Principles of Software Engineering Fall 2021
FAU CEN 4010

OnRamp

Group: FA21-G17

Carlos Fonseca	cfonseca2018@fau.edu
Ignacio Albornoz	ialbornoz2016@fau.edu
Thiago Goncalves Vasconcelos	goncalvesva2020@fau.edu
David Moore	davidmoore2014@fau.edu
Laura Casals	lcasals2014@fau.edu

Milestone 1 Project Proposal and High-level Description
9/28/2021

Table of Contents

Table of Contents **1**

Executive Summary **3**

Use Cases **5**

- Search 5
- Create an account 5
- User login 6
- Create a Journey 8
- Create a Path 8
- Create an Experience 9

Definitions **10**

Initial List of High-Level Functional Requirements **12**

- Non-Member expectation 12
- Member expectation 12

List of Non-functional requirements **15**

- Performance Requirements 15
- Ease of Use 16
- Interoperability Requirements 16
- Expected Load 16
- Security Requirements 17
- Portability Requirements 17
- Supportability Requirements 18
- Storage Requirements 18
- Survivability 18
- Availability Requirements 18
- Fault Tolerance 19

High-Level System Architecture **20**

- Frontend Technologies 20
- Backend Technologies 20
- Other Technologies 21
- Browser Compatibility 21

Competitive Analysis **22**

- Expectations 22
- Competitors 23
  - OnRamp (4.50) 23
  - LinkedIn (3.33) 24
  - Handshake (3.67) 24

Indeed (1.83) 24
Sumry (3.67) 24
AngelList (2.00) 25
Planned Advantages 25
Team Roles **26**
Team Lead, Front End Team Lead/Developer: 26
Github Master, Front End Developer: 26
Scrum Master, Back End Developer: 26
Back End Team Lead/Developer: 26
Back End Developer: 26

Executive Summary

There is a gap between education credentials and job attainment, among other modern concerns for at least three groups of learners pursuing careers in technology; college students (undergraduate and graduate), primarily self-taught students, and post-college career changers. CITDL aims to help people build their unique paths that will bridge the gap. To accomplish this, we want to research and develop a platform that facilitates the answers to at least three implicit questions of modern learners. First, what is the most immediate thing to creatively fulfill the experience gap besides internships? Second, what essential knowledge domains and domain skills do I want to master in the technology field? How can I put together and translate the knowledge, skills, and experience into an overall credible background of meaningful experiences while not, for example, yet job-ready and between internships (or before college graduation)? Everybody now knows that going to college and getting a degree is not enough. Experience and proof of expertise now compete with mere credentials.

Our platform is OnRamp. OnRamp is for every learner that provides a workspace that we call a Journey[1] While goal orientation is good, there is presently an overemphasis on attaining our goals that crowd out the creative process for establishing our learning path. For example, by graduation year, a pivotal question is left unanswered: "Why did I learn all of this and for what?"

Much of what we need to know for a job is available to consume, get the dream job, and earn money quickly and ideally. There is nothing wrong with this approach. However, there is so much that we need to know that it is challenging to engage in significant and creative activities as a byproduct of our fast learning process. And how often do leading learning institutions challenge learners to slow down long enough to think about their thinking and learning as they create? Meaning and purpose has been consigned to the backseat, figuratively speaking, if not also sacrificed at the altar of prestige and credentialing.

OnRamp goes against the grain by providing a place not simply to build a profile of posted credentials and a friends list of references, but to provide a means to formulate job-relevant vision and action. The Triple Graph is like a timeline that runs linearly and in the online space (think of a workspace or canvas) called a Journey. Build a Path within set time constraints between each point of experience. Or the user can

[1] Metaspace is used uniquely here with the knowledge that the "metaverse" is the future of the internet for social networking platforms. In a similar sense, OnRamp will embrace what we call Source-knowledge Agnosticism. If you learned something from a reliable source and built something valuable with your knowledge, why not hire you?

