CareerAdvisor Final Report

CS4474/CS9552: Human-Computer Interaction

Team Members of Team 4:

Divigalpitiya, Ryan
Li, Xin
Wright, Alexander
Xie, Kun

Submission Date:

April, 9, 2018

Ryan Divigalpitiya

Xin Li

Alexander Wright

Kun Xie

Table of Contents

Executive Summary

30 Design Principles

Heuristics, Evaluation & Improvements

Navigational Map & Attached Prototype

CareerAdvisor Final Report

Executive-Level Description: User Goals

The **primary goal** of the user of our application, called *CareerAdvisor*, is to discover what careers are right for them based on their interests, take the necessary actions to be qualified for those career fields well before they graduate and, eventually, apply for those jobs.

This primary goal applies to **two user types**:
(a) Users that are currently enrolled in post-secondary programs or
(b) Users planning on (and deciding upon) what post-secondary programs to enter.

The primary goal consists of **three sub-goals**:

1. **Discover** what kind of careers they can get with the completion of their current or future post secondary program.

2. **Create** a plan or model plan to become fully qualified for a career of interest by the time they graduate (while also tracking their progress) and,

3. **Search** and apply for online job postings (or gauge the availability of jobs) that are related to their post-secondary program, view job projections and, finally, view career salary and benefits information.

These three sub-goals apply to both **(a)** and **(b)** users, as they have similar concerns regarding career planning. Take note that **(b)** users have the added benefit of earlier execution of these goals, as CareerAdvisor will help them make more suitable post-secondary program selections, which in turn will translate into career choices that are more in line with their interests.

The three sub-goals map cleanly onto **three corresponding tasks** that *CareerAdvisor* aids the user in achieving:

Task 1	Searching for & choosing from careers/jobs that are related to what user is studying
Task 2	Creating a plan that will result in user becoming fully qualified for applying for chosen career/job by the time they graduate
Task 3	Tracking user's progress of acquiring all qualifications in time for graduation & applying for related job postings when nearing graduation

Task #s are referred to throughout this submission.

CareerAdvisor Final Report

Executive Summary
Complex Tasks that Serve Users' Goals

A breakdown of the Executive-Level Description has been done both **visually** and **textually**. We recommend viewing the visual breakdown to aid the textual breakdown that is presented here.

0-Visual Breakdown of Executive Summary.pdf

The following will break down each of the three tasks and their corresponding sub-tasks in greater detail, as well as commenting on how user goals are served by their operation.

Task 1: Searching for and choosing from careers/jobs that are related to what user is studying or are interested in studying.

Step 1 Sub-task

- **Input by User:** Inputs area they are studying/interested in studying or post-secondary program they are pursuing/wish to pursue.
- **Output by App:** App displays 2 things in return:
    - 1) All Career Choices related to previous input —> Each choice is click-able.
    - 2) A summary of all the career choices, industries, average salaries, etc as an overview of what their degree/area of study yields.

Commentary: *This sub-task is central to the user's sub-goal of discovering careers of interest. (b) users can also use this function to map out career paths available to them from certain post-secondary programs in question.*

Step 2 Sub-task

- **Input by User**: User clicks on a Career Choice they want to know more information about.
- **Output by App:** App displays a description of the selected career, the industry the career is in, and other relevant career information such as salary averages and ranges, usual benefits and available career growth projections —> User can select a career choice as a "Career They Are Interested In".

Commentary: *This sub-task allows the user to further drill for information they would like to know about a career to deem it a "Career of Interest".*

Step 3 Sub-task

- **Input by User:** Select a listed Career Choice as a "Career They are Interested In".
- **Output by App:** App will add it to their list of careers they are interested in —> the app then allows the user to create a plan for this career choice.

Commentary: *This sub-task is the first step in taking the leap from discovering career options to creating/modelling a plan for a selected career option.*

Step 4 Sub-task

- **Input by User:** Selects "Create a Plan" for a chosen career.
- **Output by App:** App will guide user to create a plan that will result in user becoming fully qualified for applying for chosen job by the time they graduate (Task 2 + 3).

Commentary: *This plan can be used for a user who is fully committed to follow the plan, or a user who is modelling career planning paths that stem from certain post-secondary program choices they are considering or have chosen.*

Task 2: Creating a Plan that will result in user becoming fully qualified for applying for chosen career by the time they graduate.

Step 1 Sub-task

- **Output by App:** App requires user to fill out the following information before guiding them to create a plan:
  - Year they've started or will start their post-secondary program.
  - Expected graduation date (can automatically generate this info if user inputs how long their post-secondary program is).
  - A Yes or No input of whether their post-secondary program offers internship options or not.
  - A Yes or No input on if they'll be doing their post-secondary program is part-time or full-time.
  - A Yes or No input on if they have summer breaks.
- **Input by User:** Fills out relevant information listed above and submits their information

Commentary: *It is important to keep these options free from the actual constraints the listed post-secondary programs specify, as these constraints can change in the future.*

Step 2 Sub-task

- **Output by App:** App displays the following:
  - Generate List of Required
  - Breaks down list of Required Qualifications in to categories and may offer various details of information on each item
- **Input by User:**
  - User can edit/add to generated list of required qualifications.
  - User adds items from list of Qualification Items on to Timeline to plan when to acquire each Qualification Item by (in the form of an Action)

Commentary: *This sub-task can be framed as an actual course of committed action, or for a career plan modelling scenario.*

Step 3 Sub-task

- **Output by App**: App highlights in red or whatever colour to indicate remaining items that still need to be planned —> user completes planning everything else.
- **Input by User:** Time passes by and user's point in time in the real word starts to approach a "Take Action" deadline as saved in the Plan's Timeline

Commentary: *This sub-task can be framed as an actual course of committed action, or for a career plan modelling scenario.*

Task 3: Tracking user's progress of acquiring all qualifications in time for graduation & applying for related job postings when nearing graduation.

