

ConnActiv – Software Plan

Group 7 – David Johnson, Kim Cooperrider, Ray Wang, Rob Filippi, Vincent Tran

CS 1530 - Software Engineering

I) Scope

ConnActiv is a social network meant to connect people who participate in physical activities. Its goal is to allow users to meet other people in an area based on a mutual interest in sports, running, hiking, etc. Users will be able to subscribe to different activities based on their interest (e.g. a running “activity”). Each activity feeds into its own public stream containing posts that mention or “tag” the activity, as well as any other relevant updates. The stream is able to be seen in a user’s “Home” view if he or she is subscribed to the particular activity. This public stream is intended for users to post what activity they are doing and when. This allows other users to join them if the original user allows invitations to his or her posts.

For example, Jon posts the following message on the site’s “Running” section: “Hey, going out for a run in Oakland at 9AM.” If Jon has allowed it, Stacy may ask to join Jon in running that day.

Finally, users may give other users recommendations after they have done an activity together. This gives other users an opportunity to see if the user in question would be a suitable activity partner; however, because this could potentially lead to cyber bullying in the form of malicious reviews, users are limited to one review per activity with the person in question.

a. Functions

User login – Allows the user to login to the site.

User info – A database entry is stored with all user-inputted info about him or her.

Public stream – Pulls the latest entries from an activity’s database and displays them to users who are subscribed. Stream entries will be interactive -- if the posting user specified that the activity is open, subscribers of the activity may ask to join. For better usability, tabs will be implemented to separate streams of activities and networks to which the user subscribes. There will be tabs that correspond with the activities and networks that the user subscribes to.

Recommendations – Users will be able to recommend other users based on their performance, helpfulness, friendliness, etc. when they participate in activities together. A user may write a short review on the user and he/she may simply provide a “thumbs up” or “thumbs down.” A user's ratio of thumbs up/down is displayed in a simple, graphic form on the profile below the profile picture, and when clicked links to a more detailed Recommendations view.

Requests – Requests will come in two forms: activity and connection. When a user asks to join another user to participate in an activity, this will be an activity request. A request to be buddies will be a connection request.

b. Performances

We aim to be able to serve up to 200 concurrent users.

c. Limitations

We predict that anything more than 200 concurrent users will slow down server responsiveness dramatically.

II) Tasks

Database creation and management – The backbone of our system relies on databases to store user information and interaction.

Clean and modern UI – In order to fit with the growing world of social networks, our user interface has to look modern and be simple to navigate.

Android app – An android app will be developed as another way for users to access the website.

III) Resources

a. Hardware

Besides computers to program and maintain the site, a web server is needed to host the site. Because we are creating a largely PHP-based web application, our server needs to support the use of PHP and MySQL for our databases.

b. Software

Each developer will be using his or her preferred program to develop the code. We will be programming the front-end of our web application with HTML, CSS (supplemented by LESS, a CSS extension to simplify and optimize stylesheets), and JavaScript (notably jQuery). For back-end, we will be utilizing PHP with MySQL and an Apache web server to manage and manipulate the site's databases.

c. People

Vincent Tran – Project Manager, Salesperson, Android Specialist

David Johnson – Database Specialist

Kim Cooperrider – Front-end Specialist, Document Specialist

Ray Wang – Tester

Rob Filippi – Tester, PHP Specialist

IV) Cost

Function	Optimistic	Most	Pessimistic	Expected	Deviation	\$/Line	Line/Month	Cost
Login	100	200	300	200	33	1	500	500
Registration	300	400	550	408	42	1	500	500
Subscribe	400	500	700	516	50	1	500	500
Approve	500	700	800	683	50	1	500	500
Posting	300	400	600	416	50	1	500	500
Join	20	250	400	266	33	1	500	500
News Feed	300	400	600	416	50	1	500	500
Favorites	100	200	300	200	33	1	500	500

V) Scheduling

	1	2	3	4	5	6	7	8
Login	O	X						
Register	O	X						
Subscribe		O	X					
Approve			O	X				
Posting				O	X			
Join					O	X		
News Feed						O	X	
Favorites							O	X

Person	Date Modified	Section	Description
Vince	9/22	1, 2	First draft
Kim	9/22	3	First draft
Vince	10/6	4, 5	First draft
Vince	12/10	1	Reworked scope to reflect changes made to the website