# Abstract

Game management system is the one of the best project which allows admin to manage each game indivisibly. It’s provide each game record details information about current game or upcoming tournaments event, also Track all specific user on the system. It’s track all user on map to show the user profile, prizes, reviews information on the current game.

Game management system used all system recourses to registration new user who will later involved, participate in upcoming events, its helps to add game, delete and update game information, add new tournament videos, invite user to join event, track all participate user dynamically, reward user, view all game list, view user list, view all game post, keep secure reward transaction details.

User easily modifies their profile, registration email, password, update, and delete personal information related to user who wills already registered on system. Admin allow user to access Tournament details, view user track location, view reward details, and play individual game.

Each user get reward point when they join game or review post which created by admin for registered user, player used their reward point to collect prizes.

In this system I have used android framework which support java programming language which run JVM machine which allow administrator to complete shorten task. All task done genuinely using **MVC pattern**. Project will done in waterfall approach which gives more benefits to consume time which declared in WBS.

Database help to interconnect the system with backend service which store the application data. I used online **JSON** structure database which store all date on Google console firebase real time database, whole project source code follow **object oriented programming** concept, which gives much dynamically function to run the application.

JSON help to connect or create virtual database on internet server. Its help to exchange data throw API. Application programming interface (**API**) build the connection to access the features or data of an application, or other services.

This system implement on the **Android studio** which is open source software development platform which gives huge benefits in upcoming future mobile development to modify project recourses.

# Chapter 1

## Introduction

Game management system is web based android application, its help to manage game related issues, manage game details user score, reward, user profile, track user Location, and add update delete users, publish game post, tournament events. Its works on any lollipop version android device which support this application, application allow user fully functional with web base database system.

In this development phase all system implements in MCV Patten which separate system with model controller and view part, its helps to implement parents attributes to their each child. It’s managing resource file systemically, so each method has been override when it’s needed.

Java programming language is used for developing the android application, android studio used to make the application; it is open source android mobile development tools which making responsive android mobile programs.

The main goal to developing this tool to update user about game information, also give more fun to play some default game which help to get reward from system, track location at the same time while using this application.

# Scope

Game management system helps to maintain game details, easily access tournament information, easily access track user location, profile, live discuss, system manage game to get reward point manage high score point. It’s gives friendly interface for each user maintain user details securely on Google database.

## Aims

1. **Efficiency**

Its easily access game post, play game view user details in real time.

1. **Time**

Easily maintain game post, user details, user point information in short period of time update delete view details in real time.

1. **Flexibility**

All database sources online, Data easily convert for any device.

1. **Security**

Game manager tools used API database that gives secure connection between view and model controller.

## Objectives

This system manage all the requirement that follow basic concept of making short score reward program, store all the user information in backend database with full details.

1. **Efficiency**

This system automatically updates current score in database, its update score when player play game as well tracks all users at same time.

1. **Accuracy**

According to game play user automatically upgrade program online as well, its gives full same time accuracy functional on all over users.

1. **Security**

This system used online API firebase database store data more securely which provided user restriction for access data on database server.

1. **Time Saving**

This automated system provide each different user interface in same time which help to grand access in same time which create some basic stuff to consume time processing.

1. **User Friendly**

Game management provides basic game environment which provides user suitable game platform also which provide smooth android interface which attracts user to lead the program.

# Development Methods

This system used waterfall design process which done step by step method which provide this tool a basic development progress which help to maintain project zero to finalize report which allow requirement gathering , analysis , design etc , this is custom method to protect project form failure. Action tacking in different phases on this method given below:

* Requirement specification
* Design
* Implementation
* Verification
* Maintenance
* Deployment



The following steps play their own role which provide best outcome for end users.

* All required document and data recourses collected simultaneously according to user requirement.
* Basic object oriented concept followed to finalize the product.
* Its react with program performance which redirected functional requirement.
* Object oriented play vital role to maintain project resources in future.

# Chapter 2

# Analysis Specification

## Introduction

This is a first identification of all basic needs of user requirement, identification problems which occur in development phase, also collecting all the raw data and developing prototypes which give blueprint for the program background. Game management system manages the game, as same time its track user, score, also provide game information tournament details in brief. Also provide reward facilities which reward user with shorten amount of prize.

Analysis gives full information about the steps which followed continued to deploy the project; it’s determining the project goal. Analysis helps to find problems identification, requirement gathering, and system prototype designing. This system operates on android, so it’s easy to focus on user aspect more than technical aspect which makes user-friendly GUI. For more organize database we draw primary class diagram, use cases which clear our requirement essential for this system. This requirement cover all accepts which can understand by all technical and non technical users.

On this analysis state I choose soft approach methodology which collect core information that allows us to maintain our system systemically also mark problem issue, & figure out core database flow diagrams.

## Need for analysis

* Customer’s needs should be identified.
* System performance evaluation for feasibility.
* Perform all economic technical analysis.
* Establish functions then allocate system elements.
* Establish schedule for project.
* Establish constraints and create system definitions.

## End-user development objectives plans follow

* Software features should be specific and written in clear, concise and understandable terms.
* Objectives that are impossible to accomplish are not realistic and not attainable objective should be centered in reality.
* Objectives should have a time frame with end date assigned to them.
* Project should be measurable we should know when you will successfully complete the task.
* The software should direct the project scope efficiently.
* The software features should replace the legacy software system.
* The software features should clearly describe the plan assigned with responsibilities and accountabilities.
* The software features should mitigate the risk and focus on adoption.

## End user involved Software development problems

* Multiple complicated user level requirements.
* Globalization causing extremely high competition.
* Legacy system issues.
* Third party integration and interface issues.
* Software may not deliver consistently correct result or cannot be developed to work correctly each time in use.
* Inadequate software performance – slow system response time and transaction rate.
* Misleading and confusing data - data shown to the user may be correct but user might not fully understand how to interpret the data.
* Obsolete software – software can stop working due to new hardware or software changes.
* Inconsistent processing – software which works correctly in one environment may not work in another environment.
* Inadequate security control – unauthorized access to the system is not adequately controlled.
* Incorrect implementation of business rule – the mistake that occurs between what is indented to be developed or implemented and what is actually developed.
* May not accepted globally due to limited boundary.
* Less quality assurance.
* Can be more time consuming because of coordinates and compromise.

## Solutions apply for end user developments

* Maintaining good communication.
* Understanding the user, to make user centric we should know what an user want.
* Testing products.
* Keeping up with the latest threads.
* Learning new coding practices and tools which mean you will get better at creating code and can develop more innovative products.
* Correct implementation of business rule.
* Proper time estimation.
* Breaking down tasks so that the tasks can be more manageable and less complicated.
* Avoiding working with others code /spend more time reading and understanding codes.

## Object oriented analysis

This is the process which interacts with each other groups’ item, by class, data or behavior to create a model object that represents accurately whole system implementation who developed by model. Object oriented analysis help to determine actual diagrams in form of class diagram, ER-diagram, use cases. It’s identified clear view of system class which operates the function on the program.

## Advantage of project analysis

* Learn about new skills and technique.
* Achieve project goal effectively
* Develop new skills for organization projects
* Team management analysis
* Risk analysis and management.
* Human influence interacted core communication.

## Disadvantage of project analysis

* Report all goals which makes complex determine extract outcome.
* All Requirements data not clear to process.
* It does not prevent all risk shown in project development.
* Effects on project management enlarge time period.
* Cost prioritization not to be set clearly.

## Functional Requirement

Requirement includes all the background business cases, program goals and objectives, functional, non-functional requirements. This project follows some basic functions which supports user requirement, terms condition which operates programs fundamentals.

**Device requirement for my program**

Basic needs for running my android application

|  |  |
| --- | --- |
| **Operating system** | Android mobile, supports lollipop version |
| **Ram** | Minimum 512MB |
| **Memory space** | Minimum 50MB |
| **Program version** | V.1.0 |
| **Android Package name** | Com.gamepoints |

## Functional and non-functional

Functional requirement are those technical functionality of system, non functional requirement is identification of function to judge the operation of system in some condition, more than specific behaviors.

|  |  |  |  |
| --- | --- | --- | --- |
| **SN** | **FUNCTIONAL** | **SN** | **NON-FUNCTIONAL** |
| F1 | Login user | NF23 | Update admin profile |
| F2 | Login admin | NF24 | Download game |
| F3 | Tournament video view | NF25 | Update user profile |
| F4 | New registration | NF26 | View admin list |
| F5 | Save play score | NF27 | View reward list |
| F6 | Update tournament information | NF28 | Delete reward |
| F7 | Delete user | NF29 | Block user |
| F8 | Update location | NF30 | Secure game |
| F9 | View win user | NF31 | Delete attend tournament |
| F10 | Delete game | NF32 | Get payment |
| F11 | Delete tournament | NF33 | Set payment |
| F12 | Update game time | NF34 | Delete payment |
| F13 | Update tournament time | NF35 | User password change |
| F14 | View user score list | NF36 | Upgrade user level |
| F15 | View game list | NF37 | View another user profile |
| F16 | View profile | NF38 | Bank transfer |
| F17 | View total play score | NF39 | No admin access for user |
| F18 | Delete old tournament post | NF40 | Pause game |
| F19 | view reward details |  |  |
| F20 | Update reward |  |  |
| F21 | Play game |  |  |
| F22 | Track user |  |  |
| F23 | Logout user |  |  |
| F24 | Exit game |  |  |

## MOSCOW requirement prioritization

Moscow apply DSDM approach where time has been fixed, it’s understand relative importance of work done in order progress and deploy program within deadlines. This prioritization technique helps to understand and manage priorities upon program requirements. Moscow stands for:

* **M**ust Have
* Not without it, unsafe without it, cannot deliver without it.
* **S**hould Have
* Important but not vital this time, painful to leave but solution is still viable.
* **C**ould Have
* Give less impact if left, wanted but less important.
* **W**on’t Have this time
* Project requirement listed but not used in related in current project.