Step 1 Sub-task

- **Output by App:** App sends notification to user via email to notify them of a Required Action that is part of their plan.
- **Input by User:** After further passage of time and user acquires a Qualification Item, user checks it off
- **Output by App:** App updates itself to show newly acquired Qualification Item.

Commentary: This sub-task is key to verifying that a user is making actual progress towards being ready to apply for selected careers upon graduation, and have a competitive chance at being hired.

Step 2 Sub-task

- **Input by User:** Time passes by and user's point in time in the real word starts to approach Graduation. User clicks on "Apply for Jobs"
- **Output by App:** App shows search results + links from Monster, LinkedIn + other Job Search Platforms with available developer APIs —> User applies to desired postings

Commentary*: The ultimate goal of the App, is for a user to find full employment. Proper integration of the user's entered information and selections to job search platforms is most important. User's who are still considering careers of interests chosen from the previous tasks can then proceed to this last sub-task to gauge current job availability in their interested career field.*

CareerAdvisor Final Report

30 Design Principles

1. Alignment (UDP p. 24)

- The text in our app, is aligned as follows:
    - Left aligned text for paragraph form content (e.g. in the career summary descriptions and full descriptions).
    - All career search results are displayed in row alignment.
    - Nearly all user information inputs are displayed in column alignment.

2. Colour (UDP p. 48)

- Number of Colours:
    - Our app deliberately is limited to white and shades of blue, violet, and grey. This was done to reduce overload on eye processes.

- Saturation:

    - We have chosen to limit the instances of saturated colour (i.e. the blue-violet spectrum scheme), for important features of our app, such as:
        - Marking the stage of the app, in the navigation bar (i.e. Explore, Pinned Careers, My Plan)
        - Marking “Pin This” or further “Explore” functionality on returned career results
        - Marking “Qualifications Checklist”, “Add Action Item”, “Add to Timeline”, “Close”, and “Save Changes” functionality in the Create Plan function.
        - The “Overview” dashboard.
        - The app’s upper margin bar, that contains User Account options.
        - The Job Search, Monster, Linkedin, Indeed, job results summary.

- Desaturation:

    - For the remainder of the app, we have chosen white or light grey as the colours of choice, this can be especially seen in the information inputs, and career search

results. This in our minds, maximizes the professional look and friendliness of the app (and does not overload a user's visual senses).

3. Consistency (UDP p. 56)

- The entire app has been designed to be consistent both visually (e.g. uniform colour schemes across all three main functions), and functionally (e.g. radio buttons, toggles, dropdown menus, sliding scales) through the Create Plan functions.

4. Five Hat Racks (UDP p. 100)

- Career requirement information is organized by category (i.e. "Education", "Skills", and "Experience").

- The user inputs in the Create Plan function, are organized by category (i.e. "Education", "Skills", and "Experience").

- Once the user has finished inputting the desired actions and submissions related to education, skills, and experience, this information is displayed in yearly timeline (i.e. the information is organized by time).

5. Highlighting (UDP p. 126)

- Bold text highlighting is utilized:
    - In the career full descriptions, that highlight the "What Do They Do?" and How Do I Get This Job?", headings. As well as the choices under "Education", "Skills", and "Experience". This was done to emphasize important and relevant items.

    - In the Create Plan Required Qualifications auto generation, the heading, and "Education", "Skills", "Experience" choices are bolded to draw the focus of the user to them (which signals their importance in regards to the Career Plan Timeline).

- In the Career Plan Timeline, text in the “Add Action Item” sub-function is bolded (i.e. to signal the what, how, and when options of the sub-function).

- In the Career Plan Timeline, popups for both Submission Items, and Action Items, use bolded text to highlight item title, qualifications in that item, start dates (which include a noticeable size increase), and deadlines.

- Coloured text is utilized:

  - In the Career Plan Timeline function:
    - To indicate Submission Items (magenta) and Action Items (violet).

    - To indicate the status of education, skills, and experience selections within an action item (i.e. violet for “In Progress”, green for “Already Added to Timeline”, gold for “About to Add This :)” red for “Not Planned Yet”.

  - Colours highlight crucial attributes user’s must accumulate in order to realize their career plan.

6. Iconic Representation (UDP p. 132)

- Our app utilizes a number of symbolic and arbitrary icons.
  - In the career full descriptions, three symbolic icons (i.e. clock, for commitment, dollar-bill, for average salary, and briefcase with magnifying glass, for number of job postings), and one arbitrary icon (i.e. up and down arrows, for salary range) are included.

  - Symbolic icons for education (“degree” icon), skills (crossed paintbrush and pen icon) and experience (checklist icon) are utilized in both career full descriptions and the Career Plan function to aid in user recognition

  - In the Career Plan Timeline function, symbolic icons denote the “Qualifications Checklist” (i.e. target sign), and “Add Action Item” (i.e. lighting bolt sign) sub-

functions. Submission Items are denoted by an arbitrary icon (i.e. an arrow pointing upwards, grounded by a representation of a letter tray). The icons for action items and submission items repeat themselves throughout sub-functions and the timeline, for the purpose of user comprehension (i.e. sparing the user the time to read "action item" or "submission item" over and over again). Also the symbolic pencil icon is utilized to denote edit sub-functions.

