Running head: LAB 2 – STROME INFUSION PROTOTYPE SPECIFICATION

Lab 2 – Strome InFusion Prototype Specification

Patrick Ferate

Old Dominion University

CS411W

Janet Brunelle

November 22, 2015

Version 1.1

Table of Contents

List of Figures

List of Tables

1   Introduction

Old Dominion University (ODU) has many ways of preparing Monarchs for the future. Some of their futures involve entrepreneurship and ODU's Strome Entrepreneurial Center (SEC) is a valuable resource available to the Monarch family wanting to go down that path. The SEC is a place for students, faculty, staff, and alumni receive guidance on starting a business or improving an existing business. The SEC staff bring in speakers from the business community and host workshops to help future entrepreneurs develop skills that can help in there endeavors. The SEC lacks the ability to easily connect with students, maintain those connections, and build connections within the center's members; and therefore, many more opportunities to help future entrepreneurs are missed.

1.1   Purpose

Strome InFusion is a way to improve the potential missed opportunities that the SEC faces. InFusion gives its members the ability to easily connect with the SEC, other Monarchs, and local businesses. With this network, entrepreneurs will be able to find the necessary skillsets and resources to make their ventures a success. Figure 1 shows the current process that students experience when interacting the SEC and the missed opportunities that Strome InFusion is designed to address.

InFusion addresses these opportunities to help improve everyone's experience with the SEC. InFusion has a simple registration process, tracks Monarch ventures, improved communication between Monarchs, the SEC, and other members of InFusion. Through the InFusion connections, entrepreneurs will be able to find the resources to make their ventures a success.
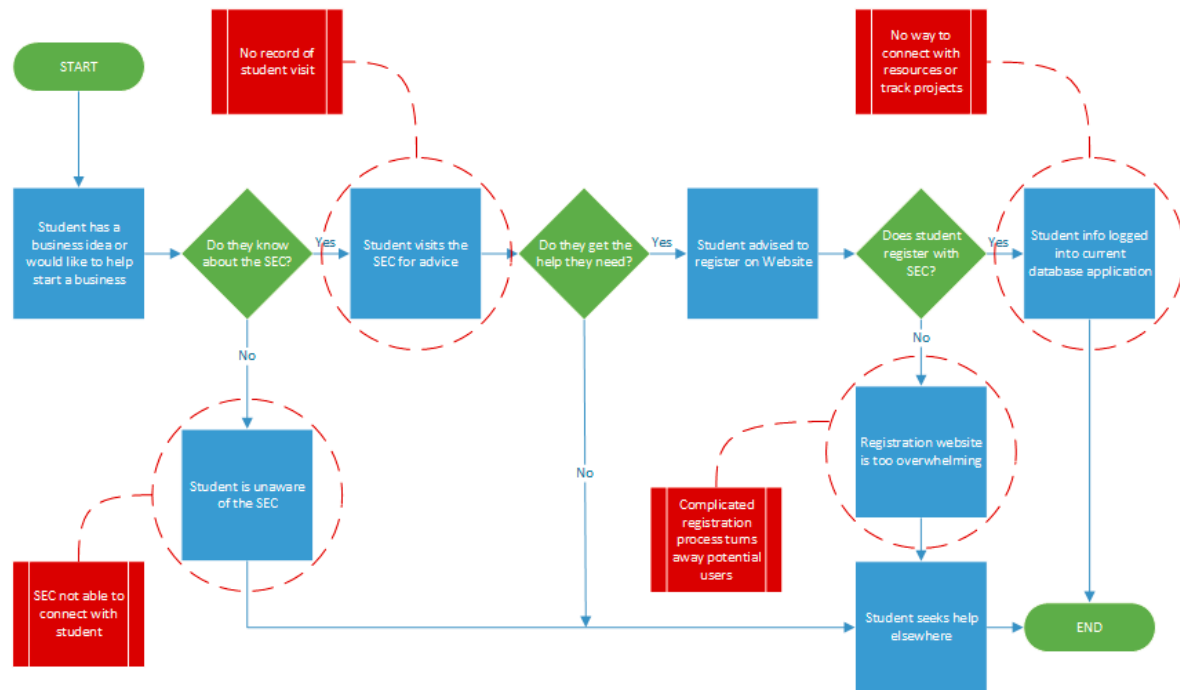
Figure 1. Current Process and Missed Opportunities

1.2 Scope

The prototype for Strome InFusion represents the major components of the full product, but will differ in terms of hosting, scale, and data. Testers and data will be simulated by the Strome InFusion development team. More feature differences are listed in Table 1.

