Cairo University  
Faculty of Computers and Information



**CS251**

**Software Engineering I**

OnlineSchool

Software Design

A3

December & 2016

Contents

[Team 3](#_Toc468575261)

[Document Purpose and Audience 3](#_Toc468575262)

[System Models 4](#_Toc468575263)

[I. Class diagrams 4](#_Toc468575264)

[Important Algorithm 7](#_Toc468575265)

[II. Sequence diagrams 7](#_Toc468575266)

[Class - Sequence Usage Table 10](#_Toc468575267)

[Ownership Report 12](#_Toc468575268)

# 

# Team

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Email** | **Mobile** |
| **20140192** | **Amr Saeed Hosney** | **amrshosny@gmail.com** | **01125376468** |
| **20140196** | **Amr Magdy Ibrahim** | **a.magdy7077@gmail.com** | **01141148336** |
| **20140064** | **Ashrakat Mokhtar** | **ashrakatelnow@gmail.com** | **01114178143** |

# 

# Document Purpose and Audience

**1 Document purpose:**

Educational platform where educational games are created by teachers and played by students to enhance their skills.

**2 Audience that is interested in this document:**

1- Client: Computek Company.

2- Supervisor Teacher Assistant: Mohamed Samir.

# System Models

# I. Class diagrams

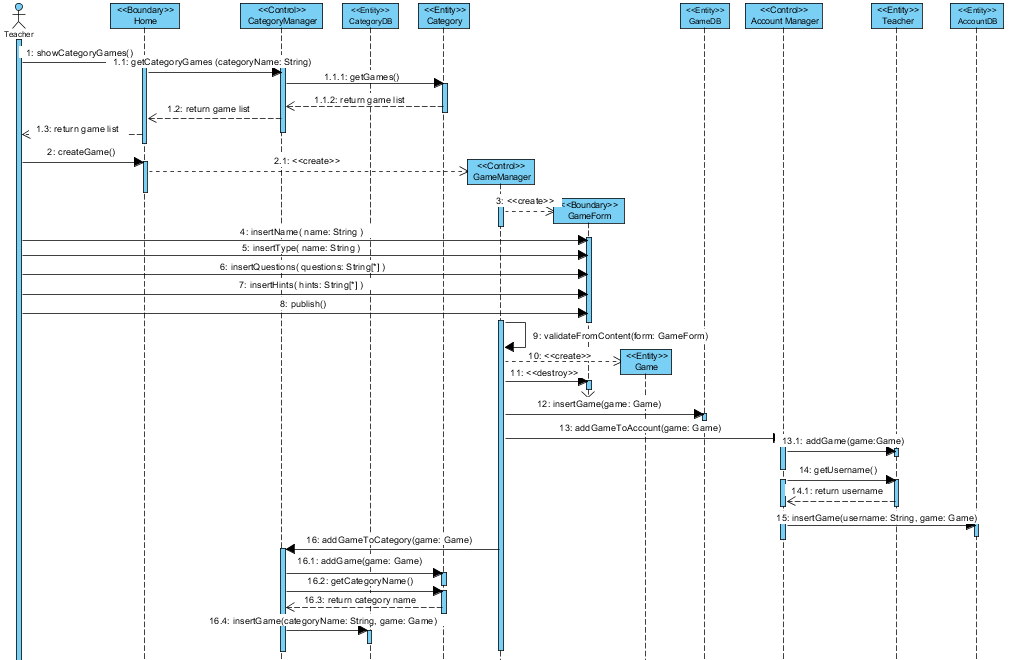
| **Class ID** | **Class Name** | **Description & Responsibility** |
| --- | --- | --- |
| **001** | CategoryManager | Controller class. Manages Categories' updates, Adds new games to category objects. Gets games of specific category by its name. Prevent any different type of objects to access "Category" object. "Act as mediator between Category object and any other objects". Deals with CategoryDB |
| **002** | AccountManager | Controller class. Manages Accounts' updates. Add tournaments/games to the account object "teacher", update score of a player. "Act as mediator between Account object and any other objects". Deals with AccountDB |
| **003** | GameManager | Controller class. Responsible for creating Game Form and validating its content to create game object. Also activates GameInterface object. Deals with GameDB |
| **004** | TournamentManager | Controller class. Responsible for managing tournaments and validating the content of tournament forms |
| **005** | Home | Boundary class. Mediator between user and other classes (UI). Its job is to view categories, games belong to it, tournaments created by teachers. Also through it teacher can create both tournaments and games |
| **006** | Form | Boundary class. UI teachers deal with during creating game or tournament. Responsible to take data from them to complete the creation process. In any case (game or tournament) it takes Name, type (T/F, MCQ), Questions and confirmation from teacher. |
| **007** | GameForm | Boundary class. Inherited from Form class. Mainly for creating games and taking data from teachers. e.g. taking hint |
| **008** | TournamentForm | Boundary class. Inherited from Form class. Mainly for creating tournaments and taking data from teachers. e.g. taking date and duration |
| **009** | GameInterface | Boundary class. UI which user deals with when to choose to play any game. It is responsible for everything associated to the game (running a game, exit a game, showing scoreboard, getting score…etc.) |
| **010** | Category | Entity class. Acts as container for games. It has a name and collection of games. We can add more games in it and get one of them |
| **011** | Game | Entity class. Contains all information of a game. Its name, type, its questions and hints, player score, scoreboard and help. With its functions the player can get hint for specific question, play, get its score, see the scoreboard and also can see description of a game( getHelp() ) |
| **012** | Tournament | Entity class. Contains all information of a tournament. Its duration, date, name, type and Question |
| **013** | Account | Entity class. Parent class Contains all general information of a user. Its name, user name, email, password, Gender, Birthdate, score, and by it we can add his scores to his profile, with it we can get the user name |
| **014** | Teacher | Entity class. Child class inherited from Account class contains information belongs only to teachers. e.g. Professional reference. It allows teachers to add game and tournaments created by their own |
| **015** | Student | Entity class. Child class inherited from Account class contains information belongs only to teachers. e.g. educational level |
| **016** | CategoryDB | Entity class. Responsible for holding data of the categories and deals only with categoryManager<<Controller>>. By it we can retrieve any peace of information concerning category object and update it |
| **017** | GameDB | Entity class. Responsible for holding data of all Game objects has been created. Deals only with GameManager<<Controller>>. With it we store each detail about games. Can add new objects, update existing ones, retrieve objects of class Game |
| **018** | TournamentDB | Entity class. Responsible for holding data of all tournaments has been created. So, we can insert new tournaments and updates existing ones. By it we can retrieve all tournaments or specific ones |
| **019** | AccountDB | Entity class. Responsible for holding each piece of information concerning Account objects "Teacher and Student ". can retrieve specific object from it or all objects, update them and insert new ones |

