Project Proposal

GigHub:

A Social Web Application for Musicians, Promoters and Fans

Niall Ryan

12114812

niall.ryan1@student.ncirl.ie

Higher Diploma in Science in Web Technologies

22 February 2013

Objectives

GigHub will be a social media application which will allow music fans to connect with each other and with bands and musicians who wish to promote their own events, as well as with larger promoter who wish to share their events with a wider community, an alternative to traditional means of promotion i.e. TV/radio advertising, posters etc.

Users will be able to create accounts, add other users as friends, share status updates, join groups to interact with users who they might not wish to connect with directly, and post about events that they will be or would like to attend.

Bands and promoters will also be able to create accounts and make use of the standard functionality, while also being able to use the app to create event pages to share with other users which will contain info on the venue, ticket prices and availability.

Promoters will have access to other advanced features, allowing them to tailor the promotion of their events by sharing them with users that they are not directly connected to based on user preferences and the kind of events they have specified they are interested in or will be attending.

Advanced functionality where users can purchase tickets for events, as well as music merchandise directly from the app through an integrated e-commerce system could be implemented too, making **GigHub** a one-stop-shop for all music fans.

The app will be developed using **Ruby on Rails** on the back-end while the front-end will use **HTML5**, **CSS3, jQuery, Ajax** or perhaps a JavaScript framework such as **Backbone.js**, **Ember.js** or **Batman.js**, all of which integrate well with Rails, to create a rich, user-driven application that will be both easy-to-use and professional in execution.

Background

GigHub will tap into the community-driven event promotion market and could become a valuable tool for general music fans as well as promoters, bands, record labels, artist management and marketing executives.

Obviously, the major competitor for GigHub will be social media behemoth Facebook. While many would, probably correctly, argue that to take on Facebook is a waste of time, I hope GigHub will offer a unique alternative for music fans who have become jaded by the many constant gimmicky changes that Facebook have inflicted on their users.

When FB introduced their Post Promotion features for Pages, there was uproar among independent musicians and promoters in Ireland, for whom Facebook had become an invaluable tool for promoting their music and interacting with their fan base. By cutting post visibility and encouraging users to pay what many considered extortionate rates to push content out to fans, Facebook managed to alienate a large proportion of their user base.

It is hoped that by providing a unique, easy-to-use application, created specifically for the music community by developers who understand how and why musicians and promoters use social media tools, GigHub can manage to attract a large and vibrant community that will set it apart from Facebook and become a valuable presence in the online music scene.

Technical Approach

Methodology

This project will be developed using the Rapid Application Development Methodology due to the short time-scale and small development team (in this case, just one person). Following RAD principles will allow me to develop a prototype very quickly and add and remove functionality on the fly as deadlines start to come nearer.

System Architecture

As specified above, the application will be created using the Ruby on Rails framework which uses the Model-View-Controller system architecture pattern. The MVC pattern allows for the seperation of concerns, removing the modelling of the domain from the user's interaction with it and the presentation of information. Models will include Users (which will be broken down further into different types such as Fan, Band, Promoter etc.), Events and Groups but further details analysis will be required, as well as some trial and error, before all models necessary can be listed.

Technical Details

As stated previously, the application will be created using Ruby on Rails, and will incorporate some Gems such as Devise or Authlogic in the prototyping stage. As the project approaches its final iteration, it is hoped that most of the functionality will be custom built but that may not be possible depending on time constraints.

Rails uses sqlite3 as a database during development and I see know need to change this but if/when the application is deployed to Heroku, the database will be migrated to PostgreSQL, as this is required by Heroku.

Most of the HTML will be generated by Rails but the styling will be customised after the prototyping phase and will be implemented using CSS3 techniques.

JQuery will also be used to enhance the user experience on the front-end, while a JavaScript framework such as Backbone.js, Ember.js or Batman.js may also be used, again depending on time constraints and comfort with the technology.

Evaluation

The system will be developed incorporating elements of test-driven development to ensure that the system functions correctly as it is being developed. Writing system and integration tests in Rails is relatively easy and should enhance the development process without slowing down development too much.

I hope to test the application with a small group of users from the target market once the system is in beta and if time allows. By deploying the app with Heroku, I will allow this group to completely test the application remotely, without having to have the testers use the system on a local server which will hopefully speed up the process, and they will be able to provide me with feedback on bugs and any features that could be implemented, as well as comments and criticism on functionality and design.

Consultations with Academic Staff

Awaiting replies from Mr. Frank McArdle and Mr. David Hamill

Project Plan

Please see Gantt Chart uploaded along with this file.