### Inf 43 – Homework 1

##### Student Name: Max A. Villa

#### Student Number: 89938096

|  |  |  |
| --- | --- | --- |
| Awarded Points | Maximum Points | Document Aspect |
|  | 15 | Clarity of writing (spelling, grammar, sentence construction) and Clarity of expression (flow, structure, making logical arguments). Roughly 7.5 each. |
|  | 15 | Introduction / Executive Summary (can be different sections or combined into one) |
|  | 7.5 | Application Context / Environmental Constraints (can be different sections or combined into one) |
|  | 35 | Functional Requirements, including use-case diagram and description of each use case. |
|  | 7.5 | Software Qualities and Non-functional Requirements |
|  | 5 (+5) | Other Requirements and Other Items. At least a Glossary of Terms. You can earn up to 5 points Extra Credit if you go beyond Glossary |
|  | 7.5 | Assumptions / Risks (can be different sections or combined into one) |
|  | 7.5 | Priorities / Implementation Phases;  Future Directions and Expected Changes |
|  | **100** | TOTAL |

ARCHIE System Requirements

###### October 26, 2017

Max A. Villa

##### Introduction

Anteater Recreation Center Health Information and Events (ARCHIE) is a software project commissioned as an extension of the UC Irvine’s ARC center app. The current ARC app is limited to viewing information and does not allow for deep user engagement. With the extension of features, the goal is to improve the health of the UCI student body by increasing awareness of personal health metrics as well as health and fitness related classes and events.

This document specifies the requirements and other pertinent information surrounding the project. It is organized according to the requirements template provided by the client. Each section of the document contains a header describing its purpose and contents.

|  |  |
| --- | --- |
| **Table of Contents** | |
| Introduction | [This Section] Description of project purpose and this document’s purpose and organization. |
| Overview/ Executive  Summary | Addresses major intentions, features, parties involved, and context of ARCHIE. |
| Application Context/ Environmental  Constraints | Provides more information on development context of ARCHIE |
| Functional  Requirements | Details ARCHIE’s features and usage, including by whom (what type of user) and for what function. Describes all aspects of the software in detail, including capabilities and attributes. |
| Software Qualities and  Non-Functional  Requirements | Discusses necessary non-functional requirement details. |
| Other Requirements | All requirements not mentioned in previous categories. Includes a glossary of terms. |
| Assumptions/ Risks | Addresses all assumptions made not covered earlier, as well as any known risks to the project. |
| Priorities/ Implementation Phases | Identifies which of the system’s functionalities has the highest priority to be implemented first. |
| Future Directions and  Expected Changes | Provides inside and guidance to the system designer and programmers by discussing possible changes made to the  system in the future. |

##### Overview / Executive Summary

ARCHIE’s major intentions are not for profit but to improve UCI student health outcomes and increase student awareness of health and fitness events. It is intended to replace the current ARC app.

Its major features will include step counting, food logging, ARC class and event information and interaction, user-event information and interaction, and aggregate data report analysis.

ARCHIE will be directly associated with the ARC and its current website, which must be extended to provide a section for ARCHIE’s aggregate data reports. It will also provide means of direct payment for ARC classes from within the app which requires integration with the ARC’s current payment processing system. Both ARC admins and ARC webpage admins work requirements must be adapted to include integration with ARCHIE. Furthermore, restaurants, outlets, and other food vendors around campus will have their meals available for logging within ARCHIE.

It is assumed that mobile phone devices are a popular and familiar interface for the majority of the UCI student body, that the majority of smartphone users have either an up-to-date Android or iOS device, and that Facebook is available for external sharing.

##### Application Context / Environmental Constraints

ARCHIE is a mobile phone app whose main function and context revolves around the University of California Irvine. Its general user-base is required to only consist of current, registered UCI members with a valid UCI ID. Graduating or otherwise leaving UCI will result in lack of valid credentials for a given user. The app must provide food journaling abilities specific to the area around UCI, i.e. meal vendors on the campus, UTC, and Campus Plaza.