[This section intentionally left blank]

Features/Components	Real World Product	Prototype
Students	Actual students at ODU who wish to collaborate on a business venture	Actual student and Strome InFusion team members using virtual machines using Strome InFusion team data along with special test functionality
SEC Staff	Actual SEC staff and mentors who will provide guidance for the project	Director Grden as well as Strome InFusion team members simulating Strome Entrepreneurial Center staff
Internal Resources	Actual students who want to utilize their skills to work on a business venture	Strome InFusion team members connecting as a venture to test functionality
External Resources	External investors and mentors who are willing to contribute to the success of the venture	Test data provided by Director Grden and the Strome InFusion team
Mobile Application	Mobile application will be available natively for all major mobile OS platforms	Will only be available natively for Android devices
Web Server	An independent web server hosting Strome InFusion web application	Virtually hosted server at ODU
Database Server	An independent database server hosting Strome InFusion database	Virtually hosted server at ODU

Table 1. Feature comparison between full product and prototype

Instead of independent systems for the web and database servers, both services will be running on the same virtual machine, which is hosted by the Systems group in ODU's Computer Science department. InFusion users will be simulated by members of the development team and Director Grden. This will involve acting as users of all types.

1.3   Definitions, Acronyms, and Abbreviations

<u>ACID compliance:</u> Atomicity, consistency, isolation, and durability are the properties that

guarantee that a database transaction is processed reliably.

Agile (Software Development): A group of software development methods in which solutions

evolve through collaboration between self-organizing, cross-functional teams. It

promotes adaptive planning, evolutionary development, early delivery, continuous

improvement, and encourages rapid and flexible response to change.

AngularJS: A structural framework for dynamic web applications, based off of JavaScript.

Apache Cordova: An open source project that provides a set of device APIs that allow a mobile

application to utilize native device functions from JavaScript.

Apache web server: The world's most widely used web server software.

API server: A computer that provides API data over a network.

Application Programming Interface (API): An application programming interface is a set of

routines, protocols, and tools for building software applications.

Application server: The middleware of a server centric architecture.

Backend: The server, the web application, and the database.

Business Canvas: A strategic management and lean startup template for developing new

business models.

Company: Any business that is owned or operated outside of ODU.

Company resource: A resource an external company registers in Strome InFusion to assist with

student ventures.

Concurrency: Allowing multiple processes to go at once.

Cordova: See Apache Cordova.

CSS: Cascading style sheets, gives form to the HTML page.

Data integrity: The accuracy and consistency of data stored in the database.

Database: A comprehensive collection of related data organized for convenient access.

DOM: The Document Object Model is a logical structured document applied to HTML, it

represents separate and distinct browsers and interfaces to view and interact with the web

application.

Employer: A person or company that hires at least one or more employees.

Encryption: Encoding information in a way that is not readable except by authorized entities.

Entrepreneur: A person who organizes and operates a business or businesses, taking on greater

than normal financial risks in order to do so.

Firewall: Software that prevents hackers from gaining access to systems with sensitive data.

Flask: A python web application framework.

Frontend: The HTML, CSS, and JavaScript of a web page.

GUI: Graphical user interface, is an interface that allows the user to interact with the application.

Horizontal Scaling: Adding more nodes to a system, such as adding a new computer to a

distributed software application.

HTML: Hypertext Markup Language, a standardized system for tagging text files to achieve

font, color, graphic, and hyperlink effects.

InFusion: An alternative name for Strome InFusion.

JavaScript: An object-oriented computer programming language commonly used to create

interactive effects within web browsers.

JSON: JavaScript Object Notation is a minimal, readable format of structured data.

Linux: A Unix-like and mostly POSIX-compliant computer operating system assembled under

the model of free and open-source software development and distribution.

Linux distributions: Operating systems based off of the Linux kernel.

Load balancing: The distribution of processes across multiple available systems to improve performance.

Mobile application: An application developed to be operated and interacted within a mobile operating system.

Monarch: Students, faculty, staff, and alumni of Old Dominion University.

Monarch Family: See Monarch.

Monarch Community: An individual or organization that is involved on campus, but is not a Monarch themselves.

MVC: Model View Controller is an architectural pattern for building applications.

MySQL: An open-source relational database management system.

Native application: An application that has access to system functionality (i.e. the camera on a cell phone).

Nginx: A web server with a strong focus on high concurrency, performance and low memory usage.

ODU: Acronym for Old Dominion University.

Object-relational database: Is a database that uses an object-oriented model: objects, classes, and inheritance are directly supported in the database schema and in the query language.

Object-Relationship Mapper (ORM): A programming technique for converting data between incompatible type systems in object-oriented programming languages.

Open source: Software where the source code is freely available.

Operating system: The software that supports a computer's basic functions (i.e. launching applications and handling peripherals).

PostgreSQL: An open sourced, object-relational database system.

Push Notification: The delivery of information from a software application to a computing

   device without a specific request from the client.

Python: A widely used general-purpose, high-level programming language.

Representational State Transfer (REST): Software architectural style of the World Wide Web.

RESTful framework: Representational state transfer is an architectural style framework for the

   development of web services.

Reverse proxy server: A server that can retrieve resources from one or more servers.

SEC: Acronym for Strome Entrepreneurial Center.

SI: Acronym for Strome Infusion.

