**Cairo University  
Faculty of Computers and Information** 

**CS251**

**Software Engineering I**

GET BACK

Software Requirements Specifications

Tawfik

october & 2018

**Contents**

**Team**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Email** | **Mobile** |
| 20120473 | Muhammed Ahmed Tawfik | muhammedtaowfik@gmail.com | 01159701722 |
| 20120527 | Mansour mohamed mansour | Kingmano2008@yahoo.com | 01210404042 |

**Document Purpose and Audience**

* **This Document is the guideline for Get Back Application with the expected requirements and specifications.**
* **This document is for customers that need to understand what Get Back App do and for developers who wanted to edit or even know about our Get Back App.**

**Introduction**

**Software Purpose**

**The purpose of our App is to help people find their lost things regardless the type of that things and make sure that these things is got back to the right place.**

**Software Scope**

* **Our app contain people that searching for their lost items including evidences that prove that that item belongs to us ,contain people that found somthing and trying to search for the owners and our software trying to communicate between those people throw posts and pictures of lost items.**

**Definitions, acronyms, and abbreviations**

* **Finder : the one who finds the lost items.**
* **Item : the lost thing.**
* **Owner : the owner of the item.**
* **Fake Owner : the fake owner of the item.**
* **Post : the place that contain information about the item including picture of the item and tags related to that item.**
* **Tag : the word that binds the finder with the owner using word that describe the item itself.**
* **System : the system that integrate functions to reach to the goal of the app.**
* **Security check : the check between the owner and fakr owner to deliver the item to the right place.**
* **Profile : each of Owner,Fake Owner and Finder have a profile with information about themselves including name, address,phone number and verification notationa.**
* **Spam : a notation to define the Fake Owner .**
* **Verified : a notation to define the Owner .**

**Requirements**

**Functional Requirements**

* **Finder :**
* **Sign up**
* **Sign in**
* **Log out**
* **Upload photo**
* **Delete photo**
* **Check within the owner is correct or fake owner**

**Owner:**

**Search for photos**

**Sign up**

**Sign in**

**Log out**

**Send message to finder**

**Upload a photo for his missing things**

**Non Functional Requirements**

|  |  |
| --- | --- |
| **type** | **Details** |
| **System scalability** | Capability adding more finder and owners |
| **System security** | Checking correct user name and password on Login and force the user to enter a strong password on signup to secure his account. |
| **Reliability** | The server is not down and reliable to the users. |
| **Usability** | Provide the user with help screens to let him understand what each form does and name buttons with meaningful names and provide easy navigation between screens by making small number of forms. |
| **Robustness** | The system should cope with any errors that can happen at runtime like trying to sign up with a fake email or weak password. |

**System Models**

**Use Case Model**



**Use Case Tables**

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 1 | |
| Use Case Name: | Make post | |
| Actors: | Finder | |
| Pre-conditions: |  | |
| Post-conditions: |  | |
| Flow of events: | **User Action** | **System Action** |
| 1- user take a photo of the item. |  |
|  | 2- system approve that photo |
| 3- user set tags. |  |
| 4- user click post. |  |
|  | 5- system save the post on database and put it with the posts of the same tag. |
| Exceptions: | **User Action** | **System Action** |
| 1- User put photo only |  |
| 3-user put tags only | 2- you should put tags .  4- you should take a photo for the item. |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 2 | |
| Use Case Name: | Search tags | |
| Actors: | Owner , Fake Owner | |
| Pre-conditions: |  | |
| Post-conditions: | Approve post | |
| Flow of events: | **User Action** | **System Action** |
| 1- User enter tags in search bar. |  |
|  | 2- System collect the posts that contain same tags and print that posts |
| 3- User Select the post that is searching for |  |
|  | 4- System print the post that is selected |
| and so on |  |
| Exceptions: | **User Action** | **System Action** |
| 1- user enters too many tags |  |
|  | 2- it shouldn't more than five tags. |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 3 | |
| Use Case Name: | Approve post | |
| Actors: | System | |
| Pre-conditions: | Search tag | |
| Post-conditions: | Security check | |
| Flow of events: | **User Action** | **System Action** |
| 1- user click approve post |  |
|  | 2- System go to the profile of the finder.  3-System send notification to finder that the post is approved |
|  |  |
|  |  |
|  |  |
| Exceptions: | **User Action** | **System Action** |
|  |  |
|  |  |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 4 | |
| Use Case Name: | Security Check | |
| Actors: | System | |
| Pre-conditions: | Approve request | |
| Post-conditions: | Open profile | |
| Flow of events: | **User Action** | **System Action** |
|  |  |
|  | 2- System check the verification of the user.  3- if verified , the system go to the profile of the finder.  4- if spam , the system print "you can't enter the profile " and go back to the search tags. |
|  |  |
|  |  |
|  |  |
| Exceptions: | **User Action** | **System Action** |
|  |  |
|  |  |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 5 | |
| Use Case Name: | Delete post | |
| Actors: | Finder | |
| Pre-conditions: | Verification test | |
| Post-conditions: |  | |
| Flow of events: | **User Action** | **System Action** |
| 1- user tells the system that the owner of the item is found and the connection is made between the finder and the owner. |  |
|  | 2- System approved that and delete post from database. |
|  |  |
|  |  |
|  |  |
| Exceptions: | **User Action** | **System Action** |
|  |  |
|  |  |
| Includes: |  | |
| Notes and Issues: |  | |

**Ownership Report**

* **Remove the following notes and any red notes**

|  |  |
| --- | --- |
| **Item** | **Owners** |
| functional and nonfunctional | *mansour* |
| the rest of file | *tawfik* |

Policy Regarding Plagiarism**:**

**Students have collective ownership and responsibility of their project. Any violation of academic honesty will have severe consequences and punishment for ALL team members.**

* تشجع الكلية على مناقشة الأفكار و تبادل المعلومات و مناقشات الطلاب حيث يعتبر هذا جوهريا لعملية تعليمية سليمة
* ساعد زملاءك على قدر ما تستطيع و حل لهم مشاكلهم فى الكود و لكن تبادل الحلول غير مقبول و يعتبر غشا.
* أى حل يتشابه مع أى حل آخر بدرجة تقطع بأنهما منقولان من نفس المصدر سيعتبر أن صاحبيهما قد قاما بالغش.
* قد توجد على النت برامج مشابهة لما نكتبه هنا أى نسخ من على النت يعتبر غشا يحاسب عليه صاحبه.
* إذا لم تكن متأكدا أن فعلا ما يعد غشا فلتسأل المعيد أو أستاذ المادة.
* فى حالة ثبوت الغش سيأخذ الطالب سالب درجة المسألة ، و فى حالة تكرار الغش سيرسب الطالب فى المقرر.