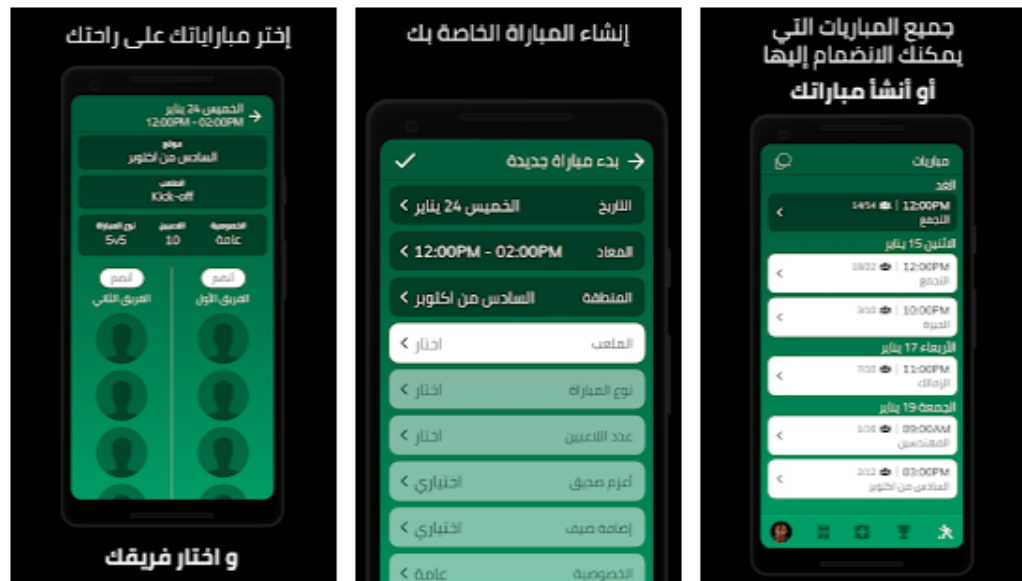


Figure 9. Find Me Football UI

## 7. Sport Easy

This app helps already organized and formed teams to manage their sports events and next training session by tracing the training attendance using attendance sheets, also by providing performance rating for each team member. But it's only an app for the formed team. It does not allow them to communicate with other groups or schedule matches [8].

### - App UI.

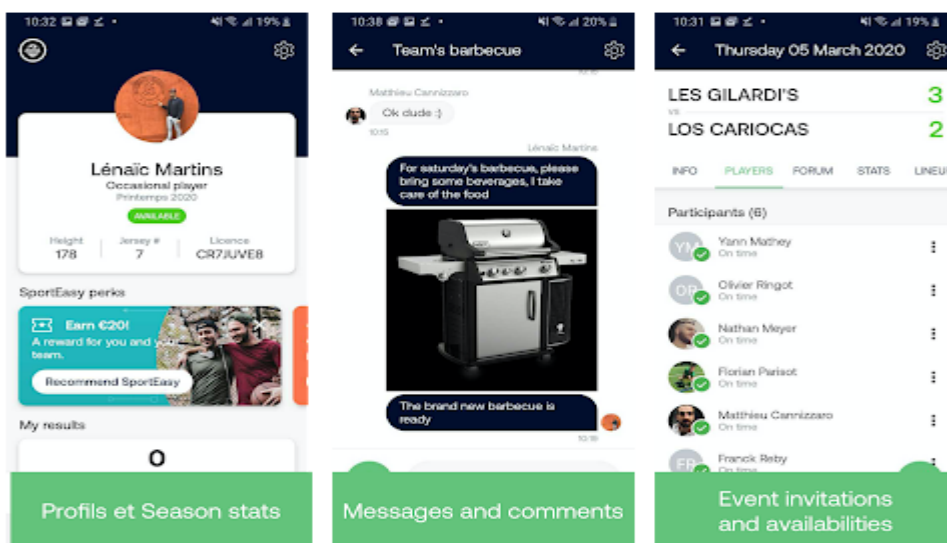


Figure 10. Sport Easy UI

## 8. Padel Qatar

This app focuses on how a player can book and schedule a Padel sports game and ranking system. So, it has a process that we need, like our app [9].

### - App UI.

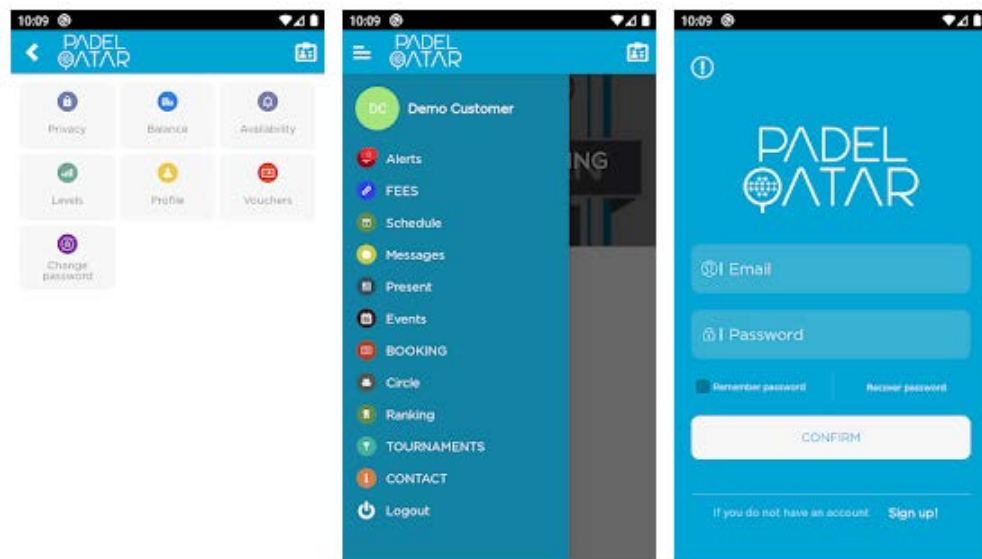


Figure 11. Padel Qatar UI

## 9. WhatsApp

WhatsApp Is a chatting app in the first place. However, some people create groups to organize their sports teams and schedule matches, which is a functionality that will be provided in our app when it is fully implemented [10].

### - App UI.

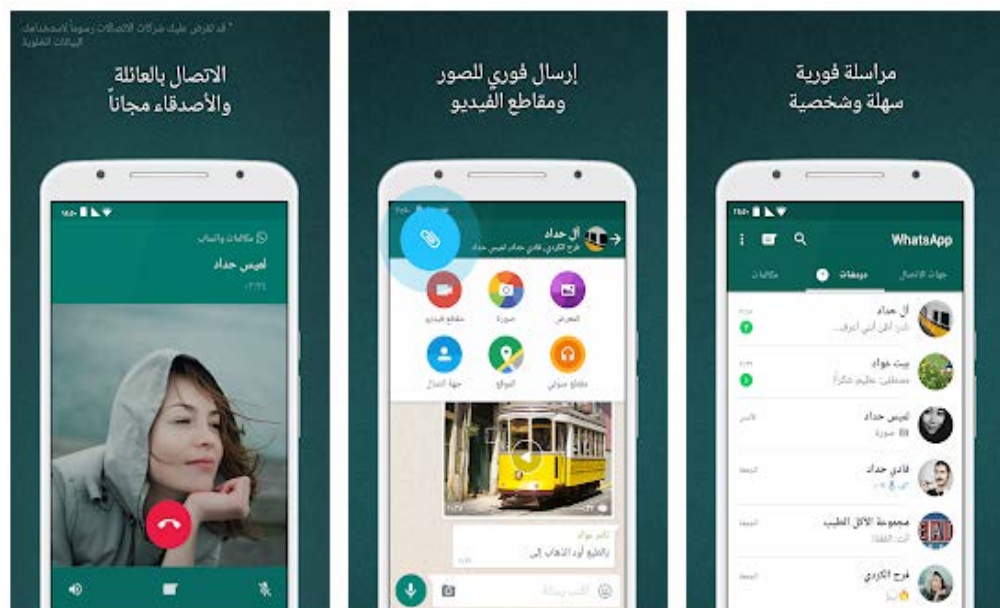


Figure 12. WhatsApp UI

Table 1. Comparison Between Apps

Name	Downloads	Direct	UI	Features	Ease of use	Age	Platform	Cost	Rating	Location	Size
الرياضة للجميع	50 + Thousand	Yes	modern and smooth UI	1- Possibility to create your own sports group. 2- Invite people to participate in your favorite sport. 3- Tips and information about healthy living to encourage everyone. 4- "Support Request" feature to find out how Sports for All can support you in the best possible way	Easy and clear functionality	12+	Android, iPhone	Free	4.3	Saudi Arabia	47MB
Malaeb	50 + Thousand	No	normal UI	1- View the stadiums near your area and check the available time. 2- See the facilities available in each stadium. 3- Join the matches that are happening in your neighborhood. 4- "Customer Support" The application usually responds to inquiries in less than 5 minutes.	Easy and clear functionality	4+	Android, iPhone	Free	3.8	Saudi Arabia	50MB
Meet Up	10 + million	Yes	modern and smooth UI	1- Discover local and online events. 2- Save events you're interested in and revisit them later. 3- Have discussions and use direct messaging to stay in touch with the people you meet. 4- Host your events online, Easily personalize and schedule events on-the-fly. 5- Host your events online and in-person by creating a group on your favorite topic.	Easy and clear functionality	12+	Android, iPhone	Free	4.7	Origin: USA, globally	60MB

QSFA	50 + Thousand	No	Basic UI	1- find out the best locations to practice sports. 2-The application is integrated with Fitbit App to get your daily activity parameters such as steps and distance.	Not easy to use because it crashed many times.	24h/7d	3+	Android, iPhone	Free	4.5	Qatar	8.8MB
Sporty	1000+	Yes	good UI	1- participate in nearby sports events and games 2- organizing the sport courses enrollments 3- Paying solutions 4- Booking system for sports places 5- Rent sports equipment 6- Team management solution 7- Ticketing solution for events, including QR codes to scan tickets and products that can be sold together with keys. And digital wallet. 8- Accounting Reports.	According to most of 12 reviews in app store, that it is fast and easy to use	24h/7d	4+	Android, iPhone	Free Fees on each bank transaction from customers	4.1	Oslo and San Francisco Can't open it from Qatar	104.5 MB
Find Me Football	50,000	Yes	Basic UI, almost two colors, with simple components	1- Create to join football matches. 2- Play with new people. 3- Join a tournament with your team for a chance to win prizes. 4- Find information about football pitches. 5- Add your football pitches as the owner of them.	Easy and clear functionality	24h/7d	4+	Android, iPhone	Free	4	Egypt	24 MB

Sport Easy	500,000 +	No	clear UI	1- Manage sports armatures teams with helping tools such as attendance sheet, assigning tasks, activities scheduler, and collecting memberships. 2- Chat feature. 3- Rating each team member and generating reports.	Easy and clear functionality	24h/7d	4+	Android, iPhone	Free to download there is 3 subscription plans 1- free with limited feature. 2- Team: €5/monthly per team. 3- CLUB: €0.25 per member per month.	4.7	Globally	50 MB
Padel Qatar	100+	Yes	not attractive, need to be developed	1- Players can book and schedule for Padel sports game. 2- Ranking system. 3- Friends and news. 4- Owners of Padel sports places can promote and make championships through the app and offer discounts.	must sign up before using it, not easy to use	24h/7d	4+	Android, iPhone	Free	4	Qatar	15 MB
WhatsApp	5,000,000,000+	No	clear UI	1- Creating groups for your team or with the group who are interested in the same sport 2- Join the interested groups 3- Communicate with people, event organizers, sport places owners quickly and free	Easy and clear functionality	limited	12 +	Android, iPhone	Free	4.4	Globally	185 MB

## Our solution

In the mentioned related apps, we have found that no app combines helping people reserve a sports facility and at the same time a social app dedicated to the sports. Our app is a social media for sports amateurs to create an athletic environment and enable them to reach and book the sports facilities around them. In addition, we have found that other apps focus on only one kind of sport. On the other side, our app serves a variety of sports which enables it to reach a wider domain of users and satisfies their needs.

Regarding matching people with each other, we have studied some scheduling algorithms (connecting users with the same interest algorithms) that could be applied to our project. Some of them had an advantage over others. Such as Correlational Learning, which is mainly used in the AI field and its approach and needs data set to work efficiently. Also, we looked for the algorithms which depend on friends of friends and static interest tags entered by users.

After that, we decided to use the Object-Oriented Sociality algorithm, linking people dynamically through shared objects of interest like jobs, workplaces, sports, and hobbies. This technique involves analyzing content posted by a user on their homepage. Content analysis will provide the system with information about the user's interest levels in various fields. Similar annotations are then searched through the system that determines the interests of other users. The result is further analyzed to determine the users' interest level in that subject and create a compatibility meter. A friend list is then prepared and suggested to the user [11].

## 3. Requirement's analysis

### 3.1. Software development process.

Based on the project, a Sports community App, the suitable software development process is agile. Our project is classified as a startup, so the available information and risks are not that clear from the beginning. Especially the customer needs and risk evaluation, as no similar projects have been applied in the same environment. Therefore, agile provides us with a process that focuses on building and maintaining software with flexibility and the ability to deal with unexpected changes. Also, if any modification is needed, it will be easy to apply it. In addition, the development process will be divided into small phases; each will be tested separately. Therefore, it minimizes complex errors, and each step will be consistent with the whole system. Moreover, one of the advantages of the agile development process is that it considers users' feedback in each phase, so the modification can be done before going into the next step.

Firstly, we will specify the requirements of the users and stakeholders. After that, we will divide the central system's requirements into phases. Then, we will implement each step and test it consecutively. [14]

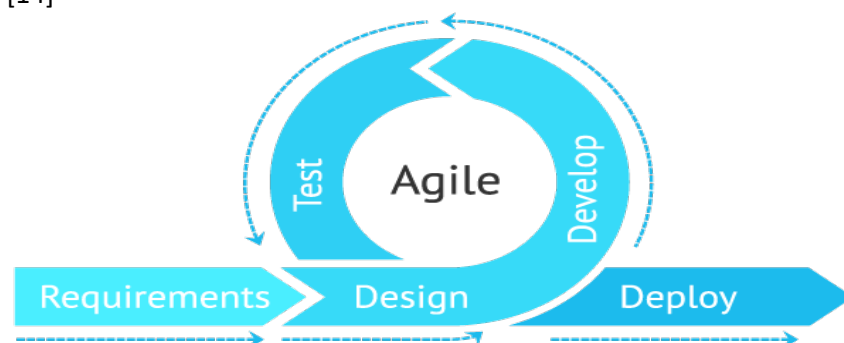


Figure 13. Agile development process



## 3.2. Functional requirements

The functional requirements are the core functionalities that describe the system's intended behavior. We have defined and elaborated our core use cases in Figure.1 and Table 2, as seen below. The actors of our system are the users, the venue owner, the admin, and the credit payment service.

