LinkedOut

Requirements Analysis and System Design

Group 14

Authors

Luke Moss

Mark Madden

Scott Strothmann

Eric Song

Yaoxuan Luan

Version: 1.0

03-20-2016

Table of Contents

Changelog

V0.1   2-25-2016   Requirement analysis rough draft created

V1.0   3-20-2016   Initial requirement analysis document completed

Glossary

Activity Feed - Displays activity from your network, such as comments and profile changes.

Company Page - A page for companies to share information about their business and to make updates.

Connection - An indication that someone has accepted an invitation to connect, allowing the viewing of each other's profiles.

Degrees – How you are connected to others through people or groups.

Follow – A way to see updates in your news feed from people you are interested in.

Group - Allows users to come together and meet on a single page to communicate about a common subject.

Headline – Informative and concise information about your prospects.

Inbox - A place where messages can be sent, received, and viewed.

Introduction - A way for someone to bring two people who are not connected together.

Invitation - A notification to a user to join a network and form a new connection.

Mention - A tag that links a person or company in a post to their corresponding page.

Message - A private form of communication between two connected users.

Network - A group of users that one is connected to and can contact.

Profile - A page for users to share information about themselves, their job history, and experience.

Recommendation - A comment intended to commend someone, based on their experience, to viewers of their profile.

Update - Status changes and posts that appear on activity feeds.

Project Summary

The project's name is LinkedOut and it is a career-based social network designed for the professional business community. The goal of this site is to have users establish connections with other users on a professional level. A user has the option to create a profile page where they can display their professional experience and achievement, such as education and employment history. A user can then connect to other users and share this information with them.

LinkedOut has many features, but one of its main features is a messaging tool that allows users who are connected to communicate with each other privately. The application as a whole can be used in such a way that a user can create a professional brand for themselves. By doing so, a user can pursue their career passion by browsing profiles and connecting with professionals in their field, turning professional relationships into opportunities.

Functional & Non-Functional Requirements

Functional requirements:

- Accounts Create:
  - o Account Type
  - o Profile Edit
- Job board:
  - o Post jobs:
    - ▪ Job Descriptions/Requirements
  - o Display:
    - ▪ Sort by Time/Type
    - ▪ Save Jobs
- Apply:
  - o Send Resume

Non-functional requirements:

- Security:
  - o User Identification
- Notifications:
  - o Email
- Search engine:
  - o University Search
  - o Interests Search
  - o Location Search
- File Review Online:
  - o Resume Review/Edit
  - o Job Descriptions Review/Edit
- Comments:
  - o Company
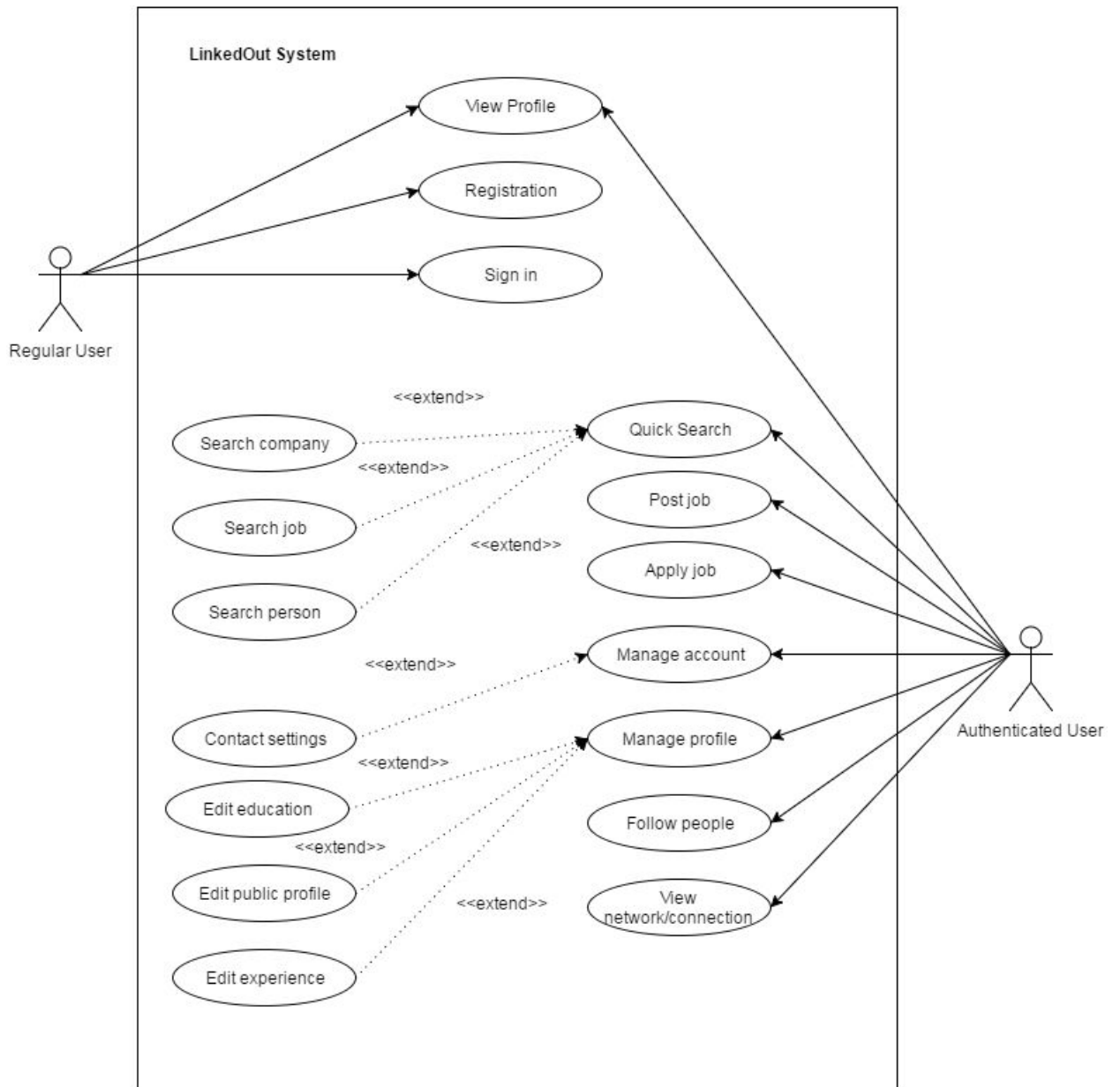 - o Person