The platforms it must support are the most recent OS versions of iOS and Android smartphones. The device itself must support a mechanism for ARCHIE’s step counting. The app must interface with Facebook to allow simple sharing of ARCHIE events, and also provide a link to the pre-existing ARC website which should be extended to support a section where aggregate data reports of metrics collected from the userbase can be viewed.

The UI must be simple and intuitive, i.e. not requiring any specialized knowledge or training to use. The main metaphors for displaying event and social information should be a “wall”, “feed”, and “timeline” present in other popular social media apps like Facebook and Instagram.

##### Functional Requirements

This section specifies the ways in which each user role will interact with ARCHIE. The section starts with general- purpose use cases for all users and is then organized into subsections addressing the use cases organized by user role. Overlap between use cases are reiterated for the sake thoroughness. UI requirements are addressed in the above section (Application Context/Environment constraints). See the glossary of terms for further definitions of these components as they are referenced in this document.

Figure 1 shows the use case diagram for ARCHIE. The remainder of this section describes each use case in further detail.

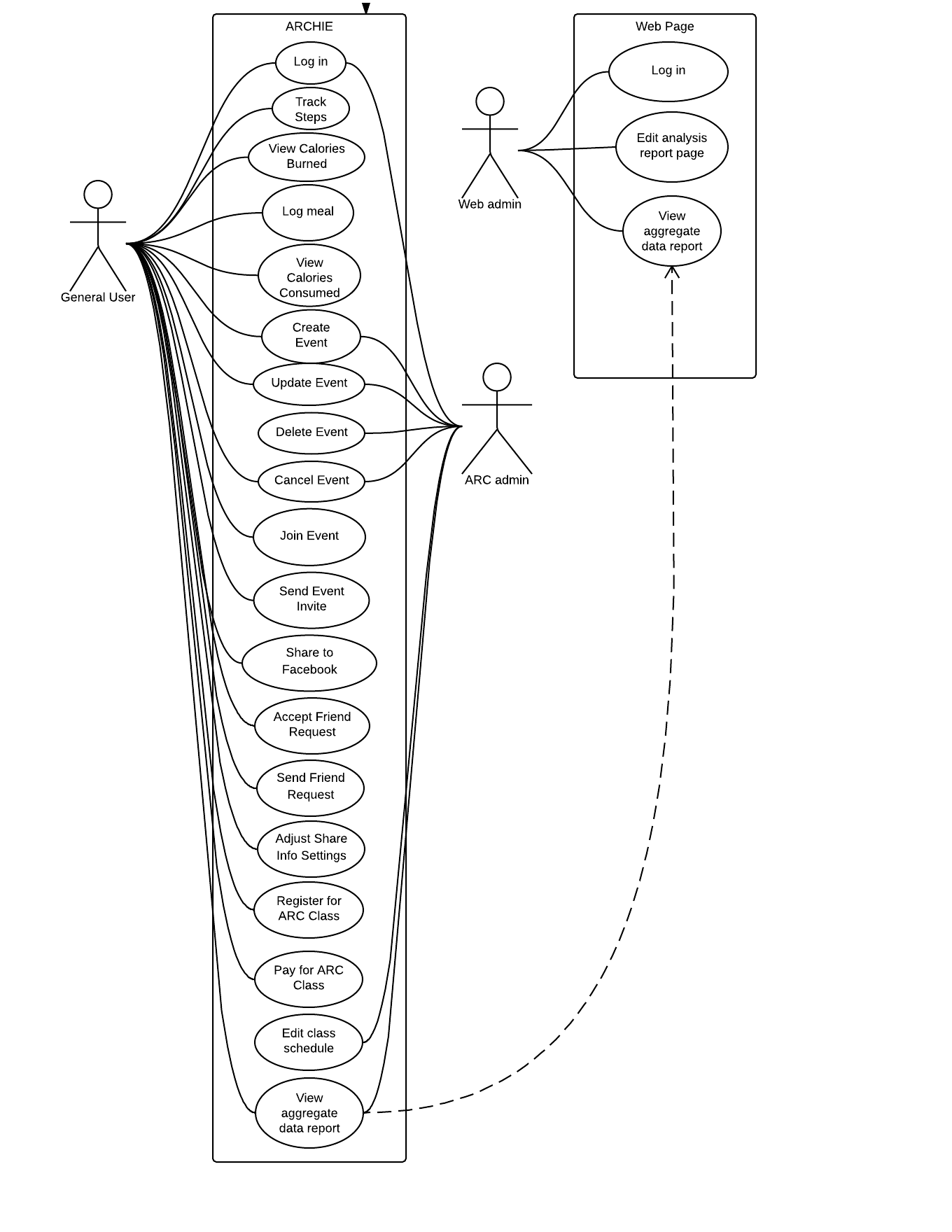
****

Figure : Use Case Diagram for ARCHIE

**4a Functionality Included for All User Roles**

***4a.1a – Logging In (ARCHIE APP):***

Login information is requested from each user to access features of the app.

- Preconditions: User must already have login credentials, i.e. a valid UCI ID or otherwise ARC admin credentials

Basic Flow:

1. User opens app in a compatible device

2. ARCHIE prompts user for login credentials

3. User fills in their login credentials

4. System validates user credentials and directs them to the main user interface

***4a.1b – Logging In (Web page):***

Login information is requested from Web admin to access features of the app.

- Preconditions: User must already have login credentials, i.e. a valid ARC web admin account

Basic Flow:

1. Web admin opens app in compatible browser

2. System prompts user for login credentials

3. User fills in their login credentials

4. System validates user credentials and directs them to the main user interface