- The Apply for Jobs function, utilizes both an arbitrary icon (pin on board, for location) and symbolic icon (radar screen, for radius).

7. Layering (UDP p. 146)

- Two-dimensional linear layering is present in both:
  - The navigation bar (i.e. tabs for Explore, Pinned Careers, My Plan), which underscores the narrative direction of the app's purpose (i.e. from exploring careers to entering a chosen career).
  - The completed timeline visualization (that includes all relevant action items and submission items), in the Career Plan Timeline function. Is a linear layering since it is segmented by academic years. It is helpful to view a career plan in this way, since the overall representation of all the required career tasks, is less complex.

8. Legibility (UDP p. 148)

- Standard 12-point font is utilized throughout the entire app (larger text is utilized for Overview, header, and item start dates).
- Sentence case text is utilized for readability.
- Most important text is arrayed on lightest background, to optimize reading performance.

9. Mapping (UDP p. 152)

- Throughout our app, control functions are single control-single effect (e.g. for career choice selections, pinning, and exploring a represented as separate control buttons).

- This mapping, maximizes user expectations regarding action and reaction, and avoids confusion.

10. Proximity (UDP p. 196)

- Our app groups information by the Gestalt principle of Proximity, in some cases.

- It can be especially seen in the Career Plan Timeline function, where the timeline itself functions as to group action and submission items by year (the years are then interspaced), this reduces information overload for the user, and promotes relatedness of the items to the yearly plan.

11. Signal-to-Noise Ratio (UDP p. 224)

- The overall look and functionality of our app minimizes the application of extraneous elements.

- The design is content (e.g. job postings, career plan elements) and functionality driven (e.g. its primary output is to create a career plan that guides users along to entry into a career field). Any content or functionality that does not aid in this regard, has not been included. This produces the effect of a streamlined design, which has a second order effect of a high signal-to-noise-ratio.

12. Visibility (UDP p. 250)

- The status and functions of our app are clearly visible, throughout its operation.

- This is seen in the app's navigation bar, which shows what functional stage the app is in (i.e. tabs for Explore, Pinned Careers, My Plan).

- Top 10 career scrolling functionality notice.

- In the app's Explore function, once a user selects "Pin This" for a respective career, a notification fades in and out (as well as the initial app level popup).

- Also in the app's Career Plan Timeline, the effects of a user's actions can be clearly seen in the four-year timeline breakdown, as it displays action and submission items immediately after they are added in a career plan.

13. Accessibility (UDP p. 16)

- Our app emphasizes simplicity and forgiveness (more information on this aspect is listed under the following Forgiveness principle).

- Simplicity (in the context of accessibility) is emphasized because our app uses simple and consistent language (e.g. "Education", "Skills", and "Experience" headings), which is accessible to high school graduates of many walks of experience. Functionality (such as "Pin This", further "Explore" and the action item buttons) provide clear visible prompting as well as feedback when executed (e.g. careers being pinned instantly, and action items being added instantly to a respective career plan timeline).

14. Comparison (UDP p. 52)

- Our app's Explore functionality displays career search results, and respective information such as average salary, industry, and number of job postings. These metrics allow for a user to make an "apples to apples" comparison of their searched for and pinned career results.

- Pinned careers include salary ranges as well.

15. Forgiveness (UDP p. 104)

- Our app's interface promotes "Good Affordances", because its input features are designed to ensure correct use. For example, the selection radio buttons in the "Add Action Item", are clearly marked and emphasized by back shading (to prevent users from selecting the wrong skills/attributes).

- Our app includes functionality that ensures full "Reversibility of Actions", as far as pinning and unpinning searched for careers, and editing and deleting action items and submission items (in the case of user mistake or change of mind).

- Our app is planned to have a help function (i.e. instructions)

16. Garbage In-Garbage Out (UDP p. 112)

- Our app's good affordances (e.g. visibility of user control functionality, and confirmation functions), allows for the minimization of input "garbage". This is closely tied to the forgiveness aspect of our app, which should also minimize user error (e.g. action reversibility).

- The type of erroneous input we are most concerned with here are "Problems of type", such as mistaken skill selection (in action items) or date selection (in submission items) inputs.

17. Hierarchy (UDP p. 122)

- Our app can be broadly defined as a hierarchical stair structure in how it lays out its organizes its information for viewing and interaction by the user.

- For example, on the highest level our app's functionality is collected around three "steps"; "Explore", "Pinned Careers", and "My Plan". From each of these steps, more narrower functionality flows (i.e. searching careers, viewing information on pinned careers, and creating a detailed plan based on a chosen career).

- This organization strategy, allows for users to always have a good sense of where they are in the app's interface.

18. Immersion (UDP p. 134)

- Our app is designed to be seamless (as possible) in its transitions between core functionality. Its response time is less than 0.1 seconds, when user actions are executed. Thus, there is no gulf between the user's perception of the results from their executed actions.

- This results in an immersive experience for the user, which affords them greater focus when making and planning career choices.

- Also, when the user chooses to further explore selected careers, an in-app window appears (and does not open a new tab), this helps immersion, and does not break the user's attention.