SDK: Software Development Kit, a set of software development tools that allow the creation of

   applications for a certain piece of software.

Server: A computer system that manages access to a central resource or service.

Small Business: A business that has less than 500 employees.

SQL: Structured Query Language is used to communicate with relational databases.

SQLAlchemy: A Python SQL toolkit and Object Relational Mapper that gives application

   developers the full power and flexibility of SQL.

Stakeholder: A person who will be affected by the outcome of Strome Infusions development..

Toolkit: A set of software tools.

Ubuntu: An open source operating system based off of Linux.

User experience: The overall experience of a person using a product.

User skill: A skill a user registers in Strome InFusion to assist with ventures.

uWSGI: Deploys the application server and implements WSGI protocols.

Venture: A student run business enterprise with a dedicated profile within Strome InFusion.

Vertical Scaling: Adding resources to a single node in a system, typically involving the addition

of CPUs or memory to a single computer.

Virtual Machine or VM: An operating system OS or application environment that is installed

on software which imitates dedicated hardware.

Web application: Client-server software application in which the client or GUI runs in a web

browser.

Web application framework: A framework for software that supports web applications,

services, and resources.

Web server: Systems that deliver web pages.

Web Server Gateway Interface (WSGI): Specification for simple and universal interface

between web servers and web applications for the Python programming language.

1.4   References

Ferate, P. (2015). *Lab 1 – Strome InFusion Product Description.* Norfolk.

Strome Entrepreneurial Center. (n.d.). *Strome Entrepreneurial Center: Frequently Asked*

Questions. Retrieved September 18, 2015, from Strome Entrepreneurial Center Web site:

https://www.odu.edu/impact/entrepreneurship/strome/answers

Council, J. (2014). Alumnus Mark Strome's $11M gift to ODU is not simple charity. Retrieved

October 21, 2015, from *Inside Business: The Hampton Roads Business Journal*:

http://insidebiz.com/news/alumnus-mark-stromes-11m-gift-odu-not-simple-charity

Lascara, V. (2014). ODU opens Strome Entrepreneurial Center. Retrieved September 18, 2015,

from *Inside Business: The Hampton Roads Business Journal*:

http://insidebiz.com/topstories2014/news/odu-opens-strome-entrepreneurial-center

Noyes, K. (2010). Five Reasons Linux Beats Windows for Servers. Retrieved September 18,

2015, from PCWorld Magazine Security Web site:

http://www.pcworld.com/article/204423/why_linux_beats_windows_for_servers.html

Netcraft. (2015). September 2015 Web Server Survey. Retrieved September 19, 2015, from

Netcraft News Web site: http://news.netcraft.com/archives/2015/09/16/september-2015-

web-server-survey.html

Ellingwood, J. (2015). Apache vs Nginx: Practical Considerations. Retrieved September 19,

2015, from Digital Ocean Community Web site:

https://www.digitalocean.com/community/tutorials/apache-vs-nginx-practical-

considerations

The Organic Agency. (2013*).* Apache vs Nginx {Performance Comparison}. Retrieved

September 19, 2015, from The Organic Agency Web site:

http://www.theorganicagency.com/apache-vs-nginx-performance-comparison/

Thousand Eyes. (2013). Improving End User Experience with AngularJS. Retrieved September

19, 2015, from Thousand Eyes Web Site: https://blog.thousandeyes.com/improving-end-

user-experience-with-angularjs/

Tezer, O.S. (2014*).* SQLite vs MySQL vs PostgreSQL: A Comparison of Relational Database

Management Systems. Retrieved September 19, 2015, from Digital Ocean Community

Web site: https://www.digitalocean.com/community/tutorials/sqlite-vs-mysql-vs-

postgresql-a-comparison-of-relational-database-management-systems

Cordova. (n.d.). Getting Started with iOS. Retrieved September 20, 2015, from Apache Cordova

Documentation Web site: https://cordova.apache.org/docs/en/2.5.0/guide_getting-

started_ios_index.md.html

1.5   Overview

The architecture of InFusion is similar to other web based applications, the details of which will be described in this document. The details of the prototype include hardware and software configurations, features, and capabilities.

2   General Description

The following sections describe the prototype in detail. The prototype's architecture is described in section 2.1. The prototype's functionality is described in 2.2. The user experience for the website and mobile application will be described in 2.3.

2.1   Prototype Architecture

The Strome InFusion prototype uses the same core software as the full product. All of the services are running on an Ubuntu Linux virtual machine. When users connect to InFusion, the web server, Nginx, directs all incoming requests. Any static data is quickly served directly by Nginx. Other requests get passed down to the back-end web application through the uWSGI gateway. The back-end web application is a Flask application, written in Python. The Flask application serves multiple purposes; it handles user authentication and serve basic HTML pages that utilize AngularJS. The AngularJS code accesses a RESTful API, which is also served by Flask. The data for InFusion is stored in a local PostgreSQL database.

