**Cairo University  
Faculty of Computers and Information**



**CS251**

**Software Engineering I**

Game to Gain

Software Design

Nada Ashraf

Michael Wageuh

Mina Nabil

Monica Millad

December 2016

**Contents**

[Team 3](#_Toc468575261)

[Document Purpose and Audience](#_Toc468575262) 4

[System Models](#_Toc468575263) 5

[I. Class diagrams](#_Toc468575264) 5

[II. Sequence diagrams](#_Toc468575266) 13

[Class - Sequence Usage Table](#_Toc468575267) 21

[Ownership Report](#_Toc468575268) 23

**Team**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Email** | **Mobile** |
| 20140289 | Nada Ashraf Ahmed | nadanodyy@ymail.com | 01100591922 |
| 20140213 | Michael Wageuh Ramzy | michaelwageuh12@hotmail.com | 01200634332 |
| 20140287 | Mina Nabil William | mnabil405@gmail.com | 01142642703 |
| 20140281 | Monica Millad Aziz | Monicamillad18@gmail.com | 01271223662 |

**Software Design Specification Document**

**1. Document purpose and audience**

**1.1. Purpose**

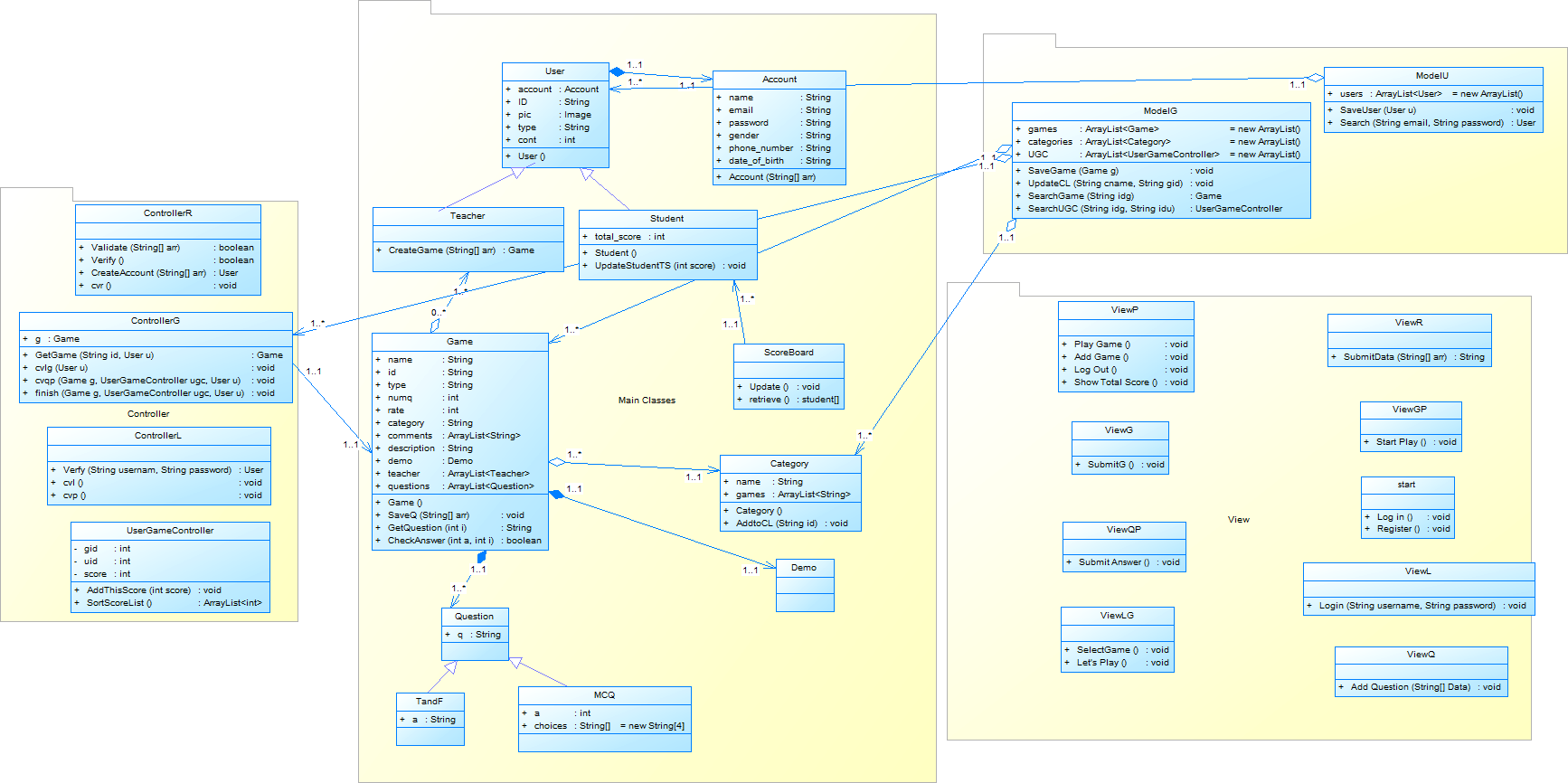
This document describe the architecture and the design of the software. It describes the dependencies of the classes and the relation between them. It also include description for most of the main functions of the software.