19. Performance Load (UDP p. 178)

- Our app's primary purpose is to aid in a person's career aspirations and goals. One of the ways it accomplishes this purpose, is by greatly reducing the cognitive load relating to the activities and tasks of career planning, on the medium of a computing device.

- This is thanks to the visible and well organized structure of our app's functions and the information it offers.

- For example, the tasks concerning the activities of searching for, selecting, and planning for a careers are entirely performed graphically, and there is no need for the user to memorize any detail contained within the app (besides their password of course).

20. Progressive Disclosure (UDP p. 188)

- Our app, implements the principle of progressive disclosure in various ways.

- For example in the “Explore” careers functionality, returned career results are displayed with only key important information (i.e. average salary, salary range, and industry sector), even top 10 career results are returned with a shorter description. If the user desires to find out about more about any of the returned careers, they can either click the relevant “Explore” button or “Pin This” (in the case of multiple desired returned results). This reduces information overload, and complexity especially for novice users of the app.

- Also in the Career Plan Timeline, once a user’s career plan is completed, the potential plethora of career related tasks could be overwhelming if all displayed in the same place, this is why each career related task is itemized, so a user can click on them to be “progressively disclosed” with more detailed information.

21. Readability (UDP p. 198)

- All text in our app has been written to correspond with basic english as much as possible, along with sentences being kept short.

- It avoids jargon, and complex words especially relating to functionality descriptions.

- Industry related jargon and complex words may however be tied to career postings, and their respective required skills/experience.

22. 80/20 Rule (UDP p. 14)

- This principle was considered in the conceptualization and design phase of our app.

- For example, we have planned the three main functionalities of our app (i.e. career searching and selection, career plan creation, career plan progress tracking and job application integration), to be as streamlined (i.e. focusing on accomplishing these functions in the plainest possible meaning) as possible.

- This results in an interface that is easy to understand and use, since its core functions are emphasized (i.e. the 20%, that is probably responsible for over 80% of the output), and there is no superfluous or “nice to have” additional sub-functions, that may interfere with a users goal of finding/choosing a career, and following a career plan to completion.

23. Aesthetic-Usability Effect (UDP p. 20)

- We like to think that our app’s aesthetic appearance has been carefully chosen to harness this effect (e.g. the choice of complimenting colours, shading, and overall clean (vs. cluttered) design).

- Since we were focused on the usability of our core functionality, we made sure the graphical design was considered to be appealing to the eye. This would aid in the user’s perception of usability, before they would begin to interact with the app’s interface.

24. Confirmation (UDP p. 54)

- Most of the app’s functionality, operates in the manner of “two-step operation” to confirm user inputs.

- This can be seen in the initial “Create Plan” options, as well as the “Add Action Item” and “Submission Item” options. In all cases there is a final confirmation button, to close the active window, and confirm the timeline/action item/submission item selections.

- This promotes the behaviour of a user double checking their inputs, as opposed to just inputting selections, without a confirmation button (i.e. the window would close, after the final selection is made in the sequence).

25. Control (UDP p. 64)

- Our app approaches the design principle of control by having a simplified design, since we assume that all users are first time users.

- This can be explained in two ways:
    - Users that do indeed experience the app for the first time, need to be able to effectively interact with it, thus control functionality is simplified.
    - Users who have interacted with the app before, to create a plan, would not revisit the app too frequently once their career plan has been fully created (also since the app would send out notifications). The only times they would revisit the app, would be to change their career plan, edit or complete an action item, and finally apply for the chosen career.
    - Thus, user interaction with our app has been viewed as overall occasional, and controls have been created to view all users (experienced or not) as at the same group (i.e. control options and range do not vary based on user experience).

26. Entry Point (UDP p. 80)

- Our app's entry point (i.e. the splash page), is designed to be inviting (based on its aesthetics, which minimizes user perceived barriers), and to inspire curiosity (i.e. a progressive lure, due its minimalist presentation).

27. Errors (UDP p. 82)

- Our app minimizes errors primarily through the implementation of button controls (to minimize slips) (also a confirmation pop-up appears to confirm initial career plan details) and a clean, non-cluttered interface that minimizes noise (which helps prevent mistakes).

28. Fitts' Law (UDP p. 98)

- This design principle, is considered and implemented throughout our app.
- It can be seen in many places, including: the large buttons for the "Explore" button or "Pin This" functions, as well as smaller controls and selectors, in the "Add Action Item" and

"Submission Item" being more closely grouped together for ease of use.

29. Personas (UDP p. 182)

- When we originally envisioned the purpose of our app, we planned it to support all types of post-secondary programs for theoretically any type of career.

- Since then, we have refined and narrowed the scope, to just focus on university 4 year programs, with plans to expand on this, once this segment group or persona has proven the utility of the app (i.e. by broadly adopting it for career planning activities).

30. Wayfinding (UDP p. 260)

- Our app incorporates wayfinding, primarily in two ways.

- In the app's navigation bar, users can orientate themselves as far as what core functional areas they are interacting with (i.e. the notation of Explore, Pinned Careers, My Plan). This promotes both user orientation and route monitoring.

- In the Career plan timeline, the effects of a user's actions can be clearly seen in the four-year timeline breakdown. This also promotes both orientation (i.e. see where they are and where they are going), route monitoring (i.e. see what action items relate to, and their corresponding submission items), and destination recognition (i.e. the plan timeline, ends in a apply for jobs function).

- Happy face signifies users relative position on their timeline.

