Cairo University Faculty of Computers and Information



**CS352 – Software Engineering II**

**Phase 3 Template**

**2017**

**Project Team**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Email** | **Mobile** |
| 20140015 | Ahmed Hussein Karam | ahkcsit@gmail.com | 01023355193 |
| 20140035 | Ahmed Mohamed Ahmed | aabdelmeged70@yahoo.com | 01200875427 |
| 20140263 | Mariam Ashraf Fekry | mariamashraf096@gmail.com |  |
| 20140081 | Andrew Emad Nassif | andrewen2010@yahoo.com | 01271122945 |

**Eng Ragia Mohamed r.mohamed@fci-cu.edu.eg**

Contents

[Restful API 3](#_Toc414459278)

[Class diagram design 11](#_Toc414459279)

[Trello workspace screenshot 13](#_Toc414459281)

[Github repository link 13](#_Toc414459282)

# Restful API

|  |  |
| --- | --- |
| **Service name** | **createAccount** |
| **Input type** | **POST** |
| **Service URL** | **localhost:8090/st-comm.com/signup** |
| **Parameters** | |  |  | | --- | --- | | **NAME** | **Description** | | **name** | **The user name for the new account** | | **birthdate** | **Provide user birthdate** | | **gender** | **Either male or female** | | **mail** | **Email to which verification link will be sent** | | **country** | **Provide user country or country code** | | **password** | **Provide user password** | | **type** | **Either “student” or “teacher”** | |
| **Example request** | **POST**  **name: STCommUser**  **birthdate: 04/24/2014**  **gender: male**  **mail: stcomm.education@gmail.com** |
| **Example response** | **true** |

|  |  |
| --- | --- |
| **Service name** | **authenticate** |
| **Input type** | **POST** |
| **Service URL** | **localhost:8090/st-comm.com/login** |
| **Parameters** | |  |  | | --- | --- | | **NAME** | **Description** | | **name** | **Provided user name** | | **Password** | **Provided user password** | |
| **Example request** | **name: STCommUser**  **password: admin** |
| **Example response** | **true** |

|  |  |
| --- | --- |
| **Service name** | **getRegistrantType** |
| **Input type** | **GET** |
| **Service URL** | **localhost:8090/st-comm.com/query/registrant-type** |
| **Parameters** | |  |  | | --- | --- | | **NAME** | **Description** | | **name** | **Provided the name of the registrant whose type is queried** | |
| **Example request** | **localhost:8090/st-comm.com/query/registrant-type?name=STCommUser** |
| **Example response** | **2**  **1 means “student” and 2 means “teacher”** |

|  |  |
| --- | --- |
| **Service name** | **playGame** |
| **Input type** | **GET** |
| **Service URL** | **localhost:8090/st-comm.com/games/play** |
| **Parameters** | |  |  | | --- | --- | | **NAME** | **Description** | | **gameName** | **Provided the name of the game you want to play** | |
| **Example request** | **localhost:8090/st-comm.com/games/play?gameName=goodGame** |
| **Example response** | **{"name":"goodGame","courseName":"courseName","numOfQuestions":2,"teacherName":"teacherName","questions":[{"choices":["choice1","choice2"],"correctAnswer":1,"questionStatement":"questionStatement1","time":{"hour":0,"minute":0,"second":0,"nano":0}},{"choices":["choice3","choice4"],"correctAnswer":2,"questionStatement":"questionStatement2","time":{"hour":0,"minute":0,"second":0,"nano":0}}]}** |

|  |  |
| --- | --- |
| **Service name** | **createGame** |
| **Input type** | **GET** |
| **Service URL** | **localhost:8090/st-comm.com/games/new** |
| **Parameters** | |  |  | | --- | --- | | **NAME** | **Description** | | **gameName** | **Provided the name of new game** | | **courseName** | **The name of the course in which the new game will be added** | | **teacherName** | **The name of the teacher who is creating the game** | | **wrapper** | **This is a JSON object that wraps up the collection of questions that the game contains** | |
| **Example request** | [**http://localhost:8090/st-comm.com/games/new?courseName=angularCourse&gameName=angularGame&teacherName=angularTeacher&wrapper=%7B%22choices%22:%5B%220%22,%221%22%5D,%22correctAnswer%22:0,%22questionStatement%22:%22AngularQuestionStatement0%22,%22time%22:%2200:00:01%22%7D&wrapper=%7B%22choices%22:%5B%222%22,%223%22%5D,%22correctAnswer%22:1,%22questionStatement%22:%22AngularQuestionStatement1%22,%22time%22:%2200:00:01%22%7D**](http://localhost:8090/st-comm.com/games/new?courseName=angularCourse&gameName=angularGame&teacherName=angularTeacher&wrapper=%7B%22choices%22:%5B%220%22,%221%22%5D,%22correctAnswer%22:0,%22questionStatement%22:%22AngularQuestionStatement0%22,%22time%22:%2200:00:01%22%7D&wrapper=%7B%22choices%22:%5B%222%22,%223%22%5D,%22correctAnswer%22:1,%22questionStatement%22:%22AngularQuestionStatement1%22,%22time%22:%2200:00:01%22%7D)  **Note: the parameter “wrapper” should be sent as JSON, and the wrapper value you see in the URL corresponds to the following JSON array:**  **[{choices:["0","1"],correctAnswer:0,questionStatement:"AngularQuestionStatement0",time:"00:00:01"},{choices:["2","3"]** |
| **Example response** | **true** |

