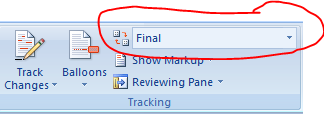
***Notes to the Team***

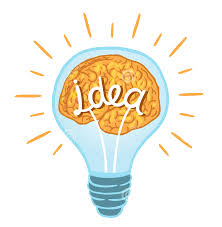
This is a template of a Requirements Document for your project with explanatory notes included as comments.

**When using this template, the following steps are recommended:**

1. Replace all text enclosed in angle brackets (e.g., <Project Name>) with the correct values. These angle brackets appear either in the body of the document or in headers and footers.
2. To update the Table of Contents, right-click on it and select “Update field” and choose the option - “Update entire table”.
3. Turn off Tracking. Go to the Review tab; make sure Display for Review is set to “Final”.



1. Before submission of this document, delete this instruction section “Notes to the Team”.



Stats and Stacks

Requirements Document

*BucketList*

Team Members:

|  |
| --- |
| 217143512, Bernard Mc Donald  217198392, Mu’Minah Daniels  213348292, Tiffany de Vos |

Date Submitted: 2 May 2019

**TABLE OF CONTENTS**

[1 SYSTEM OVERVIEW 4](#_Toc3230019)

[2 FUNCTIONAL REQUIREMENTS 5](#_Toc3230020)

[2.1 Business Use Case Model 5](#_Toc3230021)

[2.2 Use Case Glossary 6](#_Toc3230022)

[2.3 Use Case Narratives (User Stories) 8](#_Toc3230023)

[2.3.1 Package A: [insert package name] 8](#_Toc3230024)

[2.3.2 Package B: [insert package name] 8](#_Toc3230025)

[2.3.3 Package C: [insert package name] 8](#_Toc3230026)

[2.3.4 Package D: [insert package name] 9](#_Toc3230027)

[3 NON-FUNCTIONAL REQUIREMENTS 10](#_Toc3230028)

[3.1 Interface Requirements 10](#_Toc3230029)

[3.2 Performance Requirements 10](#_Toc3230030)

[3.3 Security Requirements 10](#_Toc3230031)

[3.4 Operational Requirements 10](#_Toc3230032)

[4 DATA REQUIREMENTS 11](#_Toc3230033)

[5 Project Matters 12](#_Toc3230034)

[5.1 Open Issues 12](#_Toc3230035)

[5.2 Preliminary Schedule 12](#_Toc3230036)

[6 Researching the Requirements 13](#_Toc3230037)

[6.1 10 User Interview Questions 13](#_Toc3230038)

# SYSTEM OVERVIEW

The main objective of this system is to provide adventurous people with an app that encourages them to accomplish more out of life’s diverse activities than just sitting at home tied to their phones.

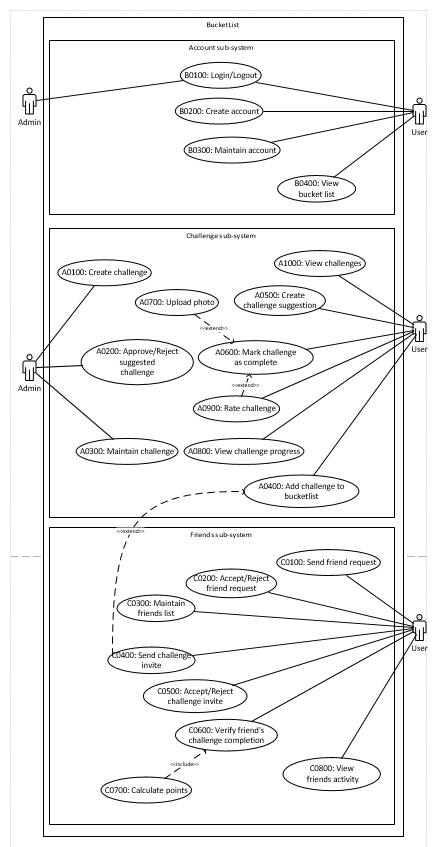
An adventurer using the app is introduced to a wide range of activities in the form of a bucket list. The user can complete the challenges that they add to their personal bucket list and simply upload a photo of the activity to receive points and gain achievements. The user’s goal is to collect all the predefined achievements. The challenges are of a wide variety and also vary in difficulty. Examples of simpler challenges are: “Talk to someone new”, “Photobomb people you don’t know”, “Go for a jog” or “Play a prank on someone”. More complicated challenges could be: “Climb a mountain” or “Go on a road trip”. Users can rate and review each challenge to ensure adventurers find the challenge best suited for them.