build a Path with no time constraints at all. A user can also do a combination of the two. Time or no time constraints, the user can create an Experience to place on a Path on their Triple Graph. An Experience is like a point on a timeline. While Paths naturally meander, they are still rooted in time and place. The Experience is composed of three sub-components: Knowledge, Creation, and Documentation. The user can build a record of knowledge learned (domain knowledge). Then they are encouraged to apply and present their knowledge in a completed and purpose-driven project (creation). Finally, the user is encouraged to reflect, modify, and package their work in narrative use-case form (documentation). Within each subcomponent are experience guidelines in the form of questions. Before: Why do I want to learn a subject? What are the ways I want to learn it?; During: What am I getting from this learning experience? Would I like to learn more through to completion? Do I want to develop my project solo or on a team? Who can/should use what I/we create?; After: What do I know now that I did not know before? How can I apply what I learned? What can I do now that I could not do before? What do I appreciate or want to improve in my project?

Experiences allow media-rich integrations for the learning sources (college courses, YouTube, Coursera, Free Code Camp, etc.) and import, and a place to present the completed project. As the user completes an experience and adds more to their Triple Graph, OnRamp can predictively create a graphical presentation. OnRamp will show the user possible areas of interest within the technology field and graphically draw a connection from key functional and non-functional aspects of their completed project to real-world requirements in job descriptions without having to leave the OnRamp application. From there the user can create a visual data-driven path from one completed Experience to a newly created Experience.

Use Cases

Search

1. **Description:**
   The user can search for public-level user accounts and their Experiences.[2]

2. **Actors**:
   2.1. User
   2.2. System

3. **Preconditions:**
   3.1. The user has internet access.
   3.2. The website is accessible.

4. **Primary Flow of Events:**
   4.1. The user goes to the website.
   4.2. The user enters search criteria.
   4.3. The system will display relevant information.
   4.4. The user can sort only public search results by username, Experience, Path, and Experience-Career relationship.

5. **Alternate Flow: Follower-Only Search**
   5.1. The user must initiate **Login**.
   5.2. The user enters search criteria.
   5.3. The system will display relevant information.
   5.4. The user can sort public and follower-only search results by username, Experience, Path, and site-wide Triple Graph data based on said search results.

Create an account

1. **Description:**
   The user will input an email address and unique password to create a new account and secure it.

[2] See the definitions table for “Experience.”

2. **Actors**:
    2.1. User
    2.2. System

3. **Preconditions:**
    3.1. The user has internet access.
    3.2. The website is accessible.

4. **Primary Flow of Events:**
    4.1. The user will input a valid email address.
    4.2. The user will input a unique password. The password must satisfy current best practices for password length and complexity to pass system input validation.
    4.3. The system highlights the data fields in green for valid input.
    4.4. The user clicks on submit.
    4.5. The system will present notice that the user must verify their new account through an automated email.
    4.6. The user clicks the confirmation button in the email and is taken to the website to log in.

5. **Alternate Flow: Invalid Email Address or Password Format**
    5.1. Start at Step 4.2.
    5.2. The user inputs an invalid email address or password.
    5.3. The system highlights the data fields in red containing invalid input and a message to correct input errors. In addition, the system mutes the submit button until the user input is valid.
    5.4. The user inputs correct information.
    5.5. Continue at Step 4.3.

User login

1. **Description:**
    The user will use their email address and password to access their account.

2. **Actors**:
    2.1. User

2.2. System

3. **Preconditions:**

3.1. The user has internet access.
3.2. The website is accessible.

4. **Primary Flow of Events:**

4.1. The user will go to the web website and click the login button.
4.2. The system will take the user to the login page.
4.3. The user will input their correct email address and password.
4.4. The system will highlight the data fields green for proper input formatting.
4.5. The system will confirm the user input.
4.6. Finally, the system will take the user to their account.