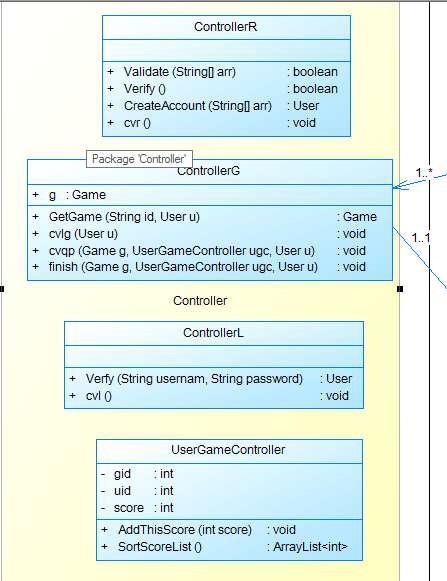
**1.2. Audience**

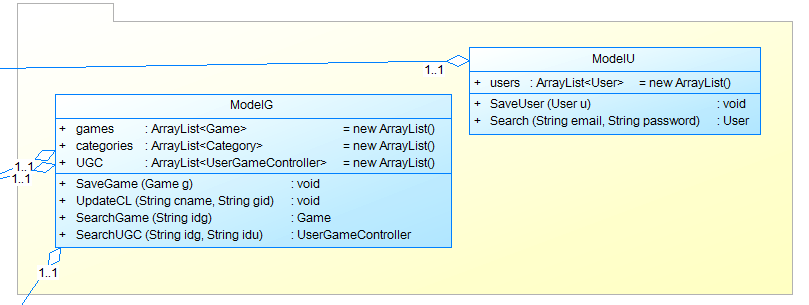
The audience of this documents are the engineers that are going to implement this software, as it will be a reference to facilitate understanding the functionality of the software. Also the engineers that may modify or extend the software after building it.