The system allows users to connect with friends on the app where they can view each other’s progress and achievements. It is easy to search for a friend by username and send them a friend invite which they can accept or decline. The system places especial emphasis on in-person social interaction as the challenges can be completed in groups with their friends and each photo a user uploads of a completed challenge need to be validated by a friend of the user.

Users and admins will log in with their personal passwords to keep their account secure. Users can manage their profile and list of friends on the app. Although users can suggest challenges, all the challenges and their details are finalised by an admin. The admin ensures a challenge is ethical, safe and assigned a fair difficulty level.

# FUNCTIONAL REQUIREMENTS

## Business Use Case Model



## Use Case Glossary

|  |  |  |
| --- | --- | --- |
| **Package Id: Package Name: Challenges sub-system** | | |
| **Use Case Id** | **Use Case Name** | **Actors** |
| A0100 | Create challenge | Admin |
| A0200 | Approve/Reject suggested challenge | Admin |
| A0300 | Maintain challenge | Admin |
| A0400 | Add challenge to bucket list | User |
| A0500 | Create challenge suggestion | User |
| A0600 | Mark challenge as completed | User |
| A0700 | Upload photo | User |
| A0800 | View challenge progress | User |
| A0900 | Rate challenge | User |
| **Queries/Reports** | | |
| A1000 | View challenges | Admin, User |

|  |  |  |
| --- | --- | --- |
| **Package Id: Package Name: Account sub-system** | | |
| **Use Case Id** | **Use Case Name** | **Actors** |
| B0100 | Login/Logout | Admin, User |
| B0200 | Create account | User |
| B0300 | Maintain account | User |
| **Queries/Reports** | | |
| B0400 | View bucket list | User |

|  |  |  |
| --- | --- | --- |
| **Package Id: Package Name: Friends sub-system** | | |
| **Use Case Id** | **Use Case Name** | **Actors** |
| C0100 | Send friend request | User |
| C0200 | Accept/Reject friend request | User |
| C0300 | Maintain friends list | User |
| C0400 | Send challenge invite | User |
| C0500 | Accept/Reject challenge invite | User |
| C0600 | Verify friend’s challenge completion | User |
| C0700 | Calculate points | System |
| **Queries/Reports** | | |
| C0800 | View friend’s activity | User |