### Important Algorithm

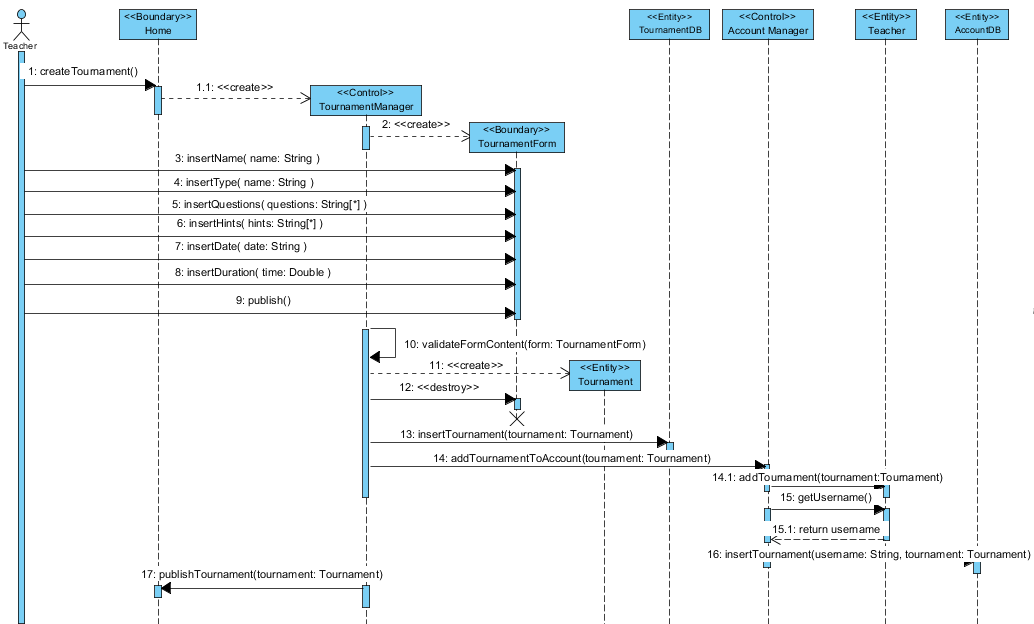
## In create tournament we will need to count the real time online since Declaration till tournament's end. Also the actual start of the tournament.