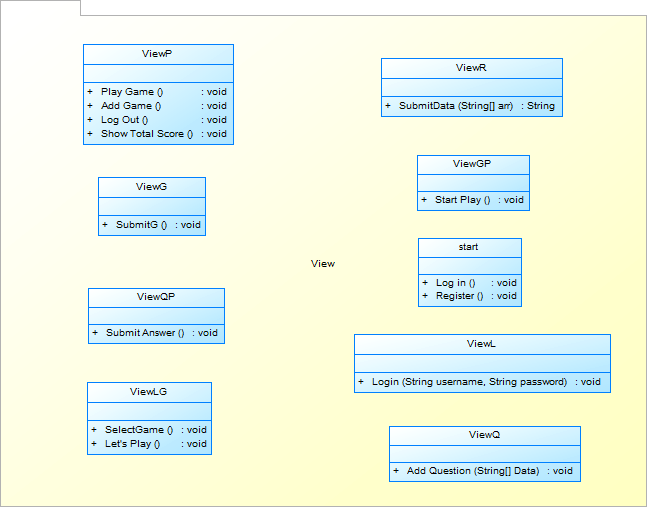
**2. System Model**

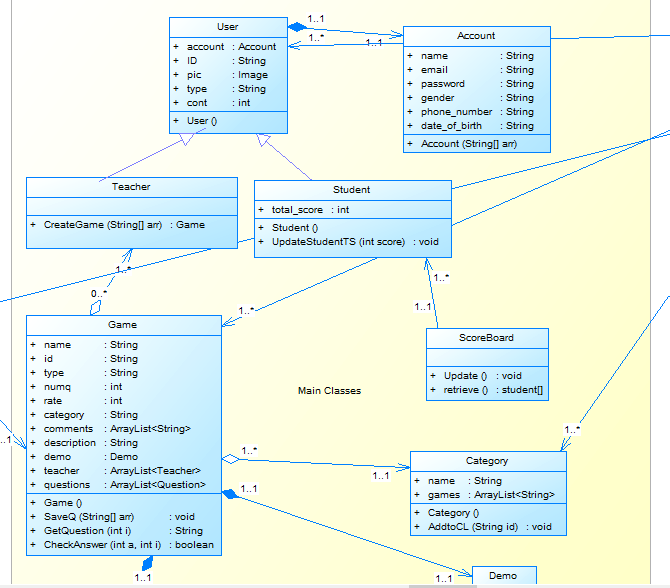
**2.1. Class diagrams**

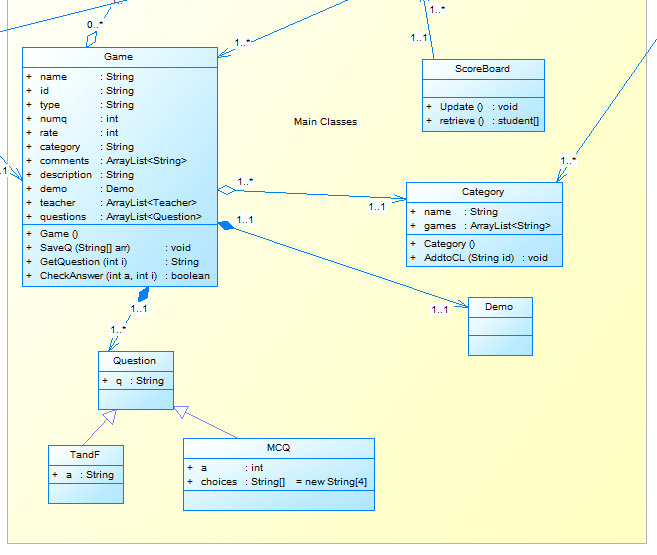
****

****

****

****

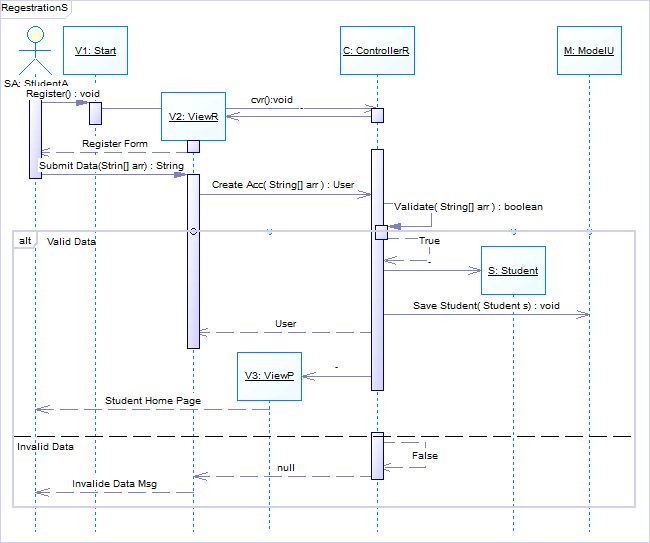
****

****

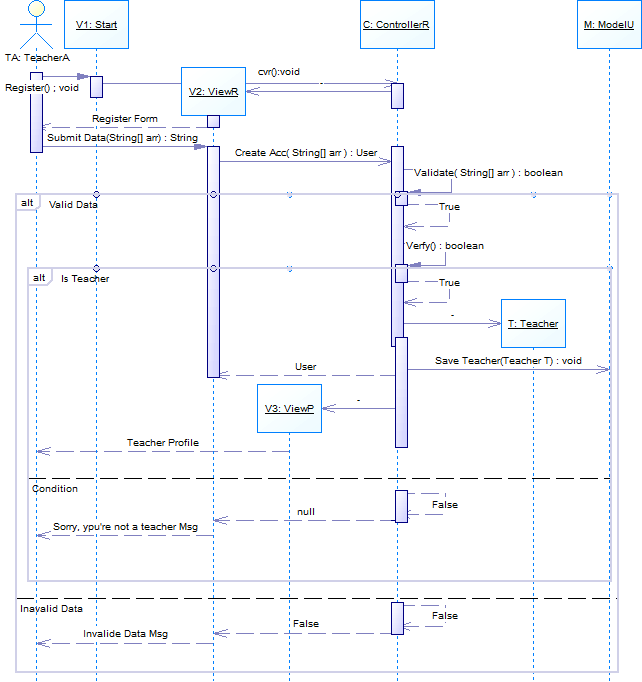
|  |  |  |
| --- | --- | --- |
| **Class ID** | **Class Name** | **Description & Responsibility** |
| 1 | Account | It includes the common information between all the user such as; name, password, email …etc. |
| 2 | User | It includes the common information between all the users that are related to the user profile such as the pic, it also include the account information. |
| 3 | Student | It includes the total score of the player.  It is responsible for updating student total score. |
| 4 | Teacher | It is responsible for creating a game. |
| 5 | Game | It contains information about the game such as the author, game description and so on.  It is responsible for:   * Saving questions * Returning the questions * Checking answers |
| 6 | Question | It is made for abstraction so we could unify any return type related to the questions regardless to its type. |
| 7 | TandF | It is one of the question types, it contains the true and false question standard format. |
| 8 | MCQ | It is one of the question types, it contains the multiple choice question standard format. |
| 9 | Category | It contains the list of game for the specific category as there is an instance from it for each category contains its list of games.  And it’s responsible for adding a game to this category. |
| 10 | ScoreBoard | It is responsible for updating the list of the top players on the website and retrieving it. |
| 11 | Demo | It contains the demo of the game as each game have a demo and it is generated by the controller after the teacher creates new game. |
| 12 | ControllerL | It is responsible for:   * Verifying the user’s username and password before linked him/her on his/her home page. * Creating the log in view. |
| **Class ID** | **Class Name** | **Description & Responsibility** |
| 13 | ControllerR | It is responsible for:   * Validating the user information while the registration. * Verifying the teachers’ accounts before creating them. * Create the users’ accounts. * Create the registration view. |