## Use Case Narratives (User Stories)

### Package A: Challenge sub-system

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A0100 | Create challenge | |
| **Primary Business Actors** | | **Other participating Actors** |
| Admin | | User |
| **Description** | The admin creates a challenge that is made available to users to add to their bucket list. | |
| **Pre-Conditions** | The admin must be logged in. | |
| **Triggers** | The admin selects the create challenge option. | |
| **Post-Conditions** | A challenge is created and saved. | |
| **Basic Flow of Events** | 1. Admin selects “create challenge” option 2. System displays new challenge form 3. Admin fills in form containing all information regarding challenge 4. Admin selects “save challenge” option 5. Challenge is saved | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A0200 | Approve/Reject suggested challenge | |
| **Primary Business Actors** | | **Other participating Actors** |
| Admin | | User |
| **Description** | The admin approves or rejects a challenge that has been suggested by a user. The admin can edit the challenge before approving it. | |
| **Pre-Conditions** | The admin must be logged in and a challenge must be suggested by a user. | |
| **Triggers** | A challenge is suggested by a user (A0500) and the admin selects the suggested challenges option. | |
| **Post-Conditions** | The challenge is either approved, saved and made available to users for use or rejected and discarded. | |
| **Basic Flow of Events** | 1. Admin selects “suggested challenges” option 2. System displays list of suggested challenges 3. Admin selects a suggested challenge 4. If admin approves challenge, admin selects “approve challenge” option and the challenge is saved | |
| **Alternate Flow** | * 1. Admin selects “edit challenge” option   2. System displays edit challenge form   3. Admin edits form containing all information regarding challenge   4. Admin selects “save changes” option   5. Changes are saved   6. If admin rejects challenge, admin selects “reject challenge” option and the challenge is discarded | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A0300 | Maintain challenge | |
| **Primary Business Actors** | | **Other participating Actors** |
| Admin | |  |
| **Description** | The admin makes changes to an existing or suggested challenge. | |
| **Pre-Conditions** | The admin must be logged in and the challenge must already exist (created, approved or suggested). | |
| **Triggers** | The admin selects the edit challenge option. | |
| **Post-Conditions** | The changes to the challenge are saved. | |
| **Basic Flow of Events** | 1. Admin selects “views challenges” option 2. System displays all challenges (A1000) 3. Admin selects challenge 4. Admin selects “edit challenge” option 5. System displays edit challenge form 6. Admin edits form containing all information regarding challenge 7. Admin selects “save changes” option 8. Changes are saved | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A0400 | Add challenge to bucket list | |
| **Primary Business Actors** | | **Other participating Actors** |
| User | |  |
| **Description** | The user adds a challenge to their bucket list. | |
| **Pre-Conditions** | User must be logged in and the challenge must exist. | |
| **Triggers** | The user selects the add challenge option. | |
| **Post-Conditions** | The challenge is added to their bucket list. | |
| **Basic Flow of Events** | 1. User selects “view challenges” option 2. System displays all challenges (A1000) 3. User selects challenge 4. User selects “add challenge” option 5. Challenge is added to their bucket list | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A0500 | Create challenge suggestion | |
| **Primary Business Actors** | | **Other participating Actors** |
| User | | Admin |
| **Description** | The user creates a challenge suggestion which is either approved or declined by an admin. | |
| **Pre-Conditions** | The user must be logged in. | |
| **Triggers** | The user selects the create challenge suggestion option. | |
| **Post-Conditions** | The suggested challenge is saved and made available to the admin for review (A0200). | |
| **Basic Flow of Events** | 1. User selects the “create challenge suggestion” option 2. System displays new challenge form 3. User fills in form containing all information regarding challenge 4. User selects “submit challenge suggestion” option 5. Challenge is saved | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A0600 | Mark challenge as completed | |
| **Primary Business Actors** | | **Other participating Actors** |
| User | |  |
| **Description** | The user marks a challenge as complete and it is made available for verification by friends. | |
| **Pre-Conditions** | The user must be logged in, the challenge must exist, and the challenge must have been added to their bucket list. | |
| **Triggers** | The user completes the challenge and selects the completed challenge option. | |
| **Post-Conditions** | The challenge is marked as complete and is made available for verification by friends. | |
| **Basic Flow of Events** | 1. User selects “view bucket list” option 2. System displays bucket list (B0400) 3. User selects challenge 4. User selects “completed challenge” option 5. Completed challenge is made available for verification by friends (C0500) | |
| **Alternate flow** | 4.1) If challenge requires photo, system calls A0700 | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A0700 | Upload photo | |
| **Primary Business Actors** | | **Other participating Actors** |
| User | |  |
| **Description** | The user uploads a photo of them completing a challenge. | |
| **Pre-Conditions** | The user must be logged in, the challenge must exist, the challenge must have been added to their bucket list, and the user must have a photo of them completing the challenge. | |
| **Triggers** | The user completes the challenge and selects the upload photo option (some challenges require a photo, others do not). | |
| **Post-Conditions** | The photo is uploaded, and the completed challenge is made available for verification by friends (C0500). | |
| **Basic Flow of Events** | 1. System displays a dialog box to select the photo to upload 2. User selects the photo 3. Photo is uploaded | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A0800 | View challenge progress | |
| **Primary Business Actors** | | **Other participating Actors** |
| User | |  |
| **Description** | The user can view the progress of an individual or group challenge. | |
| **Pre-Conditions** | The user must be logged in, the challenge must exist, and the challenge must be added to their bucket list. | |
| **Triggers** | The user selects the challenge. | |
| **Post-Conditions** | The system displays the progress of the challenge to the user. | |
| **Basic Flow of Events** | 1. User selects the “view bucket list” option 2. System displays bucket list (B0600) 3. User selects challenge 4. System displays challenge progress | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A0900 | Rate challenge | |
| **Primary Business Actors** | | **Other participating Actors** |
| User | |  |
| **Description** | The user can rate a challenge that they have completed. | |
| **Pre-Conditions** | The user must be logged in, the challenge must exist, and the user must have marked the challenge as complete (A0600). | |
| **Triggers** | The user selects the “rate challenge” option. | |
| **Post-Conditions** | The challenge rating is saved. | |
| **Basic Flow of Events** | 1. User selects “view challenges” option 2. User views challenges (A1000) 3. User selects challenge 4. User selects “rate challenge” option 5. System displays rate challenge form 6. User fills in form containing all information regarding rating and reviewing challenge 7. Rating is saved | |
| **Alternate Flow** | 1. User selects “completed challenge” option (A0600) 2. User selects “rate challenge” option 3. System displays rate challenge form 4. User fills in form containing all information regarding rating and reviewing challenge 5. Rating is saved | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A1000 | View challenges | |
| **Primary Business Actors** | | **Other participating Actors** |
| Admin, User | |  |
| **Description** | The user can view and filter through all available challenges. | |
| **Pre-Conditions** | The user must be logged in and at least one challenge must exist. | |
| **Triggers** | The user selects the “view challenges” option. | |
| **Post-Conditions** | The available challenges are displayed to the user. | |
| **Basic Flow of Events** | 1. User selects “view challenges” option 2. System displays all challenges 3. System displays filter form 4. User fills in form containing all information regarding filtering the displayed challenges 5. User selects “apply filter” option 6. System displays all challenges matching filter criteria | |