5. **Alternate Flow: Invalid Email Address or Password Format**

5.1. The system initiates **Alternate Flow: Invalid Email Address or Password Format.**

6. Alternate Flow: Invalid Login Combination (Email Address and Password)

6.1. The system will prompt the user that the email address and password combination are incorrect.
6.2. The system will also prompt the user to
6.2.1. Correct their input and try again;
6.2.2. Or to click *Reset Password*.
6.3. The user enters the correct email/password combination.
6.4. Continue with Step 4.5

7. Alternate Flow: Reset Password

7.1. The system will take the user to the *Reset Password* webpage.
7.2. The system will prompt the user to enter their email address and to click submit.
7.3. The system will email the user with a unique one-time link to reset the password.
7.4. The user will click the link in the email.
7.5. The system will take the user to a webpage to enter a new password.
7.6. The user will enter a new password twice in two separate data fields.
7.7. The system will highlight the data fields in green for proper password length and complexity or highlight them in red otherwise. (Similar to Steps 5.2-5.4.)

7.8. The system confirms valid input and sends a confirmation email to the user for the change.

Create a Journey

1. **Description:**
   The user will create a workspace called a Journey.

2. **Actors**:
   2.1. User
   2.2. System

3. **Preconditions:**
   3.1. The user has internet access.
   3.2. The website is accessible.

4. **Primary Flow of Events:**
   4.1. The user can click the add button to add a Journey to the list in their account.

Create a Path

1. **Description:**
   The user will add a Path.

2. **Actors**:
   2.1. User
   2.2. System

3. **Preconditions:**
   3.1. The user has internet access.
   3.2. The website is accessible.

4. **Primary Flow of Events:**
   4.1. The user must **Create a Journey.**
   4.2. The user can click add to place a path into the Journey.

Create an Experience

1. **Description:**
   The user will start an experience.

2. **Actors**:
   2.1. User
   2.2. System

3. **Preconditions:**
   3.1. The user has internet access.
   3.2. The website is accessible.

4. **Primary Flow of Events:**
   4.1. The user must **Create a Journey.**
   4.2. The user must **Create a Path.**
   4.3. The user can click add to place an Experience onto a Path.

Definitions

Name	Meaning	Usage	Description
Path	Data	Activity Type	A path consists of user input (descriptions and keywords), leading to other potential areas of interest in the technology field.
Journey	Data	Activity Type	The workspace the user can open creates a path.
OnRamp	Domain Name	use case scenario	Name of our website. Describes the user's journey. one possible name of our application.
Accounts	Data	Activity Type	Stores all of the member's information.
Search	Service	Member Service	Members can find other members.
Login	Service	Site User Service	Allows users to begin/display their journey by choosing a path. Will also gain the ability to upload content and connect with other users
Website	User Experience	User Interface	Ease of use and accessibility for the user.
Collaboration	Service	Member Service	Members can request to collaborate with other members.
Timeline	Service	Member Service	Presents all of the courses, experiences, accomplishments, etc., in chronological order.
Gap	Data	User Interface	A term to capture the critical hurdles our target learner types; requisite job experience, skills, and skill knowledge.
Bridge	Service	System	OnRamp itself is the symbolic bridge or solution to the gap.
User	Actor	Use Case scenario	Must be registered in the system; otherwise, they don't have access to anything.
System	Hardware/Service	Use Case scenario	The database, all code, frontend design, and backend services.
Member	Actor	Use Case scenario	A registered user.
Non-Member	Actor	Use Case scenario	An unregistered user.
Homepage	User Interface	User Interface	The first page of the site.
Upload	Service	Site User Service	Allows users to upload pictures,

			videos, links, pdfs, etc., to their profile.
Guide	Service	Site User Service	It helps new users to familiarize themselves with how to utilize our platform to its full extent.
Triple Graph	Data	User Interface	A graphical composite of Experiences created by the user.

Initial List of High-Level Functional Requirements

Non-Member expectation

1. **Creating an account**

   The user will be able to register for an account. The system will complete this by storing the user's preferred username and password. Next, the user will input their first name, last name, date of birth, phone number, and email address.