CareerAdvisor Final Report

Heuristics, Evaluation & Improvements

1. Interaction & user activity :

1. Metaphors

- Which basic metaphor for interaction does the app use? Manipulating, navigating or conversing?
- Any suggested improvements?

Alex W:

- Our app primarily uses the conversing interaction metaphors. This is because the user uses natural language to search for desired careers, as well as the use of text-menu selections and button commands to create a career plan around a selected career. Manipulating, could also be a metaphor, especially if the app was implemented on a touch screen.
- Suggested improvements for the app include an augmented reality interface that could be implemented with augmented reality technology, this could incorporate the “navigating” metaphor, and be a more immersive experience (assuming the original concept works, and augmented reality becomes more mainstream).

Kun Xie:

- Conversing is the main metaphor we used, since we are using detailed and natural words to communicate with the user

Xin Li:

- The metaphor we used in the app is conversing. In the search feature of the app, users use natural-language speech to interact with the app.

Ryan D:

- As highlighted by the Explore tasks where the user searches and explores for a career of interest, our app mainly uses conversing where the user uses natural language to hone in on a career that peaks their interest. The app’s design achieves this well where it is difficult to find areas of improvement on this.

2. Interface styles

- Which interface styles does the app use? (3D, gesture, VR)
- Any suggested improvements?

Alex W:

- Our app primarily uses a WIMP interface. This is because it was designed to function as a web app. It also incorporates a natural language interface, for its very first function (i.e. searching for careers). A direct manipulation interface could also be easily implemented for touch devices.
- Suggested improvements in this regard include an augmented reality interface if the technology becomes mainstream.

Kun Xie:

- As a web based application, our app uses standard WIMP interface
- Implementing gesture function to our app can improve more accessibility

Xin Li:

- The app use WIMP interface because this is a web application.
- Enable touch feature.

Ryan D:

- The application employs a standard WIMP interface design which is a consequence of implementing the app as a web application that runs in the browser. For touch devices, a direct manipulation interface could be developed in addition.

<u>3. Interface styles</u>

- Which interface styles does the app use? (DOM, DCM, RDCM)
- Any suggested improvements?

Alex W:

- Our app primarily uses Direct Object Manipulation (DOM) where applicable, this can be seen when a user makes their career choice selections, and selects their action and submission items. In the app's Career Timeline, one could say it is an example of Reflective

Direct Concept Manipulation (RDCM), since the “time blocks” (ie. the semesters and summer periods) are created through the reflective selections of the users (i.e. through careful selections that collate into action and submission item timeline blocks).

- Suggested improvements include a hover feature, that depicts distilled information of the makeup of each action and submission item, whenever a user hovers their cursor over the timeline block. This could assist in deeper understanding, since users can quickly go back and forth between timeline blocks, instead of selecting each one.

Kun Xie:

- We used html and css with javascript DOM enabled interaction to design our web page interaction system

- Making another special mobile or tablet view of our app will be more user friendly

Xin Li:

- The main interface the app uses is DOM, and RDCM is also used. DOM is used during job selection stage, RDCM is is used during plan creation stage.

Ryan D:

- Our application employs more than one interface design style - (1) Direct Object Manipulation (DOM) and (2) Reflective Direct Concept Manipulation (RDCM). (1) DOM can be seen in how we allow the user to directly pin and aggregate careers of interest along with placing Qualification Items from their checklist onto their Plan’s timeline. (2) RDCM design is employed via the collection of the user’s degree information (when the user fills out a form asking for info on their degree) that directly affects how their plan will be created.

<u>4. User Experience</u>

- Does the user experience of the app contain enough movement, tension and relationship?

- Any suggested improvements?

Alex W:

- Our app does create enough movement for a user, when they are navigating each of the three main functions (i.e. the tabs for Explore, Pinned Careers, My Plan). This is primarily because of the aesthetics of the app, which in my mind creates a welcome atmosphere of freedom, in which users can realize their career goals (i.e. searching for any career they want, and making a custom career plan). Tension is created in the app, along the lines of the user still being constrained by the Explore >> Pinned Careers >> My Plan sequence of app functionality. This is a good thing however, since it is just enough tension to guide the user along, and not waste their time. Our app invokes a relationship with the user, this is because it serves as a handy cognitive support tool, in order for the user to diarize and follow through on career plans and goals.

- Suggested improvements include an add custom career option, since there is a possibility that our app will not always include all types and titles of careers. This would improve the relationships with users, who have more obscure career aspirations.

Kun Xie:

- The user experience movement in our app is explained in detail as fundamental functions of our app

- Implement another compare function for similar jobs will be better

Xin Li:

- Use could experience movement in our app. The app use 3 differents tabs, step by step guide the user to create a plan appropriately. User could experience relationship and tension as well, user is asked to provide detailed and correct information or a inappropriate

plan will be made, also each page provides many useful information and attract user stay the right tab.

Ryan D:

- Our app facilitates intuitive movement throughout the application through the 3 main tabs situated at the top of the app at all times. The order of the tabs imply a natural and intuitive direction of movement through the application where the user starts out in Tab 1 (Explore) and moves from left to right onto Tabs 2 and 3 (Pinned Careers and My Plan, respectively). Each tab also promotes tension in a positive way by functioning as a way to restrict the user to a specific main task within the tab area before moving on to the next tab (or next major task). We also believe that a strong relationship will form between the app and the user for two reasons: 1) they can potentially spend up to 4 or 5 years using the app to hopefully make it easier to plan their academic path along with 2) deriving great value out of the app as a result of the app's attempt to help the user plan their life's future in an easier and smarter way than before - something that can highly impact the user significantly.