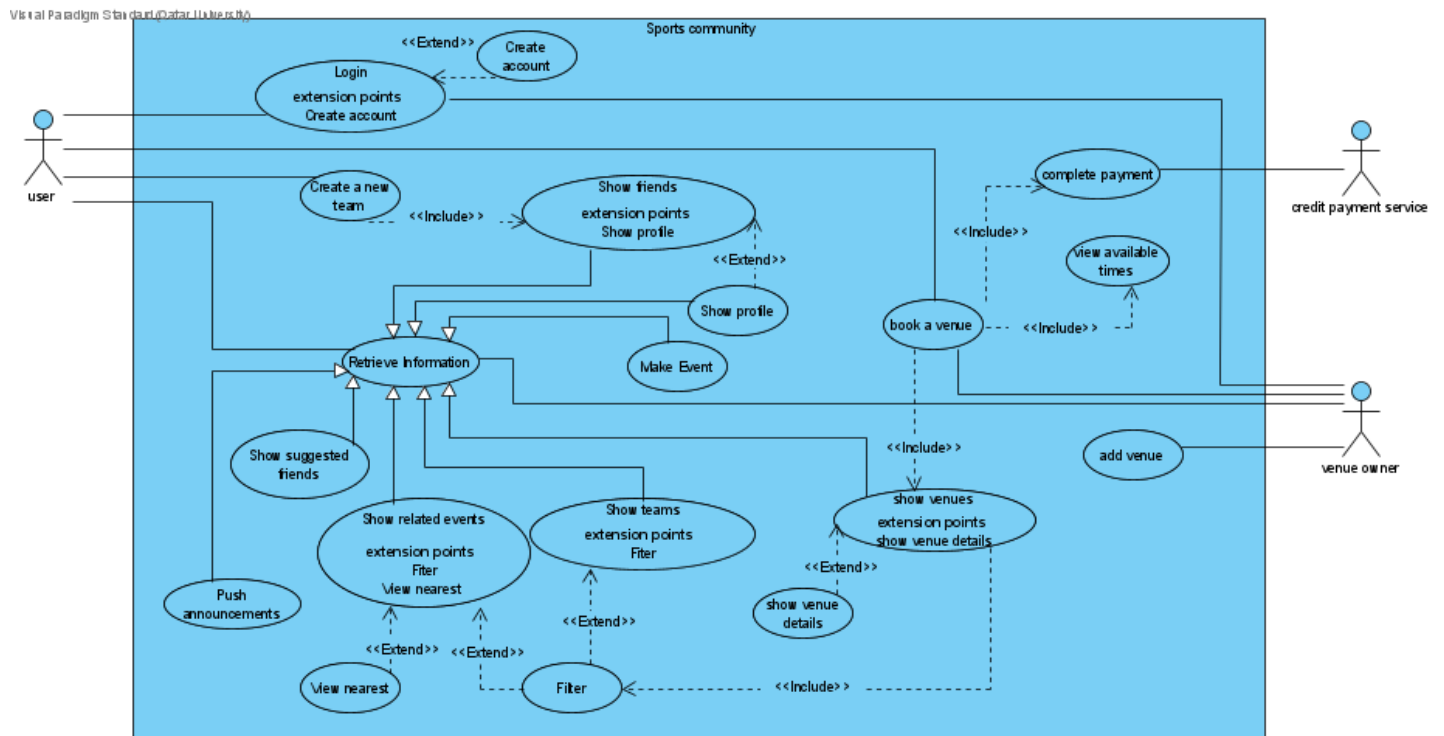


Figure 14. use case diagram

**Table 2. Use cases summary**

<b>Use case</b>	<b>Brief description</b>
Login	Whether he is a regular user or venue owner, the user can log in with his username and password if he has created an account previously. Once the user enters his credential, the system will check its validity.
Create account	The user or venue owner can create an account with his mobile/username and password if he hasn't one.
Create new team	Users can create teams for multiplayer games, and every team will have a ranking based on their activity.
Book venue	The users can book venues and sports places registered in the system. Also, they can show all the available media and the available times. Moreover, they can pay online for the booking.
Show venues	The system shows all the available venues which are registered in the system. The nearest ones will show first. And the user can filter them.
Add venue	The user of type venue owner can add his venue and its information. After that, he can manage it. So, the regular users can book their venues.
Complete payment	The use case will be rendered whenever the user books a venue to do the payment process.
View available times	The use case will be rendered to show the available times for the venue.
Show friends	The user can show all the friends list who added by him.
Show profile	The use case will be extended if the user wants to retrieve information about a certain team member.
Push announcement	The use case will retrieve any changes happened in the system related to a user and notify him about it.
Show suggested friends	The system shows all friends who are registered in the system and the user have their mobile number on the contact records of his phone.
Show related events	The system shows the events that related to the sport user chooses.
Make event	The use can make an event about the sport that registered in the system.
Show teams	The system shows the available teams related to the sport user chooses.
View nearest	The system shows the nearest places according to the user's location in kilometers.
Filter	The system will filter the items chosen from the user and put them in a list.
Show venue details	The system shows all the information related to a selected venue.
Retrieve Information	The system connects the critical use cases and presents them to the user.

### 3.3. Non-functional requirements

Table 3. Non-functional requirements

Name	Description	Evaluation Plan
1- Modifiability	Additional functionality must be added to the app, such as providing the app on the iOS platform. Similarly, some existing services can be modified or deleted to improve the application's capabilities.	We will use the MVVM architecture that separates UI from Model and View Model. Thus, modifying UI or adding new features to the application will not affect other layers and provide ease of modification.
2- Security	The system achieves the CIA triad (confidentiality, integrity, availability) to ensure that the user has a secure good user experience.	<ul style="list-style-type: none"><li>- The user data should be encrypted with a well-known unbreakable cryptosystem.</li><li>- The user data security should 't prevents service availability by deploying a secure dedicated server.</li></ul>
3- Reliability	The user and venue owners' information and the available bookings are continuously stored in the database, so the data will not be lost.	Save the new data, such as the further user information in the database, and check if it is immediately saved in the database. Try to retrieve the data stored in the database and check if it can be recovered. Ensure that there is no data loss.
4- Usability	According to Wikipedia, Usability can be described as the capacity of a system to provide a condition for its users to perform the tasks safely, effectively, and efficiently while enjoying the experience.	Take the user feedback about how helpful the app server is for the app problems and bugs and continuously provide the users with updates.
5- Performance	How the system performs under anticipated workload in terms of responsiveness and stability.	<ul style="list-style-type: none"><li>- The users should take at most 5 seconds to log in and at most a second to create a new account in the database.</li><li>- Testability of the system to communicate with the booking system of venues.</li><li>- The App should provide almost 10 seconds response time for 90% of requests.</li><li>- Perform load testing using software tester such as meter.</li></ul>

### 3.4. Assumptions

The project assumptions are:

- Internet connection is available.
- The user interface is user-friendly.
- Database containing user and venue owners' information and the available bookings are constantly updated.
- We will have all the required information for the different features, such as location, photo for the playground, and contact number for the venue's owners.
- We have all the events details. For example, when the event will happen, such as the date, time, and place.
- The application is bug-free and regularly monitored for improvements.

### 3.5. Ethics

Table 4. Ethics

Principle	Code	Project perspective
<b>PUBLIC</b>	Accept the full responsibility and approve the software if only it is safe	Apply the software to tests to ensure it's safe and work properly, and use a trusted database, such as firebase.
<b>CLIENT AND EMPLOYER</b>	Be honest on what software engineer limitation of experience and education	Assure that the project is fully developed by the project owners, and if the develop team used other code, they will provide acknowledgment.
<b>PRODUCT</b>	Ensure the suitable means were used in the project and the project has clear achievable objectives. In addition, high quality has been sought.	<ul style="list-style-type: none"> <li>- Ensure that the team will use the best practice in terms of code architecture and implementation.</li> <li>- Apply project milestones and specify time to market strategy with SMART goals.</li> </ul>
<b>JUDGMENT</b>	4.04. Not engage in deceptive financial practices such as bribery, double billing, or other improper financial practices.	- Every finance process will be endorsed and documented.
<b>MANAGEMENT</b>	5.01 Ensure good management for any project on which they work, including effective procedures for promotion of quality and reduction of risk. 5.02. Ensure that software engineers are informed of standards before being held to them.	<ul style="list-style-type: none"> <li>- Make sure to hire a qualified managers and supervisors for quality control.</li> <li>- Educate the team members about the quality standards</li> </ul>
<b>PROFESSION</b>	Extend software engineering knowledge by appropriate participation in professional organizations, meetings, and publications	- Participate irrelevant events that could improve team skills and knowledge to rise the quality of output product.
<b>COLLEAGUES</b>	7.03. Credit fully the work of others and refrain from taking undue credit. 7.05. Give a fair hearing to the opinions, concerns, or complaints of a colleague	<ul style="list-style-type: none"> <li>- Reward the hard workers in the team and notify the lazy ones.</li> <li>- Weekly meetings to discuss the different opinion and complaints.</li> </ul>
<b>Self</b>	8.04. Improve their understanding of the software and related documents on which they work and of the environment in which they will be used.	- Every team member should participate and join different activities in self-improvements and in related environment courses

## 4. Solution Design

### 4.1. Overview

Choosing the software approach or architecture we are going to implement in our project is very important because it will later affect the maintainability, usability, and performance of the product. Therefore, in this section, we are going to present some of the architecture we can implement in our projects, and we will discuss some of its advantages and its drawbacks.

#### **Selected solution overview**

During our system development process, we used the latest technology to develop our app. Moreover, we followed the most recent architectural styles and best practices that most mobile apps use nowadays. This provided an excellent user experience, and it made it easy for any programmer to grasp the system components to update it or to do any kind of software maintenance. We have found the MVVM is the most suitable architecture to design our system since it enables us to separate the system's main parts into individualistic components that are used for a certain purpose and collaborate with other system parts. In addition, it enables us to create the UI easily and separate it from the data source and logic functions.

MVVM has a massive impact on teamwork, as it is suitable for dividing the developing tasks. As a result, every member can handle certain parts of the app, and eventually, we gather the system components together to have a fully working application in a limited time slack and limited financial resources.

## 4.2. Architecture Diagram

The MVVM architecture design is one of the common design patterns used in implementing mobile applications. Since the mobile applications mainly consist of a user and interface and a functionality the needs programming logic to be implemented the MVVM architecture can be the perfect choice since it separates between the application UI design and the application programming logic. and there is a class that connect the UI with the updated data coming from any sort of data base temporary or permanent storage. So, there is a complete separation between the data source, the functions logic and the user UI which serves the purpose of maintainability. In MVVM, only necessary data is sent between the view and view model, which makes the process of sending data smooth.

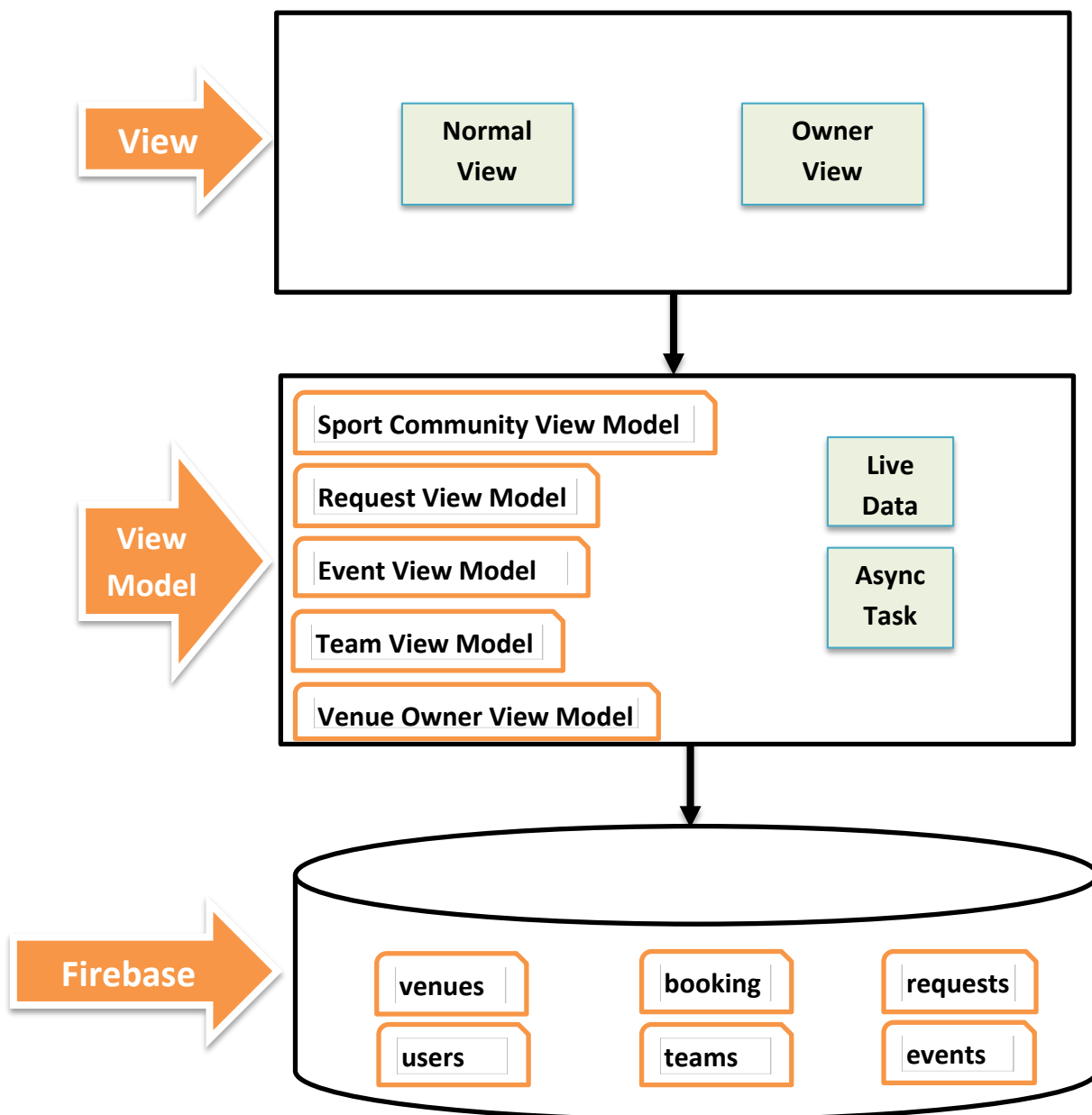


Figure 15. MVVM

### 4.3. Structural model

Visual Paradigm Standard (Qatar University)

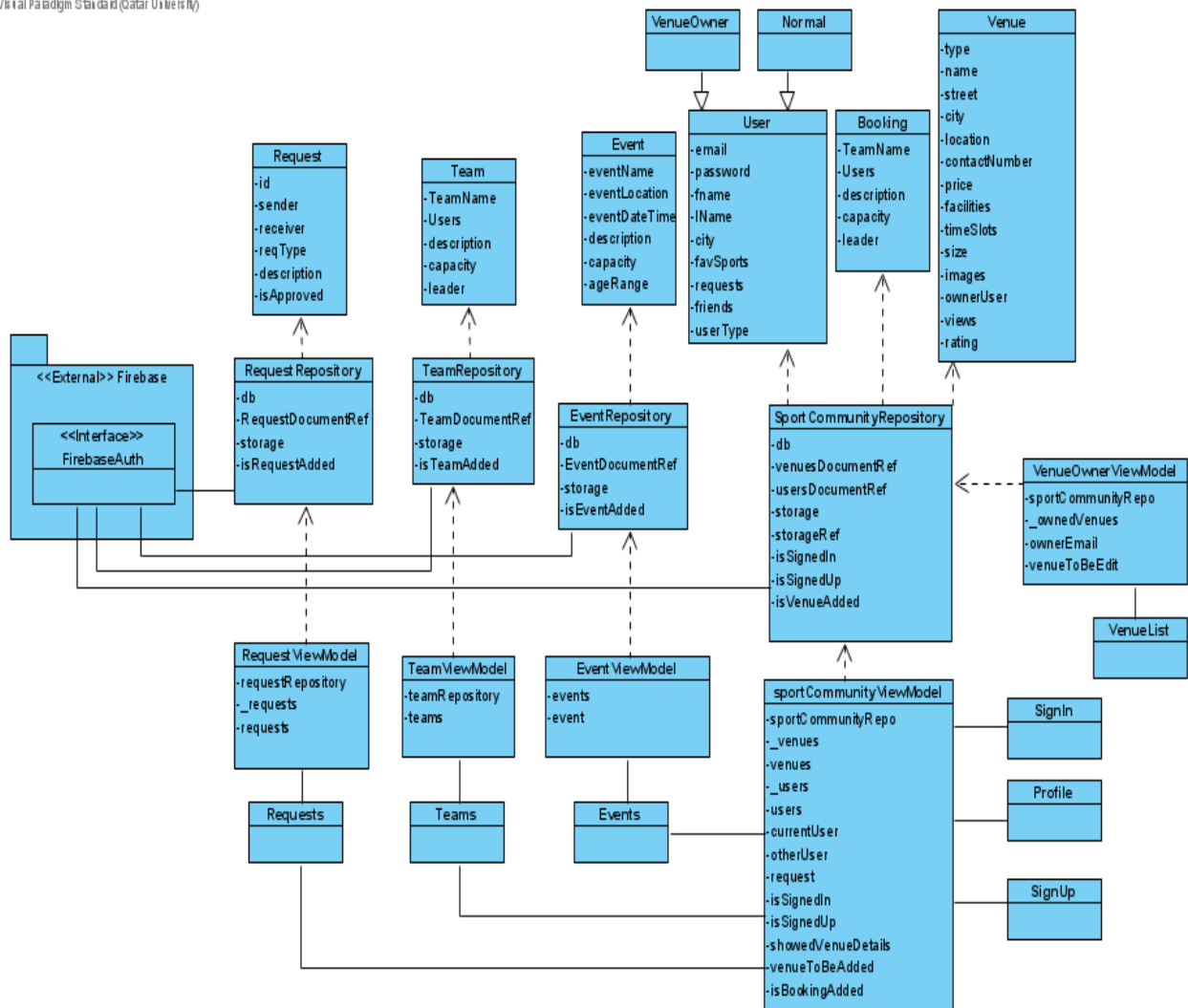


Figure 16. Class Diagram

In the sports community app, there is one interface which is firebase. The firebase is used to authenticate the users (Venue Owners and Normal Users) and store data of events, teams, booking venues, and all requests and venues.