Figure 2 visualizes the major components of the Strome InFusion prototype, as well as the interconnections between the end users and Strome InFusion. Instead of having separate, dedicated systems for the web and database servers, the processes will all be running on a virtual machine that is hosted by the Computer Science department at Old Dominion University.
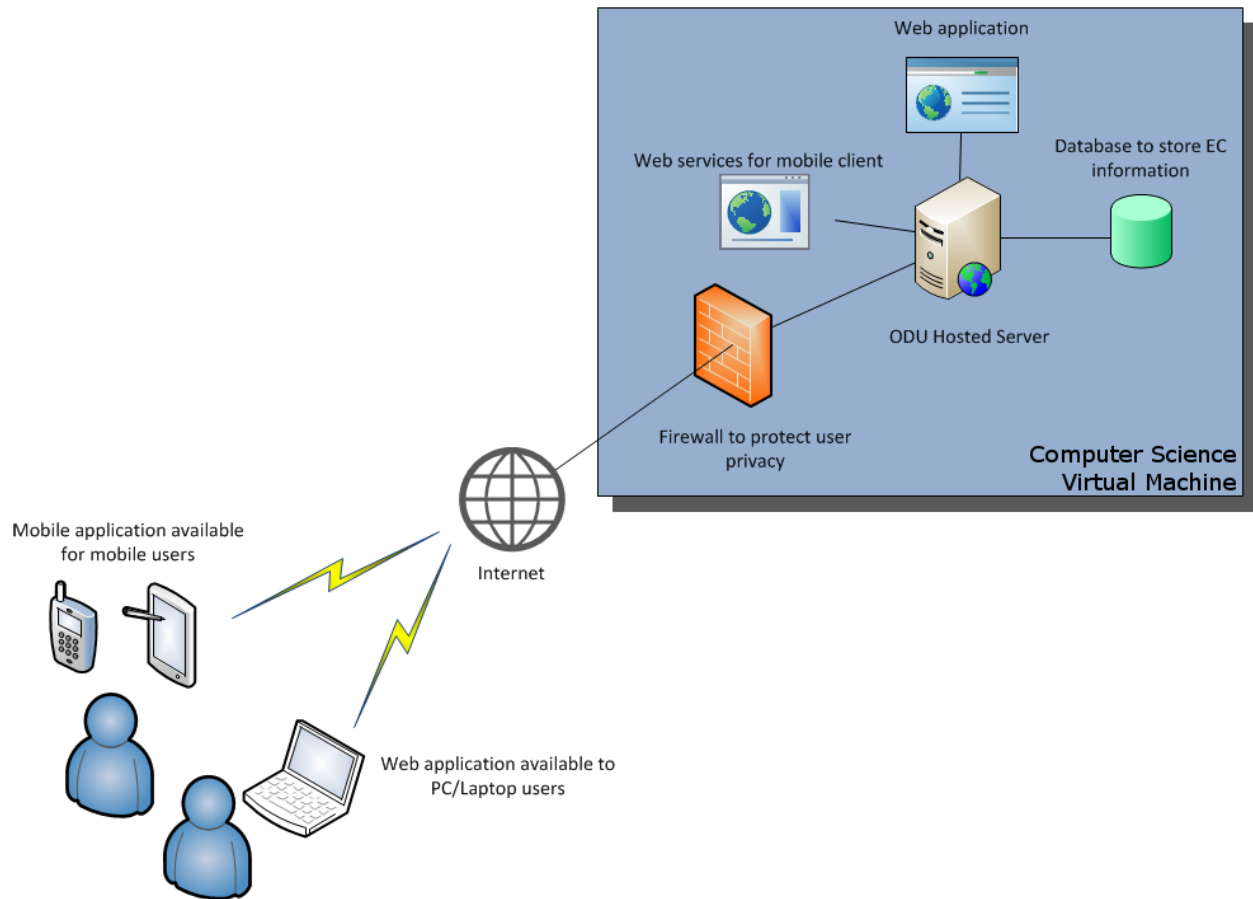
Figure 2. Major Component Diagram

Users will interact with InFusion either through the InFusion website or the downloadable mobile application. All of the user functionality in the website will be available on the mobile platforms, but the mobile apps will also have system notifications so that users can see new notifications without needing to open the app first. The Strome InFusion site-map is shown in figure 3. The site-map shows the major access points and how they relate to each other.