   If the username entered is already in use, the system will tell the user that the ID already exists and prompt them to enter a different one.

2. **Account Access**

   The user gains immediate access to their account when they register. The user can log in and out of the account.

3. **Account confirmation**

   When the user is successfully registered, they will receive an email confirmation.

4. **Recovering Access**

   During the account creation process, the user will create two-factor authentication security questions. When a user forgets their username or password, the system uses the security questions for login recovery.

Member expectation

5. **Journey**

   The user will receive a journey or workspace and an option to create a new path to connect their experiences. Each experience will be a point on the timeline, and they can provide personal insights about why they decided to learn a subject, what they will learn, and what they produce. The user will also have the option to upload projects that showcase their knowledge of the material.

6. **Timeline**

   The system will generate a "triple graph," similar to a timeline to illustrate the user's journey on their profile. It presents their work, courses taken, accomplishments, etc., which the user achieved in chronological order. Thus, providing them with a visual representation of what they finished and are currently working on. If they do not want time constraints on their graph, they can turn off the feature.

7. **Profile**

   The user will be able to update and edit their profile, entailing standard account information: first name, last name, date of birth, personal summary, and profile picture.

8. **Search**

   The user will be able to search for other account members. Each account is private by default unless the account's visibility is set to public. If the account they are searching for is private, the user must request to connect to see the member's journey. Once the request is accepted, they will be able to view the member's profile. If the account is public, they can view the profile immediately.

9. **Member Collaborations**

   A user will be able to connect with other account members. By following other account members, users can collaborate to share experiences and work together on real-world problems. If a user unfollows a member with a private account, they will lose access to viewing their profile.

10. **Messaging**

    The user will be able to message account members that they follow.

11. **Block**

    The user will have the ability to block an account member from viewing and messaging their profile or report an account member.

12. **Notifications**

    The user will receive notifications regarding new followers, messages, and reminders to visit their account if a specific amount of time has passed since their last login. The user will be able to manage how they want to receive notifications.

13. Help desk

There will be a FAQ page with the most common issues and ways to fix them. Users can email OnRamp directly for assistance when solutions are not in the FAQ. The user must include their username, full name, email address, and their queries. User help emails will receive a response within 24 hours.

List of Non-functional requirements

Performance Requirements

1. **Responsiveness**

   The system will be operating on various monitor sizes and a wide variety of resolutions, from 1024 x 600 through 1900 x 1200, and responsive with all the variations.

2. ***Cycle Time***

   The cycle time at expected performance will be 1.0-1.2 with 5-10 concurrent users. After that, the system will operate with a 1.21 – 1.30 load of 11-25 simultaneous users or a moderate lag. With a user base of 26-45, the system will operate with a 1.31-1.50 or a heavy lag. Finally, the system will work with 1.51 – 1.70 with a load of 46-50 concurrent users or a very heavy lag. Any number of simultaneous users over 50 will cause system performance to halt until another user finishes.

3. ***Speed Per Transaction***

   The speed per transaction will be between 20 – 100 milliseconds, depending on cycle time. The system will be able to process anything between 10 – 50 transactions per second.

4. ***Test Requirements***

   The test requirements will mandate a standard load test, testing all functional specifications and their speed per transaction.

5. ***Reliability***

   Average time of 1 hour or less of downtime within three months. Downtime is for maintenance and system updates. On release, the system should be 100% operational for the first year of operation.

6. ***Minimum Bug Counts***

   - No more than ten bugs in the system during integration and testing.
   - No more than five bugs remain in the system after delivery.

7. ***Execution Speed***

The execution speed of the home page using a high-speed internet connection should load within 100-200 milliseconds, depending on cycle time.

8. **Storage Utilization**

Storage utilization should be within 75%-85% of available storage to avoid using all storage and causing technical issues if emergency storage is needed.