<u>5. Action</u>

- Is it used effectively to attract attention, group elements, indicate meaning, and enhance aesthetics?

- Any suggested improvements?

Alex W:

- Actions (at least the buttons that implement core functional actions), are designed (especially with saturated colour) and a "pop-out" animation to attract user attention. User actions surrounding the creation of their career plan, result in the indication of meaning, i.e. the creation of a meaningful and organized (aesthetical) career plan.

- For example, the buttons make actions explicit and direct in focus, and are manipulated manually (as far as agency is concerned). Actions relating to the career timeline are often of

composite granularity, since career plan items often have an action and submission component. Timing is both user and system paced (i.e. users are free to operate within their own time constraints when selecting and pinning careers, but when a career plan is initiated the app keeps track of accomplishments and submissions, so then it is system paced by user choice).

- Overall, our app implements these action types relatively well, the only improvements concerned may be further improvements in the app's aesthetics (discussed below).

Kun Xie:

- We used many arrows, text buttons and explanation text and popup window to guide and attract users.

Xin Li:

- Buttons and explanation text indicate the function correctly, popup window and highlight make sure the user focus on the right information.

Ryan D:

- Our app utilizes pop-ups, notifications and high-contrasting buttons to attract attention effectively, indicate meaning and enhance aesthetics.

2. Interactivity:

<u>6. Interactivity Level</u>

- Is the right level of interactivity to best facilitate the user's tasks present in the application? (ie. Highly interactive vs. Static / No interaction)

- Any suggested improvements?

Alex W:

- The app helps facilitate the user's more abstract understanding of actions, through clearly identifying activities (i.e. Explore, Pinned Careers, My Plan). This way users, can have an expectation of what levels of interaction they are capable with the app.

Ryan D:

- Our app focuses on only including a high-level of interaction on components that the user will expect to be interactive in order for them to achieve their goals. For example, all "careers" that are displayed in the search results are interactive with enough options the user would expect to have (and no more than that).

- Room for improvements would include making the Plan's timeline more directly interactive - more specifically, a drag and drop interface to help the user add Action Items onto the timeline along with editing or moving the Items around the timeline.

Xin Li:

- The app interact with users in many ways, like buttons and pinned job.

- We can improve the timeline by adding a drag features to adjust the length of each stage.

Kun Xie:

- Yes, most components (buttons, tabs and career boxes) in our app create interactions with users.

<u>7. Reaction: Visibility of State Changes</u>

- How visible are the conditions of the interface, once the reaction process is complete and the interface reaches equilibrium for the 3 different types: Creation, Alteration and Deletion?

- Any suggested improvements?

Alex W:

- The conditions of the interface are quite visible, as far as reactions that result in creation, alteration, and deletion. For example, in the app's "My Plan" function, action and submission item blocks are readily created, altered, deleted, based on user wish and circumstances. Also the pinned career function, is an example of creation and deletion in that space.

- No suggested improvements in this regard, as I believe there is an equilibrium based on initial design intentions.

Xin Li:

- We have pinned job and popup notification to show the state is changed.

Ryan D:

- For Creation: examples include pinning careers into the Pinned Careers tab and adding Action Items to the Plan's timeline. Our app ensures a very high visibility of state changes with both occur. For example, notifications telling the user that a career has been pinned are present to ensure the user sees the result of their actions of pinning the career.

- For Alteration: collecting a list of pinned careers through adding or removing pinned careers is once again highly visible through the above mentioned notification system.

- For Deletion: Once again, the above explanations for Creation and Alteration apply here as well.

Kun Xie:

- For add and deleting functions, we have a popup notification box that notifies user, which is quite noticeable.

8. Reaction: Transitions

- How is change in system states displayed when the system is reacting to user input?

- Are animations used and are they used effectively?

- Any suggested improvements?

Alex W:

- Transition between system states is accomplished quite smoothly, and allows for more seamless integration of the three core functions ("Explore", "Pinned Careers", and "My Plan"). Fade in and out animations are utilized, and non modal application level pop-up appears the first time a user decides to pin a career (after than a notification fades in and out in the top right portion of the screen).

- Transitions are done masterfully in my opinion.

Ryan D:

- Animations were used to facilitate the user's reaction and perception of when the system is transitioning from one state to the other such as fading and opacity animations. These animations are effective at making the app's overall use and the user's reactions to the state changes more intuitive and pleasant.

Xin Li:

- We have pinned job and popup notification to show the state is changed.

Kun Xie:

- We used fade in effects when showing the results after user inputs.

9. Action: Focus

- Focus is concerned with the representation to which the user attends to, in order to act upon a representation of interest - there are 2 types: Direct, where the user acts on the representation of interest directly and, Indirect, where the user acts on an intermediary representational element. Does the app utilize the right Focus type at the right times? (ie. Direct. vs. Indirect).

- Any suggested improvements?

Alex W:

- Direct focus is used when a user, especially when a user pinns a career to their "Pinned Careers" space. While indirect focus is used when a user creates an action/submission item representation on their career plan timeline, since they are indirectly manipulating the creation of these representations through menu selections. This in my mind is logically implemented, and done at the right time, that is easy for a user to expect and understand (given our design metaphor).

- No suggested improvements in this regard.

Xin Li:

- Direct focus: pinned careers

- Indirect focus: When the plan is created, the app indirectly make the use to check the plan.