### Package B: Account sub-system

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| B0100 | Login/Logout | |
| **Primary Business Actors** | | **Other participating Actors** |
| Admin, User | |  |
| **Description** | The admin/user can log into their account using their credentials or log out of their account. | |
| **Pre-Conditions** | The admin/user must have created an account. | |
| **Triggers** | Admin/User selects the “login”/”log out” option. | |
| **Post-Conditions** | The admin/user is logged into or logged out of their account. | |
| **Basic Flow of Events** | 1. Admin/User selects the “login” in option 2. Admin/User enters account credentials 3. Admin/User is logged in | |
| **Alternate Flow** | * 1. Admin/User selects the “log out” option   2. Admin/User is logged out of their account | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| B0200 | Create account | |
| **Primary Business Actors** | | **Other participating Actors** |
| User | |  |
| **Description** | The user can create an account of their own. | |
| **Pre-Conditions** |  | |
| **Triggers** | The user selects the “create account” option. | |
| **Post-Conditions** | An account is created. | |
| **Basic Flow of Events** | 1. User selects “create account” option 2. System displays new account form 3. User fills in form containing all information regarding account 4. Account is created 5. System calls B0100 (Login) | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| B0300 | Maintain account | |
| **Primary Business Actors** | | **Other participating Actors** |
| User | |  |
| **Description** | The user can make changes to their account. | |
| **Pre-Conditions** | The user must be logged in. | |
| **Triggers** | The user selects the “maintain account” option. | |
| **Post-Conditions** | The changes to the account are saved. | |
| **Basic Flow of Events** | 1. User selects “maintain account” option 2. System displays edit account form 3. User edits form containing all information regarding account 4. User selects “save changes” option 5. Changes are saved | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| B0400 | View bucket list | |
| **Primary Business Actors** | | **Other participating Actors** |
| User | |  |
| **Description** | The user can view and filter through their bucket list containing all added challenges. | |
| **Pre-Conditions** | The user must be logged in. | |
| **Triggers** | The user selects the view bucket list option | |
| **Post-Conditions** | The completed and pending challenges are displayed to the user. | |
| **Basic Flow of Events** | 1. User selects “view bucket list” option 2. System displays users bucket list 3. System displays filter form 4. User fills in form containing all information regarding filtering the displayed bucket list 5. User selects “apply filter” option 6. System displays bucket list matching filter criteria | |