User & System Requirements

User Requirements

- The software needs to limit what users can or cannot see based on their being logged in or not and/or linked with someone or not.
    - Use $_SESSION variables to track users' status across application usage.
    - Data - usernames, hashed passwords, links table ("friends").
- The software needs to provide registered users the ability to post resume-like work experience and education information.
    - Work experience table
    - Education table
- The software needs to allow user to search jobs.
    - The job information table
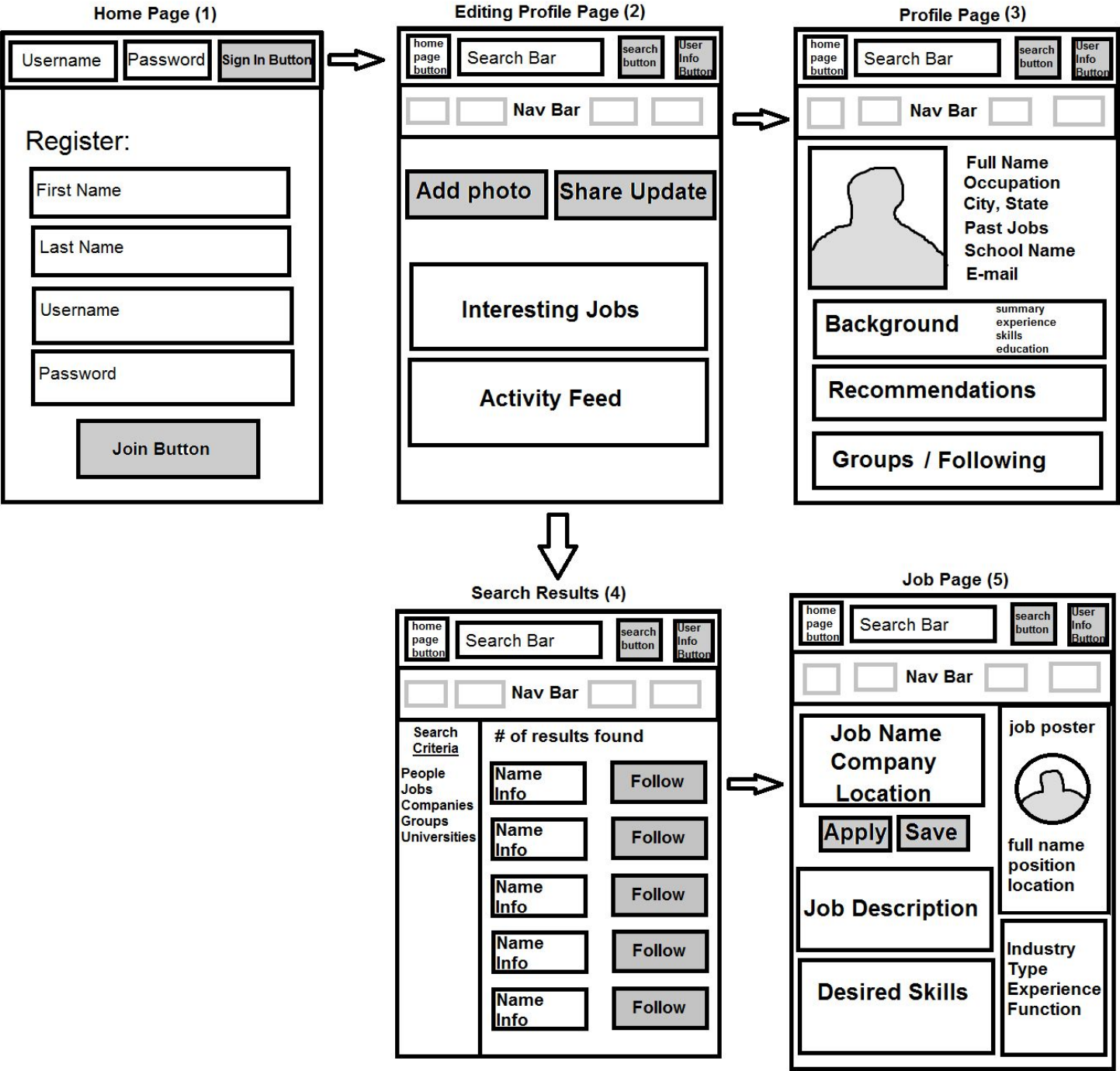- The software needs to allow to use part of functionality without login