|  |  |
| --- | --- |
| **Service name** | **saveScore** |
| **Input type** | **GET** |
| **Service URL** | **localhost:8090/st-comm.com/games/scores/save** |
| **Parameters** | |  |  | | --- | --- | | **NAME** | **Description** | | **name** | **Provide the username of the player** | | **score** | **Provide achieved score** | | **gameName** | **The name of the game of which the score was achieved by the player** | |
| **Example request** | **http://localhost:8090/st-comm.com/games/scores/save?name=myName&score=500&gameName=goodGame** |
| **Example response** | **true** |

|  |  |
| --- | --- |
| **Service name** | **exists** |
| **Input type** | **GET** |
| **Service URL** | **localhost:8090/st-comm.com/games/exists** |
| **Parameters** | |  |  | | --- | --- | | **NAME** | **Description** | | **gameName** | **Provide the name of the game that you want to know whether it exists in the system** | |
| **Example request** | **http://localhost:8090/st-comm.com/games/exists?gameName=goodGame** |
| **Example response** | **false** |

|  |  |
| --- | --- |
| **Service name** | **getGames** |
| **Input type** | **GET** |
| **Service URL** | **localhost:8090/st-comm.com/games/courses/list-by-course** |
| **Parameters** | |  |  | | --- | --- | | **NAME** | **Description** | | **courseName** | **Provide the name of the course that you want to get the names of its games** | |
| **Example request** | **http://localhost:8090/st-comm.com/games/courses/list-by-course?courseName=myCourseName** |
| **Example response** | **["goodGame","happyGame"]** |

|  |  |
| --- | --- |
| **Service name** | **createCourse** |
| **Input type** | **GET** |
| **Service URL** | **localhost:8090/st-comm.com/courses/new** |
| **Parameters** | |  |  | | --- | --- | | **NAME** | **Description** | | **courseName** | **Provide the name of the new course** | | **teacherName** | **The name of the teacher who is creating the course** | |
| **Example request** | **http://localhost:8090/st-comm.com/courses/new?courseName=myCourseName&teacherName=myTeacherName** |
| **Example response** | **true** |

|  |  |
| --- | --- |
| **Service name** | **getOtherCourses** |
| **Input type** | **GET** |
| **Service URL** | **localhost:8090/st-comm.com/courses/list-all** |
| **Parameters** | |  |  | | --- | --- | | **NAME** | **Description** | | **registrantName** | **Provide the name of the registrant for whom you want to get the names of the courses he/ she did not yet enroll in** | |
| **Example request** | **localhost:8090/st-comm.com/courses/list-all?registrantName=myRegistrantName** |
| **Example response** | **["course1","course2"]** |