[This section intentionally left blank]
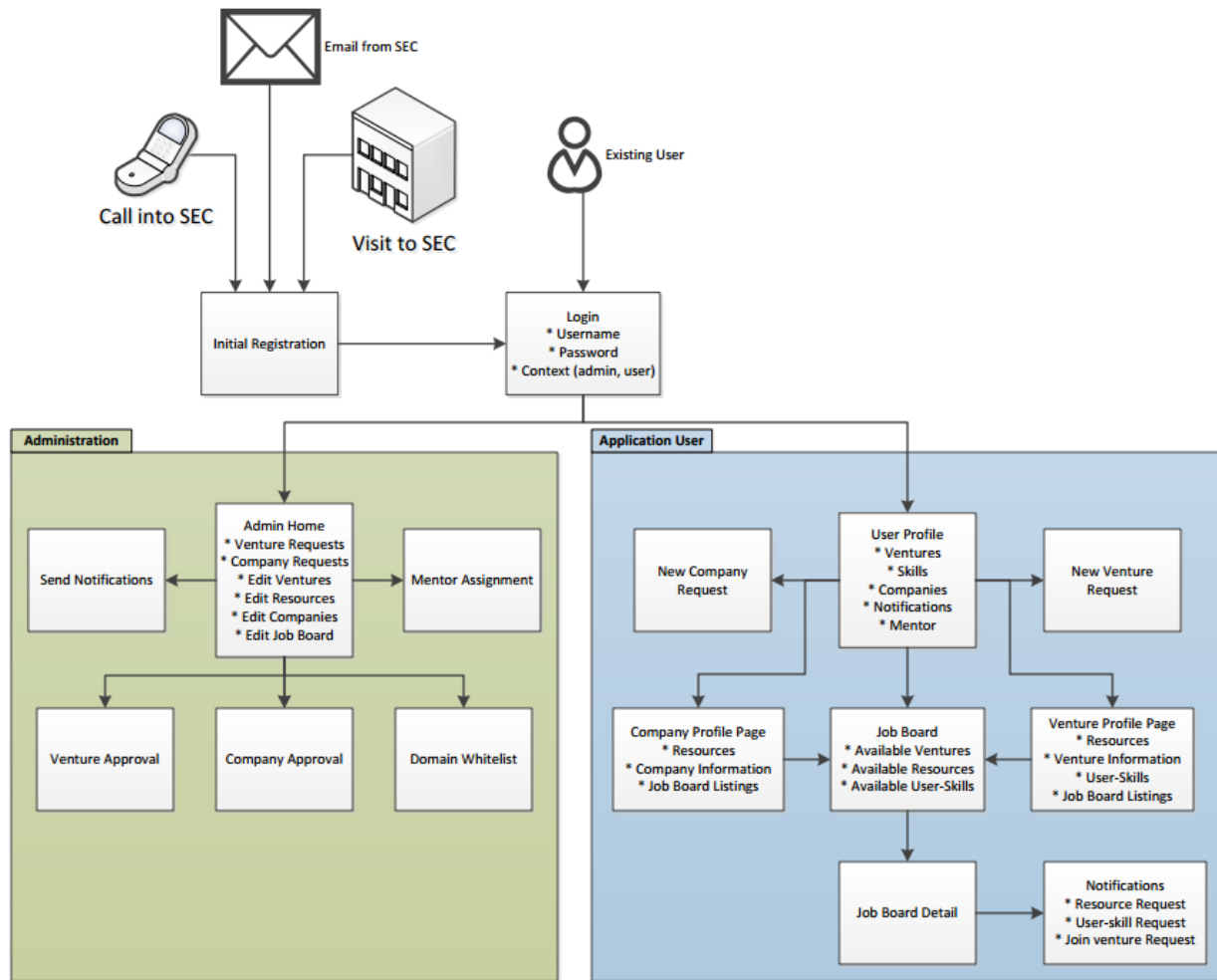
Figure 3. Application Site Map