### Package C: Friends sub-system

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| C0100 | Send friend request | |
| **Primary Business Actors** | | **Other participating Actors** |
| User | |  |
| **Description** | The user can attempt to add another user to their friends list by sending a friend request. | |
| **Pre-Conditions** | The user must be logged in, the other user must exist, and the user must know the other users’ username. | |
| **Triggers** | The user selects the “add friend” option. | |
| **Post-Conditions** | The other user is added to the users’ friends list. | |
| **Basic Flow of Events** | 1. User selects “add friend” option 2. User searches for the other user by username 3. System displays search results 4. User selects friend from results 5. User selects “send friend request” option 6. Friend request is sent | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| C0200 | Accept/Reject friend request | |
| **Primary Business Actors** | | **Other participating Actors** |
| User | |  |
| **Description** | The user accepts or rejects a friend request from another user. | |
| **Pre-Conditions** | The user must be logged in and must have received a friend request from another user. | |
| **Triggers** | The user receives a friend request and selects the “received invites” option. | |
| **Post-Conditions** | If the friend request is accepted, the other user is added to the user’s friend list. If rejected, the friend request is discarded. | |
| **Basic Flow of Events** | 1. User selects “received invites” option 2. System displays all invites (friend request or challenge invite) 3. User selects friend request 4. If user accepts friend request, user selects “accept” option and other user is added to their friends list | |
| **Alternate Flow** | * 1. If user rejects friend request, user selects “reject” option and friend request is discarded | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| C0300 | Maintain friends list | |
| **Primary Business Actors** | | **Other participating Actors** |
| User | |  |
| **Description** | The user can maintain their friends list by viewing and removing friends. | |
| **Pre-Conditions** | The user must be logged in and have another user added to their friends list. | |
| **Triggers** | The user selects the “maintain friends” option. | |
| **Post-Conditions** | Changes to the friends list is saved. | |
| **Basic Flow of Events** | 1. User selects “maintain friends” option 2. System displays all friends | |
| **Alternate Flow** | 1. User selects “maintain friends” option 2. System displays all friends 3. User selects friend 4. User selects “remove friend” option 5. Other user (friend) is removed from the friends list | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| C0400 | Send challenge invite to friend | |
| **Primary Business Actors** | | **Other participating Actors** |
| User | |  |
| **Description** | The user can invite a friend to a group challenge. | |
| **Pre-Conditions** | The user must be logged in, the challenge must exist, and the other user must be a friend. | |
| **Triggers** | The user selects the invite friend option when selecting a group challenge from view challenges (A1000). | |
| **Post-Conditions** | The friend is added to the group challenge. | |
| **Basic Flow of Events** | 1. User selects “view challenges” option 2. System displays all challenges (A1000) 3. User selects group challenge 4. User selects “invite friend” option 5. User selects friend from friend list 6. Challenge invite is sent to friend | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| C0500 | Accept/Reject challenge invite from friend | |
| **Primary Business Actors** | | **Other participating Actors** |
| User | |  |
| **Description** | The user can accept an invite to a group challenge. | |
| **Pre-Conditions** | The user must be logged in, and the user must’ve received a challenge invite from a friend (C0400). | |
| **Triggers** | The user receives an invite to a group challenge from a friend and selects the “received invites” option. | |
| **Post-Conditions** | If the challenge invite is accepted, the challenge is added to the user’s bucket list. If rejected, the challenge invite is discarded. | |
| **Basic Flow of Events** | 1. User selects “received invites” option 2. System displays all invites (friend request or challenge invite) 3. User selects challenge request 4. If user accepts challenge request, user selects “accept” option and challenge is added to users bucket list (A0400). | |
| **Alternate Flow** | * 1. If user rejects challenge request, user selects “reject” option and challenge request is discarded. | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| C0600 | Verify friend’s challenge completion | |
| **Primary Business Actors** | | **Other participating Actors** |
| User | |  |
| **Description** | The user can verify that a friend has completed a challenge. | |
| **Pre-Conditions** | The user must be logged in, the other user must be added as a friend, and the friend must’ve marked the challenge as completed (A0600). | |
| **Triggers** | The user sees a friend has marked a challenge as completed and the user selects the “verify” option on a challenge in the view friends activity (C0800). | |
| **Post-Conditions** | The challenge completion is verified, and the friend receives points. | |
| **Basic Flow of Events** | 1. User selects “view friends activity” option 2. System displays all friends activity (C0800) 3. User selects challenge 4. User selects “verify” option 5. System calls C0700 | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| C0700 | Calculate points | |
| **Primary Business Actors** | | **Other participating Actors** |
| System | |  |
| **Description** | System allocates points to user for a verified completed challenge. | |
| **Pre-Conditions** | The user must’ve marked a challenge as complete and a friend must’ve selected the “verify” option on the challenge (C0600). | |
| **Triggers** | A friend selects the “verify” option on a challenge the user marked as complete (C0600). | |
| **Post-Conditions** | The system allocates points to the user. | |
| **Basic Flow of Events** | 1. Use case C0600 occurs 2. System calculates the amount of points that should be allocated to user 3. System allocates points to user 4. Changes are saved | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| C0800 | View friends activity | |
| **Primary Business Actors** | | **Other participating Actors** |
| User | |  |
| **Description** | The user can view and filter through the activities of their friends such as challenges in progress and completed. | |
| **Pre-Conditions** | The user must be logged in and have at least one user added to their friends list. | |
| **Triggers** | The user selects the “view friends activity” option. | |
| **Post-Conditions** | The user’s friends activities are displayed to the user. | |
| **Basic Flow of Events** | 1. User selects “view friends activity” option 2. The system displays all friends activity 3. System displays filter form 4. User fills in form containing all information regarding filtering the displayed friends activity 5. User selects “apply filter” option 6. System displays all friends activity matching filter criteria | |