***4a.2 – View Anonymized Aggregate Data:***

All roles (General Users and Admins) can view anonymized, aggregated data of metrics collected from general users of ARCHIE via a pre-existing companion website run on the ARC. The flow of viewing differs between general users, who use a link to from the app, and admins, who can access the web site directly; so, this will be reiterated in each role’s respective section below.

**4.1 – General User (Valid UCI ID) Functionality**

4.1a – Classes

***4.1a.1 – Register for ARC Class***

Basic Flow:

1. User views ARC class list

2. User selects ARC class

3. User chooses to register from ARC class info screen

4. System confirms class choice and prompts for payment

4a. User pays via mobile phone

4b. User selects alternative (non-mobile) payment method

4c. User cancels registration and resumes at flow at step 6.

5. System confirms registration

6. System returns to ARC class list

Exception Flow:

[3]error: System cannot register due to class at capacity

4. System displays error message and returns to basic flow at step 6

[4a]error: Payment declined.

5a. System prompts user to try again, select alternative(non-mobile) payment method. (Loop until successful or alternative selected)

5b. User cancels registration and returns to basic flow at step 6

6. Return to basic flow at step 5.

***4.1a.2 – Pay for ARC Class within App***

Users can pay via commonly supported payment methods such as credit/debit card. This is feature is a step in the use case flow of registering for an ARC class (4.1a.1). It’s exception case is discussed there.

***4.1a.3 – View ARC class list***

Users can navigate list of ARC classes and select from this list to display more information pertaining to the selected class. The registration flow begins from this navigation view.

4.1b – Fitness

***4.1b.1 – Track Steps (Step Counting)***

Preconditions: Device is capable and step counting is enabled by user.

If preconditions are met, the app keeps count of each step taken by the user while the phone is on their person and powered on. As discussed in the Other Requirements/Glossary section below, this record should sync to aggregate data report when internet is available.

***4.1b.1 – View Calories Burned***

The user may view total calories burned based on step count.

4.1c – Food Journaling

***4.1c.1 – Log meal***

Predefined meals of food items available from campus vendors, UTC, and Campus Plaza can be logged into food journal. Alternatively, meals can be created manually or customized by selecting a base and adding items individually.