Since we use MVVM architecture, so the models will connect to the repositories which give and send data from Firebase. The repositories will connect to the View-Models that will make a connection between the repositories and views. For example, "Event" model will be connected to "Event-Repository". Then, the "Event-Repository" will be connected to the "Event-View-Model" which is the connector between "Event-Repository" and view "Event".

#### 4.4. Behavioral model

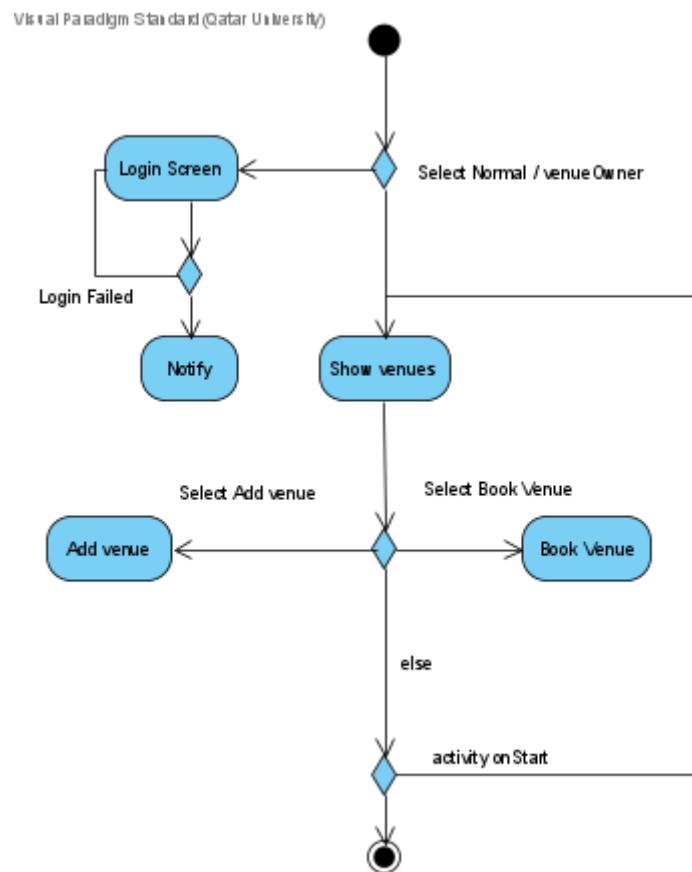


Figure 17. Activity Diagram



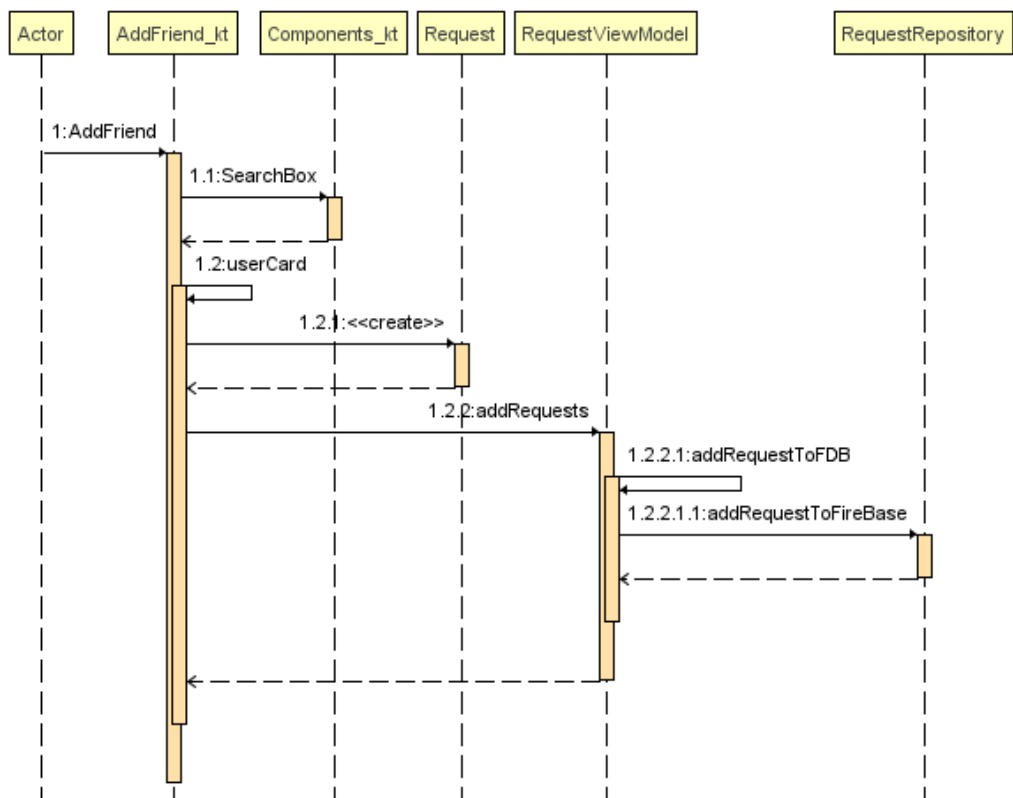


Figure 18. Sequence Diagram of addFriend

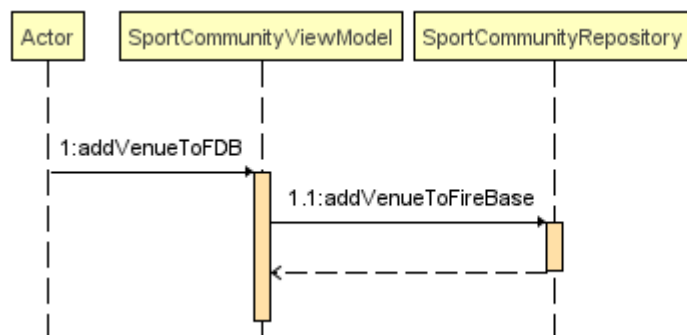


Figure 19. Sequence Diagram of addVenue

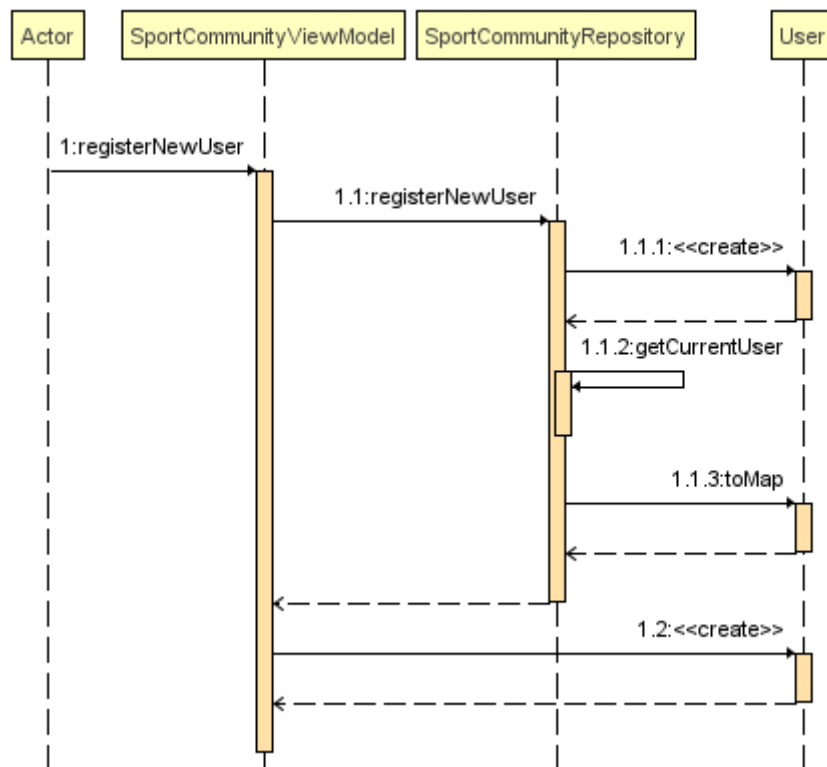


Figure 20. Sequence Diagram of Sign Up

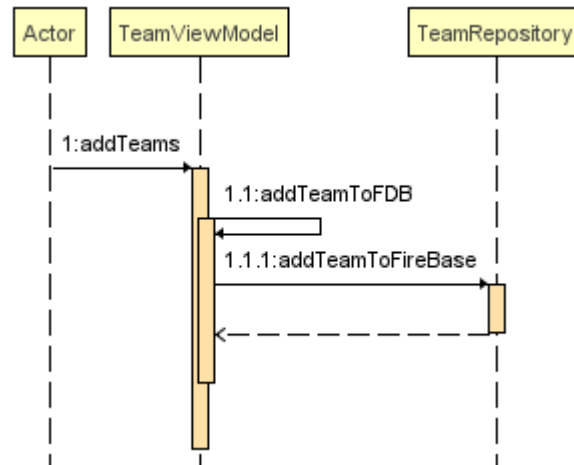


Figure 21. Sequence Diagram of addTeams

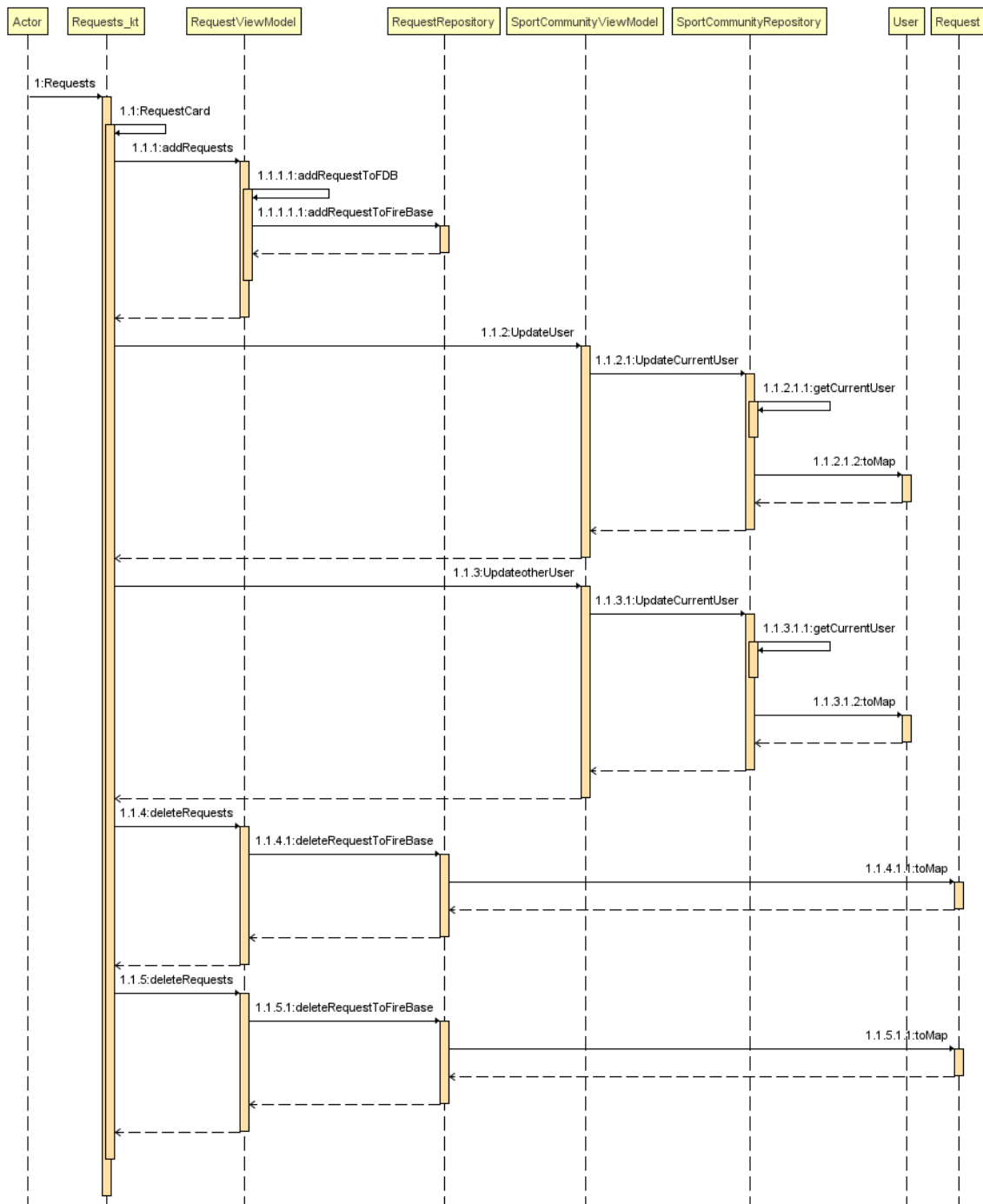


Figure 22. Sequence Diagram of notifications

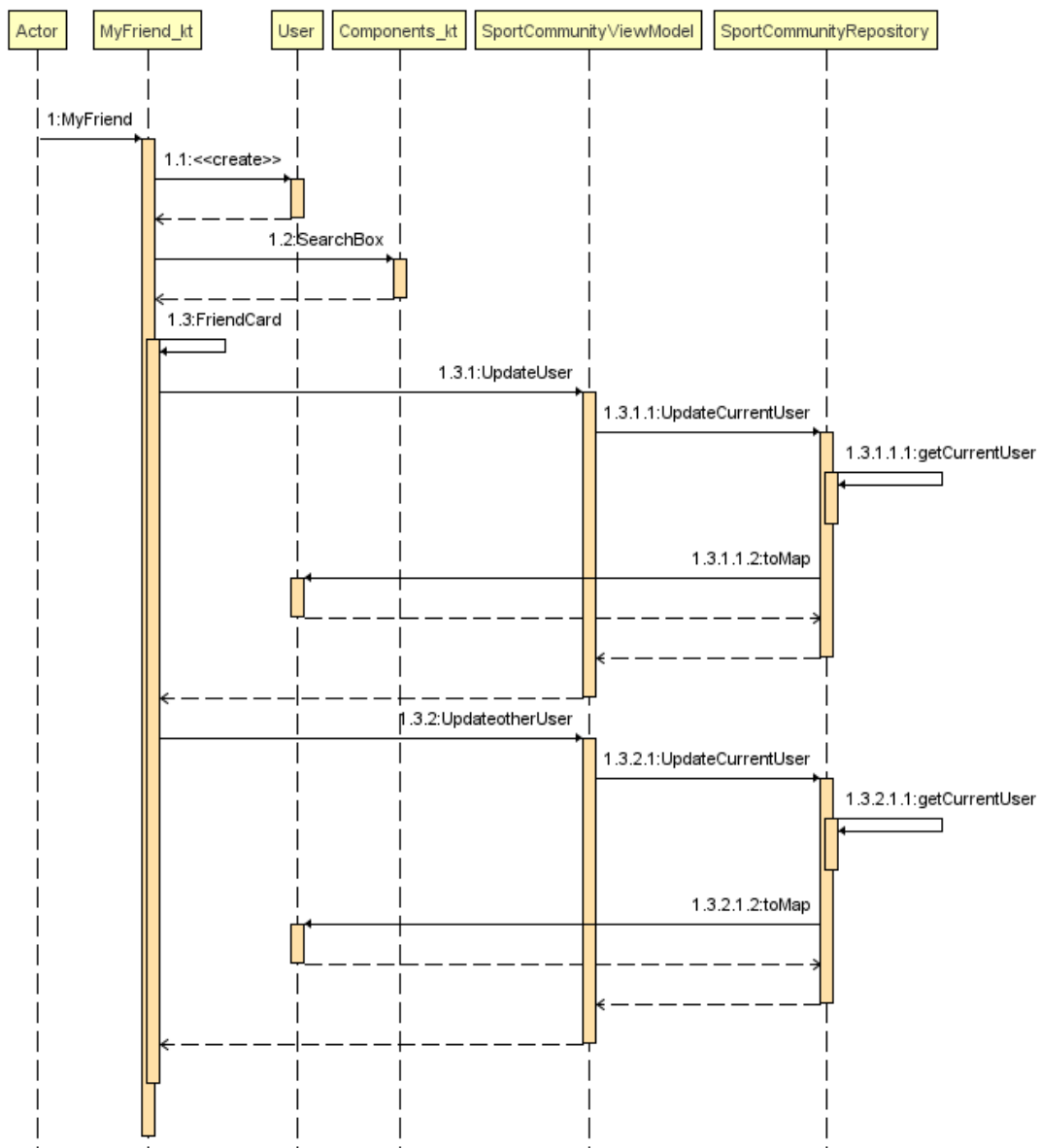


Figure 23. Sequence Diagram of ShowFriends

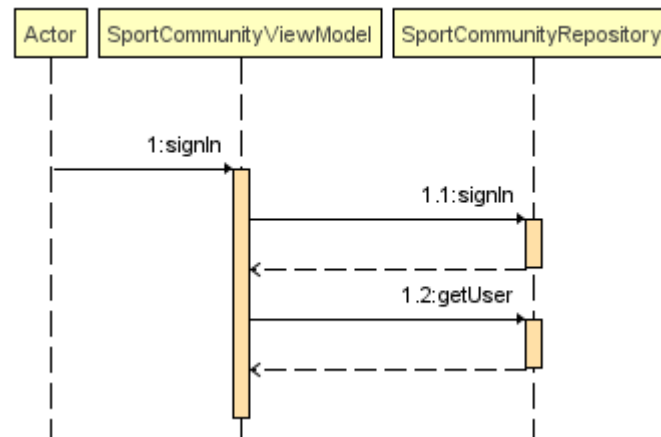


Figure 24. Sequence Diagram of Sign in

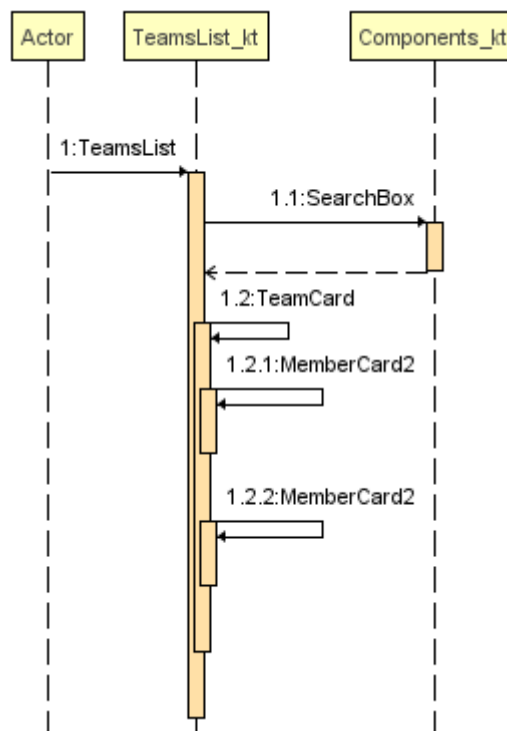