System Requirements

- MySQL Database
- Web server with PHP enabled
- Javascript Libraries - jQuery, D3.js
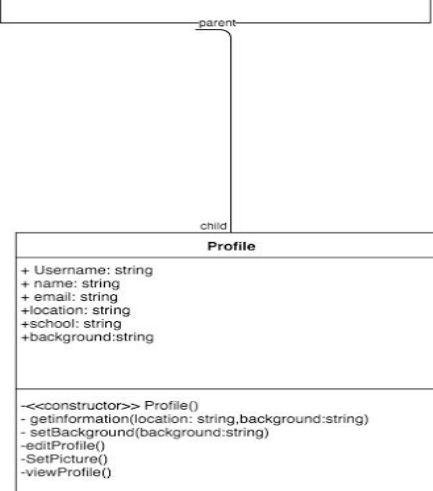- HTML, CSS

Use Case Diagram

Screen Sketches

Home Page (1)

| Username | Password | Sign In Button |

Register:

First Name

Last Name

Username

Password

Join Button

Editing Profile Page (2)

home page button | Search Bar | search button | User Info Button

Nav Bar

Add photo | Share Update

Interesting Jobs

Activity Feed

Profile Page (3)

home page button | Search Bar | search button | User Info Button

Nav Bar

Full Name
Occupation
City, State
Past Jobs
School Name
E-mail

Background — summary experience skills education

Recommendations

Groups / Following

Search Results (4)

home page button | Search Bar | search button | User Info Button

Nav Bar

Search Criteria

People
Jobs
Companies
Groups
Universities

# of results found

Name Info | Follow

Name Info | Follow

Name Info | Follow

Name Info | Follow

Name Info | Follow

Job Page (5)

home page button | Search Bar | search button | User Info Button

Nav Bar

Job Name
Company
Location

Apply | Save

Job Description

Desired Skills

job poster

full name
position
location

Industry
Type
Experience
Function

Software Component (UML Diagram)

Login

+ Username: string
+ password: string
+ email: string

-<<constructor>> Login
- getInfo()
-signIn(username,password)

Search

+ description: string
+ name: string

-<<constructor>> search()
- getresults()
-selectresults()
-search()
-follow()
-viewResults()
-select()
-NarrowResults()

Job

+ description: string
+ email: string
+company: string
+location: string
+jobName:String

- viewJob()
-getInfo()
-Select()

parent

Person

+ Username: string
+ name: string
+ email: string
+location: string
+school: string
+background:string

-<<constructor>> Person()
- getinformation(location: string,background:string)
- setBackground(background:string)
-editProfile()

parent

child

Jobopening

+ description: string
+ email: string
+company: string
+location: string
+jobName:String

-<<constructor>> Jobopening()
- getInfo()
-apply()
-save()
-viewJob()
-select()

child

Profile

+ Username: string
+ name: string
+ email: string
+location: string
+school: string
+background:string