| 14 | UserGameController | It controls the transactions between the user and the game.  It is responsible for:   * Adding the last score to the list of scores related to a specific game and a specific user. * Sorting this list. |
| 15 | ControllerG | It is responsible for:   * Returning the game to the user so he/she can play it. * Creating the list of games view. * Creating the question page view. * Finishing the game and returning the user to the home page view. |
| 16 | ModelU | It is the database that contains all the users’ accounts.  It is responsible for:   * Saving user information in the array of users. * Search for specific user in the array in the array of users. |
| 17 | ModelG | It is the database that contains all the games’ information.  It is responsible for:   * Saving the game to the database. * Updating the category list which that game belongs to. * Searching for a game in the database. * Searching for the user game controller for a specific game and a specific user. |
| **Class ID** | **Class Name** | **Description & Responsibility** |
| 18 | Start | It is the application home page.  It contains the following buttons:   * Log in. * Register. |
| 19 | ViewR | It is responsible for:   * Submitting the user data and send it to the registration controller. |
| 20 | ViewL | Responsible for taking the username and the password from the user and send them to the login controller to verify them. |
| 21 | ViewP | It is the user home page.  It is contains the following buttons:   * Play Game * Add Game * Log out * Show Total Score |
| 22 | ViewG | It is responsible for:   * Taking the submitted game data from the teacher and send it to the controller that responsible for adding a game ( the teacher). |
| 23 | ViewQ | It is the question standard format page that the teacher use to add a new question. |
| 24 | ViewLG | It’s the view that shows to the user the list of games so he/she selects one of them to start playing. |
| 25 | ViewGP | It is the view that shows the game home page. |
| 26 | ViewQP | It is the view that shows the user the question and then the user can submit the answer through it. |