Figure 25. Sequence Diagram of ShowTeams

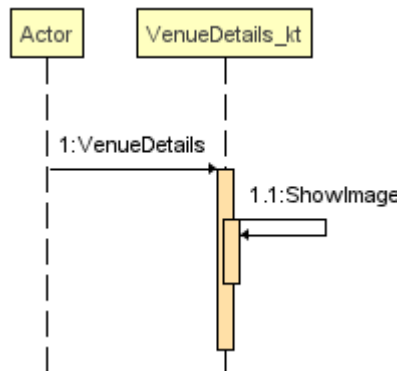


Figure 26. Sequence Diagram of ShowVenueDetails

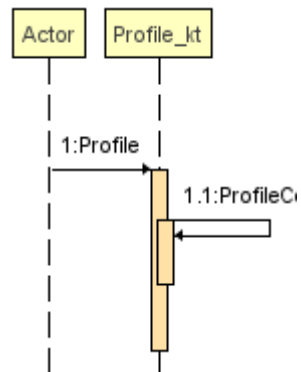


Figure 27. Sequence Diagram of ShowProfile

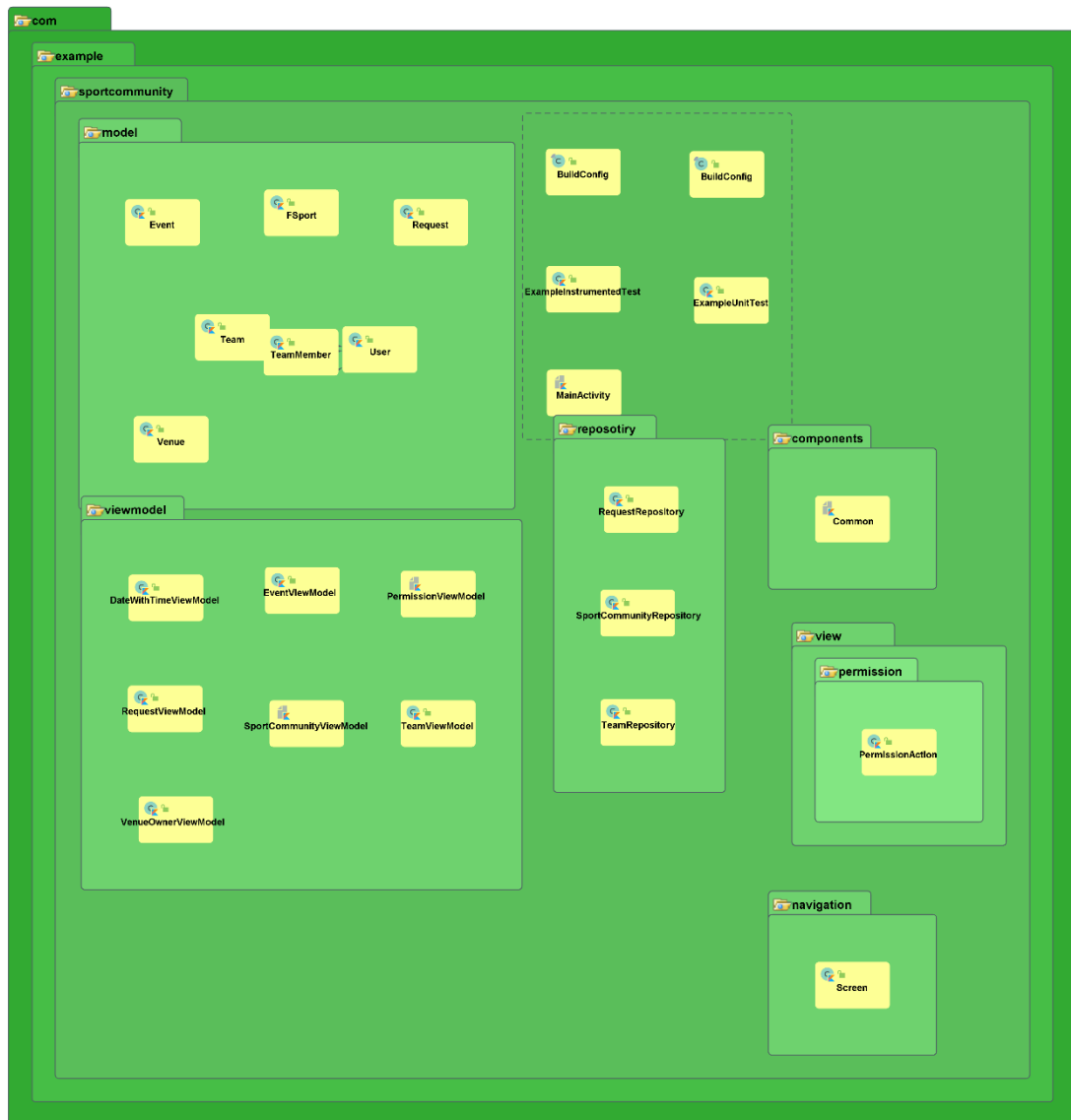


Figure 28. Package Diagram

## 4.5. Database design

An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how “entities” relate to each other within a system. entities in sport community application represent the models in the MVVM architecture. The system has 6 entities which are Venue, User, Booking, Team, Request, and Event. Each one of those entities has its own attributes.

Each user can connect to multiple users by “has friend” relationship. Also, a user can have multiple requests and can join and create many teams and events. In the upcoming versions each user will be able to create limited number of teams so that it will represent the reality. Moreover, some users will be able to own a venue in the system, those users’ type must be “owner”. Lastly for user entity, he can make a booking and that booking will be made to a venue. Lastly, every event could be in only one venue

