**Software Requirements Specification**

For

Passionate People Project

Version 1.0

Prepared by

Joshua St. Clair

Maya Adkins

David Hung

Caleb Mayerski

Gagik Movsisyan

Team 1: Video and Algorithm

Submitted for CS130 Software Engineering

April 15, 2016

Google Doc Link: (with correct format) <https://docs.google.com/document/d/1mlKfj9KTFZyA2uU9skl33eRNaDAqeSASCe2dtBuGDAw/edit?usp=sharing>

**Table of Contents**



[Introduction](https://docs.google.com/document/d/1mlKfj9KTFZyA2uU9skl33eRNaDAqeSASCe2dtBuGDAw/edit#heading=h.1ztcqc5q0bij)

[Overall Description](https://docs.google.com/document/d/1mlKfj9KTFZyA2uU9skl33eRNaDAqeSASCe2dtBuGDAw/edit#heading=h.z27x4g29wbtq)

[External Interface Requirements](https://docs.google.com/document/d/1mlKfj9KTFZyA2uU9skl33eRNaDAqeSASCe2dtBuGDAw/edit#heading=h.nxol5nuxhpr0)

[System Features](https://docs.google.com/document/d/1mlKfj9KTFZyA2uU9skl33eRNaDAqeSASCe2dtBuGDAw/edit#heading=h.facazojbo7dp)

[Other Nonfunctional Requirements](https://docs.google.com/document/d/1mlKfj9KTFZyA2uU9skl33eRNaDAqeSASCe2dtBuGDAw/edit#heading=h.40ufrhqfo7qh)

[Other Requirements](https://docs.google.com/document/d/1mlKfj9KTFZyA2uU9skl33eRNaDAqeSASCe2dtBuGDAw/edit#heading=h.27zpwcui3k4n)



1. Introduction
   1. Purpose

This software requirements specification (SRS) describes the functionality and components required to implement the Passionate People Project. The project described in this SRS is version 1.0, and only describes the video implementation and job selection algorithm. Other components of the project such as the page layout and user interface are being handled by a separate team and are explained in their SRS.

* 1. Intended Audience

The audience for this SRS is the software development team of this project, as well as the Passionate People Project team who are acting as the client.

* 1. Product Scope

The Purpose of the Passionate People Project is to construct a Webapp that will allow people to browse through the world of works and give people insights on what their careers might interest them. At the core are interview videos of people talking about their jobs and passions, and an algorithm will determine what to present to the user next as people like or dislike the videos. Browse and search features with related information can also help the user find another job based on more narrowly defined parameters such as salary.

The central task of Team 1 is to construct the video playback interface, video filtering, video viewing history, and the algorithm for queuing the next interview video. The video queuing logic is central to helping people find what jobs they are passionate about. Team 2 will be mainly working on the whole website interface, navigation, and database access. Finally, the integration will be a major task at the end of the project.

* 1. Reference

N/A

1. Overall Description
   1. Product Perspective

This software system is a component of a self-contained web application that allows users to learn about different fields of work to find a career they could possibly be passionate about.  Our component will cover the job suggestion algorithm and video functionality. We will be building on top of another team’s front-end development of the product that contains the user interface.

* 1. Product Functions
* The web application will support video playback and video controls.
* The user will have the ability to vote their preference of each video.
* Jobs will be suggested to the user based on the user’s votes and other filtering options.
  1. User Classes and Characteristics

The majority of our users will be people who are still in school at some level; those who have not yet entered full time work. The website is meant to help them find their ideal jobs by their passion.

* 1. Operating Environment (Web)

The software system will work within a web application that is compatible with all modern browsers including Chrome, Firefox, and Safari.

* 1. Design and Implementation Constraints

Open source software needs to be used as a requirement of CS 130. Video database will not be public, but implementation details will be hosted on Github.

* 1. User Documents (i.e. manuals)

No manuals would be needed for this project as it is a self-contained website. An online help section may be construction at the end of production

* 1. Assumptions and Dependencies
* We assume the other team will successfully complete the front-end user interface for the passionate people project.
* We assume the client will be able to provide us with all the relevant data needed for the website like videos and question breakdown by POPs (presentation of passion).
* We assume that the client will provide us with the detail of each career (average salary, skills, career outlook), so we will be able to provide proper filtering capability

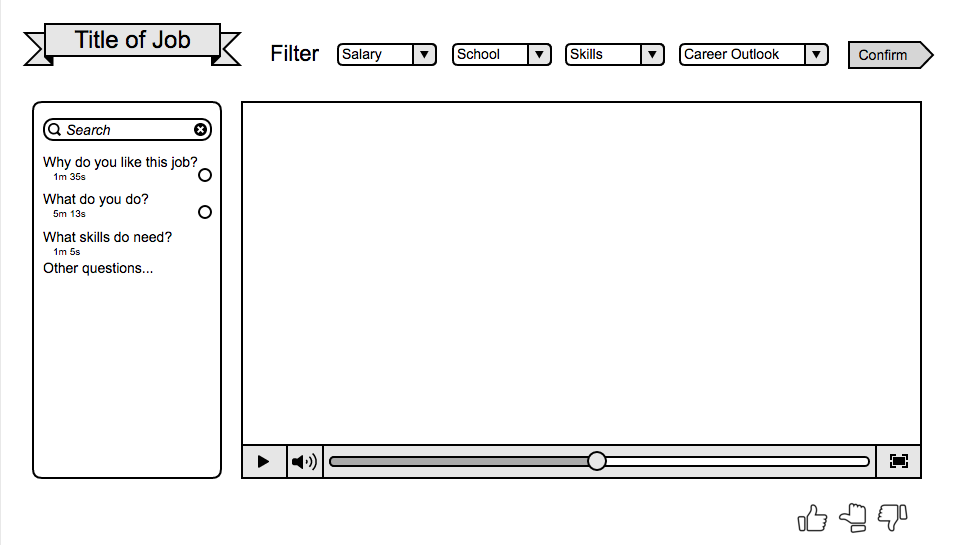
1. External Interface Requirements
   1. User Interfaces (Web Interface)

The video player will have play, pause, scrub, volume, and screen size controls. At the end of each video, it will auto-play to the next question in the list. After all videos have been exhausted, it will ask the user to rate the job before showing a new one.

For each video, there will be three voting options reflecting the user’s opinion of the job: positive, negative, and unsure. These votes will directly affect the results of the job suggestion algorithm.

At the top of the screen, there will be filter parameters the user can specify for future videos suggestions.

On the side will be the list of questions posed in the video to allow the user to select certain questions in the interview.



* 1. Hardware Interfaces

The system is compatible with any device that supports modern browsers.

* 1. Software Interfaces (Database to UI etc)

The website will be able to communicate with the html5 video player. In addition, the website will communicate with the database storing all the videos provided by the client.

* 1. Communications Interface

Services to use are to be determined.

1. System Features
   1. Video Player
      1. Description & Priority

The video player is the principle component where people react to the passions of the people in their specific career and their jobs. It will play the short list of interview videos for all the questions asked for the job chosen. This is high priority as it is the core of the content presentation to the users.

* + 1. Stimulus Response

The video player will have normal control functions as mentioned before. The user will be able to rate the video at any time and after doing so, have the option to skip to the next job video.

* + 1. Functional Requirements

REQ-VID-1: Video will load only the specified file

REQ-VID-2: Warning will be given if player is incompatible with browser, and help instructions will be given

REQ-VID-3: Normal videos controls will work as expected (i.e. play, pause, full-screen, and volume control)

REQ-VID-4: Video scrubbing will work

REQ-VID-5: Video will scale according to website size

REQ-VID-6: Video will wait for a buffer load before playing begins

* 1. User Job Video Tracking
     1. Description & Priority

User’s video viewing history will be tracked for user profiling, to avoid playing repeats, and to get better future suggestion. This is high priority because the job search algorithm is dependent on this data.

* + 1. Stimulus Response

Videos tracking will occur as users view videos.

* + 1. Functional Requirements

REQ-TRK-1: Video will be marked as viewed if shown for more than 20% of duration

REQ-TRK-2: Video tracking history will be persistent for registered users

REQ-TRK-3: Video tracking history will also be temporary for unregistered users via cookie ID

REQ-TRK-4: Video tracking data will be accessible for the suggestion algorithm

* 1. Job Suggestion Algorithm
     1. Description & Priority

This algorithm will pick the next video to show to the user based on viewing history, user response to previous videos (positive, negative, unsure), and filtering parameters. As the user watches and responds to more and more videos, the algorithm should eventually hone in on specific areas of the world of work that correspond to the user’s interests. This is also high priority as it is crucial for finding user career preferences.

* + 1. Stimulus Response

The user will watch and respond to videos suggested by the algorithm. They can also specify filtering parameters to limit future videos suggested by the algorithm.

* + 1. Functional Requirements

REQ-ALG-1: Algorithm will take into account previous “thumbs up” and “thumbs down” history to suggest a new job.

REQ-ALG-2: The algorithm will narrow the search field of all potential jobs according to the SOC codes that each job is tied to.

REQ-ALG-3: The algorithm will allow different zones of potential jobs in the World of Work, in the case that a person has multiple interests that are very different.

REQ-ALG-4: The algorithm will not show videos outside the specified filter parameters. If there are no more videos that meet the filter, a message should display informing the user and requesting they change the filter settings if they wish to see more videos.

1. Other Nonfunctional Requirements
   1. Performance Requirements

REQ-PRF-1: The video will buffer within 10 seconds on all computers using reliable 2MBPS broadband.

REQ-PRF-2: Choosing between clips using the playlist will be completed within two seconds

REQ-PRF-3: The suggestion algorithm will queue next job video within 10 seconds.

REQ-PRF-4: Website will be able to handle 10,000 users connections at the same time. (This number should scale up on demand)

* 1. Safety Requirements

N/A

* 1. Security Requirements

REQ-SCT-1: User view history will not be viewable by unauthorized users

REQ-SCT-1: All videos will be securely stored on a database, and played on a private video server.

REQ-SCT-2: Video database cannot be directly accessed outside of the website.

* 1. Software Quality Attributes

REQ-QLT-1: The database admin will be able to upload additional interview videos in the future.

* 1. Business Rules

N/A

1. Other Requirements
   1. Appendix: Glossary

|  |  |
| --- | --- |
| **Word** | **Equivalent to** |
| Job Suggestion Algorithm | Video Queuing Algorithm |
| PoP | Presentation of Passion |
| WoW | World of Work |

* 1. Timeline
* Week 4: Video Player
  + Video player interface with usual button features and playlists
  + Videos can be accessed
* Week 5: Page interface
  + Page interface surrounding the video player.
  + Display Information
  + Filter features
* Week 6-7: Algorithm
  + Algorithm for suggestions
  + 3 suggestions
  + 5 suggestions (depending on speed of vote response)
* Week 8-9: Integration
  + Integration with Team 2
  + Experimental interface