**2.2. Sequence diagrams**

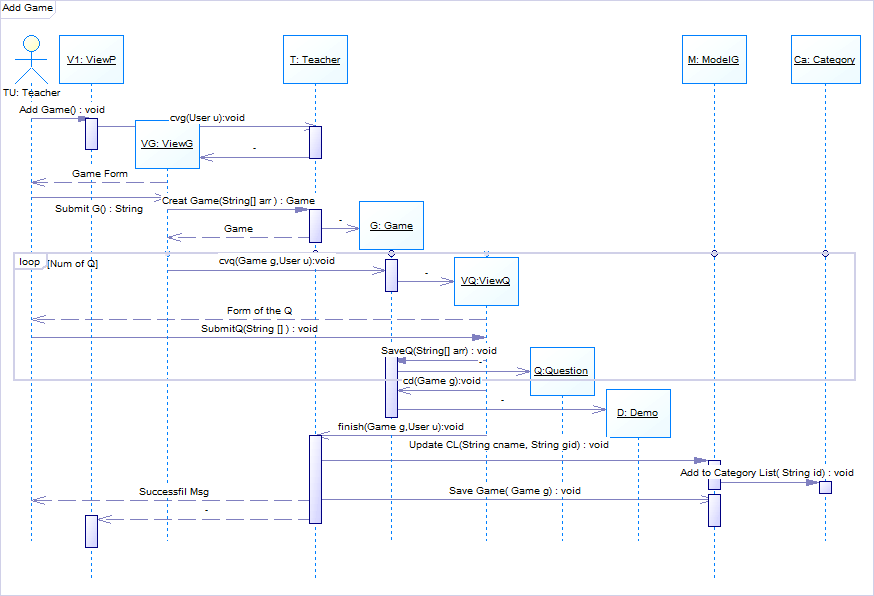
2.2.1. RegestrationS



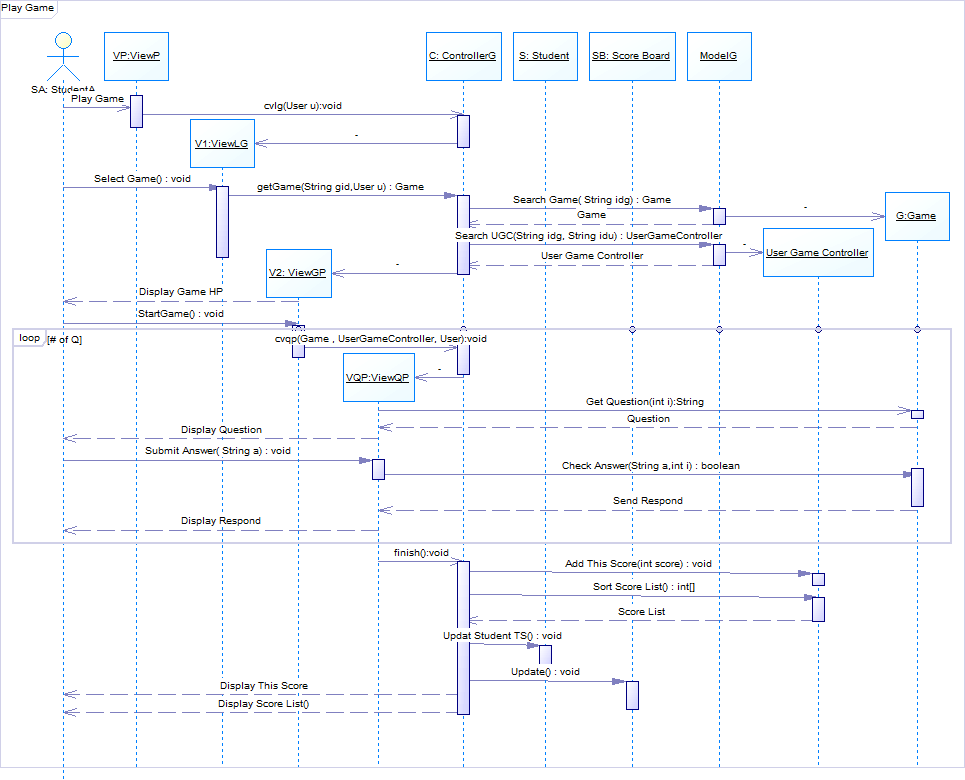
2.2.2 RegestrationT



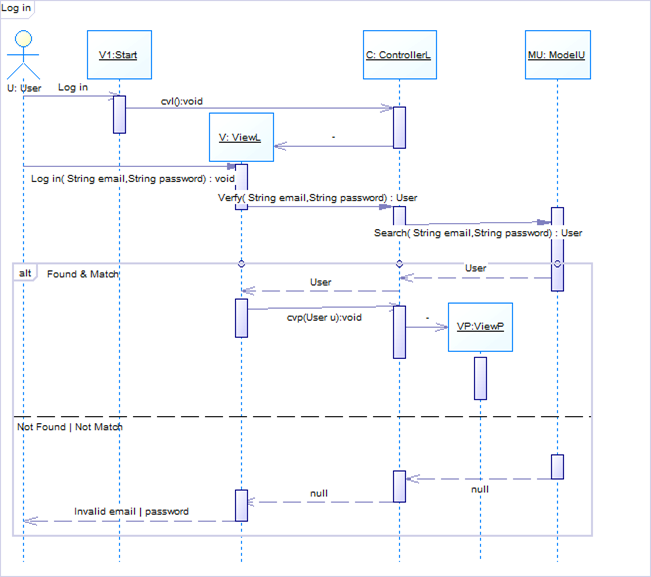
2.2.3 Add Game

****

2.2.4 Play Game



2.2.4 Log in

****

* **Class - Sequence Usage Table**

|  |  |  |
| --- | --- | --- |
| **Class Name** | **Sequence Diagrams** | **Overall used methods** |
| 1. Category | 3 | Add to Category List( String id) : void |
| 2. ControllerG | 4 | cvlg(User u):void  getGame(String gid,User u) : Game  cvqp(Game , UserGameController, User):void  finish():void |
| 3. ControllerL | 5 | Verfy( String username, String password) : boolean  vcl():void  cvp(User u):void |
| 4. ControllerR | 1, 2 | cvr():void  Create Acc( String[] arr ) : User  Validate( String[] arr ) : Boolean  Verfy() : boolean |
| 5. Demo | 3 |  |
| 6. Game | 3, 4 | cvq(Game g,User u):void  SaveQ(String[] arr) : void  cd(Game g):void  Get Question(int i):String  Check Answer(String a,int i) : boolean |
| 7. ModelG | 3, 4 | Update CL(String cid, String gid) : void  Save Game( Game g) : void  Search Game( String idg) : Game  Search UGC(String idg, String idu) : UserGameController |