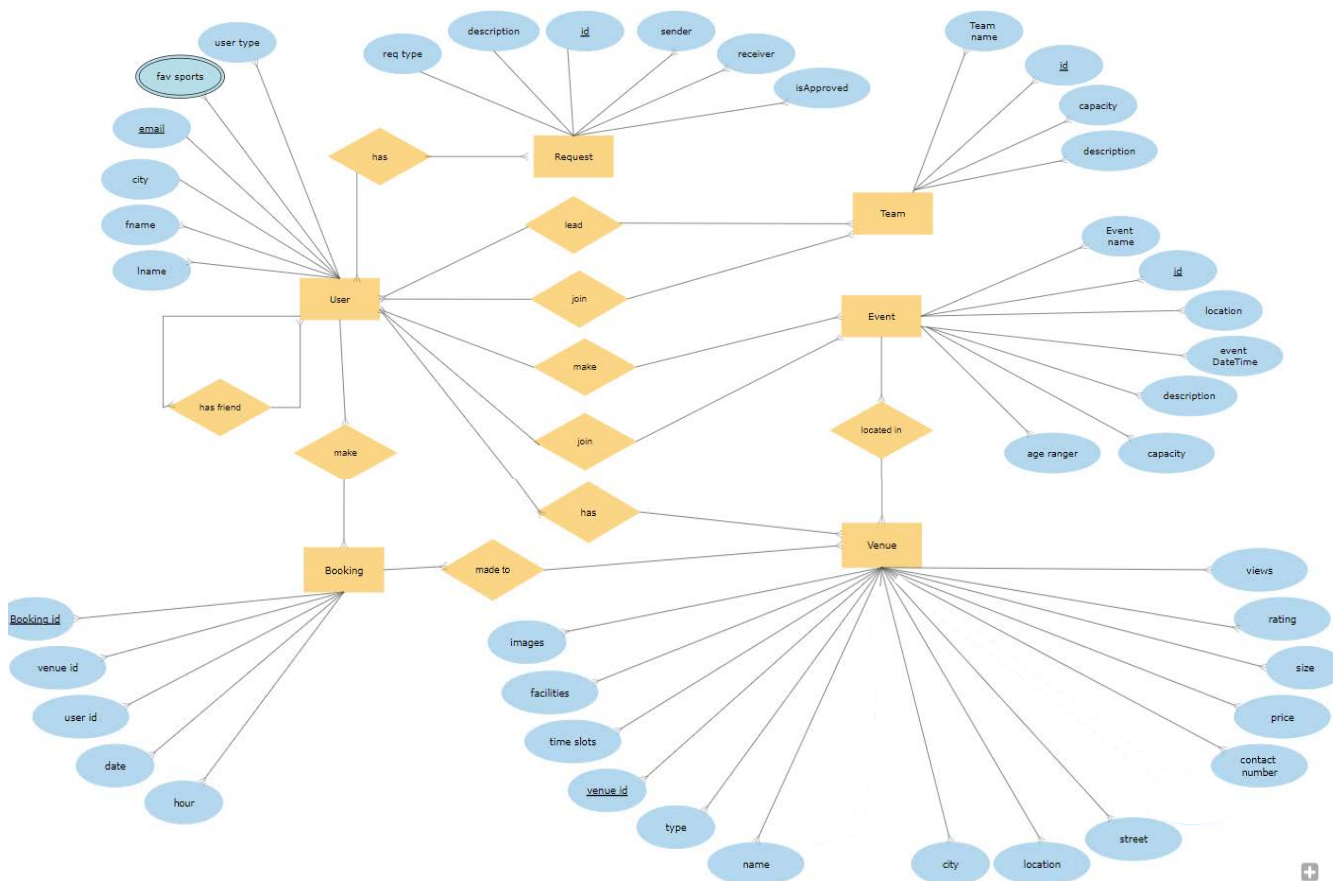


Figure 29. Database Entity Relation Diagram



## 4.6. User interface design

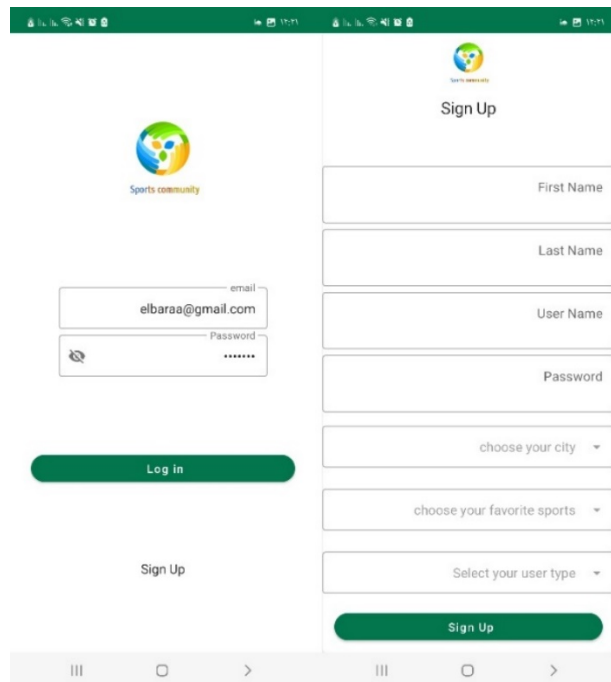


Figure 30. Sign In and Sign-Up Interface

➤ Normal user:-

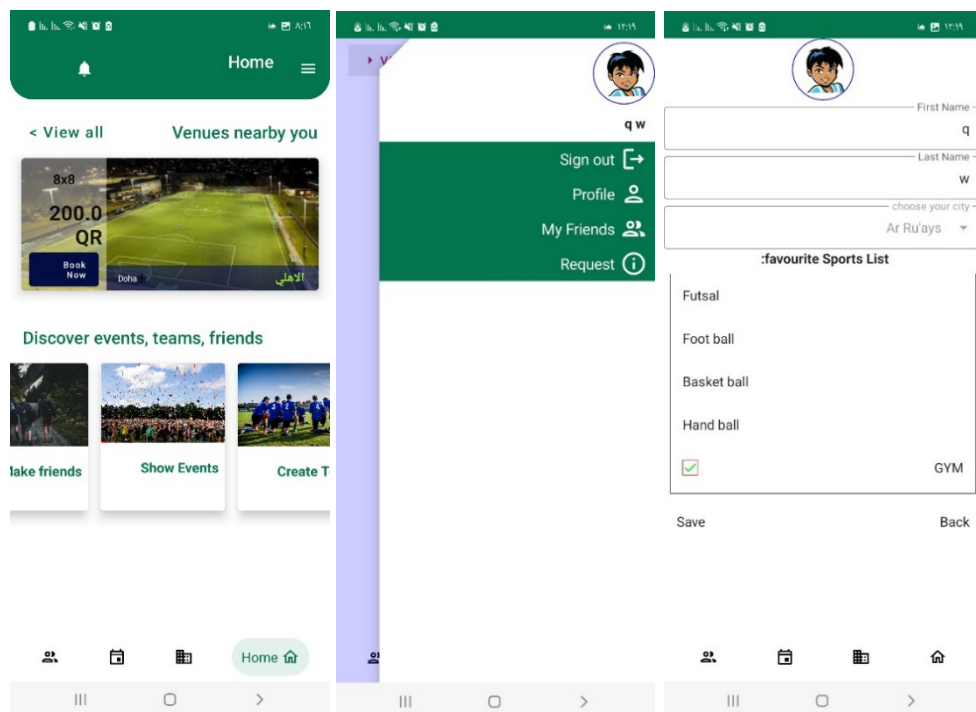


Figure 31. Normal User Interface (Home, Drawer, edit profile)

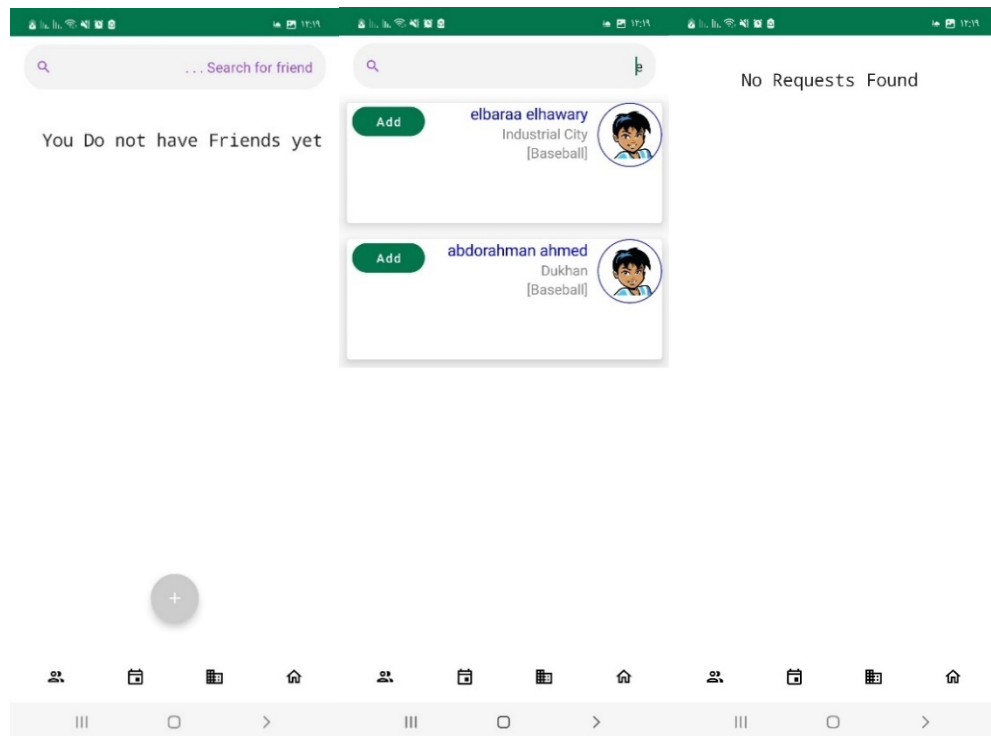


Figure 32. Normal User Interface(ShowFriends, addFriend, ShowRequestes)

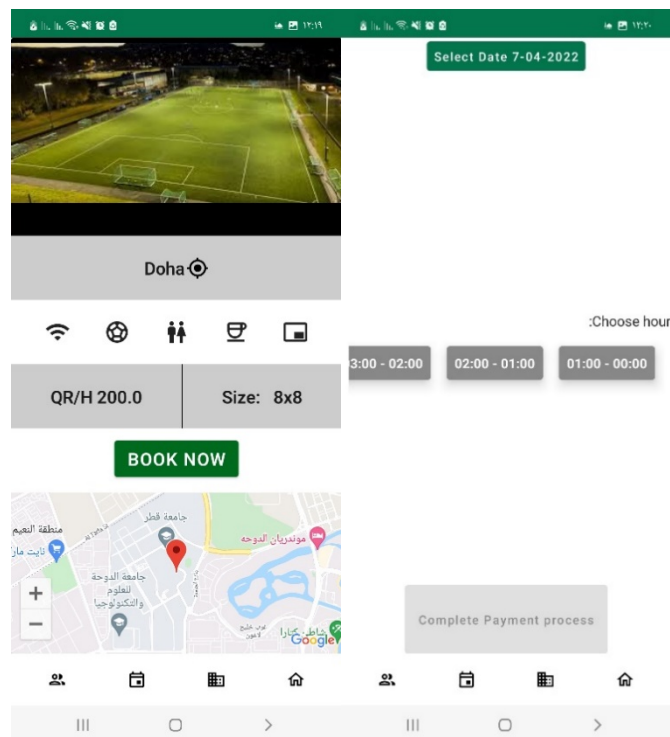


Figure 33. Normal User Interface(showVenueDetails, bookVenue)

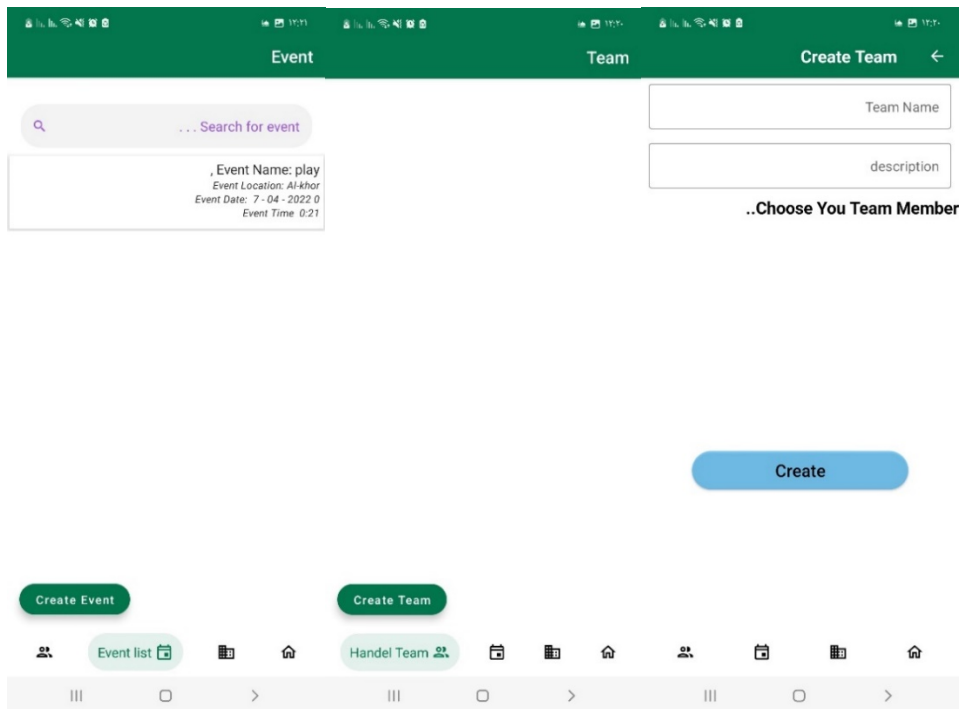


Figure 34. Normal User Interface(ShowEvent, ShowTeams, CreateTeam)

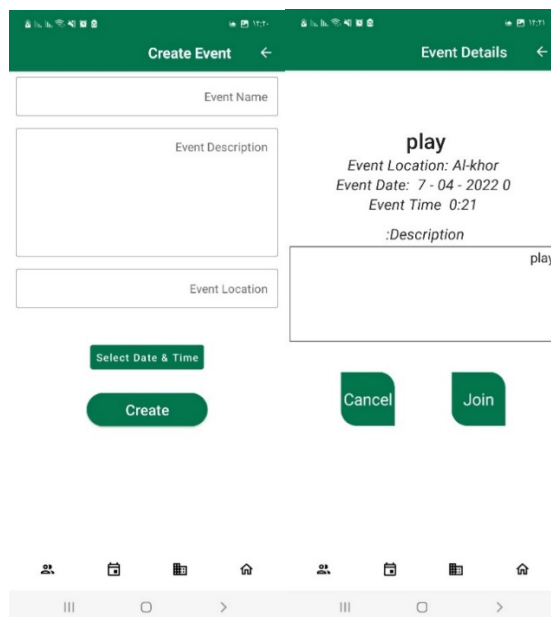


Figure 35. Normal User Interface(CreateEvent, ShowEventDetals)

➤ Venue Owner user:-

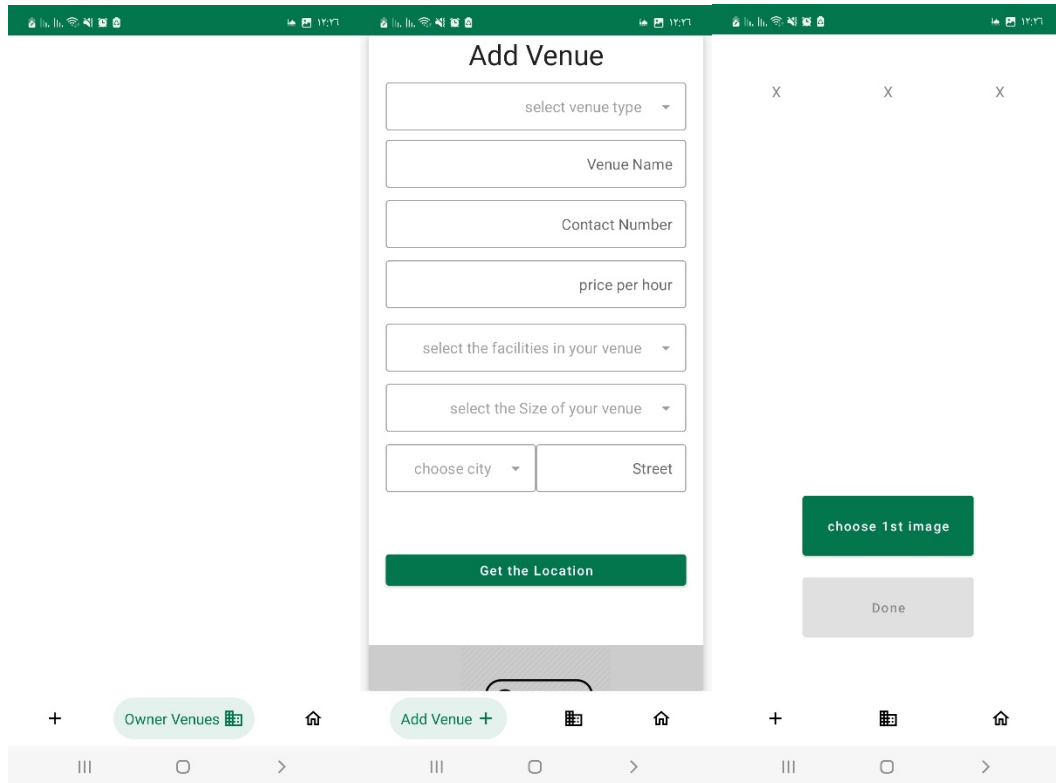


Figure 36. Venue Owner User Interface(ShowVenues, addVenue, addImage)

sport-community-67d94	booking	22nTMnEcZHhuo3KiU2S2
<a href="#">+ Start collection</a>	<a href="#">+ Add document</a>	<a href="#">+ Start collection</a>
<a href="#">booking</a> >	<a href="#">22nTMnEcZHhuo3KiU2S2</a> >	<a href="#">+ Add field</a>
events requests teams users venues	DVGmVeyChT3VfwRcoJ5C tc1q9iAXdRx10s95942P teV7UfIz1PJbVVn8ZZ0V	bookingId: "" date: "20-04-2022" hour: "01:00 - 02:00" userId: "faisal@email.com" venueId: "AsOG9Z4SObuoTN5ns0UO"

Figure 37. Firebase

## 4.7. Design patterns

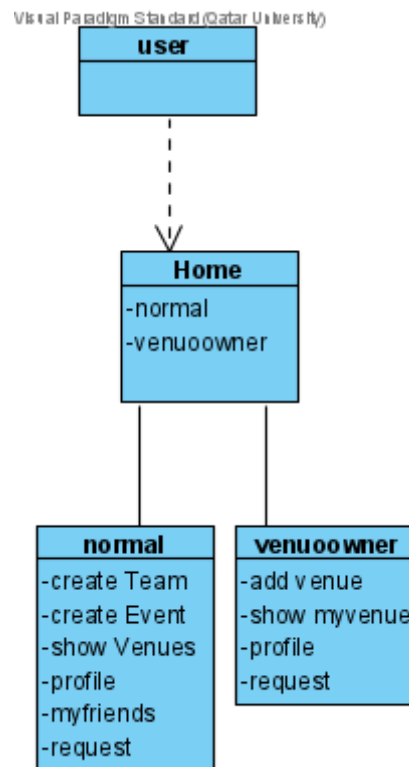


Figure 38. Design Patterns Diagram

The Model View Controller pattern was chosen for our project, in which the user begins on the home page and selects from the available buttons, after which the view changes and the App moves on to the next page so that the user achieves the target page. The Model View Controller pattern is ideal for our project since it allows us to change or update the application's view without affecting other aspects of its controllers as shown in figure 38.

## 4.8. Hardware/software to be used

- Android Studio as the IDEs for developing and testing App.

Android Studio is the official integrated development environment for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development.

Version 2020.3.1

- Kotlin as the programming languages.

Kotlin is a cross-platform, statically typed, general-purpose programming language with type inference. Kotlin is designed to interoperate fully with Java, and the JVM version of Kotlin's standard library depends on the Java Class Library, but type inference allows its syntax to be more concise.

Version 1.6.0

- Firebase as backend.

Cloud Firestore is a cloud-hosted, NoSQL database that your Apple, Android, and web apps can access directly via native SDKs.

<https://firebase.google.com/docs/firestore>

- Ktor library as REST API.