# NON-FUNCTIONAL REQUIREMENTS

## Interface Requirements

BucketList is intended to be a web-based application with user-centered design making it easy to learn and navigate through the system. The GUI should be visually appealing, minimalistic with the goal to encourage users to add more to their bucket list, complete challenges and invite more friends to join. English will be used throughout the system.

**Usability goals:**

The system will be an effective tool in enticing users to get up and do the things they have always wanted to do but never had the motivation or had the time for. Navigating through should be fun, easy, straight forward, precise, intuitive for all ages and easy to remember. GUI design across the website will be consistent between the main bucket list and each individual user’s bucket list as well as their feed.

Once a new user has signed up, the system should immediately log them in, their corresponding tables efficiently updated; giving them access to all user controls such as edit profile, send friend requests etc. The system will offer simple error handling and easy reversal of actions. Not all personal information will be immediately available so that the user feels secure when using the website making the system safe to use in public spaces. There will easy access to all the system has to offer in one place, to the left of the website.

**User experience goals:**

When visiting the website, onlookers must want to immediately sign up and complete challenges as well as invite friends to join. GUI will be aesthetically pleasing, simple yet fun in order to compel users to complete challenges. There should be visual feedback when certain tasks are completed making even the most apprehensive users feel at ease and in control. Users should feel a sense of accomplishment when completing challenges and aim to complete more. By receiving points based on challenges completed users should feel motivated to continue using the system.

**User requirements:**

This system will be implemented as a website therefore access to a computing device is required, basic understanding of how computers work and an internet connection. General casual users to expert users will be able to use the system. Although the system is aimed at adventure seeking users there will be challenges available for the more cautious users.

## Performance Requirements

The system is intended to be a multi-user system thus accessing and capturing data into tables will have to be fast, accurate and reliable. Therefore, it is imperative that processing time should not depend on the number of users currently using the website. Logging into one’s profile should be done swiftly so that loading the profile and querying tables are not noticeable. Transitioning through interfaces should be swift and lag free as well as all responses should be immediate i.e. signing up, adding to bucket List.