Some parts of the InFusion application are restricted and must have certain steps completed in order for users to access them. Some of the steps may include filling out request forms or getting requests approved by SEC staff members. Figure 4 shows the main process flow for Strome InFusion.
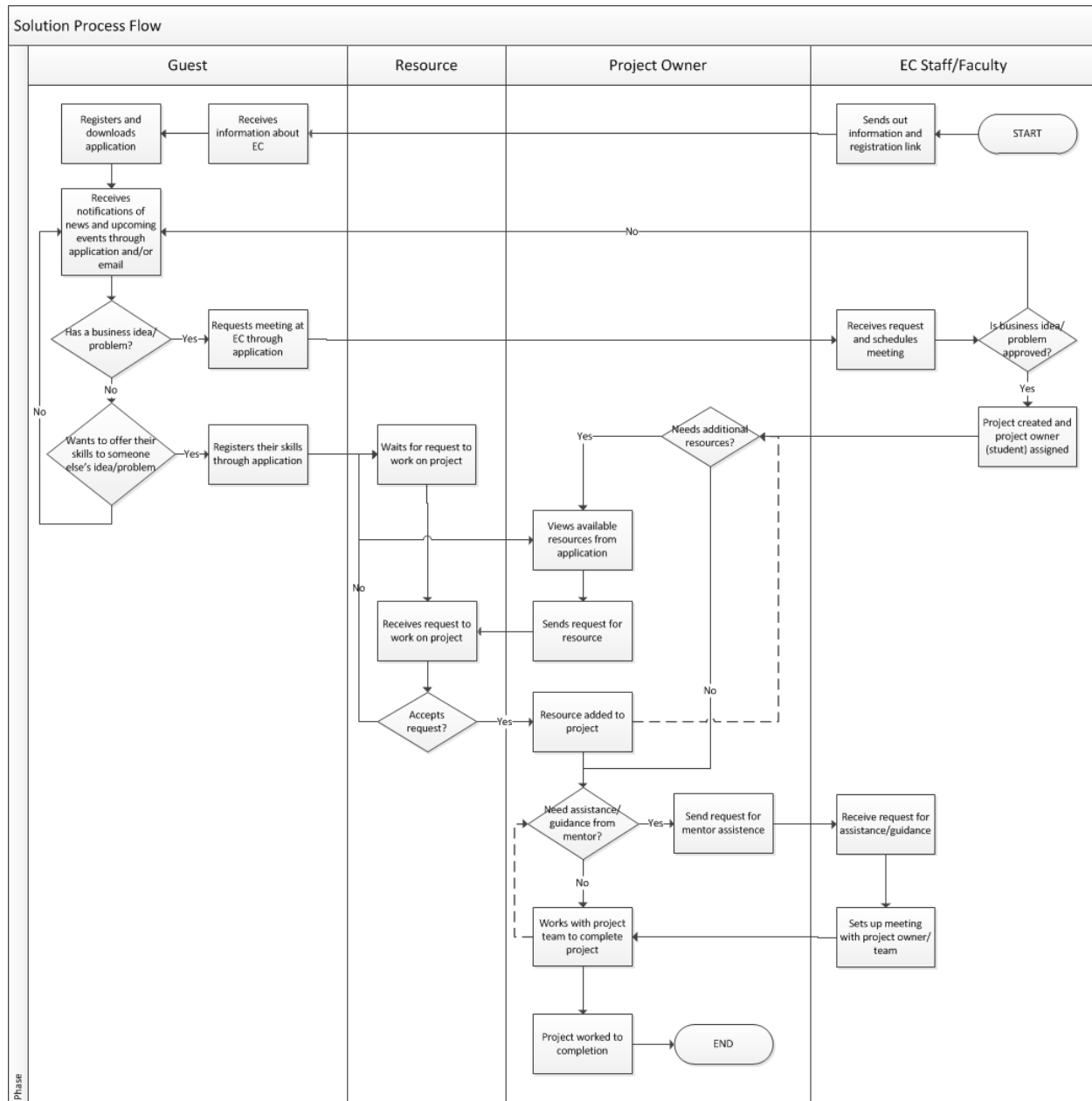
[This section intentionally left blank]

Figure 4. Solution Process Flow

2.2   Prototype Functional Descriptions

The InFusion web and mobile applications will be composed of the following pages: initial landing, registration, login, user profile, venture profile, company profile, job boards, and administration pages.

When a user registers for a new account, they must be approved before they can access the full application. While they are waiting for approval, they can fill out their personal profile page and look at public pages. Users that register with an e-mail address from a white listed domain, @odu.edu for example, the account is automatically approved; otherwise they must be vetted and approved by a member of the SEC staff. Figure 5 shows the account creation process and Figure 6 shows the user approval process.
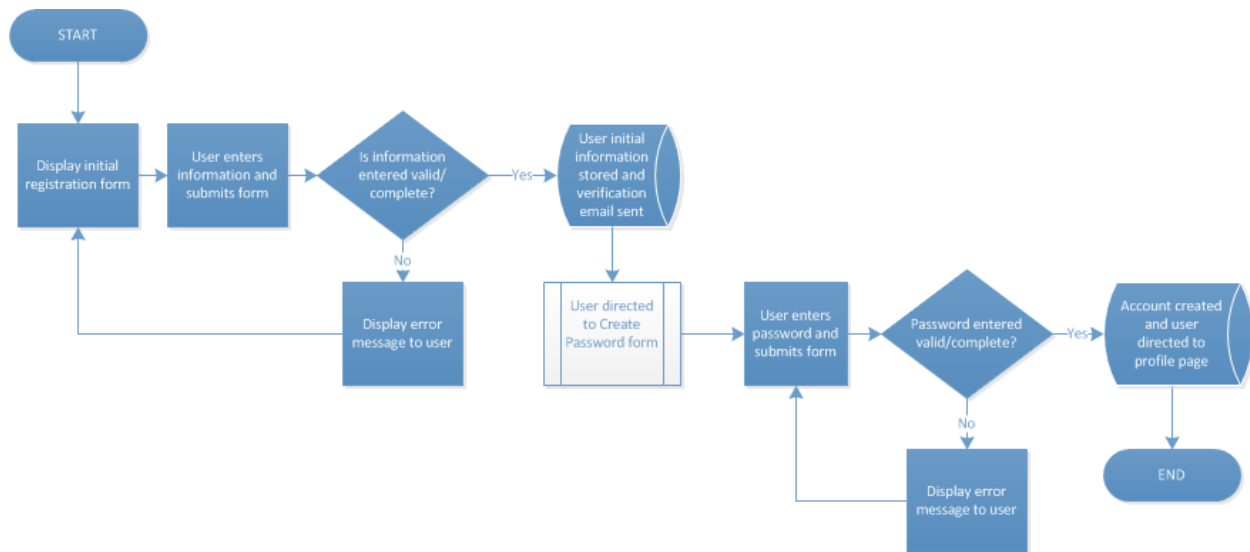