**My program functional Moscow requirement prioritization:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Must Have** | **SN** | **Purpose** | **ID** |
| Login user | 1 |  | F1 |
| Login admin | 2 |  | F2 |
| New registration | 3 |  | F4 |
| Play game | 4 |  | F21 |
| Save play score | 5 |  | F5 |
| Exit game | 6 |  | F24 |
| Logout user | 7 |  | F23 |
| **Should Have** | **SN** | **Purpose** | **ID** |
| Update reward | 1 |  | F1 |
| View profile | 2 |  | F2 |
| Update location | 3 |  | F4 |
| Delete user | 4 |  | F21 |
| View game list | 5 |  | F5 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Could Have** | **SN** | **Purpose** | **ID** |
| Login user | 1 |  | F1 |
| Login admin | 2 |  | F2 |
| New registration | 3 |  | F4 |
| Play game | 4 |  | F21 |
| Save play score | 5 |  | F5 |
| **Won’t Have** | **SN** | **Purpose** | **ID** |
| Login user | 1 |  | F1 |
| Login admin | 2 |  | F2 |
| New registration | 3 |  | F4 |
| Play game | 4 |  | F21 |
| Save play score | 5 |  | F5 |

## NLA Natural language Analysis

Natural Language Analysis is the process of classification of noun verbs used in the task that extract used able class property that’s help to indicate appropriate class for each suitable data on meeting schedule, **NLA** filtrations nouns verbs adjectives nouns required as potential class/attributes adjectives as the attributes if any possible approval required, Verbs as inherent potential methods but not as actual method.

### Step 1: Noun identification for potential classes

* Scenario brief was studied thoroughly and all nouns were identified.
* Filtration was applied on nouns to get final list of potential classes.
* Redundant nouns were removed.
* Synonymous nouns were removed.
* Data carrying capacity of noun was checked. Data assigned with one appropriate value

### Step 2 – Verb identification for potential methods/operations

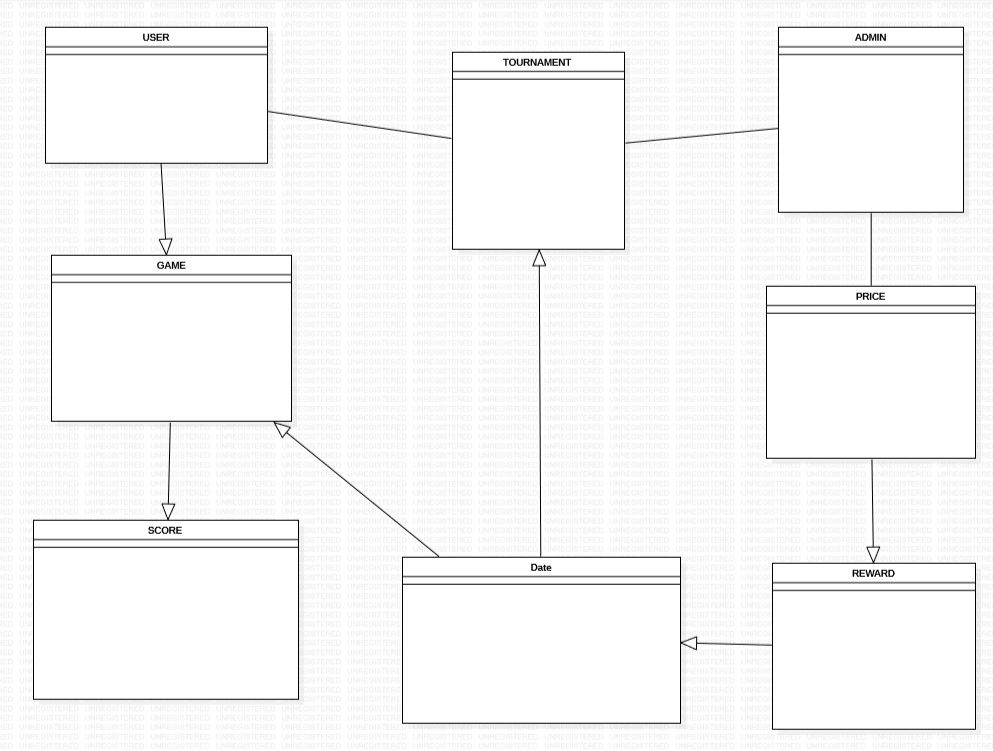
* Filtration was applied on verbs to get final list of potential methods.
* Redundant verbs were removed.
* Synonymous verbs were removed.
* Architecture level actions and objects were skipped saving to databases and creating views

### Class name NLA

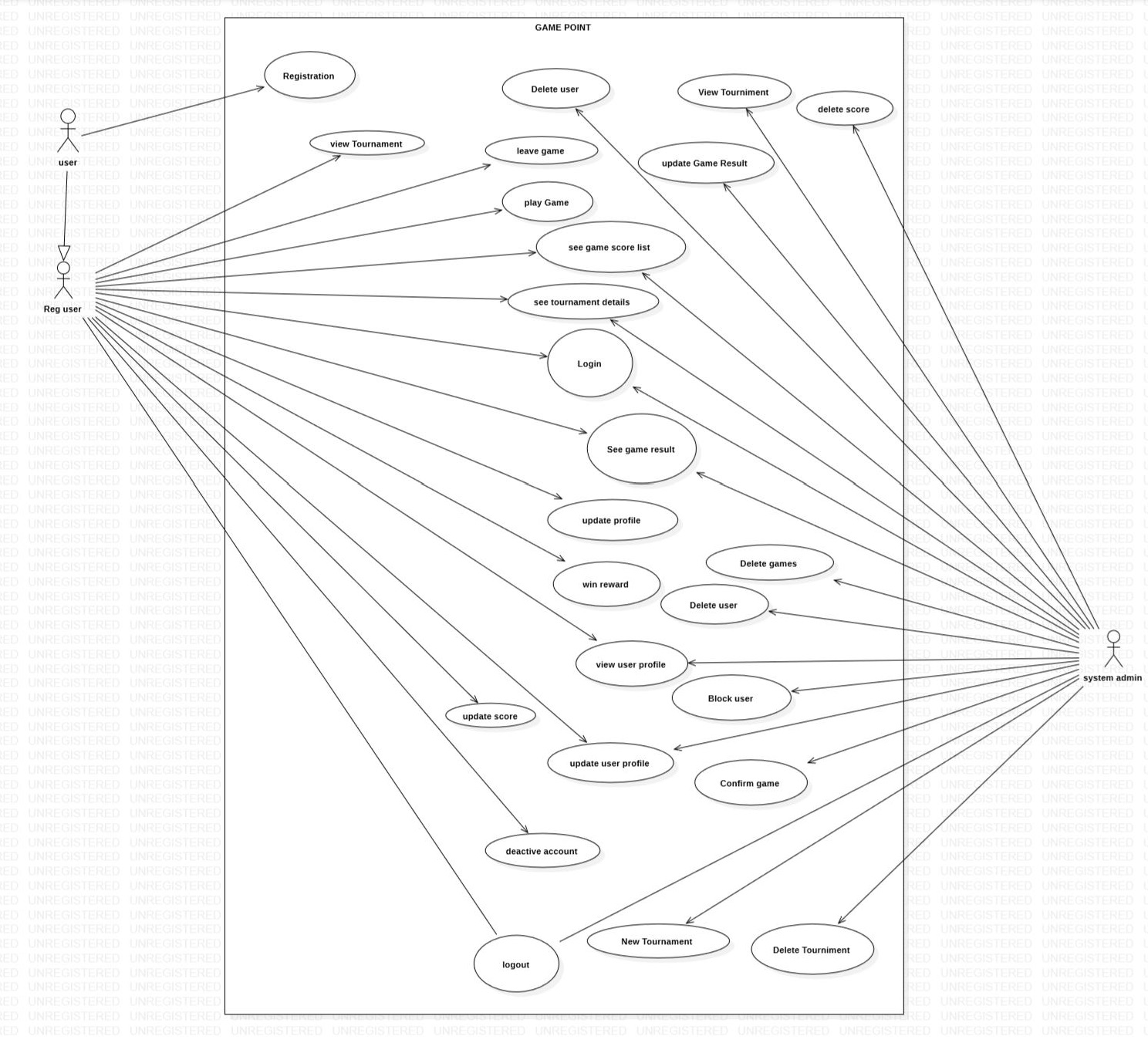
1. Admin
2. User
3. Game
4. Play
5. Score
6. Location
7. Tournament
8. Reward
9. Price
10. Coins
11. Level upgrade
12. Save score
13. User login
14. Admin login
15. Admin edit
16. User update
17. User delete
18. Game delete
19. Game update
20. Play Game
21. Tournament update
22. Tournament delete
23. Post update
24. Post delete

## Initial class diagram

Initial class diagram help to define actual class diagram which help to define some class identification. Initial class diagram is the design of the relations and their source code dependences between the classes. This is also known as structural model of given system. The real implementation is yet to come, in given diagram method and attributes are contains. Class diagram illustrations the classes of the system and interrelationship between these classes.