***4.1c.2 – View calories consumed***

The user may view how many calories were consumed based on a daily calorie requirement either calculated based on user’s height and weight or entered manually by the user.

4.1d – Events

***4.1d.1 – Create Event***

Users can create health and fitness related events, setting the type of event, description, the date and time, and location.

***4.1d.2 – Update Event***

Users can update their own created events to adjust the type, description, data and time, and/or location of event.

***4.1d.3 – Cancel Event***

Users can cancel their own created events. Canceling notifies all users who had joined the event of the cancellation.

***4.1d.5 – Join Event***

Users can join an available ARC or user-created event.

***4.1d.6 – Send Event Invite***

Users can send a direct invitation to friends of an existing, available event.

***4.1d.7 – Share to Facebook***

Users can share event as a post on Facebook.

4.1e – Social Features

***4.1e.1 – Send Friend Request***

Users can request “friend” status to another user.

***4.1e.2 – Accept Friend Request***

Users can accept “friend status” from another user’s friend request.

***4.1e.3 – Adjust Shared Info Settings***

Users can adjust what information is visible on their profile for friends to view. The information settings may be from the following list:

1. Classes user is enrolled in
2. Events user attends
3. User step count
4. User consumed foods

4.1f – Data Analysis

***4.1f.1 – View Anonymized Aggregate Data:***

General users can follow a link within the app that opens a page on pre-existing ARC website housing statistics

**4.2 – ARC Administrator Functionality**

4.2a – Events

***4.2a.1 – Create Event***

ARC admins can create an official ARC event, setting the type of event, description, the date and time, and location.

***4.2a.2 – Delete Event***

ARC admins can delete any event, including general user-created events that are inappropriate.

***4.2a.3 – Update Event***

ARC admins can update created events to adjust the type, description, data and time, and/or location of event.

***4.2a.4 – Cancel Event***

ARC admins can cancel created events. Canceling notifies all users who had joined the event of the cancellation.

***4.2a.5 – Share Event to Facebook***

Users can share event as a post on Facebook.

4.2b – Classes

***4.2b.1 – Edit Class Schedule***

ARC admins can edit schedule information of ARC classes. This includes details such as time, location, price, instructor name, type of class, and brief description.

4.2c – Data Analysis

***4.2c.1 – View Anonymized Aggregate Data:***

ARC admins can follow a link within the app that opens a page on pre-existing ARC website housing statistics of anonymized, aggregate data metrics.

**4.3 – Web Administrator Functionality**

***4.3.1 – Edit Aggregate Data Report Web Page***

Web admins can edit aggregate data analysis report page to provide views of the statistical data in the form of graphs.

***4.3.2 – View Anonymized Aggregate Data***

Web admins can view aggregate data report page directly on the web page.

##### Software Qualities and Non-functional Requirements

**Usability** – The app should be simple and intuitive, in particular in its UI and UX aspects. Its primary userbase includes a diverse student body with only typical technical experience. Thus, features should be implemented with careful regard to minimizing complexity from the user perspective.

**Security** – Login credentials for the app are the same for all other UCINet resources and data metrics will be collected from the users of the app. Food logging and user-event sharing could be used for malicious purposes as well. Thus, information security is a primary concern.

**Scalability** – The UCI student body is around 33,000 students. Thus, development should prepare for functionality of the app under extreme cases of high user engagement. Furthermore, aggregate data is expected to be cumulative, i.e. spanning years, and in the future expand to include data from the entire UC system. Thus, considerations to scalability of data analytics/reporting methods are paramount.

##### Other Requirements/ Glossary

**User Profile:**

User account creation is automated and based upon UCINet ID database. Upon logging in, a user has a profile which should contain the following:

1. Name
2. Age
3. Height and Weight
4. Profile Photo

Height and Weight (item 3) is not part of the database, so a user must enter these. They are required for default calculation of calories requirement. The other items are part of the database and need not be supplied by the user.