9. **Robustness**

The time to restart after a failure or crash should be under 1 hour. The probability of corruption of data due to failure must remain under 1%. The percentage of events that cause system failure will stay under 0.5%

Ease of Use

1. **Training Time**

   Training time should be minimal, or none at all, as the site will be user-friendly for anyone to use with ease.

Interoperability Requirements

1. **Browser Compatibility**

   The system will be a web-based app. It will use all major web browsers (e.g., Google Chrome, Mozilla, Firefox, Safari, and Internet Explorer). In addition, it will include functionality to provide alternatives if the browser does not have JavaScript installed on it.

2. **Computer and OS compatibility**

   The system will operate on various types of operating systems, including Windows, macOS, and Linux. In addition, the site is accessible with any computer which can run a browser that is supported.

Expected Load

1. We will make allowances for up to 50 concurrent users at once. Load testing will be done to record performance and cycle times during high traffic periods, which includes testing during continuous or spiked patterns.

Security Requirements

1. **Login/Password System**

The system will have a login and password system to maintain user preferences, Journey, accomplishments, and collaborators. Security will be implemented with account creation and email confirmation. In addition, there will be a password retrieval system via security questions if the user forgets their password. It will permit the user to retrieve and reset their login credentials.

2. **Encryption**

The website will not include end-to-end encryption at the stage of initial system delivery. Users are not encouraged to store personally identifying information on the platform. Moreover, there will be no sensitive information data storage features.

3. **Access Control**

The development team will have frontend and backend access. Users and visitors will have user-level access.

4. **Spam Protection**

The site will ask users to enter a string of characters shown in a picture to create an account to prevent bots from creating false user accounts.

5. **Resource Utilization**

Resources such as the MySQL database will be accessed through the usernames and passwords provided by Florida Atlantic University on-campus servers. With appropriate documentation and licensing, the system tools for use and access are Bootstrap, React, Express, Node.js, and MongoDB.

Portability Requirements

1. **Platform Compatibility**

The system will be a web-based app. Mobile and tablet versions may be implemented at a later stage.

2. Percentage of Target-Dependent Statements

This version of the system will have 0% of target-dependent statements. In a future mobile and tablet version, 30% of statements will be target-dependent.

Supportability Requirements

1. Coding Standards

The system will be developed in a range of 75-80% of coding standards for HTML 5 and CSS3. The code will be produced, reviewed, tested, and reviewed again for efficiency purposes.

2. Naming Conventions

HTML classes and id tags will be coded in lowercase except in situations where there is more than one word in a name; then camelCase will be used. The team will agree on SQL tables and data names. First letter capitalization and lowercase followed by lowercase letters will be the standard.

Storage Requirements

1. The storage for our system will be on FAU Lamp servers holding our MySQL databases within an unknown capacity and holding our files for the actual website.

Survivability

1. The system will be stored on FAU LAMP servers and backups stored in Microsoft OneDrive to prevent loss in case of any physical destruction of servers.

Availability Requirements

1. Accessible Times

Our system will be available 24 hours per day, seven days a week. In addition, the site will be running if FAU lamp servers are available.

2. Downtime Impact

Downtime will be minimal but will have a splash page informing that the system is down for maintenance. Scheduled downtime will be announced ahead of time and is expected to be minimal.

3. Support

There will be a FAQ page with the most common issues and ways to fix them. There will also be help available by email that will filter to assigned developers. Emails will receive a response within 24 hours.

Fault Tolerance

1. Exception Handling

Exception handling will be implemented in all situations where an error may occur. Users will be provided with an explanation for the exception and have a chance to input the correct answer or be taken back to the home page.

2. Self-Checking

There will not be self-checking software implemented since this system is not critical enough to warrant it.

High-Level System Architecture

Frontend Technologies

1. **HyperText Markup Language (HTML)**

   HTML will allow the browser to display the content of the website.

2. **Cascading Style Sheets (CSS)**

   CSS will be used to style web pages and enhance the user interface and user experience.