## Use case

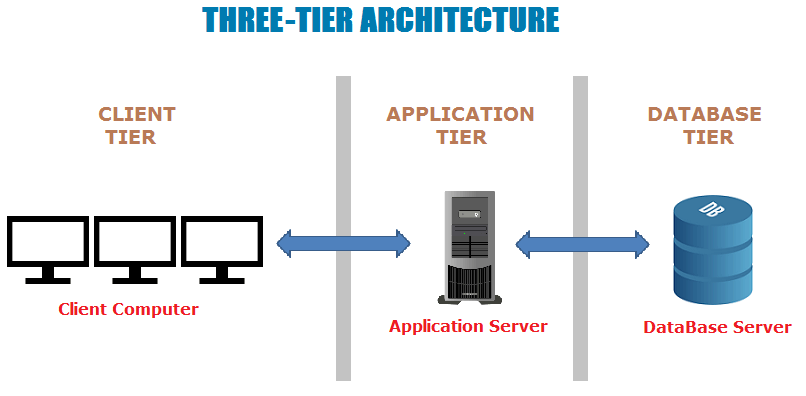


## Architecture

### Three Tier Architecture

Three tiers Architecture is a pattern used for a completely different reason. It separates the entire application into meaningful design groups UI, Business Logic, & Data storage servers. So, three tier applications refer to all the related code in the application, MVC pattern is a pattern used in the UI tier as view controller.

* Scalability: each tier can scale horizontally. For example, you can load-balance the Presentation tier among three servers to satisfy more Web requests without adding servers to the Application and Data tiers.
* Performance: because the Presentation tier can cache requests, network utilization is minimized, and the load is reduced on the Application and Data tier model controllers. If needed, you can load-balance any tier.
* Availability: If the Application tier server is down and caching is sufficient, the Presentation tier can process Web requests using the cache to collect data resources.

****

# Design Patten

### MVC Design Patten

Model View Controller (MVC) design pattern

**Model** - The model handles the business logic.

**View** – It handles the re-representation of the elements in the user interface.

**Controller** - It allows user to interact with the model. It acts as interface between the model and the view components to process all the business logic and incoming requests. It provides the ability to manipulate the system.



Fig: MVC design pattern

**Advantages of MVC design pattern:**

* MVC helps to develop an application in rapid way. It supports parallel development. A programmer can work on view and controller simultaneously.
* This makes development faster.
* There is less chance of code duplication because it separates data and business logic from the display.
* It also supports asynchronous technique, which helps developers to develop an application that loads very fast.
* Modification does not affect the entire model as model part does not depend on the view parts.

**Disadvantages of MVC design pattern:**

* Because there are three layers in this MVC pattern, it can increase the complexity.
* Developers must have knowledge on multiple technologies.
* There is lack of efficiency of data access in view layer.
* This may need multiple programmers.

# Chapter 3: Design

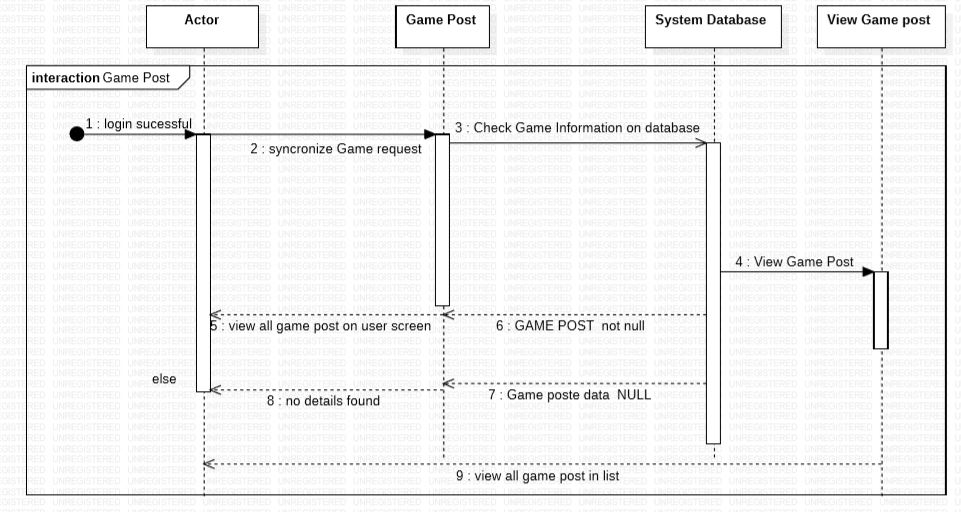
## Justification

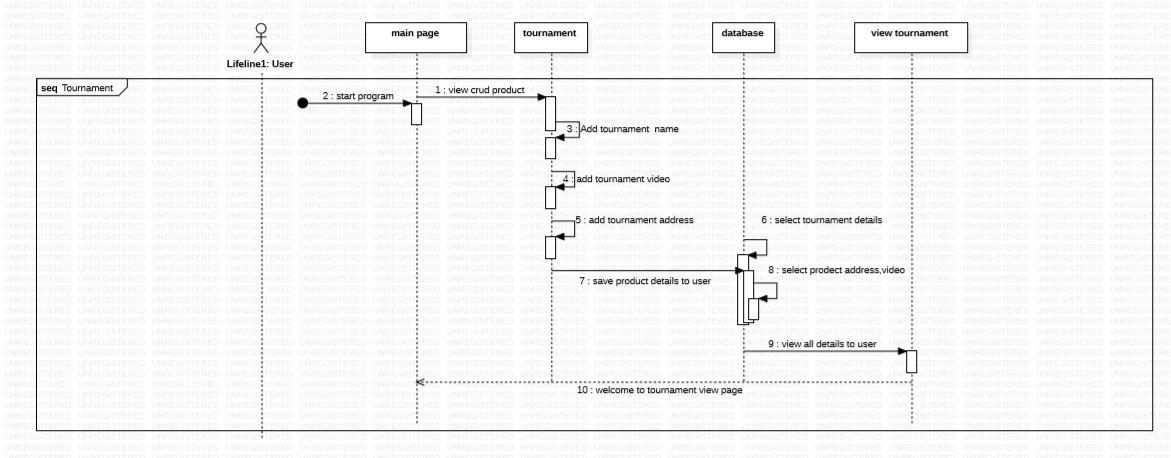
In this Design project, here provided all detail about information system characteristics of the needed project as well as detail about information of components, modules, interface and data. That information is directly as well as indirectly involved for scheme development. Design is the most important stage for any system development. In this design stage original system's blue print were created. There are many tools and methods to make design project. Design project divided into three different steps that are: structural model, behavioral model and database model.