## II. Sequence diagrams

**101 – Create Game**

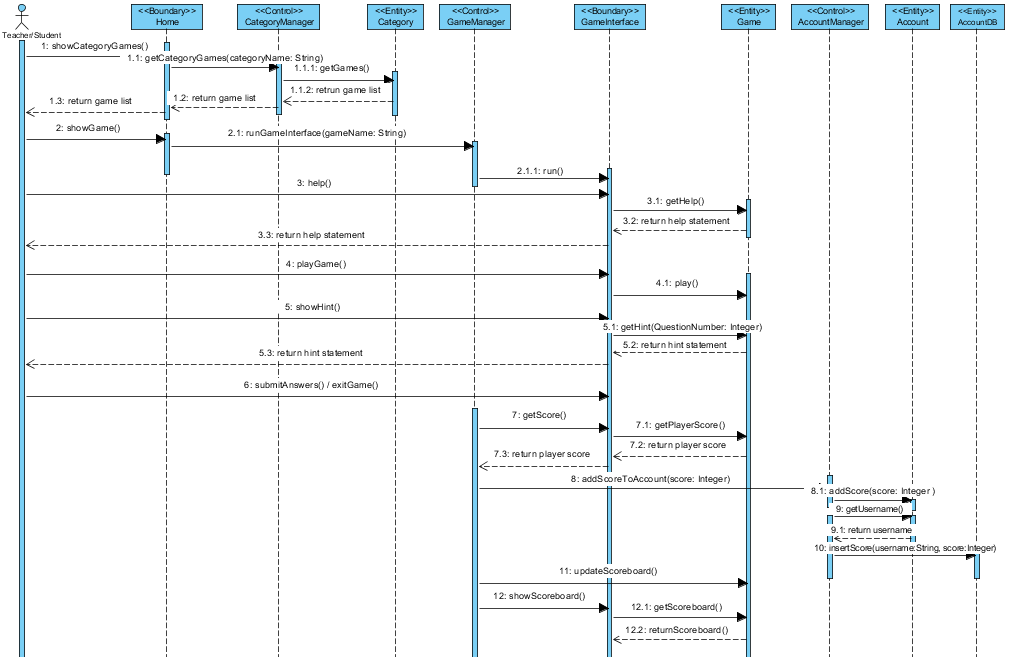
****

**102- Create Tournament**

****

### 

### 103- Play Game



### Class - Sequence Usage Table

| **Class Name** | **Sequence Diagrams** | **Overall used methods** |
| --- | --- | --- |
| AccountManager | 101 – 102 – 103 | addTournamentToAccount(Tournament)  addGameToAccount(Game)  addScoreToAccount(int) |
| CategoryManager | 101 – 103 | getCategoryGames(String)  addGameToCategory(Game) |
| GameManager | 101 – 103 | ValidateFormContent(GameForm)  runGameInterface(String) |
| TournamentManager | 102 | ValidateFormContent(TournamentForm) |
| Home | 101 – 102 – 103 | showCategoryGames()  CreateGames()  Create Tournament()  showGame()  publishTournament(Tournament) |
| GameForm | 101 | insertName(String)  insertType(String)  insertQuestions(String[])  insertHint(String[])  publish() |
| TournamentForm | 102 | insertName(String)  insertType(String)  insertQuestions(String[])  insertHint(String[])  insertDate(String)  insertDuration(Double)  publish() |
| GameInterface | 103 | run()  help()  playGame()  showHint()  submitAnswer()  getScore()  showScoreboard()  exitGame() |
| Category | 101 – 103 | getGames()  addGame(Game)  getCategoryName() |
| Game | 101 - 103 | getHelp()  play()  getHint(QuestionNumber)  getPlayerScore()  getScoreboard() |
| Tournament | 102 | - |
| Account | 101 – 102 - 103 | addScore(int)  getUsername() |
| Teacher | 101 - 102 | addGame(Game)  addTournament(Tournament) |
| Student | 103 | addScore(int)  getUsername() |
| AccountDB | 101 – 102 – 103 | insertGame(String, Game)  insertTournament(String, Tournament)  insertScore(String, int) |
| GameDB | 101 | insertGame(Game) |
| TournamentDB | 102 | insertTournament(Tournament) |
| CategoryDB | 101 | insertGame(String, Game) |

# 

# Ownership Report

|  |  |
| --- | --- |
| **Item** | **Owners** |
| All Document | *Amr saeed* |
| All Document | *Amr Magdy* |
| All Document | *Ashrakat Mokhtar* |