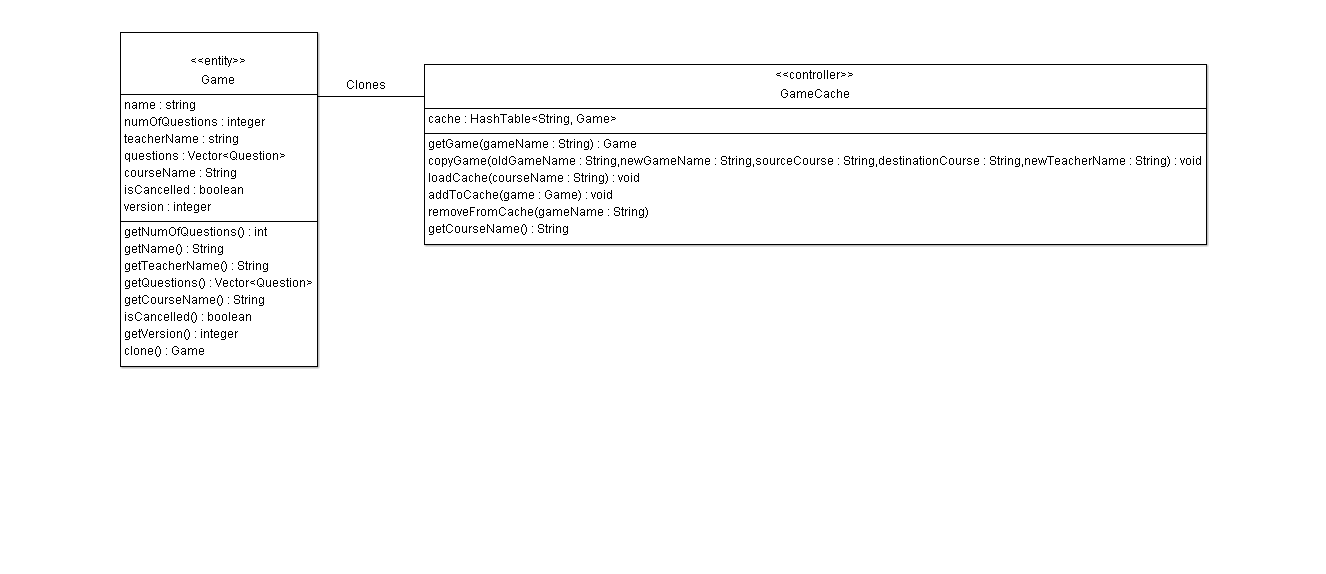
|  |  |
| --- | --- |
| **Service name** | **getCourses** |
| **Input type** | **GET** |
| **Service URL** | **localhost:8090/st-comm.com/courses/list-by-registrant** |
| **Parameters** | |  |  | | --- | --- | | **NAME** | **Description** | | **registrantName** | **Provide the name of the registrant for whom you want to get the names of the courses he/ she is enroll in** | |
| **Example request** | **localhost:8090/st-comm.com/courses/list-by-registrant?registrantName=myRegistrantName** |
| **Example response** | **["course1","course2"]** |

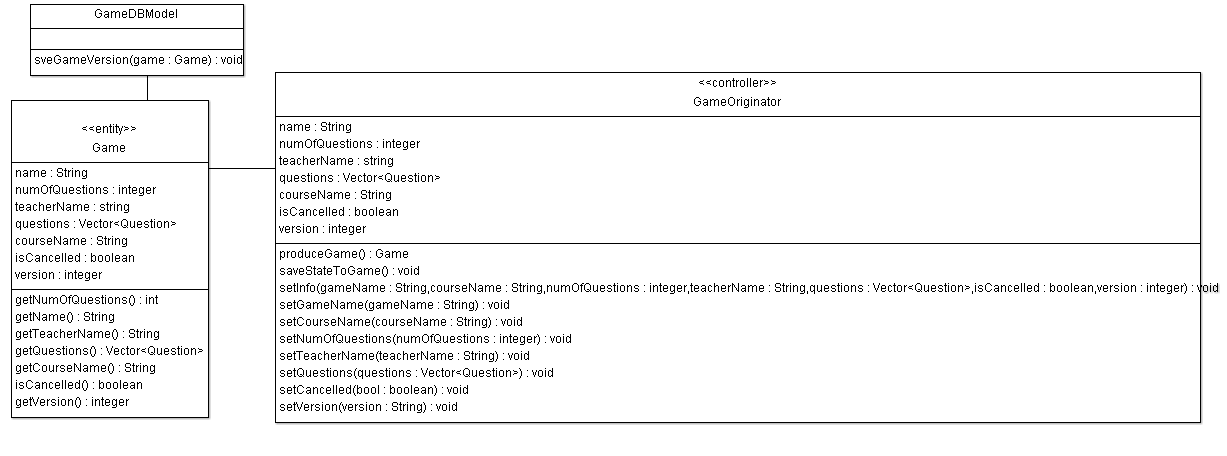
|  |  |
| --- | --- |
| **Service name** | **register** |
| **Input type** | **GET** |
| **Service URL** | **localhost:8090/st-comm.com/courses/register** |
| **Parameters** | |  |  | | --- | --- | | **NAME** | **Description** | | **registrantName** | **Provide the name of the registrant who wants to register in a course** | | **courseName** | **Provide the name of the course in which the registrant wants to register** | |
| **Example request** | **localhost:8090/st-comm.com/courses/register?registrantName=myRegistrantName&courseName=myCourseName** |
| **Example response** | **true** |