Ryan D:

- During the Explore task, the user has the ability to directly manipulate the careers they're interested in by clicking on them and saving (pinning) the ones they like. I found that this was as intuitive as the design could possibly be for this task. For the Create Plan task, the plan is quite complex and, while the ability to directly manipulate it would have been nicer,

choosing the design the interface to allow for indirect focus was still intuitive enough for me.

Kun Xie:

- We have both direct and indirect representation in our app, and we apply them for different function complexities.

10. Action: Granularity

- Does the app employ appropriate levels of granularity for tasks? (ie. Atomic vs. Composite)

- Any suggested improvements?

Alex W:

- Granularity in our app fits the function. Atomic granularity is used (relatively) in tasks such as career search. While composite granularity is very much shown in all aspects of the My Plan function (i.e. all the sub-tasks a user must perform to either set up their initial My Plan profile, or sub-tasks related to action/submission items).

Ryan D:

- For the Explore task, one does not expect a high level of granularity for such a task and the design stayed true to that - employing only atomic levels of granularity. With the Create Plan tasks, such complexity would result in one expecting to have various composite tasks to accomplish their goals and the interface was designed in a way to help the user through the various composite tasks to accomplish the their goals.

Xin Li:

- In this app, we use granularity for career search, and timeline.

Kun Xie:

- In our app, we used atomic (simple) granularity for career search and view function. For create plan, which is the timeline function, we used higher level of granularity.

3. Metaphor, semiotics, representation:

11. Metaphor

- What is our app's metaphor? Does it (in your mind) effectively bridge the gulfs of execution and evaluation with the user?

- Any suggested improvements?

Alex W:

- Our app's design metaphor is a hybrid between a career search platform, and a career planner. This combination can be clearly seen in the app's core functions of "Explore", "Pinned Careers", and "My Plan". Targeted users, in my mind our familiar with the concepts of career search platform, and career planners, since they are tools that most people encounter.

- A suggested improvement would be to perhaps state that CareerAdvisor is a combination of career searcher and planner on its splash page.

Ryan D:

- Our app uses the metaphor of time during the Create Plan task where a timeline is used to convey the user's time spent in real life during their academic years along with the implication that they must add events to their timeline (Action Items). I found this to be an intuitive way to help the user understand what is expected of them in the application and so I do believe the app bridges of gulfs of execution and evaluation with the user effectively.

Xin Li:

- We have pinned job and popup notification to show the state is changed.

Kun Xie:

- For the metaphor of our web app, we used a mixed version of a career searcher and planner, which makes our feature functions complete.

12. Representations: Overall concept

- Does our app's representations constrain thinking or aid it (i.e. by helping users find and interpret information)?

- Any suggested improvements?

Alex W:

- I never came across a situation when using the app, that constrained my thinking. The career plan timeline representation especially aided my thinking (if I was in the place of someone planning career activities over the span of four years). I would not want to see what a paper version of that plan would look like!

Ryan D:

- I found that the representations in our app all do a very good job of aiding the user in finding desired information and helping them interpret them.

Xin Li:

- The app is very intuitive and guide the user to use it in a correct order.

Kun Xie:

- Our app is designed in a way to aid and guide users, so it does not constrain thinking.

13. Representations: Cognition

- Does our app promote experiential or reflective cognition via its representations?

- Any suggested improvements?

Alex W:

- Our app very much promotes experiential cognition via its representations. This can be seen in the probing questions regarding career aspirations, the app asks the users, in order to custom tailor a career timeline to fit their goals. Having the career timeline spread out over four years in front of the user, promotes them to ask further questions as to whether their goals fit with their expectations and interests.

Ryan D:

- Our app heavily promotes reflective cognition via its representations as the questions it tries to help the user answer (what do I want to do in life?) are probably the most reflective questions a human can experience! This is mainly true during the Explore task. During the Create Plan task and Apply for Jobs task, again, a lot of heavy reflective cognition is required of the user through the app's representations as they figure out how they will want to plan their academic path along with what types of job postings they will want to apply to.

Xin Li:

- Our app promotes experiential cognition. The plan is created base on the information that user enters in.

Kun Xie:

- Yes, our app does promotes experiential cognition by generating questions to guide user create their plan step by step.

14. Semiotics: Sign meaning

- Does the app's signs convey the correct (i.e. designer intended) meanings?

- Any suggested improvements?

Alex W:

- At no point in navigating our app, was I confused with the signs used (though the text underneath them, aids in understanding).

Ryan D:

- The sign meanings used throughout the application follow very standardized signs used throughout software systems elsewhere and, thus, resulted in little to no confusion throughout my use of the app.

Xin Li:

- Yes. Buttons are designed well to let the user know their functions.

Kun Xie:

- Our app used proper signs to make sure users understand what does each button/function mean.

<u>15. Semiotics: Sign makeup</u>

- Does our app use a common type of sign (i.e. iconic, indexical, or symbolic)?

- Does the predominant type of sign make sense? Would another type be more appropriate?

Alex W:

- Our app mostly utilizes indexical signs (e.g. in career full descriptions, a clock, for commitment, dollar-bill, for average salary, and briefcase with magnifying glass, for number

of job postings).

- Using iconic signs instead would be bizarre (e.g. creating signs that physically resemble aspects such as “job commitment” in high fidelity) and using symbolic signs would be boring (e.g. just text).

Ryan D:

- Our app’s design employs mostly indexical signs throughout its interface and it completely made sense while using the application; the other types would have been inappropriate.

