Project Overview

A Collaboration Hub mobile application for Android capable of recalling user-specific routines and settings represented by records and communicating with a back-end database used for persistent storage.

The collaboration hub will bring all of the people and project resources together in one place. Similar to a web portal, the collaboration application will have links to social media, Google drive, and information such as maps, news as well as contact links for email, telephone, and video chat. This hub will also have a security certificate for secure file storage for sensitive information.

The collaboration hub will also allow members of a project to communicate easily with each other about project goals, tasks, and status. Various project details will be accessible to the public, so the collaboration hub has the potential to both help resources find projects, and projects find resources.

Team Member Names

- **Sevren Gail**
- **Christopher Rendall**
- **Natalie Calkins**
- **Nicholas Fullerton**

Team Processes

The team intends to ensure product reliability and fault-tolerance by participating in design and code review meetings, test driven development, group wide code-auditing, and implementation of well-documented standard and/or custom user and programming interfaces.

Project Communication

The team will organize and communicate through e-mail and hangout meetings taking place on an approximately bi-weekly basis or as-needed, and through repository commit comments.

Application Platform

Programming languages and tools used will be some or all of: Java, XML, Android Studio, Eclipse, Apache Tomcat, SQL (specifically MySQL).

System Overview and Requirements

Application Definitions

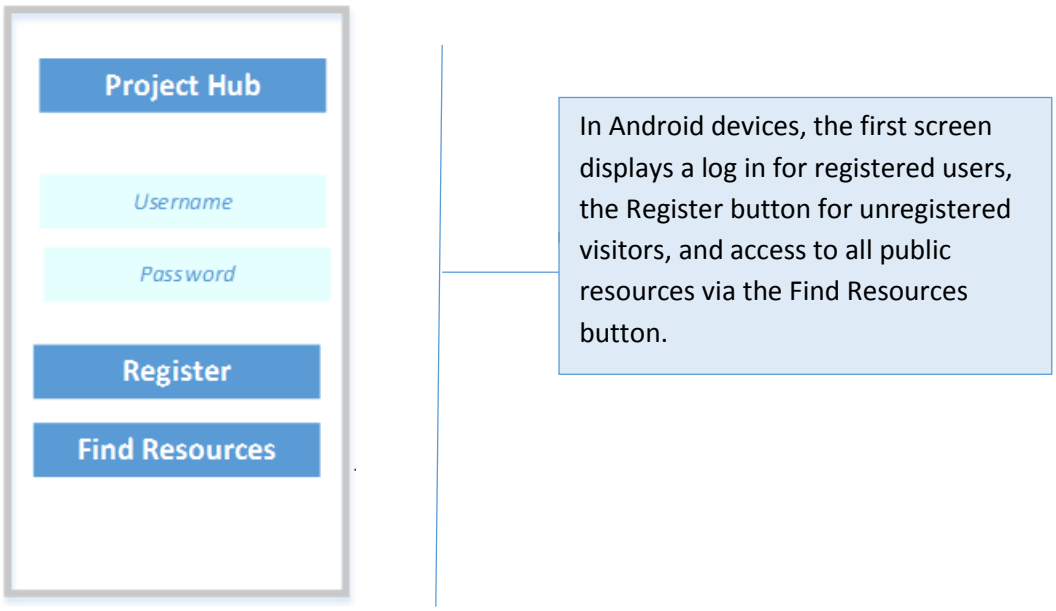
Project: Any goal that can be defined as a center focus that requires people and resources. All projects are private to the site. Not to be shown through search engines.

Resource: Any service, product or information that serves the project. A resource can be private or public. Public resources will show in search engines and to all projects when the user searches for Resources in the system.

Participant: Any person who serves a project. A participant can be private or public. Public participants will display Profiles in search engines and to all projects when the user searches Participants in the system.

Screens Overview

Design Screen 1: Home Screen



Design Screen 2: Personal Home

After logging in, the projects for the specific user are listed. From this page, the user can search for a project, scroll to find a project as well as add a project, and view global resources.

Design Screen 3: Project Detail

In the project detail page, the user can view all the resources in a project, search for a resource, add a resource, and edit the project.

Every Project receives a Forum. A Forum is called Discussion. Notifications via email are offered for all Forums in the My Profile Settings.

Design Screen 4: Add Resource to Project screen

After clicking the Add Resource option in the project detail screen, the user can create a resource and make it public or private. All public resources are available to non-registered users.