Ktor is built from the ground up using Kotlin and Coroutines. You get to use a concise, multiplatform language, as well as the power of asynchronous

Version 1.6.5

- Android devices which the app will run on them.

- Project Microsoft

Microsoft Project is a project management software product, developed and sold by Microsoft. It is designed to assist a project manager in developing a schedule, assigning resources to tasks, tracking progress, managing the budget, and analyzing workloads.

- Visual Paradigm.

Visual Paradigm supports requirements management including use cases, classes diagrams and textual analysis.

- GitHub

GitHub is a distributed version-control platform where users can collaborate on or adopt open-source code projects, fork code, share ideas and more.

- Google Map

google Maps is a Web-based service that provides detailed information about geographical regions and sites around the world

- Coil image library

An image loading library for Android backed by Kotlin Coroutines.

## 5. Implementation

The framework used to build the application is Android studio using Kotlin language and Jetpack compose for the UI part. Another tool used in the implementation is firebase. Firebase is used as the database for all our data and authentication, where we save the data in it, and retrieves them when required. The good part of using firebase cloud service is that it gives us a no-cost plan for limited storage and the number of requests for reading and writing to the cloud DB. This plan enables us to implement all the use cases of our system and connect them to the cloud service in the early stages. As for the authentication, Firebase Authentication gives back-end development services, simple-to-use SDKs, and instant UI libraries to confirm clients over the application. It supports authentication using passwords, email IDs, or usernames. In addition, we use the Location service from Google Maps API. We used the location service for locating the location for every venue and to set the location for every sports event will be created in the application, so the users can reach the places they want with ease.

One of the challenges we faced was the algorithm of the booking system, mainly how to deal with adding new days with their available times and deleting the last days to not consume the available storage in the database because it is limited. The best solution we came up with was using cloud functions in firebase, which are programming functions that allow us to trigger a code from Google Cloud or Firebase. However, we had to upgrade the price plan to the (pay as you go) plan. Therefore, we solved this problem by changing the way of storing and managing the array which holds the available times for each venue. So now, each venue owner must update the available times by himself, and when he does that, the system will save the changes, and the updated available times will be displayed for the regular users. The other challenge we have encountered was implementing the payment use case. We decided to use the Google Pay service to add a payment method for users, so they can book the time slots that suit them and pay the fees for it. However, to open an account in the Google Pay service, we must own an honest company with valid legal papers. Therefore, we solved this problem by changing the business logic of the payment process until we could get those papers. So, the user can now book the time slot that suits him and pay in the venue. Hence, our system still has the benefits of showing users the available times and advertising for the venues. Our implementation is unique in enabling anyone who has free land suitable for playing sports; can use it for getting income by adding it to our system. The users can book it and know all information they need before booking it.

## 6. Testing

### Unit testing:

We have done unit testing for all the repositories that we used in the application. As the repositories files have the most important function as they are the connection between the front-end and back-end as shown as figure 39.

Test case	result
System should not accept the request unless sender and receiver are not empty.	pass
The status of the request should be updated when the user accepts or rejects it.	pass
System should be able to delete the request after the user response it.	pass
Password should be more than 6 characters in sign up and sign in	pass
Sport community repository should be able to get the image URL of the venues to add it the venue	pass
Email should not be bad formatted in sign in	pass
Email should not be bad formatted in sign up	pass
Return null if the user signed out or not signed in	pass
Team repository should add the team when all required filled are filled in	pass



✓ Test Results	2 s	1/1
✓ TeamRepositoryTest	2 s	1/1
✓ return_true_when_team_Added_successfully	2 s	✓
Tests	Duration	samsung SM-A530F
✓ Test Results	3 s	3/3
✓ RequestRepositoryTest	3 s	3/3
✓ return_false_if_sender_or_receiver_empty	78 ms	✓
✓ return_true_when_updating_status_of_request	3 s	✓
✓ return_true_when_Deleting_request	389 ms	✓
Tests	Duration	samsung SM-A530F
✓ Test Results	145 ms	6/6
✓ SportCommunityRepositoryTest	145 ms	6/6
✓ return_false_if_password_tooShort_in_signIn	67 ms	✓
✓ return_url_of_image	52 ms	✓
✓ return_false_if_emailUser_bad_formatted_in_signIn	0 ms	✓
✓ return_false_if_emailUser_bad_formatted_in_signup	26 ms	✓
✓ return_null_if_no_user_signedIn	0 ms	✓
✓ return_false_if_password_tooShort_in_sign_up	0 ms	✓

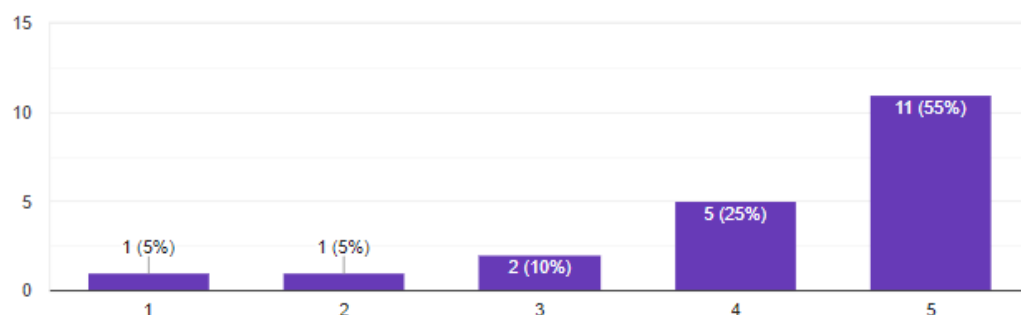
Figure 39. Unit Test

## Evaluation:

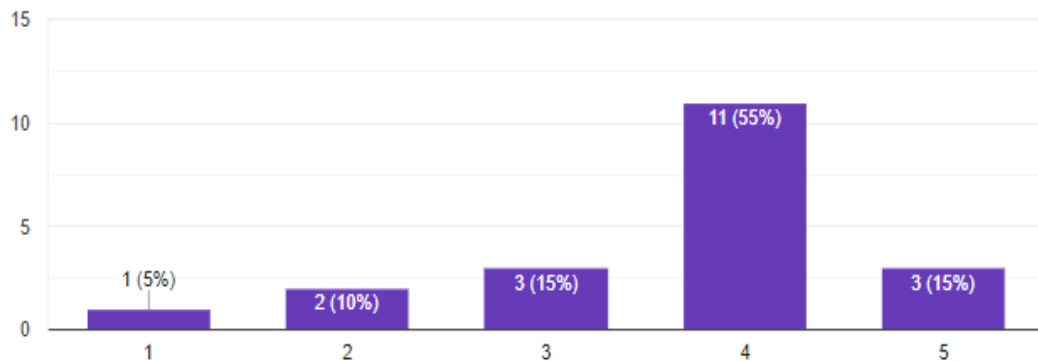
We built the evaluation for the user interface and the UX by using Shneiderman's Eight Golden Rules of Interface Design. Ben Shneiderman is an American scientist with solid expertise in human-machine interaction. Many of his works are fundamental to today's human-machine interaction. Shneiderman's eight golden rules help designers solve problems by improving usability, and the interface needs to be well designed to be "user-friendly."

The evaluation was done by letting 20 random users from Qatar university "as the community in Qatar university has the needed diversity for the target users" use the sports community app and then solving an online survey designed depending on the eight rules. The types of questions in the survey are rating scales questions.

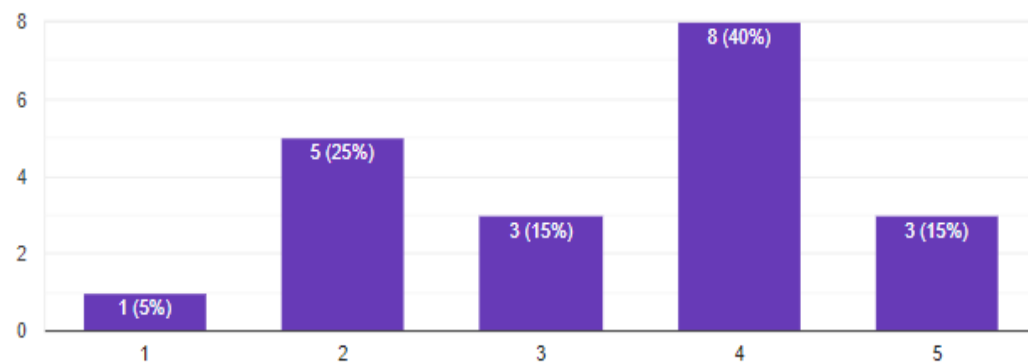
1. The layout, the size of the button, and the color code in our App are consistent. (Sports Community App strived for consistency)



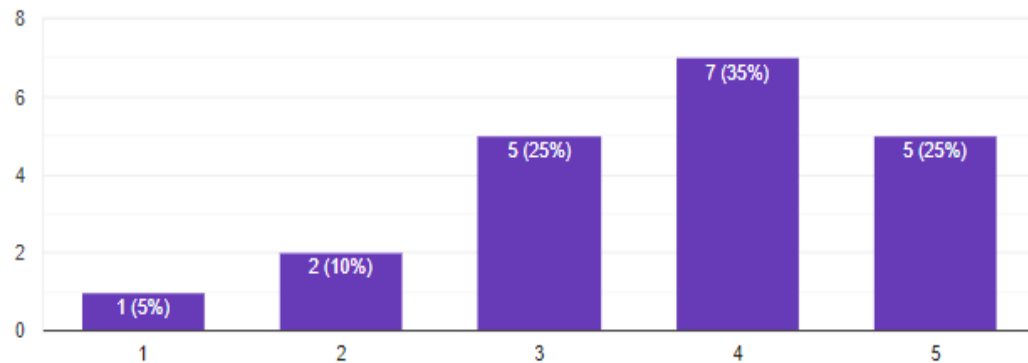
2. Sports Community App allow users to access all parts of the App with a minimum of clicks. (Enable Frequent Users to Use Shortcuts).



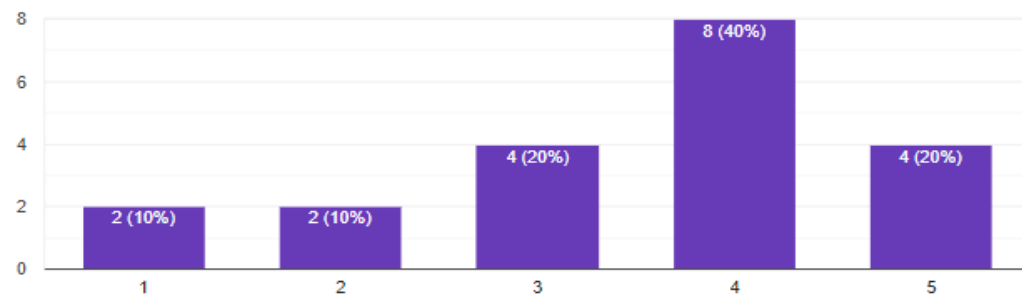
3. Sports Community App displays feedback immediately If users have performed or are performing actions.



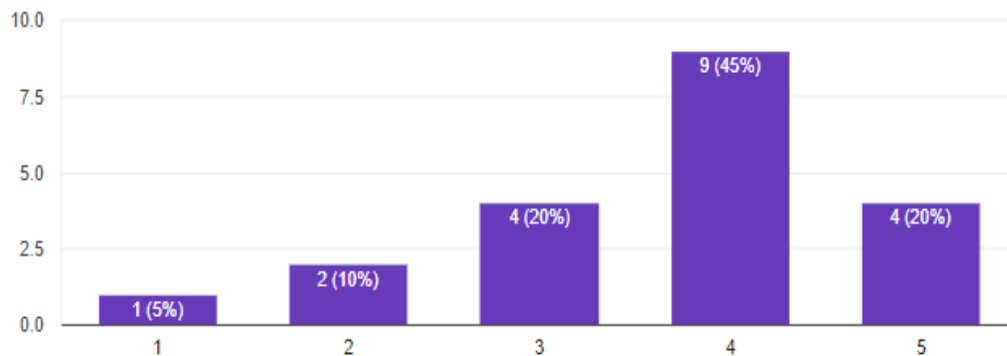
4. Sports Community App gives to the user a design dialog if he performed an action. (For example: Validation message.)



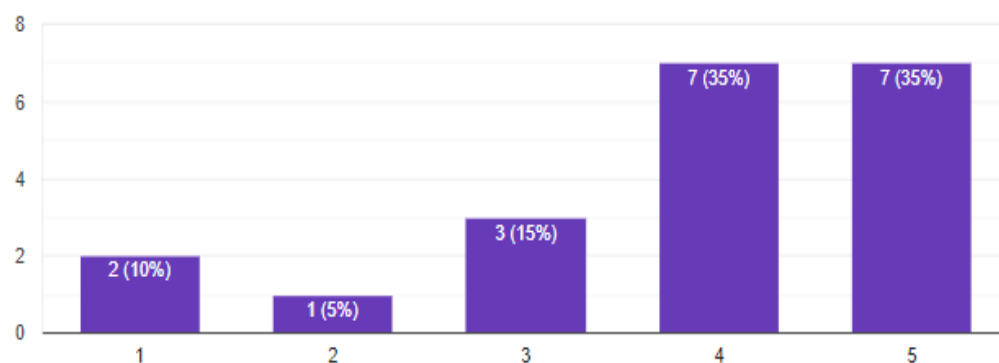
5. Sports Community App has a good interface designed to avoid errors as much as possible and allow the users to understand and resolve the problem.



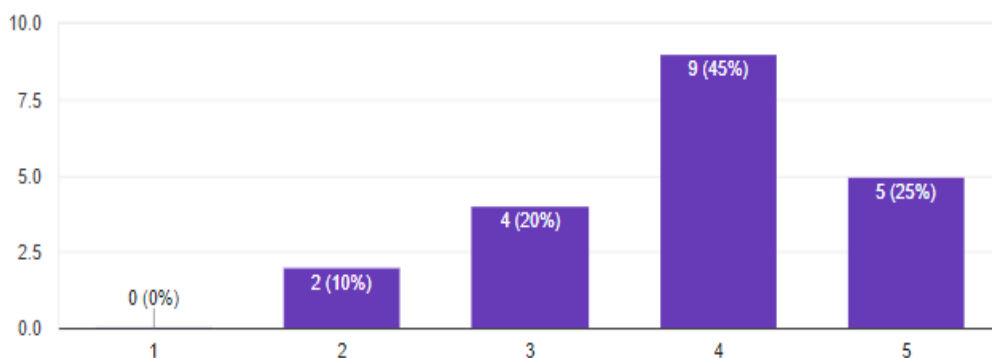
6. Sports Community App permits Easy Reversal of Actions if the user needs to cancel the action.



7. Sports Community App allow users to control many things in the App. (support an Internal Locus of Control).



8. Sports Community App is very minimalist and allows the user's attention to be focused on the main purpose. (Reduce Short-Term Memory Load).



### Non-functional requirements evaluation:

#### Test case 1:

For this test case we will test how long does it takes for the request to reach a user. The expected time from sending the request to the time receiving it is 10 seconds or less. The actual time after testing the use case is 3.1 seconds. The result shows that sending a request will be fast. We Started calculating the time from pressing add, until the request shows up on the phone screen.

#### Test case 2:

For this test case we will test the time needed to login into the system. The expected time is 5 seconds or less.

The actual time after testing the use case is 2.3 seconds.

The result shows that login into the system is less than the expected time. Therefore, it is providing good performance. We Started calculating the time from pressing log in button until the home page appeared on the phone screen.

### Test case 3:

In this test case we need to prove that firebase authentication service is secure and reliable.