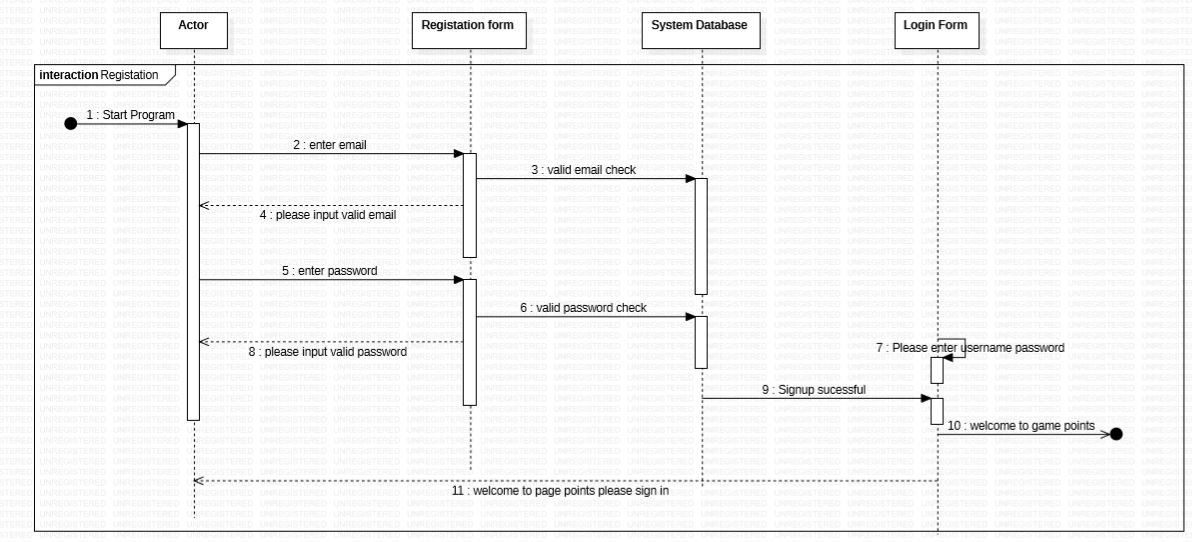
## Dynamic modeling

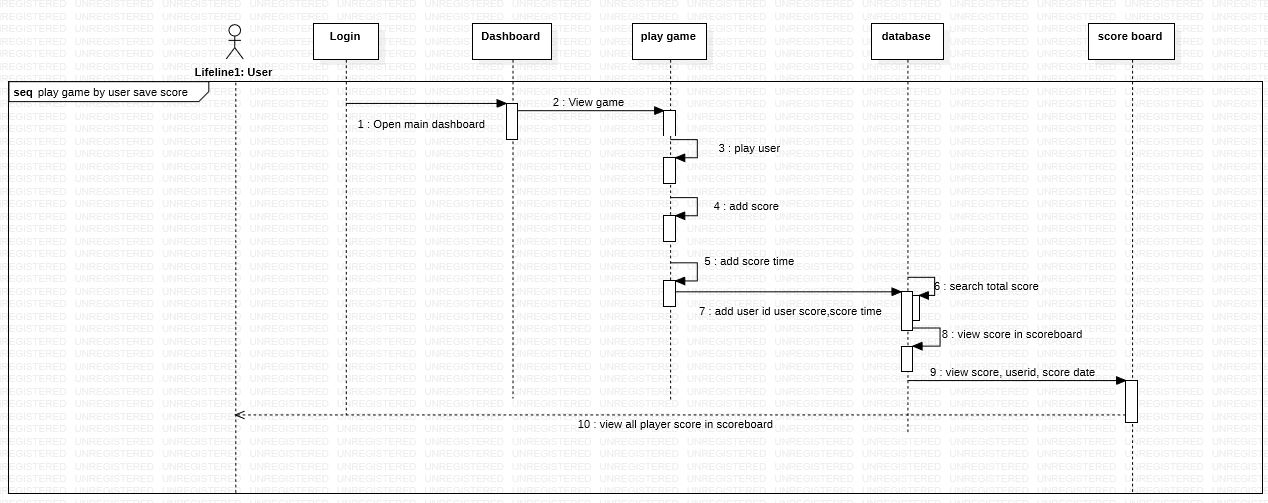
## Sequence diagram

Sequence diagram is very important UML model which displays the transaction of message between the life lines also shows the cooperation of the built object on similar time sequence. They generally does not implemented directly. This is focused on mainly dynamic modeling that categorizes every performance of the above system. Its demonstrations the orders in which approaches are raised in a system, Sequence diagram shows the dependencies of both i.e. methods as well as objects. The diagram representation of sequence is drawn and shown below:



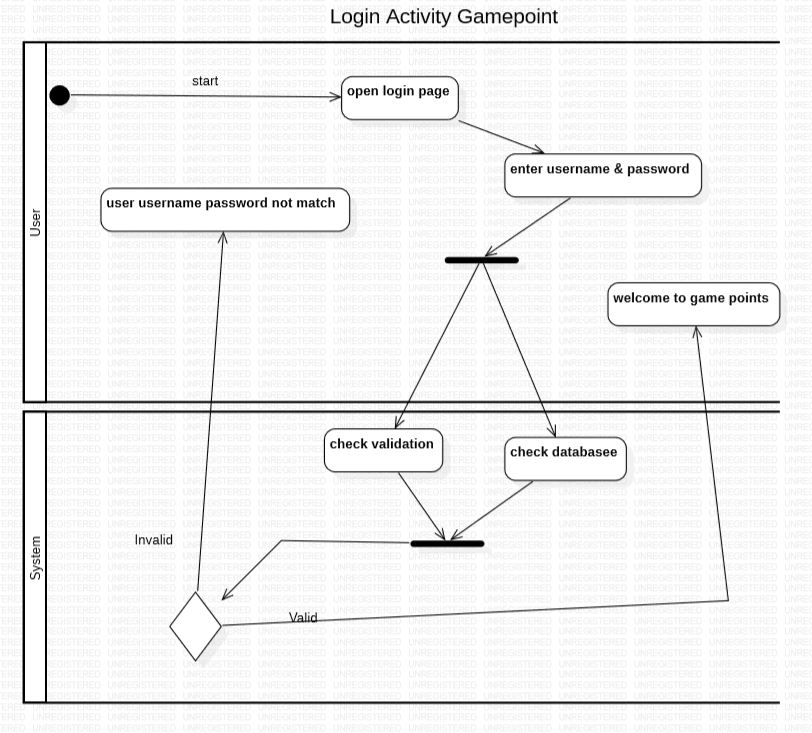


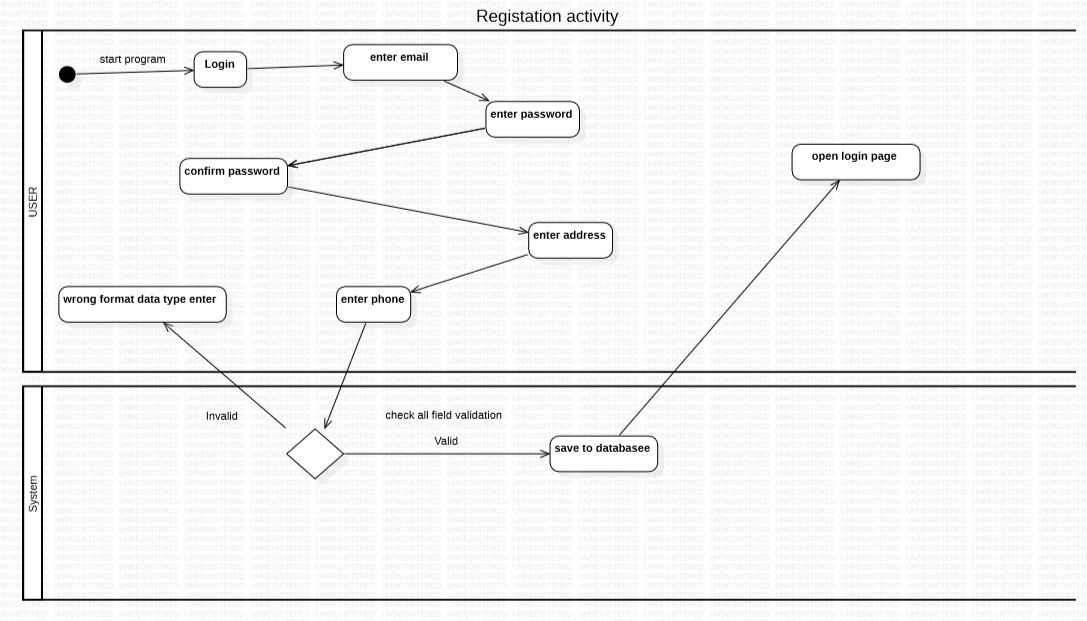


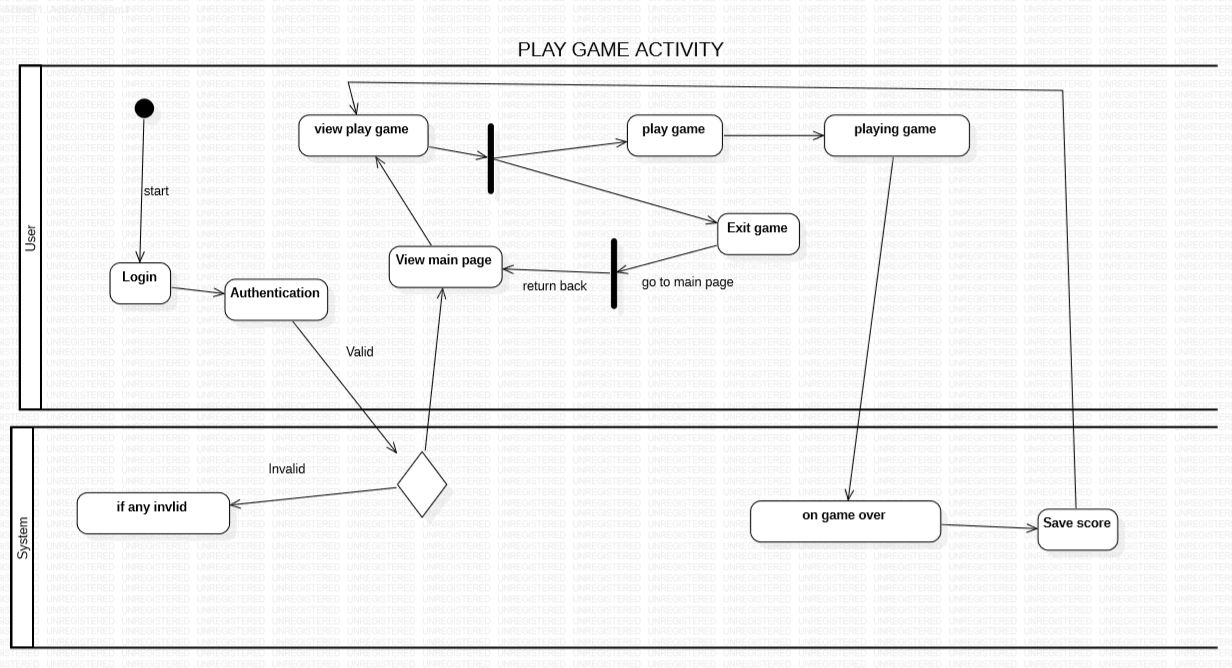


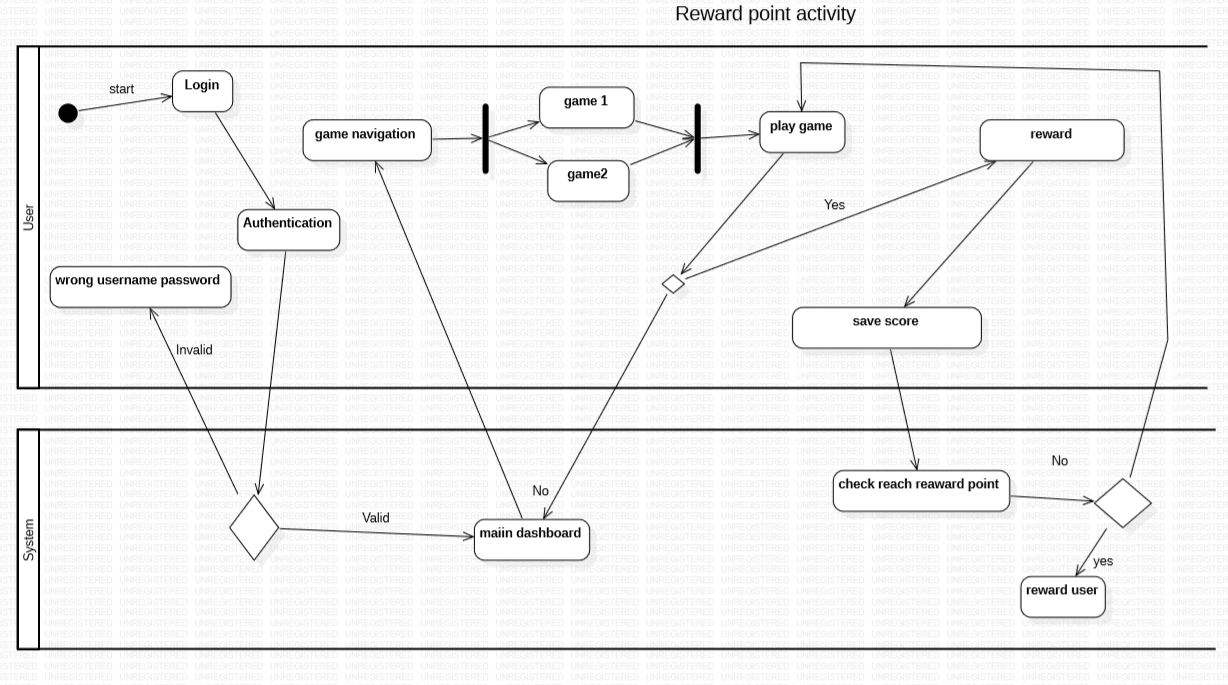
## Activity diagram

Activity diagram is also dynamic representation of system which defines different part of the system. They are most likely on flow chart so it is also called work flow diagram. It shows all control and flow which involve in game management system. In activity diagram different representation and symbols are used to describe the user action to the game management system. The diagram representation of activity is drawn and shown below:







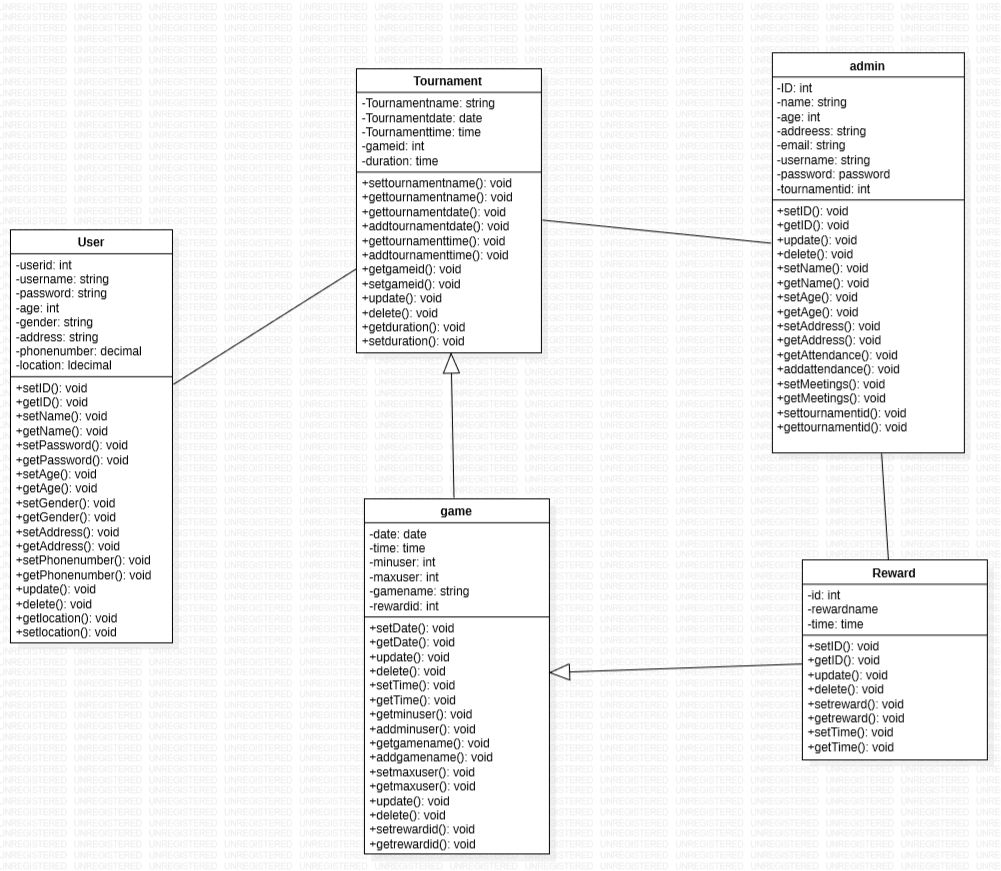


## Structural modeling

Structural model is simply known as UML Model which defines the process which shows the fixed structure of the system. Structural model is used to imply overall architecture of this system. The different level of abstraction and their relationship is presented to each other. The best example of Structural model is a Class diagram which is defined as the fixed structure of an application that labels the total understanding of the project.

