Cairo University Faculty of Computers and Information



**CS352 – Software Engineering II**

**Phase 3-b: Testing Report**

**2015**

**Project Team - Eng. Mohamed Samir**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Email** | **Mobile** |
| 20120364 | Mahmoud Mohamed Hadad | m.hadad@stud.fci-cu.edu.eg | 01117943793 |
| 20120062 | Ahmed Mohammed Fekry | ahmedfikry78@gmail.com | 0112059041 |
| 20120084 | Eslam Ebrahim Mohamed | altaiir\_brotherhood@yahoo.com | 01141731202 |
| 20120207 | Sherif Mohamed Farouk | sfarouk513@gmail.com | 01144967048 |

Contents

[1. Backend Testing 3](#_Toc418368340)

[2. Frontend testing 5](#_Toc418368341)

[Github 7](#_Toc418368342)

[Ownership report 7](#_Toc418368343)

# 1. Backend Testing

* 1. **Service class**

|  |  |  |  |
| --- | --- | --- | --- |
| **Number** | **Testing function** | **Description** | **Result** |
| 1.1.1 | addFriend() | Testing function for addFriend function in **Service**  Assumption: friend isn't in friends | Passed |
| 1.1.2 | getFriendRequests() | Testing function for getFriendRequests function in **Service** class  Assumption: friend request sent | Passed |
| 1.1.3 | getFriends() | Testing function for getFriends function in **Service** class  Assumption: show all friends | Passed |
| 1.1.4 | loginService(hadad, 12)  loginService(hadad, 123) | Testing function for loginService function in **Service** class  Assumption: logging is done correctly | Failed  Passed |
| 1.1.5 | registrationService(hadad, hadad.com, 123)  registrationService(hadad, haadad.com, 123) | Testing function for registrationService function in **Service** class  Assumption: registration is done correctly | Failed  Passed |

* 1. **NotificationService class**

|  |  |  |  |
| --- | --- | --- | --- |
| **Number** | **Testing function** | **Description** | **Result** |
| 1.2.1 | getConversationMessages() | Testing function for getConversationMessages function in **NotificationService**  Assumption: show all conversation messages | Passed |
| 1.2.2 | getFriendRequestByID() | Testing function for getFriendRequestByID function in **NotificationService**  Assumption: show all friend requests | Passed |
| 1.2.3 | getMessagesByID() | Testing function for getMessagesByID function in **NotificationService**  Assumption: show all messages for certain id | Passed |
| 1.2.4 | getUnseenConversationMessages() | Testing function for getUnseenConversationMessages function in **NotificationService**  Assumption: show all unseen conversation messages | Passed |
| 1.2.5 | getUnseenFriendRequests() | Testing function for getUnseenFriendRequests function in **NotificationService**  Assumption: show all unseen friend requests | Passed |
| 1.2.6 | getUnseenMessages() | Testing function for getUnseenMessages function in **NotificationService**  Assumption: show all unseen messages | Passed |
| 1.2.7 | sendFriendRequest(hadad, m, 123)  sendFriendRequest(hadad, hadad, 123)  sendFriendRequest(hadad, hh, 123) | Testing function for sendFriendRequest function in **NotificationService**  Assumption: friend request is sent | Passed  Failed  Same user  Failed  Wrong user name or password |

* 1. **Service class**

|  |  |  |  |
| --- | --- | --- | --- |
| **Number** | **Testing function** | **Description** | **Result** |
| 1.3.1 | addFriend() | Testing function for addFriend function in **Service**  Assumption: friend isn't in friends | Passed |
| 1.3.1 | getFriendRequests() | Testing function for getFriendRequests function in **Service** class  Assumption: friend request sent | Passed |
| 1.3.1 | getFriends() | Testing function for getFriends function in **Service** class  Assumption: show all friends | Passed |
| 1.3.1 | loginService(hadad, 12)  loginService(hadad, 123) | Testing function for loginService function in **Service** class  Assumption: logging is done correctly | Failed  Passed |
| 1.3.1 | registrationService(hadad, hadad.com, 123)  registrationService(hadad, haadad.com, 123) | Testing function for registrationService function in **Service** class  Assumption: registration is done correctly | Failed  Passed |

# 

# 2. Frontend testing

* **In the following table, you should describe each testing functions you developed in each testing class and state the result of testing after executing testing class**
* **Frontend classes numbers will use prefix 2.**
* **Number test cases in proper way, for each controller component class use number 2.1,2.2,2.3 and for each function in some controller component class (i.e. class #1) use 2.1.1, 2.1.2, 2.1.3,…**
* **Remember you should develop testing class for each component in frontend**

**Example**

* 1. **UserControllerTest**

|  |  |  |  |
| --- | --- | --- | --- |
| **Number** | **Testing function** | **Description** | **Result** |
| **2.1.1** | **Login(name, pass)** | **Testing function for login function in user controller**  **Assumption: name is unique for each user** | **Passed** |
| **2.1.2** | **signUp(name,email,pass)** | **Testing function for signup function in user controller class** | **Failed** |

**Remember: you should handle all possible cases for specific function, for example you should test search user function in user entity if user is exist and if user is not exist**

**In frontend testing you should call services and check returned json from this calling**

**Frontend Testing should depend on backend testing**

# Github

**https://github.com/ahmedfekry/FCI-CS352-MASE**

# Ownership report

|  |  |  |
| --- | --- | --- |
| **ID** | **Name** | **done** |
| 20120364 | Mahmoud | \*Refactoring the whole project to be ready for testing  Classes : 1.1 – 1.2 |
| 20120062 | Ahmed |  |
| 20120207 | Sherif | Documentation and preparation of the template  Functions : 1.1.3 - 1.2.5 - 1.2.6 |
| 20120084 | Eslam | ----------------------------------------------------------------------------- |