We have found that the authentication service which is provided by Firebase has successfully completed the ISO 27001 and SOC 1, SOC 2, and SOC 3 evaluation process. Also, it completed the ISO 27017 and ISO 27018 certification process. As for the encryption, Firebase services encrypt data in transit using HTTPS and logically isolate customer data. In addition, it encrypts its data at rest.[15].

### Test case 4:

In this test case we will ensure that MVVM architecture has been used in the developing the system, so that the principle of separation of concern is applied.

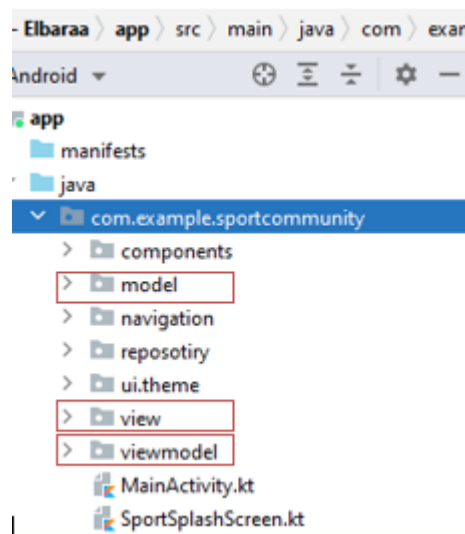


Figure 40. Activity Diagram

As shown above in the Figure 40, we separated the views and the models and the connection between them is done by the view model. Thus, modifying UI or adding new features to the application will not affect other layers and provide ease of modification.

### Evaluation of applying the Software development process

The agile development process positively impacted the quality of the solution and the ease of implementing the tasks needed to develop the Sports community application. Firstly, we specified the requirements and features included in the sports community application. That was done by taking feedback from our targeted users and the sports venue's owners. After that, we divided the system's requirements into phases. Therefore, we concentrated more in every phase on specific

tasks. In every phase, we started by designing the suggested solution. The design included the UI and database design and how it will interact with other components. After that, we implemented the tasks needed in this specific phase. Then we test the result of our implementation of that phase. Thus, every phase was separated from the others, enabling us to test every one of them separately, such as sign-in and sign-up use-cases. It was a user-centered process, so we conducted two surveys with the target users, which was the testing part of the agile process. Also, as for testing the component of every phase, we let some users try the implemented use case that had been done.

## 7. Impact of your project solution

The solution will affect many aspects of the Qatari society in the first place and the Mena region after that.

For individuals, the application will help sports amateurs to make a better connection with people who have similar interests in a sport. For example, if someone wants to practice kickboxing, he will find it easy to find someone to play with or a group to join them. In addition, the application will reduce the obesity of the individuals in our society. Many people will start to practice sport more than before, because they will be able to find all the information they need to know before practicing it, such as the locations of the venues and the prices and the availability of them. Moreover, they will get motivated when they find interesting nearby sports events in which they can participate.

For the society, our application will have an essential role in expanding the sport and sport-tech sectors in our region as the stakeholders and the owners will get more benefits throughout our application. It will open more ways to maximize the revenue of sport-related businesses, such as sports coaching and private trainers and sports facilities, and marketing events.

## 8. Conclusion

To conclude, we believe that our project will change the way of practicing sport in our society, and it will lead to the creation of local communities for every sport. Therefore, the health of the people will get better. The functionalities implemented in the application will help the sports amateurs in different ways since there are many use cases in the application, such as browsing all the sports facilities around them and booking them. Also, they can create a team or join one. Moreover, sports amateurs can find all nearby sports events and matches to join them and be involved more in the sport's local communities. On the other hand, the venue's owners can get more revenue as their platform will be available to many users who use our application. Therefore, they can expand their businesses.

## 9. Future work

In the future we have some developments and ideas that we will implement in the application. First, the payment process is currently the user who pays directly, so in the future, we will test an online credit card payment system to make it easier for the user to pay, so there are many payment options for the user. Secondly, communication between members of the same team in the future there will be a way of communication between team members through the application which will be practically a conversation between the members. Thirdly, Competition between the teams There will be a competition between the teams so that any team can challenge one of the teams through the application, and if the other team accepts the challenge, the competition will start. Fourth, we will add a new user, the admin, who will investigate the status of the owner venue and based on that accepting or rejecting the request.

## 10. Student reflections

### **Elbaraa Elhawary:**

Since this is the biggest project I have ever done, I learned a lot about managing time and dividing the work among group members. Additionally, I gained more experience with Android Studio. Activity and status graphs helped me greatly in analyzing the system and understanding the purpose of the different types of graphs. In the future, I believe these skills will assist me in software engineering and systems analysis.

### **Mahmoud Gazalh:**

This project helped me to learn many lessons, such as time management and dividing the work between the group members in effective way. I have also learned that teamwork is essential in planning and implementing the project. No matter how skilled the people are, teamwork will get the best results. Moreover, through this project I have understand how to analyze a system and how to get benefits from the different kind of diagrams. Therefore, I think these skills will help me with software engineering and system analysis in the future. In addition, I have learned more things related to Kotlin language, new trend of Jetpack Compose, and Android Studio IDE. So, I got more experience with it. As for the future work, I will ask for help from my teammate early to save time for doing other tasks, also to start with any given task as early as possible even if I have a plenty of time.

### **Mohamed Khattab:**

The process of doing this project gave me many new technical experiences and soft skills such as using new software development tools and project management software I also learned how to work in teamwork environment where team members coordinate and manage the work to achieve a certain task, I also learned about testing tools and testing techniques in the software development industry I didn't know about before.

## References

- [1] “الرياضة أهم ركائز رؤية قطر 2030” Al-watan Qatar. [Online]. Available: <https://www.al-watan.com/news-details/id/175169>
- [2] “الرياضة للجميع” Google Play. [Online]. Available: <https://play.google.com/store/apps/details?id=com.uxbert.sfa&hl=ar&gl=US>
- [3] “ملاعب” Malaeb” Google Play. [Online]. Available: <https://play.google.com/store/apps/details?id=com.malaebapp.app&hl=ar&gl=US>
- [4] “Meetup: Find events near you” Google Play. [Online]. Available: <https://play.google.com/store/apps/details?id=com.meetup&hl=ar&gl=US>
- [5] “QSFA Qatar Sports for All Federation” Google Play. [Online]. Available: <https://play.google.com/store/apps/details?id=com.applab.sportsforall&hl=ar&gl=US>
- [6] “Sporty” Google Play. [Online]. Available: <https://play.google.com/store/apps/details?id=com.elipsbilisim.sporty&hl=ar&gl=US>
- [7] “Find Me Football” Google Play. [Online]. Available: <https://play.google.com/store/apps/details?id=com.findme.football&hl=ar&gl=US>
- [8] “Sport Easy” Google Play. [Online]. Available: <https://play.google.com/store/apps/details?id=com.sporteasy.android&hl=ar&gl=US>
- [9] “Padel Qatar” Google Play. [Online]. Available: <https://play.google.com/store/apps/details?id=es.tpc.matchpoint.appclient.padelqatar&hl=ar&gl=US>
- [10] “WhatsApp messenger” Google Play. [Online]. Available: <https://play.google.com/store/apps/details?id=com.whatsapp&hl=ar&gl=US>
- [11] L. Ching-Cheng, G. Prachi, “An Object-Oriented Social Networking to Link People with Similar Interests and Activities” WorldComp12.SoftEngFinalPaper, USA. [Online]. Available: <http://worldcomp-proceedings.com/proc/p2012/SER2150.pdf>
- [12] “النشرة السنوية لإحصاءات الشباب والرياضة 2016” planning and Statistics Authority. [Online]. Available: <https://www.psa.gov.qa/ar/statistics1/Pages/LatestStats/07022018.aspx>
- [13] G. Don, M. Keith, R. Simon, “Software engineering code of ethics” ACM Digital Library. [Online]. Available: <https://dl.acm.org/doi/10.1145/265684.265699>
- [14] “The Phases Of Agile Software Development Life Cycle & Workflow And Project Management” Bitbytesoft. [Online]. Available: <https://bitbytesoft.com/phases-of-agile-software-development-life-cycle/>
- [15] “Privacy and Security in Firebase” Firebase. [Online]. Available: <https://firebase.google.com/support/privacy>

## Appendix A – Project plan

### A.1. Project milestones

We have divided the tasks in this report into manageable tasks that each member can handle, and we have set some timelines where we gather our work together and organize it in one file. We are going to use this strategy as well in the second phase of the project and work in parallel to finish as many tasks as possible in the shortest timeline. We have divided our entrepreneurship idea where we can sum up the process of creating this report in these main milestones:-

#### ❖ Phase-1

Milestone	Milestone objectives	Assigned to
<b>1) Milestone 1: system brainstorming and document analysis</b>	<ul style="list-style-type: none"><li>- Discuss the project idea and problem statement</li><li>- Discuss the project significance and the value it adds to the beneficiaries</li><li>- Write a complete background for the reader to know about the project</li></ul>	<ul style="list-style-type: none"><li>- Mohamed</li><li>- Elbaraa</li><li>- Mahmoud</li></ul>
<b>2) Milestone 2 : market-point of view</b>	<ul style="list-style-type: none"><li>- Information gathering about statistical information sports practitioners in Qatar</li><li>- Information gathering about similar apps and competitors</li><li>- Survey users and business owners' acceptance to such an app</li></ul>	<ul style="list-style-type: none"><li>- Mohamed</li><li>- Elbaraa</li><li>- Mahmoud</li></ul>
<b>3) Milestone 3: Define the possible use cases and design a use case diagram</b>	<ul style="list-style-type: none"><li>- Define the use cases that hold the functionalities that the system will provide</li><li>- Design a use case diagram that shows how the system function and interact with different users and systems</li></ul>	<ul style="list-style-type: none"><li>- Elbaraa</li><li>- Mahmoud</li><li>- Mohamed</li></ul>



<b>4) Milestone 4: Proof of concept and implantation</b>	<ul style="list-style-type: none"> <li>- Implement 20 to 30 percent of the actual design</li> </ul>	<ul style="list-style-type: none"> <li>- Mohamed</li> <li>- Elbaraa</li> <li>- Mahmoud</li> </ul>
<b>5) test the software validity and modify any errors or bugs</b>	<ul style="list-style-type: none"> <li>- implement software testing some the use cases done in the proof of concept and write feedback notes to about it</li> </ul>	<ul style="list-style-type: none"> <li>- Mohamed</li> <li>- Elbaraa</li> <li>- Mahmoud</li> </ul>
<b>6) Submission and presentation</b>	<ul style="list-style-type: none"> <li>- Take the final feedback from the supervisor instructor</li> <li>- Present the work done</li> </ul>	<ul style="list-style-type: none"> <li>- Mohamed</li> <li>- Elbaraa</li> <li>- Mahmoud</li> </ul>

❖ Phase-2

Milestone	Milestone objectives	Assigned to
<b>1) Milestone 1: Software design</b>	<ul style="list-style-type: none"> <li>- Group member will finalize the UC diagram and the database tables to make the whole project ready to be fully implemented and tested in the upcoming milestones.</li> </ul>	<ul style="list-style-type: none"> <li>- Mohamed</li> <li>- Elbaraa</li> <li>- Mahmoud</li> </ul>
<b>2) Milestone 2 : Implementation</b>	<ul style="list-style-type: none"> <li>- This is one of the most important steps in the project which is Implementing all the use cases and creating the proper database and link the m together for the testing step.</li> </ul>	<ul style="list-style-type: none"> <li>- Mohamed</li> <li>- Elbaraa</li> <li>- Mahmoud</li> </ul>
<b>3) Milestone 3: Testing</b>	<ul style="list-style-type: none"> <li>- Conducting unit testing</li> <li>- Testing each component after combining the units.</li> <li>- Verify that the implementation is meeting the functional and non-functional requirements.</li> </ul>	<ul style="list-style-type: none"> <li>- Mohamed</li> <li>- Elbaraa</li> <li>- Mahmoud</li> </ul>
<b>4) Milestone 4: Documentation</b>	<ul style="list-style-type: none"> <li>- Prepare and finalize the final version of the documents which will be submitted, such as the assessment of the built solution, the analysis and potential work.</li> <li>- Make sure all documents are ready to be submitted to the senior project committee.</li> </ul>	<ul style="list-style-type: none"> <li>- Mohamed</li> <li>- Elbaraa</li> <li>- Mahmoud</li> </ul>

<b>5) Milestone 5: Demonstrate work to examiners</b>	<ul style="list-style-type: none"> <li>- Handing over all requirements, including the presentation, video, poster, the source code.</li> <li>- All team members must be ready for the final presentation.</li> </ul>	<ul style="list-style-type: none"> <li>- Mohamed</li> <li>- Elbaraa</li> <li>- Mahmoud</li> </ul>
--	--	---

## A.2. Project timeline

### ❖ Phase-1

Task Mode						Task Name		Duration	Start	Finish	Resource Names		Aug '21				Sep '21				Oct '21				Nov '21			
													25	1	8	15	22	29	5	12	19	26	3	10	17	24	31	7
						▲ Milestone 1(system brainstorming and document analysis)		11 days	Sun 8/15/21	Mon 8/30/21			<div><div></div></div>															
						Discuss the project idea and problem statement		6 days	Sun 8/15/21	Fri 8/20/21	Mohamed,Mahmoud, Elbaraa		<div><div></div></div> Mohamed,Mahmoud,Elbaraa															
						Discuss the project significance and the vale it adds the beneficiaries		4 days	Fri 8/20/21	Wed 8/25/21	Mohamed,Mahmoud, Elbaraa		<div><div></div></div> Mohamed,Mahmoud,Elbaraa															
						Write a complete background for the reader to know about the project		4 days	Wed 8/25/21	Mon 8/30/21	Mohamed,Mahmoud, Elbaraa		<div><div></div></div> Mohamed,Mahmoud,Elbaraa															
						▲ Milestone 2(market-point-of-view)		13 days	Mon 8/30/21	Wed 9/15/21			<div><div></div></div>															
						Information gathering about statistical information sports practitioners in Qatar		6 days	Mon 8/30/21	Sat 9/4/21	Mohamed,Mahmoud, Elbaraa		<div><div></div></div> Mohamed,Mahmoud,Elbaraa															
						Information gathering about similar apps and competitors		5 days	Sat 9/4/21	Thu 9/9/21	Mohamed,Mahmoud, Elbaraa		<div><div></div></div> Mohamed,Mahmoud,Elbaraa															
						Survey users and business owners' acceptance to such an app		5 days	Thu 9/9/21	Wed 9/15/21	Mohamed,Mahmoud, Elbaraa		<div><div></div></div> Mohamed,Mahmoud,Elbaraa															

Task Mode						Sep '21	Oct '21	Nov '21	Dec '21															
Task Name						15	22	29	5	12	19	26	3	10	17	24	31	7	14	21	28	5		
Milestone 3(Define the possible use cases and design a use case diagram)																								
		Define the use cases that hold the functionalities that the system will provide	9 days	Wed 9/15/21	Sun 9/26/21	Mohamed,Mahmoud, Elbaraa																		
		Design a use case diagram that show how the system function and interact with different users and systems	9 days	Sun 9/26/21	Wed 10/6/21	Mohamed,Mahmoud, Elbaraa																		
Milestone 4(Proof of concept and implantation)																								
		Implement 20 to 30 percent of the actual design	11 days	Wed 10/6/21	Wed 10/20/21	Mohamed,Mahmoud, Elbaraa																		