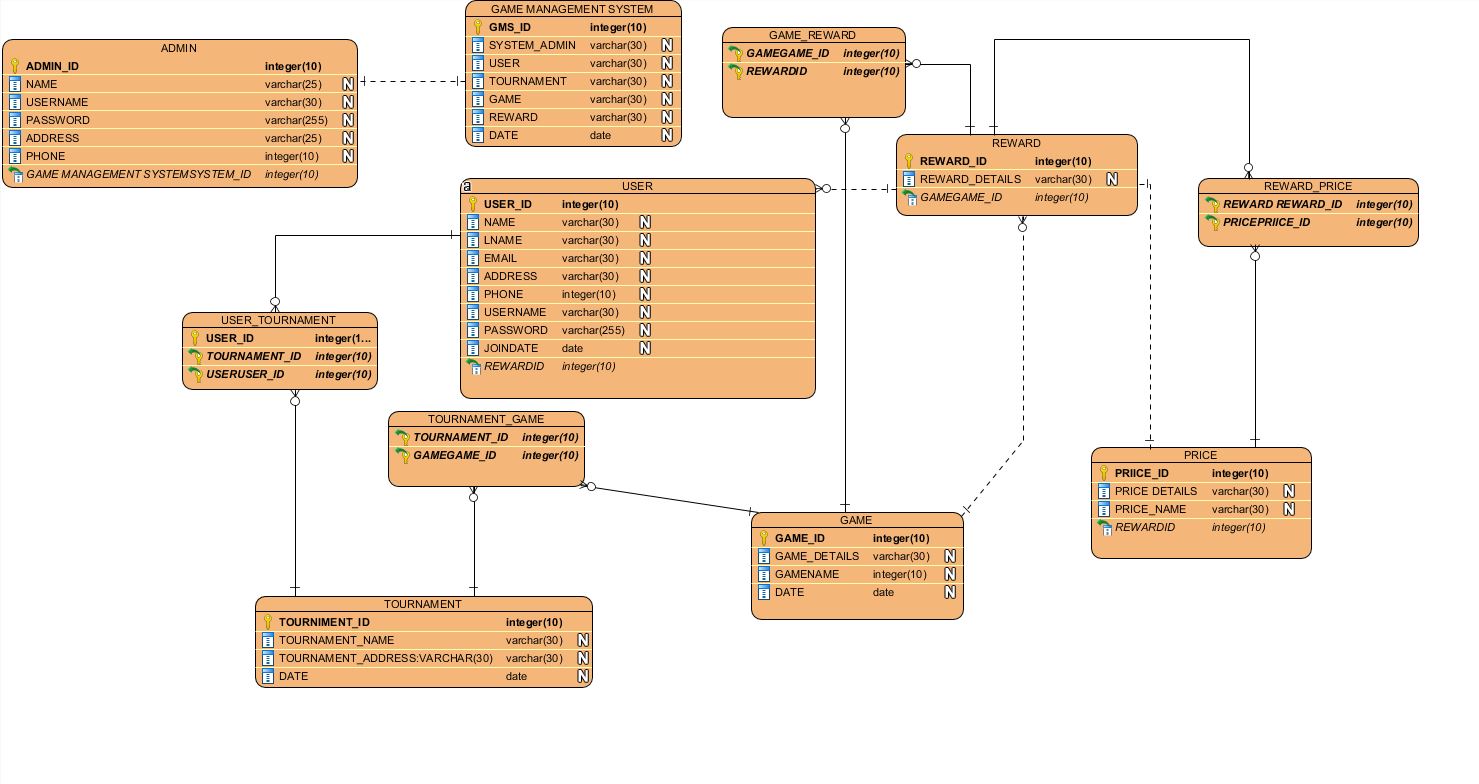
## Architecture level class diagram

Class Diagram is a static model which clarifies every class involved in the system, relation between classes as well as attributes, objects and also methods are used in same system. It is easy to understand the system as they helps to define, identify and explain all important aspect. They is pure interaction among class and objects. This is used mostly when development needs object oriented (OO) system. There is some important component in class diagram they are:

* Class: Which includes methods as well as properties
* Attributes: All the explanation of a class.
* Generalization: Relationship between classes which is super class and general class.
* Association: In this, Relation between two different classes represented.
* Multiplicity: Permits statements for the number of objects are include in relationship.
* Aggregation: Regulate fact relationship between classes.

## Database modeling ER

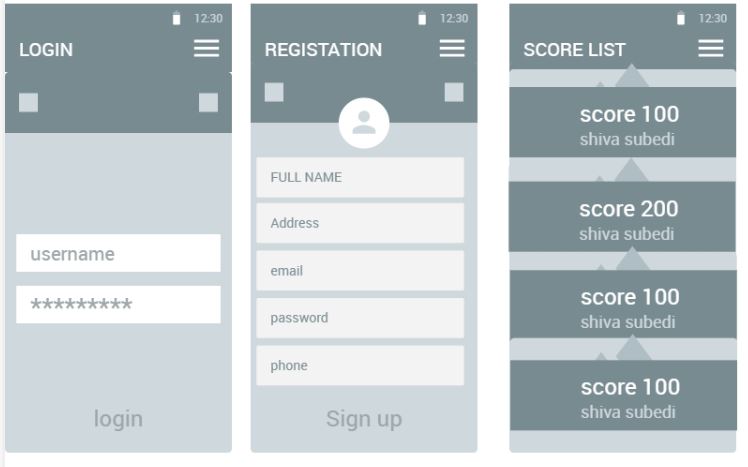
ER diagram is also a logical structure which shows the relationship among entities stored in a database. It describes how the components and entities are related together in the system.

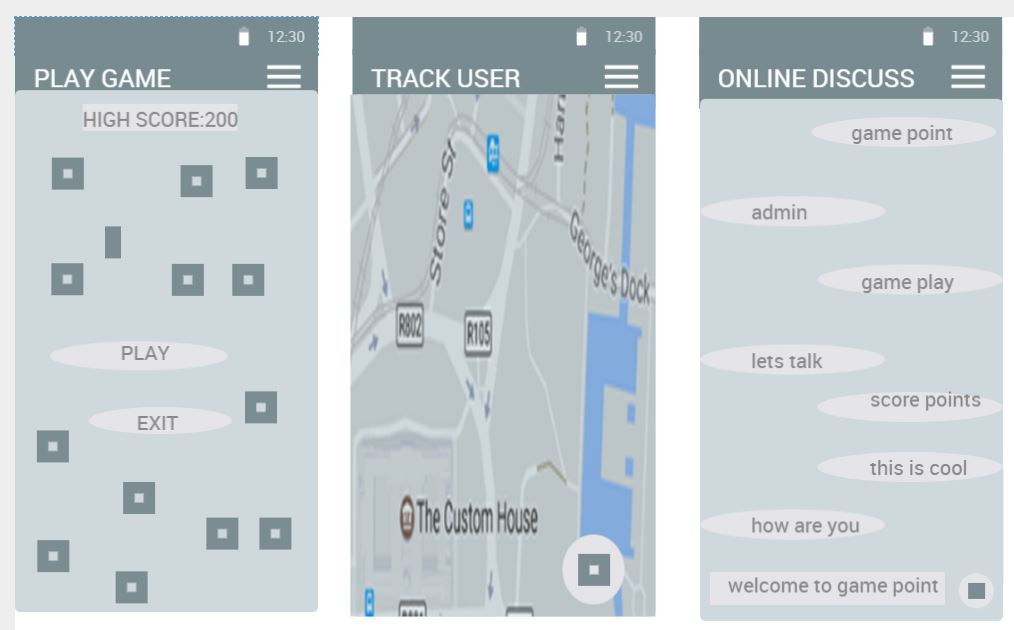


## Data dictionary

It is also known as metadata which is blue print of data. Data dictionary outlines form of simple information of metadata for game management system.

## UI-modeling (prototype)





## Chapter 4: Implementation

## Which language

Game Management System is a three tier base application which will be developed using **Android** **as MVC design pattern,** this system will be developed using **ANDROID STUDIO** because it will be easier to manage **FIREBASE DATABASE** and it is easily accessible by the help of internet. Open source android development tools manage interface & data servers to control the accessible credits. Android SDK provides the tools and APIs necessary to begin developing applications on the Android platform using the Java programming language.

## Framework snapshot

## Chapter 5: Testing

Testing is the process software evaluating or its components with core specific requirement which identify any gaps error or missing requirement in system application.

## Test cases

## Test log

## Black Box testing

## Unit testing

## 

## Chapter 6: Other project issues

Users can self register themselves in the system and can participate in the game tournament. There will be limits on number of participants in a single tournament. After that, the system collects data to make game planning and schedule game timetable. This system will also store the gaming details in the cloud database to collect information; the cloud backup provides more protective database system and also generates the output prints of each game tournament.

## Problem Statement

This system helps to manage the gaming tournament in an efficient way by doing a proper scheduling of the tournament. This helps to organize the tournament successfully without any disturbance or problem. It also helps to declare upcoming tournament as sustainable chart, each user is easily connected to the server through their devices and individual user gets easy access to their credentials. This project will access database as well as development of the entire application with help of android development tools and research the necessary information as well as build a proper application for a tournament purpose.

### Description of the project

#### 1.3.1 Feature of the project is as follows:

* Admin can add new tournament.
  + User could get new access on each game
* Admin can update the tournament details.
  + If game schedule change admin change details also user get easily notification
* Admin can schedule the tournament.
  + Admin know about all tournament on each individual game date time
* Admin manages payment details.
  + No privacy access to every user about financial statement
* Admin edit, update, and delete game information
  + If any game has been cancel admin easily update throw the system
* Users can create their profile by signing up member
  + Enter their name user id its give uniqueness on each user
* Users can login to the website
  + Using their email or username they easily get access on game tournament
* Users can participate in the tournament
  + Each user get access on one tournament on one time
* Users can see the tournaments details
  + User know about their coming game details
* Users can update their account details
  + User update their profile password user details
* User view their score and leader board
  + User get their total score total game on their profile
* User pays their game payment throw online
  + Secure payment system available for each user

## Scope and Limitation of project

The users won’t be able to pay online while participating in the tournament. This system won’t be able manage all the gaming tournaments. This system will be designed to manage few gaming tournaments. To manage more gaming tournaments, it requires

Lots of work and, it requires a lot of time and commitment. But, it can be managed in the coming future.

### 2.2 Aims and Objective

**Aims**

* To help organizers to organize a gaming tournament successfully.
* View the game result by the help of cloud servers.
* Payments done on the tournament are more securely regulated.
* Gives manageable platforms to operate game tournament events.
* Easily accessible on any android devices.
* To create a more user-friendly digital environment.

Objectives

* To perform better analysis of the system.
* To perform better design.
* To perform better coding.
* To perform different tests to find and solve the flaws in the system.
* To deploy the working system.