Verifying that a challenge has been completed should happen automatically once a friend is verified it. The feed tab needs to be automatically and immediately updated every time a user’s-friend completes a challenge, or the users opts to open the tab. All notifications received by the user should be in real-time. The system should be available to the user and the admin 24/7. There should be regular challenge update in the main Bucket list ensuring that users always have more options than they need.

## Security Requirements

The Bucket List system will be integrating two types of users namely a general user and an admin. Each user has different levels of accessibility and different means of identification and authentication Both users will have unique login details.

All users are required to sign up with a valid email address, create their own password and a user profile before gaining access to the Bucket List system. Thus, each user and admin profile will be password protected. Thereafter, they may sign in with there username and password or their username and email address.

Bucket list is an online web-based system with the following security restrictions:

* All personal/ biographical information should be encrypted.
* User challenges only visible to users who are listed as “friend”
* Unless an individual has signed up and has a username and password, they will not have access to any Bucket List information besides what the system is about.
* Users only have access to their own profiles but may view other profiles and verify their challenges.
* Only admins may create, maintain and approve or reject challenges
* Users may only verify friend challenges and not their own
* No user may make themselves admin
* Users may only complete group challenges with users that are listed as their “friend”

## Operational Requirements

Bucket List will be web-based, coded in C# using Visual Studio. A SQL Server database will be used to manage and store the data captured into the system. A SQL Server Management System will be used during development to maintain the database. The system automation boundary will be represented by a mouse, keyboard, scanner, monitor and printer.

# DATA REQUIREMENTS

|  |  |
| --- | --- |
| **Entity** | **Entity Attributes** |
| User | User ID  Username  Password  First Name  Surname  Email  User Type  Points |
| Challenge | Challenge ID  Title  Description  Difficulty Level  Points  Need Photo |
| Rating | Rating ID  Challenge ID  User ID  Rating  Review |
| Suggested Challenge | Suggested Challenge ID  User ID  Title  Description  Difficulty Level  Points  Need Photo |
| User Challenge | User ID  Challenge ID  Status  Photo |
| Friend Invite | Friend Invite ID  From User ID  To User ID |
| Friendship | Friendship ID  User1 ID  User2 ID |
| Achievement | Achievement ID  Achievement Description  Difficulty Type  NumberNeeded |
| User Achievement | Achievement ID  User ID |
| Group Challenge Invite | Challenge Invite ID  From User ID  To User ID  Challenge ID  Status |

# Project Matters

## Open Issues

Possible nice to haves:

* Google Map view to specify and share your location
* Email notifications for friend requests and challenge requests
* Letting other users know when you will be completing a challenge in a specific location so that those with shared locations can collaborate.
* Give users the option of having a reset password link sent to their email if they forget their password
* Include the option to report a user
* Users can unlock new challenges when they reach a certain level of points

## Preliminary Schedule

|  |  |  |  |
| --- | --- | --- | --- |
| **Package/Sub-System Name** | **Team Member Responsible** | | |
| **Student Number** | **Name** | **Signature** |
| Challenge sub-system | **217143512** | **Bernard Mc Donald** |  |
| Account sub-system | **217198392** | **Mu’Minah Daniels** |  |
| Friends sub-system | **213348292** | **Tiffany de Vos** |  |

# Researching the Requirements

## 10 User Interview Questions

1. Do you want to be able to view the challenges without logging in, but in order to accept a challenge or interact with friends, you need to log in?
2. Would you prefer a star system for the difficulty level (e.g. 1 star = easy, 5 stars = extreme) or would words be better (e.g. easy, difficult, extreme)?
3. Do you want admin users to be able to participate in challenges?
4. Can you befriend anyone whose username you know or does it need to be accepted by the user?
5. If a user deactivates their Bucket List account, should their details be kept?
6. What happens if a challenge is completed but nobody verifies it or a challenge is not actually completed but is verified?
7. How are fake accounts handled? For example, creating a fake account to get points for a group challenge
8. Besides completing Bucket lost challenges, what else does this app provide? E.g.: charity, monetary gain, unlocking challenges based on points
9. Would you like to receive monthly information via email about new challenges and other related products?
10. What details do we need from the users, and what are you comfortable sharing?