Task Mode	Task Name	Duration	Start	Finish	Resource Names	Sep '21	Oct '21	Nov '21	Dec '21	Jan '22
	➤ Milestone 4(Proof of concept and implantation)	2.2 wks	Wed 10/6/21	Wed 10/20/21						
	Implement 20 to 30 percent of the actual design	11 days	Wed 10/6/21	Wed 10/20/21	Mohamed,Mahmoud,Elbaraa					
	➤ test the software validity and modify any errors or bugs	6 days	Wed 10/20/21	Wed 10/27/21						
	implement software testing some the use cases done in the proof of concept and write feedback notes to about it	6 days	Wed 10/20/21	Wed 10/27/21	Mohamed,Mahmoud,Elbaraa					
	➤ Submission and presentation	17 days	Wed 10/27/21	Thu 11/18/21						
	Take the final feedback from the supervisor instructor	12 days	Wed 10/27/21	Thu 11/11/21	Mohamed,Mahmoud,Elbaraa					
	Present the work done	6 days	Thu 11/11/21	Thu 11/18/21	Mohamed,Mahmoud,Elbaraa					

## ❖ Phase-2

Task Mode	Task Name	Duration	Start	Finish	Resource Names	2021	Qtr 1, 2022	Qtr 2, 2022	Qtr 3, 2022
	➤ Milestone1(Software design)	15 days	Sat 1/8/22	Thu 1/27/22					
	Group member will finalize the UC diagram and the database tables to make the whole project ready to be fully implemented and tested in the upcoming milestones	15 days	Sat 1/8/22	Thu 1/27/22	Mohamed, Elbaraa, Mahmoud				
	➤ Milestone 2(Implementation)	40 days	Fri 1/28/22	Thu 3/24/22					
	This is one of the most important steps in the project which is Implementing all the use cases and creating the proper database and link the m together for the testing step	40 days	Sat 1/22/22	Thu 3/17/22	Mohamed, Elbaraa, Mahmoud				

	➤ Milestone 3(Testing)	17 days	Fri 3/25/22	Mon 4/18/22					
	Conducting unit testing	7 days	Sat 2/19/22	Sat 2/26/22	Mohamed,Elbar.				
	Testing each component after combining the units	7 days	Sat 2/26/22	Sun 3/6/22	Mohamed, Elbaraa,Mahmou				
	Verify that the implementation is meeting the functional and non-functional requirements	7 days	Sun 3/6/22	Mon 3/14/22	Mohamed, Elbaraa, Mahmoud				
	➤ Milestone 4(Documentation)	14 days	Tue 4/19/22	Fri 5/6/22					
	Prepare and finalize the final version of the documents which will be submitted, such as the assessment of the built solution, the analysis and potential work	8 days	Mon 3/14/22	Wed 3/23/22	Mohamed, Elbaraa, Mahmoud				

	Make sure all documents are ready to be submitted to the senior project committee	7 days	Wed 3/23/22	Thu 3/31/22	Mohamed, Elbaraa, Mahmoud				
	➤ Milestone 5(Demonstrate work to examiners)	13 days	Sat 5/7/22	Tue 5/24/22					
	Handing over all requirements, including the presentation, video, poster, the source code	7 days	Thu 3/31/22	Fri 4/8/22	Mohamed, Elbaraa, Mahmoud				
	All team members must be ready for the final presentation	7 days	Fri 4/8/22	Sat 4/16/22	Mohamed, Elbaraa, Mahmoud				

### A.3. Anticipated risks

Table 5. Anticipated risks

Possible risks	Actions to be taken
<b>Time conflict between the team members to complete the tasks at the targeted time</b>	<ul style="list-style-type: none"><li>- Reschedule the tasks time if it doesn't affect the critical path of the project and work as parallel as possible to assure there's enough slack times to handle the lateness in each task.</li></ul>
<b>Member dropping the course in the middle of the task's implementation</b>	<ul style="list-style-type: none"><li>- Disrupt the tasks over the rest members accordingly and reschedule the timeline to compact a new timeline</li></ul>
<b>Miscommunication between the team members that can lead to any kind of failure in the system development process</b>	<ul style="list-style-type: none"><li>- Keep all the members updated about all the updates that happen and always document every technical work happen in the project</li><li>- Use any task management tools that notify other members about changes in the project (GitHub , Click Up)</li></ul>
<b>User don't allow our system to access to resources</b>	<ul style="list-style-type: none"><li>- Present our code of ethics to the end user</li><li>- Use authentication and security methods to assure user data confidentiality and integrity</li></ul>

## Appendices B

Table 6. Login

Use case Id: UC001	login
Brief Description	The user login to the sport community platform and the system checks if he registered in the database
Primary actors	User, venue owner
Preconditions: 1. The user must have account	
Post-conditions: 1. The user will be logged in successfully or redirect to sign up page	
Main Success Scenario: the user signs into the sport community platform	
Actor Action	System Response
1. the user enter username and password	2. validate the username and password (see 2.a, 2.b)
	3. redirect the user to the main page
Alternative flows: 2.a. if the user entered wrong username or password, warning message will show up 2.b. if the user not registered, <<extend>> use case “Create account” (See UC002)	

Table 7. Create account

Use case Id: UC002	Create account
Brief Description	The user create account if he hasn't one.
Primary actors	user, venue owner
Preconditions: 1. User hasn't account	
Post-conditions: 1. Create account successfully	
Main Success Scenario: The user will be able to create account successfully	
Actor Action	System Response
1. the user will enter the username, password, email, phone No.	2. validate the entered information (see 2.a, 2.b)
	3. send activation code to the user's email
	4. request the activation code
5. user enter the activation code	6. validate the activation code (see 6.a)
	7. create account for the user successfully
Alternative flows: 2.a. if the user misses some required fields or entered wrong format, show warning message 2.b. if the email or username exist already, warning message will show. 6.a. if the provided code is wrong, show warning message	

**Table 8. Create new Team**

Use case Id: UC003	Create new team
Brief Description	User can create team and add existing users
Primary actors	user
Preconditions:	
1. User has account	
2. He isn't currently a team leader	
Post-conditions:	
The user creates a team successfully.	
Main Success Scenario:	
The user will be able to create team successfully.	
Actor Action	System Response
1. the user click on create team	2. redirect user to creating team page and show the creating team form
3. fill in the required fields	4. Show friends (see 4.a.)
5. add member to the team from his friends	6. create team successfully
Alternative flows:	
4.a. <<include>> use case Show friends (see UC009)	

**Table 9. Book Venue**

Use case Id: UC004	Book venue		
Brief Description	Users can book venue for the events from the application		
Primary actors	User, venue owner		
Preconditions:			
1. User has account			
Post-conditions:			
The venue is booked by user			
Main Success Scenario:			
User booked the venue he wanted			
Actor Action		System Response	
1. the user click on create venue		2. redirect user to show venues (see 2.a)	
3. user choose a venue		4. view available time for the chosen venue (see 4.a)	
5. user choose time slot from available time		6. redirect user to payment page (see 6.a)	
		7.book venue and change the chosen time slot to unavailable (see 7.a)	
Alternative flows:			
2.a. <<include>> use case show venues (UC005)			
4.a. <<include>> use case views available time.			
6.a. <<include >> complete payment (UC007)			
7.a. if the payment process didn't success, cancel the book process and redirect user to main show venues page			

**Table 10. Show Venues**

Use case Id: UC005	Show venues
Brief Description	User can show all venues that registered in the application
Primary actors	user, venue owner
Preconditions: User has account	
Post-conditions: show all venues the in the system	
Main Success Scenario: all the venues which retrieved from the database will be shown.	
Actor Action	System Response
1. the user click on show venues	2. list all venues that register in the system
	3. extend use case “Filter”.
4. user choose a venue	5. extend use case “show venue details”
Alternative flows:	

Table 11. Add Venues

Use case Id: UC006	Add venues
Brief Description	User can show all venues that registered in the application
Primary actors	Venue owner, salesman
Preconditions: User is signed in as owner	
Post-conditions: New venue request is added to the database	
Main Success Scenario: Venue owner can his venue details and system save it in the "new venues requests "	
Actor Action	System Response
1. the venue owner click on add new venue	2. show form for adding the venue details
3.venue owner fill the form and submit it	4. show success message and redirect venue owner to main page (see 4.a)
	5. save the form in "new venue requests" in DB
	6. notify sales team about the new application
7.salesman will contact venue owner and complete the new form application	8. add the venue to the database
Alternative flows: 4.a if there is missing or wrong input, system will show warning message	



**Table 12. Complete Payment**

Use case Id: UC007	complete payment
Brief Description	The use case will allow user to enter credit card details after that these details will go directly to the credit payment service that will contact the bank system to make the payment process.
Primary actors	User, credit payment service
Preconditions: User has account Valid credit card	
Post-conditions: Finish payment process.	
Main Success Scenario: The payment process finished successfully	
Actor Action	System Response
1. user enter credit card details	2. check the credit card details (see 2.a.)
	3. send credit card details to credit payment service
4. credit payment service send feedback.	5.show the feedback to the user.
Alternative flows: 2.a. if there are incorrect details send error message.	

**Table 13. Show Friends**

Use case Id: UC009	show friends		
Brief Description	The user can show all the friends list who added by him.		
Primary actors	User, venue owner		
Preconditions: 1. User has account 2. User is currently a team leader			
Post-conditions: show the members of team			
Main Success Scenario: All members of team will be shown successfully.			
Actor Action		System Response	
1. The user will click on show friends.		2. Search for users who added by the user. (See 2.a.)	
		3. show the user friends list	
		4. extend use case “Show profile” (UC010)	
Alternative flows: 2.a. if no users found, display message no friends exist.			

Table 14. Show Profile

Use case Id: UC010	Show profile
Brief Description	The system will display main information about certain user.
Primary actors	User, venue owner
Preconditions: 1. User has account 2. The user should be in his friend list page.	
Post-conditions: The system will display information about user.	
Main Success Scenario: Main information about user will be show.	
Actor Action	System Response
1. The user will click on Show profile.	2. display main information from DB
Alternative flows:	

Table 15. Push Notification

Use case Id: UC011	Push notifications
Brief Description	The system will retrieve information and notify the user about it.
Primary actors	user, venue owner
Preconditions: The user logged in.	
Post-conditions: the user gets notified.	
Main Success Scenario: The system will be able to send the user notification.	
Actor Action	System Response
	1. the system will check if the retrieve information use case is updated
	2. the system will push notification for the user he has a new notification
Alternative flows:	

**Table 16. Show suggested friends**

<b>Use case Id:</b> UC012	Show suggested friends	
<b>Brief Description</b>	The system shows all friends who are registered in the system and the user have their mobile number on the contact records of his phone.	
<b>Primary actors</b>	User, venue owner	
<b>Preconditions:</b> 1- User has account 2- User has a phone number.		
<b>Post-conditions:</b> The system will show the friends.		
<b>Main Success Scenario:</b> All friends will be shown.		
<b>Actor Action</b>		<b>System Response</b>
1- User select Show suggested friends' option.		2- Search for users by the phone number (see 2.a.)
		3- Display the users.
<b>Alternative flows:</b> 2.a. if no users found, display message no friends exist.		

**Table 17. Show related events**

Use case Id: UC013	Show related events		
Brief Description	The system shows the events that related to the sport user chose.		
Primary actors	User, venue owner		
Preconditions: 1- User has account			
Post-conditions: The system starts searching for the related events			
Main Success Scenario: The related events will be shown			
Actor Action		System Response	
1- User select Show related events option.		2- Find all events that related to the sport chosen from the user (see 2.a.).	
		3- Display all events.	
Alternative flows: 2.a. if no events found, display message no events exist.			

Table 18. Show teams

Use case Id: UC014	Show teams		
Brief Description	The system shows the available teams that related to the sport user chose.		
Primary actors	User, venue owner		
Preconditions: 1- User has account.			
Post-conditions: The system starts searching for the teams			
Main Success Scenario: The teams will be shown			
Actor Action		System Response	
1- User selects Show teams’ option.		2- Find all teams that related to the sport chosen from the user (see 2.a.).	
		3- Display all teams.	
Alternative flows: 2.a. if no teams found, display message no teams exist.			

Table 19. View nearest

Use case Id: UC015	View nearest
Brief Description	The system shows the nearest place according to the user’s location in Kilometer.
Primary actors	User, venue owner
Preconditions: 1- User has account 2- User must select Show related events.	
Post-conditions: The system starts searching for the nearest place.	
Main Success Scenario: The nearest places will be shown	
Actor Action	System Response
1- User provide his/her location.	2- Save location.
	3- Get a list of places that located within 10km (see 3.a.)
	4- Display all places within 10km.
Alternative flows: 3.a. if no place found, display message no nearest places exist.	

Table 20. Make event

<b>Use case Id: UC016</b>	<b>Make event</b>
<b>Brief Description</b>	The user can make a new event related to the own sports that saved in the system.
<b>Primary actors</b>	User
<b>Preconditions:</b> 1-User has account 2-User must select Make event.	
<b>Post-conditions:</b> The system makes new event to the user.	
<b>Main Success Scenario:</b> The system makes the event successfully.	
<b>Actor Action</b>	<b>System Response</b>
1- User press on Make event button.	2- Open make new event screen.
3- User fill all information needed.	4- Check the information(see 4.a.).
	5- Make new event.
<b>Alternative flows:</b> 4.a. if there is any missing information, the system will view error message.	

Table 21. Show venue details

<b>Use case Id: UC017</b>	Show venue details	
<b>Brief Description</b>	User can show all the information that related to a selected venue	
<b>Primary actors</b>	user, venue owner	
<b>Preconditions:</b> User has account		
<b>Post-conditions:</b> Show the selected venue information		
<b>Main Success Scenario:</b> Information about the selected venue will be shown.		
<b>Actor Action</b>	<b>System Response</b>	
1. the user click on a venue card	2. retrieve all the information for the selected venue	
	3. Show the information for the selected venue	
<b>Alternative flows:</b>		

## Appendices C

### Sports App Questionnaire

this Questionnaire is intended to collect data about users information about sport app online communities The data will be used for analysis purposes to have information about the market point of view of a software

?Do you play any sport that you have to create a team for

YES ☐

No ☐

How often you face the problem of finding a team or a partner to practice your  
\* ? sport with

	5	4	3	2	1	
Always	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Rare

how often you face the problem of finding an appropriate place of facility to  
\* ?practice your sport in

	5	4	3	2	1	
often	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Rare

Did you ever search for an app that enable you to find a team or a training  
\* ?partner

Yes ☐

No ☐

how likely you are goring to use an app that allows you to find a teammate or a  
\* ? place to practice your sport

5

4

3

2

1

Always

☐☐☐☐☐

Rare

\* ?Do you mind having a team mate you knew online

YES ☐

NO ☐

Do you mind people in your team evaluate your performance and put you into a  
\* ? rank

Yes ☐

No ☐

## Senior Project Form.

sportscommunityqa@gmail.com (لا تتم مشاركة) تعديل الحساب



\*مطلوب

1. The layout, the size of the button, and the color code in our App are consistent. ((SportsCommunity App strived for consistency \*

5	4	3	2	1
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

2. SportsCommunity App allow users to access all parts of the App with a .2 (minimum of clicks. (Enable Frequent Users to Use Shortcuts \*

5	4	3	2	1
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

3. SportsCommunity App displays feedback immediately If users have .3 performed or are performing actions \*

5	4	3	2	1
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>



SportsCommunity App gives to the user a design dialog if he performed an .4  
\* (action. (For example: Validation message

5	4	3	2	1
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

SportsCommunity App has a good interface designed to avoid errors as much .5  
\* .as possible and allow the users to understand and resolve the problem

5	4	3	2	1
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

SportsCommunity App permits Easy Reversal of Actions if the user needs to .6  
\* .cancel the action

5	4	3	2	1
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

SportsCommunity App allow users to control many things in the App. (support .7  
\* .(an Internal Locus of Control

5	4	3	2	1
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

SportsCommunity App is very minimalist and allows the user's attention to be .8  
\* .(focused on the main purpose. (reduce Short-Term Memory Load

5	4	3	2	1
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>