## Work Breakdown Structure

It means the process of dividing the whole project into small pieces of task. It is done to remove the complexity in the project and to manage the project easily. It is represented as a hierarchical subdivision of a project into work areas with the lowest generally being a work package or sometimes even an activity. The WBS provides a foundation for all the project management work including planning, cost and effort estimation, resource allocation, and scheduling. The broke down of the project is shown below:

* **Project Management**: Scoping, planning, monitoring & controlling will be done in this part.
* **Analysis:** Requirements use cases and architecture that are done in analysis part.
* **Design:** Structural model, behavior model and UI design will be done in design part.
* **Implementation:** Building database and coding will be done in implementation part.
* **Testing:** Unit testing and integration testing will be done in testing part
* **Deployment:** User manual and final report will be done in this part.

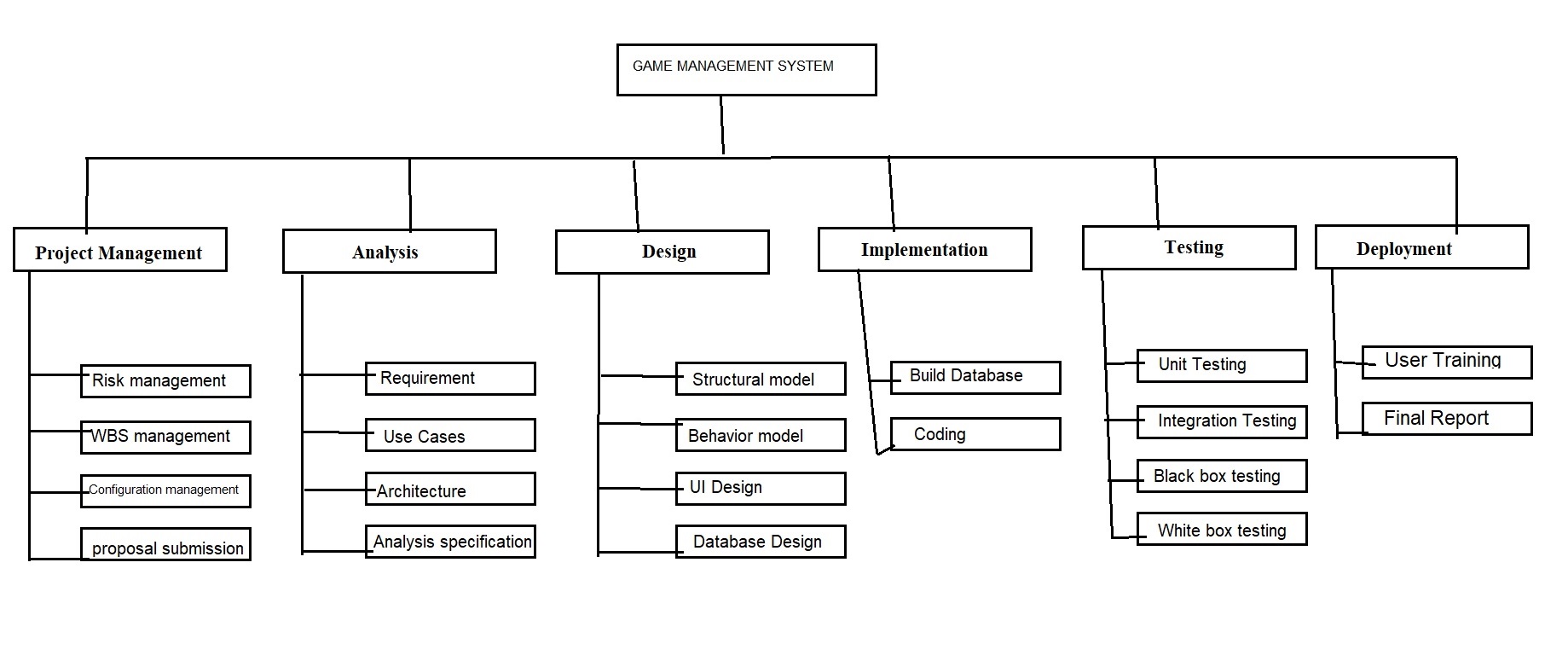


Fig: WBS of Game management system

## Milestones

|  |  |  |  |
| --- | --- | --- | --- |
| **Milestones** | **Start Date** | **End Date** | **Total days(124)** |
| **Project Management**  Risk Management  WBS  Configuration Management  Proposal Submission | 12/21/2018  12/21/2018  01/26/2018  01/01/2019  01/05/2019 | 01/08/2019  12/25/2018  12/31/2018  01/04/2019  01/08/2019 | 19  5 |
| **Analysis**  Requirement analysis  Use Case  Architecture  Analysis specification | 01/09/2019  01/09/2019  01/16/2019  01/22/2019  01/30/2019 | 02/02/2019  01/15/2019  01/21/2019  01/29/2019  02/02/2019 | 25  7  6 |
| **Design**  Structural Diagram  Behavioral Diagram  UI Design  Database Design | 02/03/2019  02/03/2019  02/08/2019  02/15/2019  02/22/2019 | 02/27/2019  02/07/2019  02/14/2019  02/21/2019  02/27/2019 | 25  5  7 |
| **Implementation**  Building Database  Coding | 02/28/2019  02/28/2019  03/05/2019 | 03/24/2019  03/04/2019  03/24/2019 | 25  5 |
| **Testing**  Unit Testing  Integration Testing  Black box Testing  White box Testing | 03/25/2019  03/25/2019  03/30/2019  04/04/2019  04/09/2019 | 04/13/2019  03/29/2019  04/03/2019  04/08/2019  04/13/2019 | 20  5 |
| **Deployment**  User Training  Final Report | 04/14/2019  04/14/2019  04/19/2019 | 04/23/2019  04/18/2019  04/23/2019 | 10  5 |

**Fig: Milestones table**

## Description of Milestones:

For Game management system, I have estimated 124 days in total. For project management 19 days, for analysis 25 days, for design 25 days, for implementation 25 days, for testing 20 days, and for Reporting 10 days. Each activities of the project are estimated which are shown below:

|  |  |  |
| --- | --- | --- |
| **No.** | **Task Name** | **Days** |
| **1** | **Project Management** | **19** |
| 1.1 | Risk management | 5 |
| 1.2 | WBS management | 6 |
| 1.3 | Configuration management | 4 |
| 1.4 | proposal submission | 4 |
| **2** | **Analysis** | **25** |
| 2.1 | Requirement | 7 |
| 2.2 | Use Cases | 6 |
| 2.3 | Architecture | 8 |
| 2.4 | Analysis specification | 4 |
| **3** | **Design** | **25** |
| 3.1 | Structural model | 5 |
| 3.2 | Behavior model | 7 |
| 3.3 | UI Design | 7 |
| 3.4 | Database Design | 6 |
| **4** | **Implementation** | **25** |
| 4.1 | Build Database | 5 |
| 4.2 | Coding | 20 |
| **5** | **Testing** | **20** |
| 5.1 | Unit Testing | 5 |
| 5.2 | Integration Testing | 5 |
| 5.3 | Black box testing | 5 |
| 5.4 | White box testing | 5 |
| 6 | **Deployment** | **10** |
| 6.1 | User Training | 5 |
| 6.2 | Final Report | 5 |