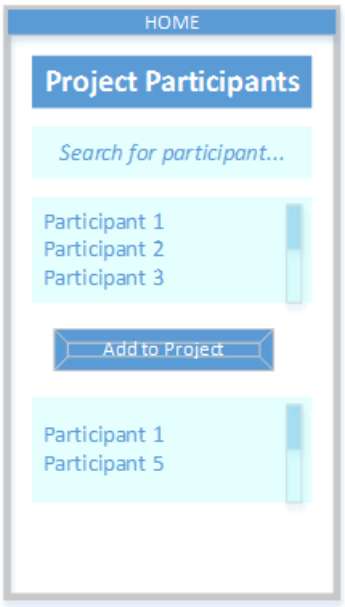
Clicking the Create Resource button takes the user back to the project's detail screen.

Design Screen 5: Add Project screen

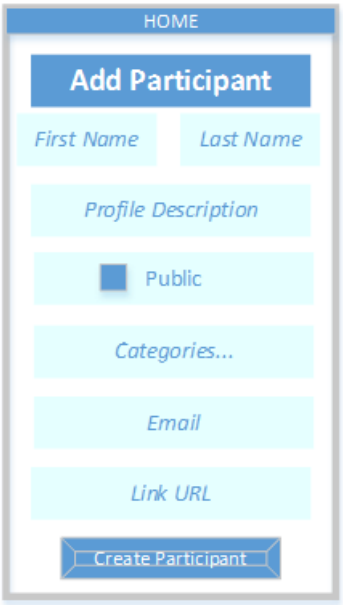
From the Personal Home screen, the user can add a project. Projects require:

- Project Title
- Description
- Project Owner: displays a list of users in the system
- Project Participants: displays a list of users in the system
- Project Resources: displays a list of resources both public and private in the system.

From the Project Detail page, a user a search for a resource and scroll through resources. The users selects, Ctl + select (Android equivalent) to identify resources to add to the projects, or right-click > Add.



From the Project Detail page, the user can select Participants global or private and add to the project following the same design for Resources. (*Note to programmers, the layout and functionality for Add Resources and Add Participants must be the same.*)



From the Project Detail screen and the Home page (*this is the page for Registration too*), the user can create a Participant:

- First Name, required
- Last Name, required
- Profile Description
- Public, required
- Categories
- Email, required (becomes username)
- Link URL

Requirements

ID	Date added	Priority	Description
REQ-1	11/10/15	ESSENTIAL	The system's user interface shall be an Android application.
REQ-2	11/10/15	ESSENTIAL	The system shall allow the user to manage projects by performing the following tasks through the user interface: (a) manage projects, (b) manage resources, and (c) manage participants.
REQ-3	11/10/15	ESSENTIAL	The system's back end will consist of a Java servlet and MySQL server. All activities performed by the user will be reflected by writing of data to the MySQL database.
REQ-4	11/10/15	ESSENTIAL	The system shall log all actions performed by the user through the Android interface.
REQ-5	11/10/15	ESSENTIAL	The Android user interface shall provide a means for the user to (a) create a new project hub account and (b) log into an existing account.
REQ-6	11/10/15	ESSENTIAL	The system shall lock a user's account after five unsuccessful login attempts.
REQ-7	11/10/15	ESSENTIAL	The system shall allow the user to communicate with a project hub administrator to unlock the user's account.

REQ-8	11/10/15	HIGH	The system should allow the user to collaborate with other project members through a messaging system.
REQ-9	11/10/15	HIGH	The system should allow the user to view and post to a project specific forum style discussion.
REQ-10	11/10/15	HIGH	The Android user interface should allow the user to manage their project hub account settings.
REQ-11	11/10/15	HIGH	The Android user interface should allow the user to manage Android application settings, including settings related to message and forum notifications.
REQ-12	11/10/15	LOW	The Android application might alert the user if new posts are made to forums of projects the user is a participant or owner of.
REQ-13	11/10/15	LOW	The Android Application might alert the user if a new message is received by the user's message inbox.

Iteration Requirements

Iteration 1

The following features will be implemented during iteration1:

- The default activity in the Android application that serves as the home screen.
- Log in functionality to authorize the user and display the user's personal home screen activity within the Android application.
- Display of dynamic content on the user's personal home screen activity.

Iteration 1 will begin with creation of acceptance tests necessary to determine successful implementation of the features listed above. Acceptance testing will include testing to ensure that the home activity is displayed when the Android application is initially opened, the user can successfully create an account by providing valid account details, the user is denied an account if invalid details are provided, the user can successfully log in by providing a valid username and password, the user cannot log in by supplying an incorrect username and password, the user is taken to their personal home screen upon successfully logging in, the user's home screen is displayed when resuming or reopening the application after logging in, and the content on the user's personal home screen is dynamically displayed based on data within the server-side SQL database.