3. **Bootstrap**

   Bootstrap is the CSS framework that will be used to provide mobile-first and responsiveness for the WebPages and templates for components, like forms, buttons, and navigation.

4. **JavaScript**

   The JavaScript language will be used to program the web pages' behavior and functionality and define how the data flows and is stored on the client side.

5. **React.js**

   A JavaScript framework will be used for modern, scalable, and modular web app development with the use of interactive and reusable components.

Backend Technologies

6. **FAU LAMP Server**

   The Fall 2021 CEN4010 term project will be temporarily hosted on a Florida Atlantic University server.

7. **Node.js**

   A JavaScript runtime environment that will be used to develop and run our web application using JavaScript. It will also provide tools to facilitate and automate the development process.

8. **Express**

   This will be used to facilitate the development of the backend API and interfacing with the database.

9. **MongoDB**

   This is a NoSQL database system that will be used to build an easily scalable database with flexible and evolving data schemas.

Other Technologies

10. **Github**

    A code-hosting platform will allow our team to work on the app and track its versions without compromising the project's integrity.

11. **Jira Software**

    Jira is a team and work-management tool that will be used to track progress and manage our team project while using agile methodologies. It will also serve as the primary means of communication for the team.

12. **Jira Confluence**

    A collaboration wiki tool that will allow our team to develop the documentation necessary for our application collaboratively. It will also serve as a workspace for sharing knowledge and notes.

Browser Compatibility

The system will be a web-based web app that operates on at least two of all of the major browsers, including Google Chrome, Mozilla Firefox, Safari, Opera, and Internet Explorer. In addition, it will have functionality that will provide alternatives if the browser does not have JavaScript installed on it.

Competitive Analysis

The competitors' websites/platforms analysis will focus on six main features (homepage, design, usability, networking, journey, personality). In addition, the competitive analysis will utilize a numerical scale (1=bad, 2=poor, 3=fair, 4=good, 5=outstanding) and five websites/platforms chosen to focus on portfolio building, personality display, and presenting knowledge through implementation.

	OnRamp	LinkedIn	Handshake	Indeed	Sumry	AngelList
Homepage	5	3	5	2	3	2
Design	4	3	5	1	3	1
Usability	4	4	2	2	3	3
Networking	4	5	4	3	3	3
Journey	5	2	3	2	5	2
Personality	5	3	3	1	5	1
Mean	4.50	3.33	3.67	1.83	3.67	2.00

Expectations

Homepage: The user interface should be user-friendly to new users. The navigation should be easy to find and use.

Design: The page(s) should catch users' attention; information should not be cluttered. Buttons and other clickable links should be easy to identify, and they should be relevant.

The site should not have unnecessary whitespace but have a uniform design throughout the platform.

Usability: The ease of account and profile creation through intuitive navigation. The platform accommodates a diverse group of users who can follow, message, and collaborate.

Networking: The platform should allow users with a common profession/ interest to connect and exchange information (their experiences) to develop professional and even social contacts.

Journey: The extent to which the platform will allow the user to express all their work from start to finish entirely. Additionally, this process should not feel like a burden to perform for the user. Instead, it should give the user a sense of fulfillment for adding this level of detail.

Personality: The platform should allow the user to make meaningful connections with their experiences. They should have the freedom to freely explain how they acquired knowledge and applied it to an authentic product. This can further be demonstrated via a file upload of their choosing (pdf, .txt, youtube video, etc.) to show what was created after (if applicable). For example, a computer engineer creates a circuit on a breadboard and makes a YouTube video displaying its functionality. Then on OnRamp, they can connect with how they applied their knowledge to be able to construct and operate said project.

Competitors

OnRamp (4.50)

The OnRamp homepage will be very user-friendly. It will explain to the user what our platform is about and a breakdown and walkthrough on starting. Our design should have little to no blank spaces, be very informative, and catch users' attention. OnRamp will allow users with common interests/professions to connect and exchange information (their experiences etc.) to develop professional and social contracts. The journey and personality feature will give that freedom and connection to the users that other platforms do not offer. They will feel invested and interested in the profile they are creating.