**Data Sync:**

Data metrics will be collected from user activity on the app. The data should be anonymized and synced. These include metrics from food logging, event, and class participation, as well as the step counter (unless the user has disabled the feature) step counting data must be recorded if available. In the event that internet connection is not available, the step tracking feature must transmit the data appropriately when it is available.

**Glossary –**

* **Anteater Recreation Center (ARC) –** The Anteater Recreation Center is an 89,000-square-foot indoor gym facility that is part of campus recreation at the University of California, Irvine that holds classes for fitness and health related activities.
* **Anteater Recreation Center Health Information and Events (ARCHIE) –** Software project (mobile phone app) described in this requirements document.
* **ARC Admin –** User role fulfilled by ARC staff with valid credentials to edit class schedules and create events.
* **Class –** ARC-organized health or fitness class, e.g. judo or cooking class
* **Event –** ARC-organized or student-organized news item or gathering focused on health and fitness, e.g. basketball pick-up game, soccer match, etc.
* **General User –** User role fulfilled by anyone with a valid UCI ID, typically but not limited to the student-body at UCI
* **Web Admin –** User role fulfilled by ARC web page administrator(s) with valid credentials to edit ARCHIE aggregate data report section of ARC site

##### Assumptions / Risks

1. **Feature disable/lack of use Assumption:**

It is assumed that users can disable or not use features of the app. The step counting feature can be assumed to be able to be enabled/disabled by the user. When data is not available from the user, such as food logging or step counting, corresponding information such as calorie consumption/use should display as “0”.

1. **Logout Assumption:**

While not discussed, logout functionality of some form is assumed to be required. An automatic system logout similar to other UCINet resources should be considered implicit.

1. **Scheduling Risk:**

The timeframe given for the development of the app is 10 weeks, assuming the budget is adequate. The project runs the risk of running over schedule to support all features in both platforms supported.

1. **Security Risk:**

The app collects data from each general user to display on an external webpage. This is supposed to be anonymized, i.e. users identity should not be associated with their metrics. However, the potential for identity exposure exists.

1. **Development Risk:**

The current ARC app does not contain the majority of the features requested and the existing ARC web site must be extended to house the new ARCHIE aggregate data reports.

1. **Other:**

The app is dependent on an internet connection for the majority of features. It is uncertain how user experience may be effected by lack of of internet connection.

##### Priorities / Implementation Phases

Must Have:

* User ID and password login using UCINetID system
* User profile
* Step counting
* All predefined meals available on campus, UTC, and campus plaza available for food logging
* Aggregate, anonymized data reports analysis and viewing (link to data analysis)
* ARC class schedule view, registration, and payment
* Ability for ARC admin and general users to create, cancel, join, update events
* ARC admin moderator ability to delete events
* ARC and user-created event info wall
* Ability to view data reports in multiple (graph) views filtered by scope such as department, age, sex, student year standing(freshman, sophomore, etc.), etc.
* Ability to view data report graphs showing average step count and calorie consumption for every day/week/month/year

Should Haves:

* User activity page
* Ability to send and accept friend requests
* Ability to adjust shared information between friends
* Ability for user to privately few health comparison against rest of student body
* Ability for user to view own calories consumed based on food logging
* Ability for user to view own calories burned based on step tracking
* Logout capability (See Assumptions/Risk, item 1)

Nice to Have:

• Facebook integration to share activity externally

##### Future Directions and Expected Changes

1. The client has planned for data metrics to expand to the entire UC system, thus planning for future data analysis should consider the future expansion of the data pool.
2. External sharing to social media apps currently only includes Facebook but should later expand to more popular social media apps and websites.
3. The app currently will be developed for English speakers, but later support for languages other than English should be expected.
4. Since security is an important consideration in the app, it should be expected that the app be able to adapt to innovations in information security as they emerge.