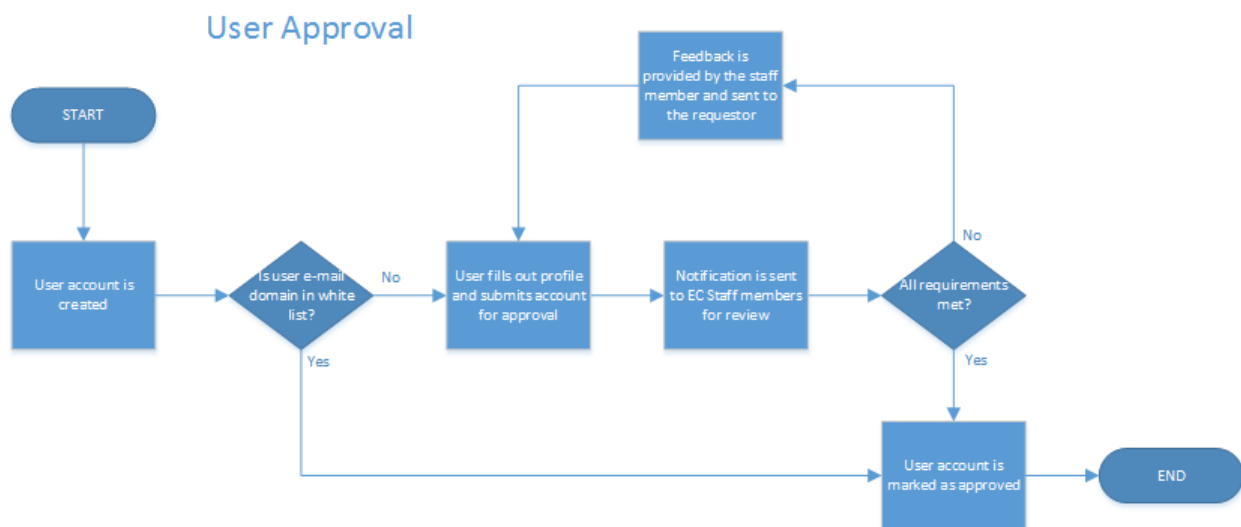*Figure 5*. User account creation process

Figure 6. User account approval process

Many of the pages on InFusion require certain permission in order to access some pages. Some examples of this are private venture pages, private user pages, and administration tools. Authenticated users must be authorized to access these restricted pages. In this case, authentication is identifying who a user is and authorization is checking if that user can access a particular resource. Figure 7 shows the authorization steps that the recourses must process before granting access.
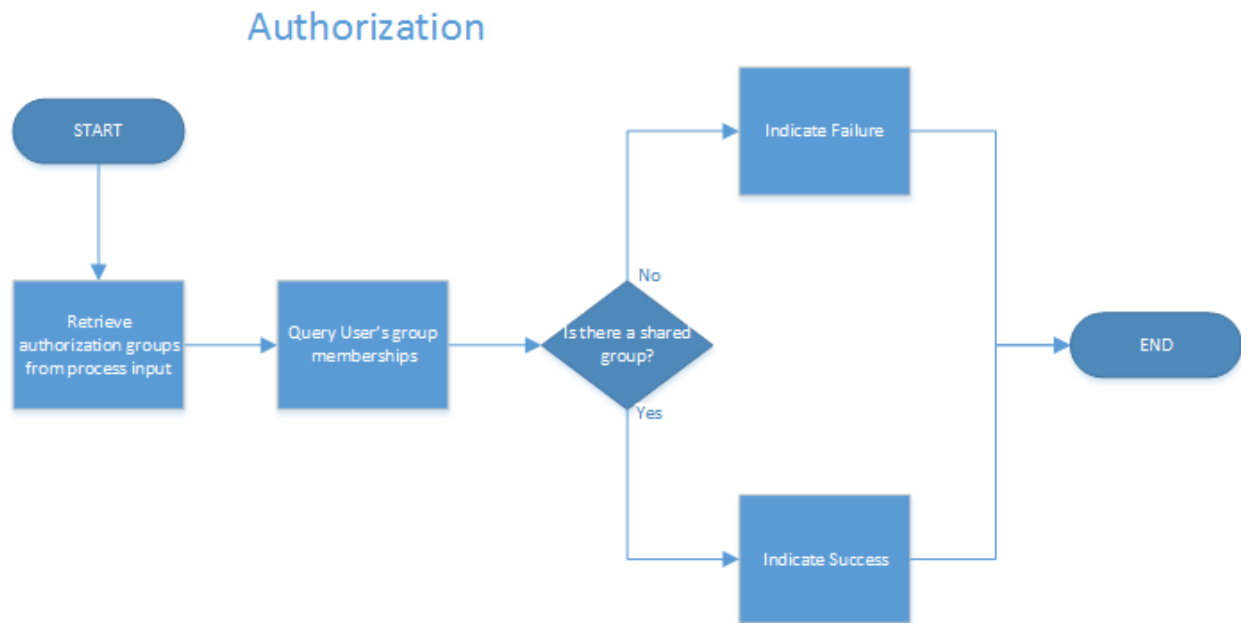