-<<constructor>> Profile()
- getinformation(location: string,background:string)
- setBackground(background:string)
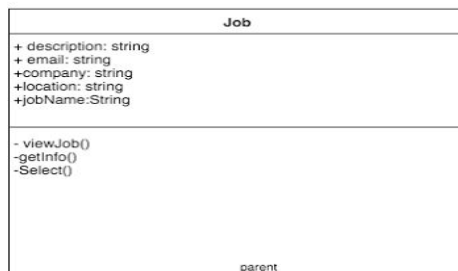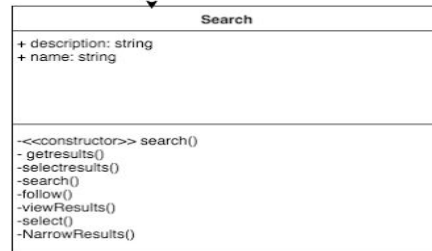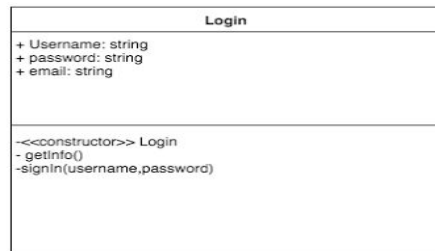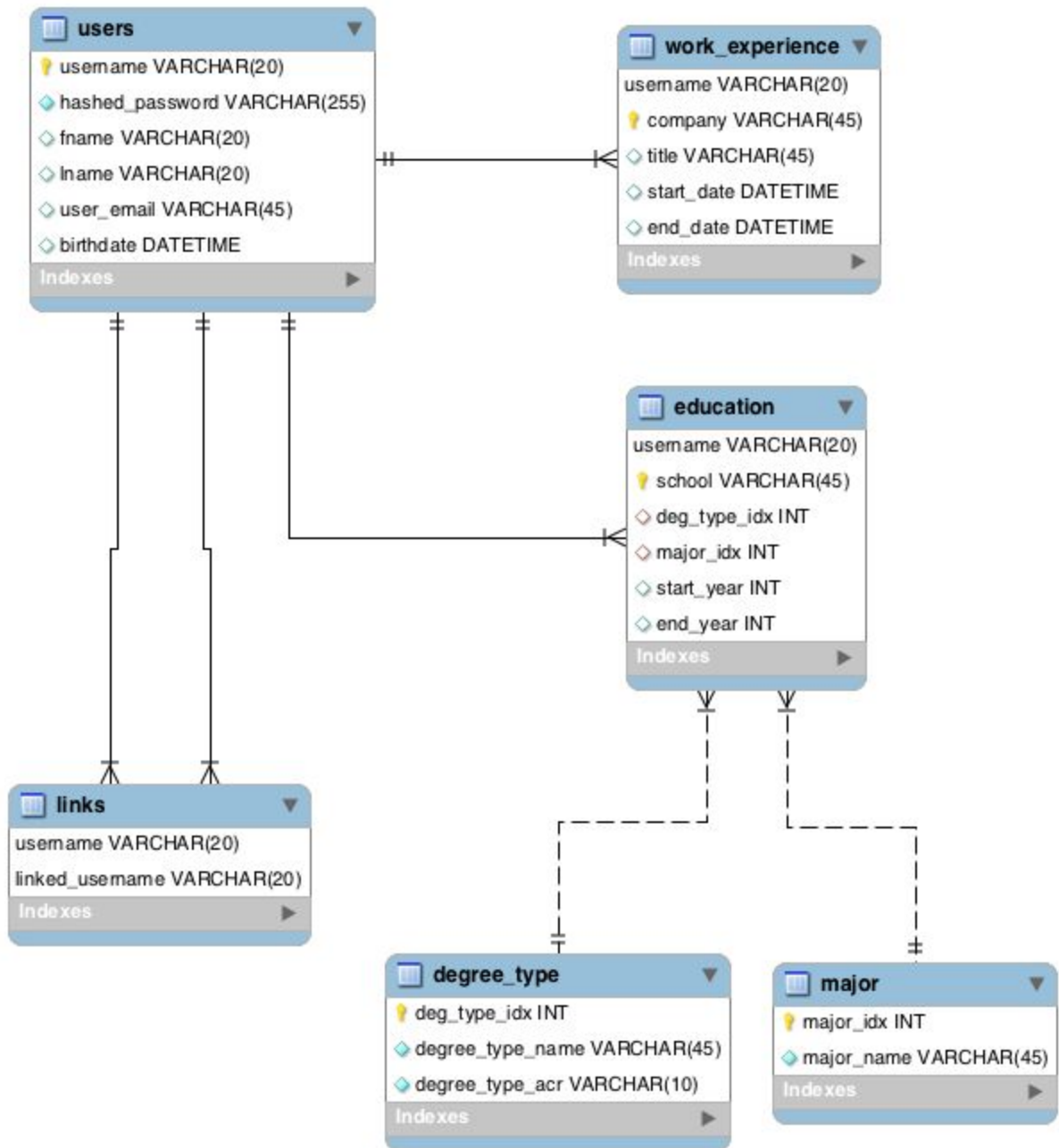-editProfile()
-SetPicture()
-viewProfile()

Table List (ERD)

Links table is a relation table for many to many relationship of users with other users. Lookup can go both ways (i.e. if user 1 is linked to user 2, no need to link user 2 to user 1 as a separate entry in the table). User 1 is linked to everyone where user 1 is the username and where user 1 is the linked_username.