Fig: Time Estimation Table

## Scheduling

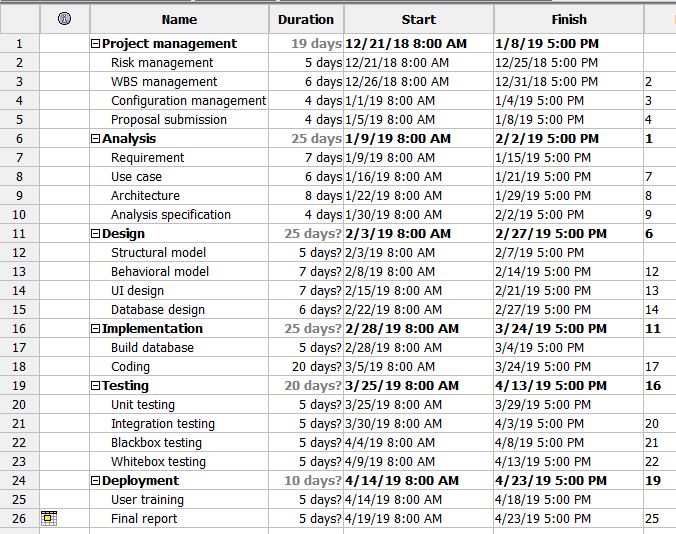
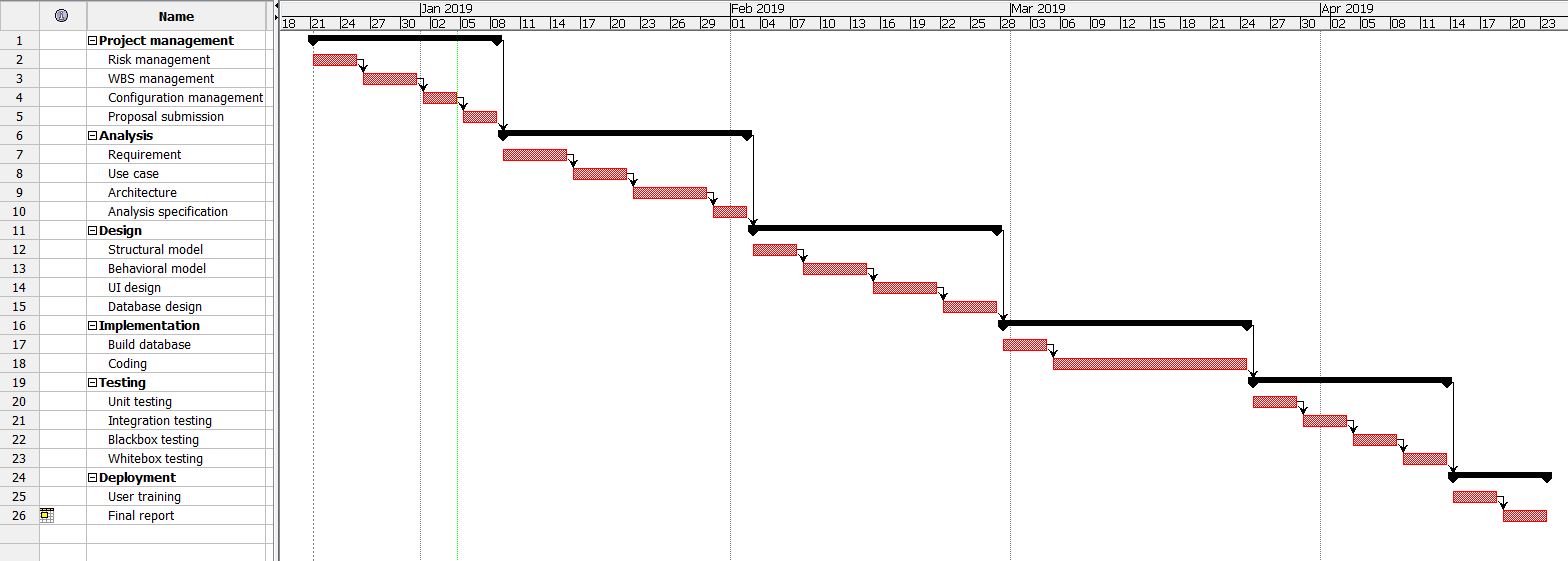


Fig: Schedule

## Gantt chart

The Gantt chart is shown below:

## Risk Management

Risk management means the process of identifying the potential risk in advance, analyzing them and taking precautionary steps to minimize the risk.

We need to do risk management plan, it helps to minimize the unexpected risks and extra costs before they happen. By doing the risk management task, we can save extra money and protect the organization’s future.

The steps of risk management are shown below:

* Risk identification – What can go wrong?
* Risk analysis – How will it affect us?
* Risk control – What should we do?

Risk treatment – If something does happen, how will you manage it?

**Likelihood**

|  |  |
| --- | --- |
| **Likelihood** | **Value** |
| Low | 1 |
| Medium | 2 |
| High | 3 |

Fig: Likelihood table

**Consequences**

|  |  |
| --- | --- |
| **Consequences** | **Value** |
| Very Low | 1 |
| Low | 2 |
| Medium | 3 |
| High | 4 |
| Very High | 5 |

Fig: Consequences table

**Impact = Likelihood \* Consequence**

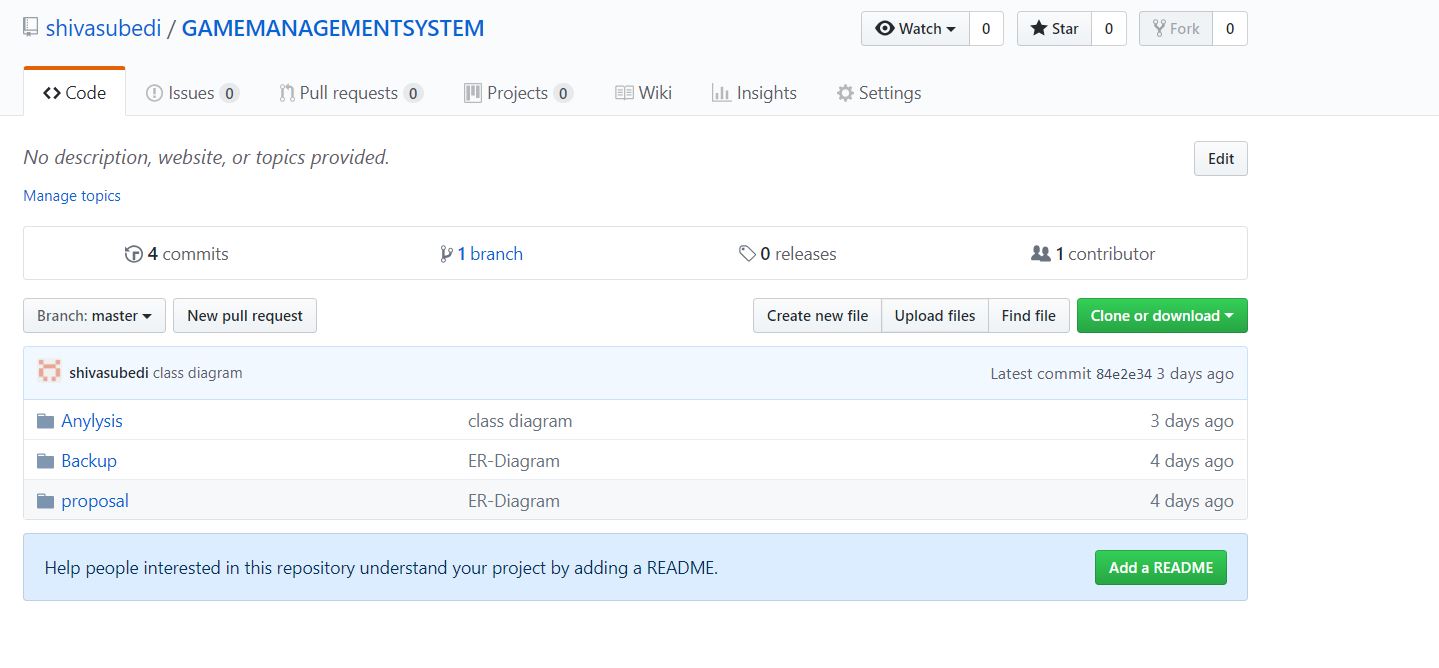
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Risk** | **Likelihood** | **Consequences** | **Impact** | **Action** |
| Lack of resources | **1** | **5** | **5** | **All the resources needed for the project should be available** |
| **Os failure** | **2** | **4** | **8** | **File backup should be done** |
| **Electricity cut** | **1** | **5** | **5** | **Ups should be set up and backup should be done** |
| **Human illness** | **3** | **2** | **6** | **Time should be manage properly extra work time should be done after healing** |
| **Human mistake** | **3** | **3** | **9** | **Testing should be done properly** |
| **Infected by virus** | **2** | **4** | **8** | **Antivirus should be installed** |
| **Hard disk crash** | **1** | **4** | **4** | **Cloud backup should be done** |
| **Natural disaster** | **1** | **5** | **5** | **File must be backup on secondary devices** |
| **Serious injury** | **1** | **5** | **5** | **Next another developer should be hired** |

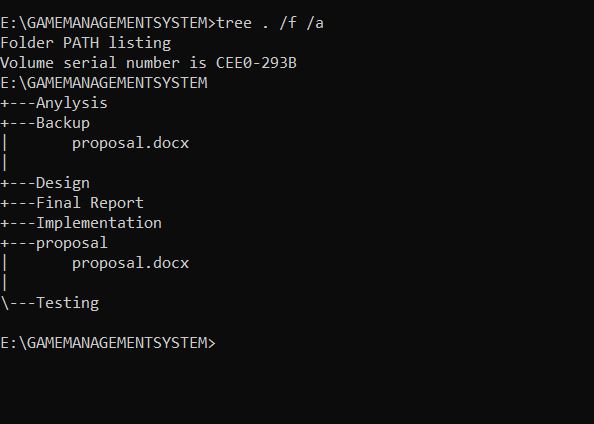
**Fig: Risk Table**

## Configuration Management

Configuration Management means identifying, tracking and protecting a project’s products from unauthorized change. It is used to increase the efficiency, performance and reliability of the system. Files are kept in an organized form so that it is easily accessible from everywhere. It’s protecting entire source code from unwanted threads.

My directory structure is shown below:





## Chapter 7: Conclusion

Hence, this project is mainly focused on managing the gaming tournament easily and properly, this helps to automate the work of the organizer. Provide secure system, bug free code. Clean lines of code with comment. Almost every work that needs to be done by the organizers can be done by using this system. This system handles all the work from registering teams, proper scheduling, and game win reporting the results of the tournament. So, this project helps to take burden from the team of organizers and get access throw online cloud database that helps to lead the organization.

## Chapter 8: Reference and bibliography

## Chapter 9: Appendix