| 8. ModelU | 1, 2, 5 | Save Student( Student s) : void  Save Teacher(Teacher T) : void  Search( String username, String password) : User |
| 9. Question | 3 |  |
| 10. ScoarBord | 4 | Update() : void |
| 11. Start | 1, 2, 5 | Register  Log in |
| 12. Student | 1, 4 | Updat Student TS() : void |
| 13. Teacher | 2, 3 | cvg(User u):void  Creat Game(String[] arr ) : Game  finish(Game g,User u):void |
| 14. UserGameController | 4 | Add This Score(int score) : void  Sort Score List() : int[] |
| 15. ViewG | 3 | Submit G() : String  SubmitQ() : void |
| 16. ViewGP | 4 | StartGame() : void |
| 17. ViewL | 5 | Log in( String username, String password) : void |
| 18.ViewLG | 4 | Select Game() : void |
| 19. ViewP | 1, 2, 3 | Add Game() : void |
| 20. ViewQ | 3 | SaveQ(String[] arr) : void |
| 21. ViewQP | 4 | Submit Answer( String a) : void |
| 22. ViewR | 1, 2 | Submit Data(Strin[] arr) : String |

**3. Ownership Report**

|  |  |
| --- | --- |
| **Item** | **Owners** |
| Document Purpose and Audience | Monica Millad |
| Class diagrams | All the team members |
| Sequence diagrams | All the team members |

**Git repository link:**

**https://github.com/NadaAshrafAhmed/SWE\_1-project**