LinkedIn (3.33)

LinkedIn seems not to have a very informative homepage (from a new user standpoint). However, the design is clean, simple, and cut up in small sections with clickable text, links, and bubbles/buttons, but it does not quickly inform a person what they need to accomplish to get up and running on their profile quickly. It has good usability because it allows users to add additional information, but it seems tedious and challenging to figure out how to add information. As for journey and personality, this platform seems not to encourage the user to become invested. On the other hand, networking on this platform is outstanding. It allows the user to search fellow users, lets you message them, and it lets you look for jobs. So, from a networking standpoint, it seems like an all-in-one.

Handshake (3.67)

The Handshake homepage (as a new user) is well-built. It is informative, and the design has high appeal. Sub-pages are not as attractive as the front page of the site. The students can access this platform, meaning a .edu profile is required. The profile building is not as elaborate and visually pleasing as LinkedIn and the designs planned for OnRamp. Networking is fair because it allows communication directly with companies about jobs. However, it does not facilitate user collaboration and connectivity. There is no method to showcase a finished product and its connection to what users have learned.

Indeed (1.83)

The homepage has many excess whitespaces. It does not guide the user to create a user profile properly and does not feel user-friendly. Design is poor and lackluster, and uninviting. Anyone can join, and page transitions are quick. It is challenging to begin profile building. The site coerces users to create a resume. New users who are inexperienced or at a pre-professional stage of learning may be confused about making a resume/Usability is poor because it is not user-friendly. For the last two features, it doesn't seem to be present here. You’re just adding info for a job instead of connecting your knowledge with your work.

Sumry (3.67)

The user interface is somewhat user-friendly, explains briefly how the platform works and has some graphics and color. However, some images are hard to see unless you click on them, they seem cluttered, and there are some big white spaces. Anyone can join, and the pages transition quickly. Users can find other user profiles, but direct messaging appears to be absent. As for the last two features, it seems you can talk about your experiences and skills and how you have applied them in the real world. You can see the personality in some examples, like Dayo Olopade.

AngelList (2.00)

The homepage has many white blank spaces, does not guide the user properly to create a profile, and does not feel user-friendly. Design is poor, lackluster, and it seems it would steer away users than bring them in. Anyone can join, and the pages transition quickly. Just like Handshake, it focuses more on connecting you to jobs instead of colleagues. The site only allows the user and the company to connect and message. For the last two features, it does not seem to be present here. You're just adding info for a job instead of connecting your knowledge with your work.

Planned Advantages

OnRamp is all about the journey. Many portfolio-creating sites just seem to ask the user to input information mindlessly. OnRamp encourages the user to form genuine creative partnerships, in what we call collaborations, to connect the knowledge they have gained, how it affected them, and how they have applied new knowledge. Most of the career support sites are an endpoint that assumes job readiness. But what pre-professional users need help translating courses (college or otherwise) into transferrable skills and experience as they progress towards applying for employment? People who are just starting or self-learners?

OnRamp allows its users to bridge the gap from *start* to *finish*. It is a holistic journey with a timeline and measurable progress. This means that the users have the freedom to start from wherever they want and begin documenting their journey. As a result, if a company is interested in you, this platform will put your personality on display. In other media, it seems complicated to have your profile stand out among the rest, but that's where the beauty of OnRamp comes into play. It allows you to keep track of what you learned, document products created with the knowledge obtained, and display what type of person you are too. This platform will make the endpoint easier to create and share with companies when the time comes to perform a job search.

Team Roles

Team Lead, Front End Team Lead/Developer:

★ *David Moore*

Github Master, Front End Developer:

★ *Laura Casals*

Scrum Master, Back End Developer:

★ *Carlos Fonseca*

Back End Team Lead/Developer:

★ *Thiago Goncalves Vasconcelos*

Back End Developer:

★ *Ignacio Albornoz*