Figure 7. Authorization process

2.3   Extended Interfaces

The user interface for Strome InFusion is a responsive website, which will be dynamic enough to easily be viewed on either a desktop or mobile device. All requests will initially be directed to the Nginx web server. Any requests for static data will be served directly by the web server, while all other requests are routed through uWSGI to the Flask application. Figure 8 shows a mockup of the Strome InFusion landing page on a desktop computer.
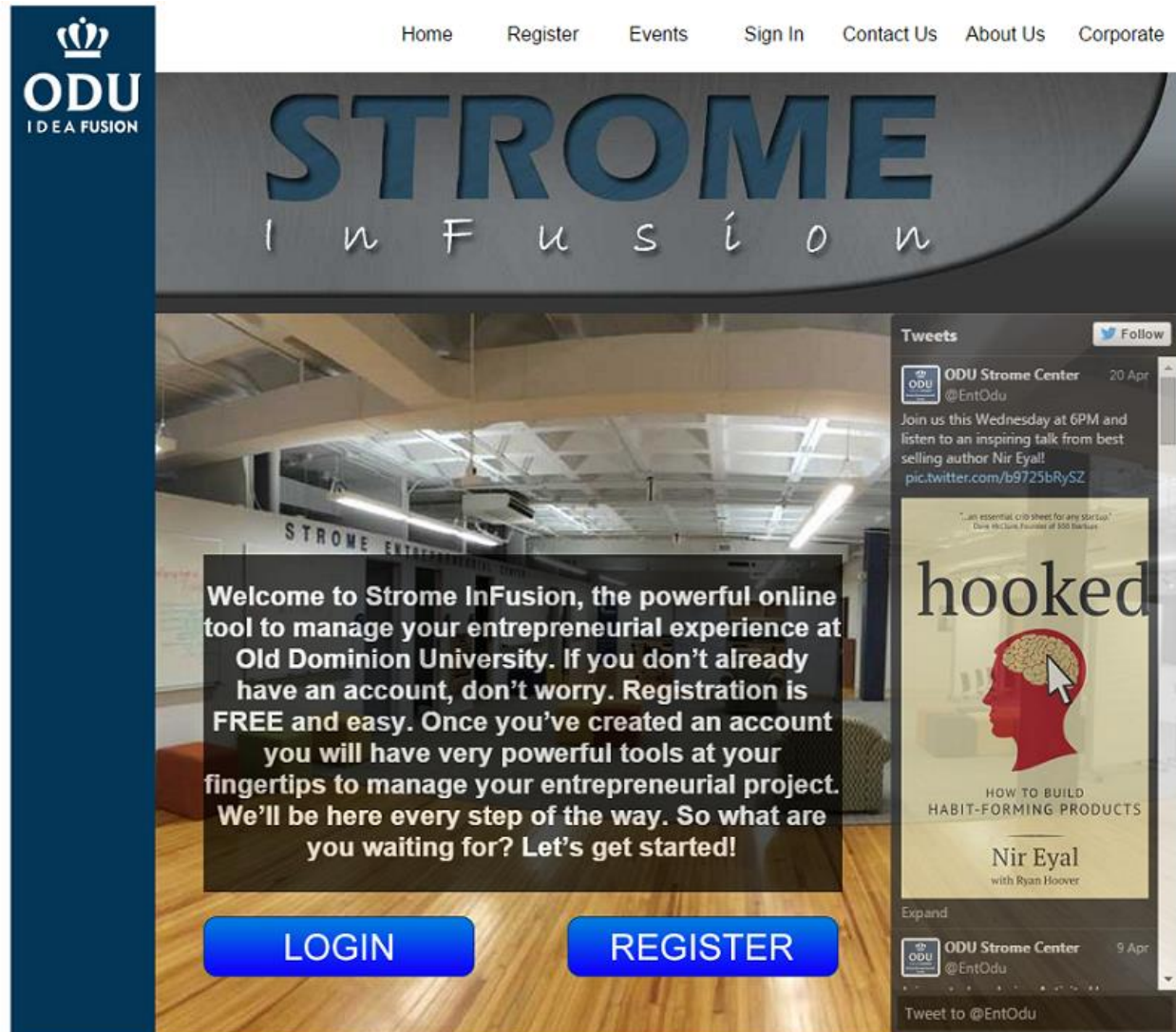
Figure 8. Mockup Landing Page