Following creation of acceptance tests, the development team will design class diagrams to develop and solidify the complex relationships between the objects involved in the features listed above. Programming will be based upon the class diagrams developed during this phase.

Programming tasks, such as the coding of the classes designed in the design phase above, will be assigned individually. Each team member will be initially responsible for developing unit tests for each function for their assigned development classes. Each team member will also be responsible for producing fully coded classes. After each user initially codes their assigned work, they will be responsible for individually testing, fixing, and refactoring their code. All code will be deposited into GitHub.

The team will work together to integrate the code into a single project. Once integrated, the team will work together to test the integrated code.

Finally, the team will perform the acceptance tests and fix any issues that arise. Upon successful acceptance testing, the iteration will be considered a success.

Requirement ID	Programmer(s) working on requirement
REQ-1	*Sevren, Nicholas*
REQ-3	Chris, Natalie
REQ-5	Chris

Iteration 1 Summary

The following requirements were delivered during iteration 1:
REQ-1: The system's user interface shall be an Android application.
REQ-3: The system's backend will consist of a Java servlet and MySQL server. (Note: All activities have not been implemented, so the second part of this requirement has not been delivered.)
REQ-5b: The Android user interface shall provide a means for the user to log into an existing account.

Code and test cases for requirement 5b were uploaded to the GitHub. Test cases fully exercised each method of class Authenticator, including using an e-mail address that exists with the wrong password, using a username that exists with the correct password, using a username that exists with another user's password, and using empty strings for the username and password.

No test cases were written for requirement 1, as it is very generic and the user interface works at this point. Launching the Android application proves that the user interface works at this point. New requirements will likely be developed regarding user interface features, and more specific test cases can be written at that point in time.

No test cases were written for REQ-3, as we are still trying to figure out how to test the doPost() function. Unit tests will be written for this in the future.

Refactoring of code was necessary. Code was first developed for the Android application and the Servlet individually. Due to using many of the same classes and methods, code was refactored into a library called ProjectHubFramework. This serves as a central repository for all classes and methods that pertain to both the Servlet and the Android application. This makes it trivial to add new classes, debug code, and implement new features.

The Singleton and Model, View, Controller design patterns were used during this iteration. The Singleton is used to enable Application scope variables for the Android application, as each Activity seems to serve as its own application, making it difficult to share variables between activities. The Singleton model allows a single instance of a configuration object to hold Application wide variables.

The Model, View, Controller design pattern is used in the login activity for the Android application. The Activity itself serves as the view, while an independent Controller class handles user input. The Model would be very basic, so the View serves as the Model as well.

Iteration 2

Implementation 2 will begin with the creation of acceptance tests necessary to determine successful implementation of the features listed above. Acceptance testing will include ensuring that the user can successfully create an account by providing valid account details, the user is denied an account if invalid details are provided, or if the user's e-mail address has already been registered, the user is taken to their home screen upon successfully logging in, and the content on the user's personal home screen is dynamically displayed based on data within the server-side SQL database.

Following creation of acceptance tests, the development team will design class diagrams to develop and solidify the complex relationships between the objects involved in the features listed above. Programming will be based upon the class diagrams developed during this phase.

Each team member will be responsible for coding the classes required to implement the requirements above. Each team member will be responsible for individually writing unit tests, testing, fixing, and refactoring their code. All code and unit tests will be deposited to GitHub.

The team will work together to integrate the code into the Servlet, Android application, and ProjectHubFramework. Once integrated, the team will work together to test the integrated code.

Finally, the team will perform the acceptance tests and fix any issues that arise. Upon successful acceptance testing, the iteration will be considered a success.

At this point, development is moving slower than planned. Now that the servlet and Android application frameworks are working, development should move more smoothly.

Requirement ID	Programmer(s) working on requirement
REQ-5b	Chris
REQ-6	Chris
REQ-4	Sevren, Nicholas, Natalie

User Stories

Iteration 1

ID	User Story	
ST-1	*As a new user, I can access the Project Collaboration Hub on my Android smartphone.*	
ST-1	*As a site visitor, I can view all public resources.*	
ST-3	*As a server admin, I can manage the system's data using MySQL server.*	
ST-5	*As a registered user, I can log in to my account and see My Projects screen.*	
ST-5	*As a registered user, I can add a new project.*	

Iteration 2

ID	User Story	
ST-1	*As a site visitor, I can create a ProjectHub account.*	
ST-2	*As a new user, I can log into my created account.*	
ST-3	*As a user, I can view my associated projects on the Android application.*	
ST-4	*As the Project Hub's database administrator, I can view any user's actions performed through the Android application by viewing the database.*	

Preliminary Diagram: