

Portfolios for Programmers

Anuhya Ghorakavi, Isabella Tromba, Patricia Saylor, Ryan Cheu

Stakeholders (Anuhya)

Site Administrators (Our Team) With all four of our members being computer science majors and individually invested in furthering ourselves within the computer science and tech industry, this product is very important to all of us. We, ourselves, find a gap in being able to accurately and efficiently demonstrate our talents to potential companies and recruiters. Therefore in planning and talking through our various project ideas, we settled on something that would not only benefit us and our careers but also hundreds of thousands of new developers and programmers whose talents are weeded out due to the inability of recruiters to read through every single padded resume their companies receive.

6.170 Staff This project can also be of interest to the 6.170 staff as a whole because it demonstrates what elements are required in a project assignment to make it presentable, applicable, and demonstrative of a student's skillset in the future, outside the scope of 6.170. This could help the staff mold what is highlighted and focused on in future projects or in modifying existing project assignments.

Programmers Programmers will be our main target for the primary users of this application. We would like programmers of all backgrounds to be able to display their coding skills through an easy-to-use interface that will allow them to showcase their projects through the form of text, images, videos, and code snippets. The programmers, upon creation of their profile, can send the link of their profile to recruiters of companies in order to better display their talents and previous work. The user's profile will highlight tangible projects and eliminate the fluff that recruiters see regularly on programmers' resumes.

Recruiters As mentioned above, upon receiving a link to a user's profile, a recruiter can navigate to a link on our site and browse through the projects of the given user. The recruiter has the ability to search a given user's profile by language in the case that they are looking to hire a candidate with strong background in a particular language and would like to view the current candidate's experience in that language.

Resources (Anuhya)

Cost Constraints:

- Storage of data on S3 servers (images, videos, user profiles)
- Font subscription
- Hosting on an external domain if the traffic becomes too much for Heroku

Computational Constraints:

- Efficiently storing data on S3 servers
- Quickly accessing and rendering data stored on the S3 servers
- Allowing for simple and easy UI customization

Time Constraints:

- Creating a clean and usable product in three weeks

Tasks (Isabella)

List of Tasks

Phase 1

- User Authentication—Sign Up, Sign In, Sign Out
- Single Template for project formatting - single color theme/font
- Two 'Block templates' - (resource on left and right)
- Code formatting with javascript
- Amazon S3 for Image uploading
- Home page image/layout/about us

Phase 2

- Font customization
- Video resource uploading/formatting
- Email verification of user accounts
- Ability to change ordering of blocks on a page
- More project templates - additional color themes
- Drag and drop image uploading
- Four 'Block templates' (resource on left, right, top, bottom)

Milestone Calendar (Isabella)

Thu Nov 14	All day	📅 Anuhya: Basic Data Models/Views/Controllers created - 5 hours
	All day	📅 Anuhya: User Authentication - Sign in/Sign out - 2 hours
Sat Nov 16	All day	📅 Patricia: Template for Project Formatting - (one color theme and font) - 3 hours
	All day	📅 Isabella: Home page layout - Image, About Us, Layout - 6 hours
	All day	📅 Ryan: Image uploading through Amazon S3 - 5 hours
Sun Nov 17	All day	📅 Patricia: Template for Blocks - Resource on left and right - 5 hours
	All day	📅 Isabella: Javascript Code formatting - 2 hour
	All day	📅 MVP Implementation
	All day	📅 Ryan: UI design on projects and portfolio page - 3 hours
Mon Nov 18	All day	📅 MVP Demo
Wed Nov 20	All day	📅 Design Critique
Sat Nov 23	All day	📅 Ryan: Video Uploading/ Embedding UI and storage - 4 hours
	All day	📅 Patricia: Additional Project Templates with more color theme options - 4 hours
	All day	📅 Isabella: Email verification of user accounts - 4 hours
	All day	📅 Anuhya: Drag and Drop Image Uploading - 4 hours
Sun Nov 24	All day	📅 Revised Design
Sat Nov 30	All day	📅 Patricia: Font Customization Implemented (multiple font options) - 5 hours
	All day	📅 Ryan: Additional Color themes for Project Templates / Final UI for Project Templates - 4 hours
	All day	📅 Anuhya: Ability to change ordering of blocks on a page / finalize formatting - 5 hours
	All day	📅 Isabella: Additional block templates with resource on top and bottom allowed - 4 hours
Mon Dec 2	All day	📅 Code Critique
Sat Dec 7	All day	📅 Isabella: Final UI Touchups with Anuhya - 3 hours
	All day	📅 Anuhya: Final UI touchups on all pages - 6 hours
	All day	📅 Patricia: All security concerns addressed - 5 hours
	All day	📅 Ryan: Implement search projects by programming language on portfolio page - 5 hours
	All day	📅 Isabella: Allow users to specify programming language of project - 3 hours
Sun Dec 8	All day	📅 Final Version of Code
Mon Dec 9	All day	📅 Final Demo

Risks (Patricia)

Misattribution/Code Plagiarism

Users may include projects or code in their portfolio that they did not actually work on in an attempt to steal credit for someone else's work. For example, they might try to include code snippets from an open source or otherwise public project and claim it as their own when they've never worked on it, or claim credit for a part of a group project that they did not work on. Most other self-reporting systems have this issue as well, such as resumes and LinkedIn.

Mitigation: Since the user can paste any code into their code snippets, there isn't really a good way to prevent them from plagiarizing. Plagiarism is a risk almost anywhere, and it is a problem we do not think we will have time to tackle in three weeks. Instead, we will mitigate this risk by encouraging recruiters and interviewers to take the self-reported information presented on Portfolio with a grain of salt as they would any personal website, and to ensure they ask pertinent questions about the projects presented therein during the recruiting/interview process. The risk of getting caught by a potential employer should dissuade users from plagiarizing code on our site.

Distraction

Recruiters or interviewers linked to someone's portfolio might get distracted by another person's portfolio while on our site, which means the intended portfolio is not getting the attention it deserves and the user is not being showcased.

Mitigation: We will not show any links to another candidate's portfolio nor to a collection of other portfolios on portfolio pages. Links such as these may be available when on the site's home page, however.

Information Overload

Users may want to put every one of their projects on our site, regardless of their current relevance, or put too much information on a project page (e.g, 100 pictures). This information overload may bloat their portfolio to the point that it's not useful. However, we may find another user with the same number of projects or the same amount of content for a project where it is all relevant and useful. Even when each piece is individually useful, the sheer quantity of information presented may devalue it in the eyes of a third-party viewer.

Mitigation: We will impose reasonable limits on the number of projects in a portfolio, and the number of pictures, videos, and/or code snippets in a project. Though the user may have 100 relevant pictures, videos, or snippets, this will force them to select a small subset of the most interesting, relevant, or salient, which will hope will improve the readability and quality of the portfolio.

Minimal Viable Product (Anuhya)

Identification of Minimum Viable Product

Though we aim to have the primary data models and their respective associations fully established for our MVP; for this phase, we will be focusing only on the successful implementation of a subset of our final list of features and customizations.

We will begin by creating a *User* and establishing authorization and access control methods for the user. For the MVP, each user will be able to create his/her own *Portfolio* and populate this portfolio with a series of *Projects*.

Upon the creation of a project (or when editing an existing project) the user will have the ability to add multiple *Resources* to this given project through the use of *Blocks*. The user can choose to either paste code or to upload an image into a block. This resource can then be augmented with a description within the block and customized through the use of two different *Templates*. The image data will be stored on Amazon S3 through its API integration with our application and the code will be saved in our database as plain text and displayed within a block as formatted code.

After the user adds all of his/her desired blocks to the project page, the user can choose a color *Theme* to customize the project with. The user can also apply this color theme to his/her main portfolio page. For the purposes of the MVP, we will only be allowing one color theme and two templates for customization of each block, but in the future this will expand to a broader range of choices.

We believe that our MVP will provide the users with all the basic functionality necessary to begin showcasing their talents and skillset in a tangible way. Programmers with minimal web development experience will be able to easily share their previous projects more efficiently even with our MVP. For our final product we aim to have a website that is easy to use, more customizable, and representative of a candidate and his/her strengths, previous experiences, and future interests. We, as creators of the website, are very open to user feedback as this site is for users to showcase who they are and what they want. And we will be looking to use all feedback from the MVP to enhance and better our final product.

Subset of Features

User:

- User can sign in/ sign out
- User can create a portfolio
- User can look through other users' portfolios

Portfolio:

- User can apply a single provided color theme to his/her portfolio
- User can create one or more new projects and add them to his/her portfolio

- User can view and edit existing projects in his/her portfolio

Project:

- User can add a resource (image or code block) along with a description of the resource to a block and add one or more blocks to the project.
- If the resource is an image it will be stored on Amazon S3
- If the resource is a block of code, we will store the code as text in the database and format the code within the block before displaying
- The user can customize the block through the use of two different templates:
 - resource on the left, description on the right
 - resource on the right, description on the left
- User can choose to apply a single provided color theme to his/her project

Issues postponed

User:

- User will have to authenticate his/her account through email verification
- User can send out a view-only link to his/her portfolio to a recruiter

Portfolio:

- User can choose between multiple color themes
- User can customize the font within his/her portfolio and projects
- User can search his/her own portfolio based on language to return a list of projects corresponding to that given programming language
 - This will be useful for when a user sends out a link to a recruiter and the recruiter wants to filter out which projects in the candidate's profile best match their programming language requirements

Project:

- User can specify a certain programming language corresponding to each project
- User can add a video resource in addition to an image or code block
- User can change the ordering of existing blocks on a given project page
- User can choose between multiple color themes and fonts
- User can choose between an additional two templates:
 - resource on top, description on bottom
 - resource on bottom, description on top
- User can drag and drop images to upload