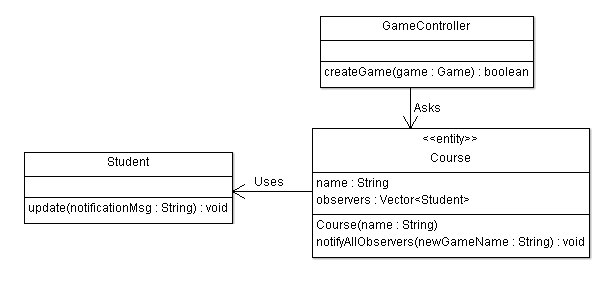
|  |  |
| --- | --- |
| **Service name** | **exists** |
| **Input type** | **GET** |
| **Service URL** | **localhost:8090/st-comm.com/courses/exists** |
| **Parameters** | |  |  | | --- | --- | | **NAME** | **Description** | | **courseName** | **Provide the name of the course that you want to know whether it exists in the system** | |
| **Example request** | **localhost:8090/st-comm.com/courses/exists?courseName=myCourseName** |
| **Example response** | **false** |

# Class diagram design

# C:\Users\Ahmed Hussein\AppData\Local\Microsoft\Windows\INetCache\Content.Word\6.png



****

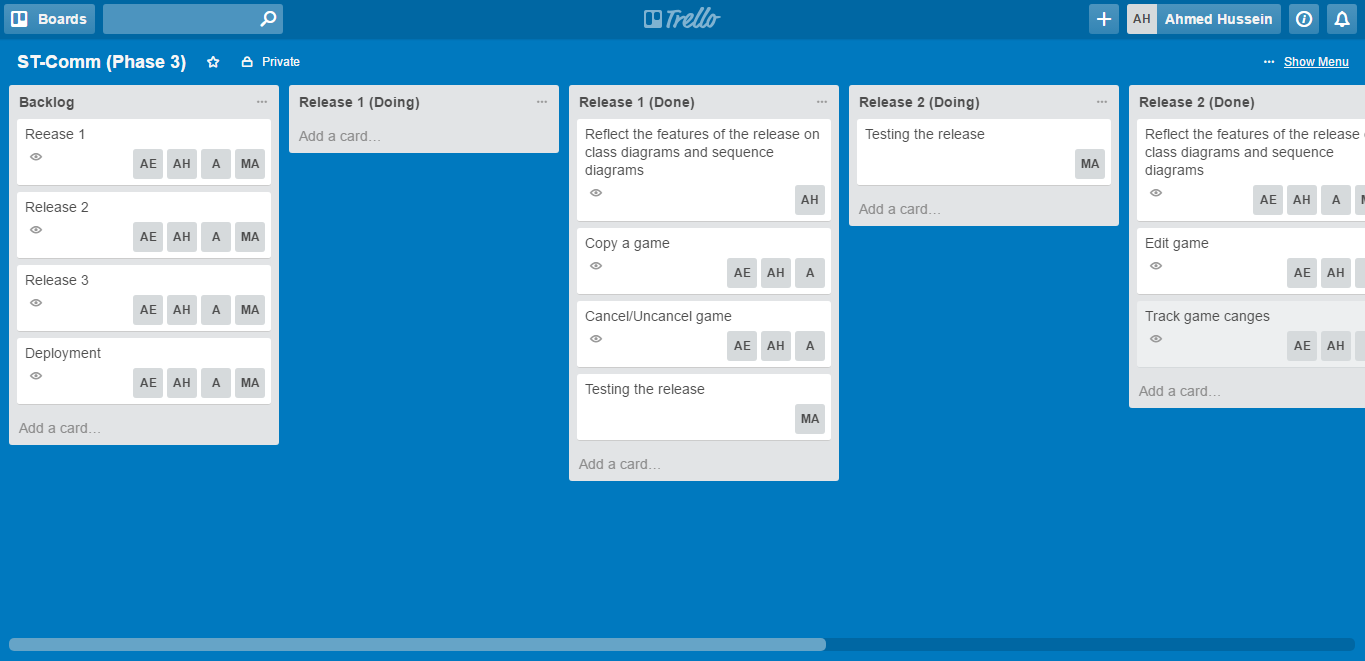
****

# 

# Used Design patterns

1. **Prototype pattern:** to solve the problem of copying a game, since Game object is quite complex.
2. **Memento pattern:** help in tracking game changes without losing versions. That is because it makes Game object immutable and manages saving new versions and retrieving old ones.
3. **Observer pattern:** to help manage notifications, since a course can be observed by many students

# Trello workspace screenshot



# Github repository link

# <https://github.com/AndrewEmad/ST-Comm.git>