Xin Li:

- Indexical signs were used a lot in our app.

Kun Xie:

- We used indexical signs in our app, and they convey proper information.

4. The way the design is perceived:

16. Colour (UDP p. 48)

- Is it used effectively to attract attention, group elements, indicate meaning, and enhance aesthetics?

- Any suggested improvements?

Alex W:

- Overall our app’s colour scheme works very well (i.e. using desaturated colours for the most part, and saturated colours, like strong violet, to draw attention to important button controls and the timeline). Reduction of overload on eye processes was the goal here.

- A suggested improvement, could be to slightly lighten the shades of blue, and violet. Just in case they are having a subtle effect on a user's ability to use the app (i.e. the important functions would be just as noticeable, without the potential for sensory overload via oversaturation).

Kun Xie:

- We used a smooth and comfortable linear gradient colour as our theme colour to make sure the user feels comfortable when using our app.

Xin Li:

- We have a light color background and black font texts, forming a high contrast and make the user focus on the texts itself.

Ryan D:

- Our app uses muted colours in the background and high contrasting colours for foreground elements that are interactive. The aesthetic achieves is very modern and inviting.

17. Consistency (UDP p. 56)

- Is the app consistent as far as colour scheme, and user interface graphics (i.e. buttons, selectors, dropdown menus)?

- Any suggested improvements?

Alex W:

- The colour and design scheme remains constant throughout the app's core functions (i.e. Explore, Pinned Careers, My Plan). White and light grey adorn the information inputs, and outputs like career search results and action/submission items.

- A suggested improvement could be to vary the consistency a small amount, to aid the user in wayfinding and maintain engagement (i.e. not to bore them).

Xin Li:

- Our background and text font stay the same, give the app a good consistency of viewing.

Kun Xie:

- Our button, text and colour stays consistent when user is viewing different links and sections in our app, which prevent them from getting lost.

Ryan D:

- Our app's design uses consistent design language throughout its interface. Buttons are consistent and display of information (such as career types) are consistently designed using a Card view or List view throughout the application.

18. Layering (UDP p. 146)

- Does the app effectively organize information into related groups that manage complexity and reinforce relationships in the information?

- Any suggested improvements?

Alex W:

- Two-dimensional linear layering is present in both navigation bar (liner by function order) and when the career timeline is completed (linear by time sequence). The navigation bar reduces complexity by highlighting the main functions of the app (i.e. Explore, Pinned Careers, My Plan), and headlines respective sub-functions (Explore: searching for and learning about careers, Pinned Careers: further learning and comparing, My Plan: creating/ editing a career plan, and applying for jobs). It functions well, in this regard, due to its clarity.

- The timeline is organized by temporal sequence, which helps a user understand where they are in their plan. This was one of our earliest design ideas, and without this concept, the career plan could have become a disorganized jumble of career steps.

- Suggested Improvements, could be to offer hover over information for each step on the navigation bar that could clearly state what each functional step does.

Kun Xie:

- We used proper layers to group related information together, such as generated job search result to let user know what belongs to which function.

- Can add things like tooltip for better explanation

Xin Li:

- The information is grouped well. For example, the job users are interested in are pinned together for users to view later and the all the information about the plan is only shown in one tab.

Ryan D:

- Our app uses layering for the search results quite heavily where common items are grouped together. It works effectively and makes the design easier to navigate and read.

19. Mapping (UDP p. 152)

- Does the app effectively map controls to corresponding effects (i.e. do the effects correspond to your expectations when controls are interacted with, is there single-to-single mapping)?

- Any suggested improvements?

Alex W:

- Most of our app, involves single control-single effect mappings (i.e. for career choice selections, pinning, and exploring). This is simplicity aids in bridging the gulf of evaluation and execution in my mind.

- However, this single control-single effect mapping is not present in the add action/ submission item control function. A suggested improvement would be to separate the controls for “add action item” and “submission item” (as they are combined right now). Having separate controls for both these items would allow for increased flexibility (i.e. if a user had to edit a submission date) and ease in comprehension. Though it is worth mentioning that different coloured icons are used to highlight these separate functions.

Xin Li:

- Single to single mapping is used to direct the user.

Kun Xie:

- We mostly used single-to-single mapping to give users directly their desired result

- Can add more guide effects for combined “add action” and “submission item” function.

Ryan D:

- Our app makes use of the one-one mapping throughout the Explore task. However, during the Create Plan task, an improvement would be to incorporate more one-one mapping.

20. <u>Signal-to-Noise Ratio (UDP p. 224)</u>

- Does the app maximize “signal” (i.e. little ambiguity, no unnecessary elements, minimized expression of necessary elements, simple designs)?

- Any suggested improvements?

Alex W:

- Extraneous elements are effectively null in our app (besides some background iconic styling). The design is overall content and functionality driven in my mind. This results in an interface that is very easy to use, and I have never been confused as to where to proceed next (my own biases aside).

- It is very hard to find a suggested improvement for this heuristic, as the maximized signal-to-noise ratio of our app, is one of its strongest design features.

Xin Li:

- We maximize the signal by organizing information well and using light color background.

Kun Xie:

- Our app is designed to show user the important and wanted contents, user can easily find functions and information they want without being disturbed by unnecessary elements.

Ryan D:

- Our app does an excellent job of filtering out the noise from the signal by delivering only enough relevant information to the user to help them decide on what to do next. This is an important design feature of our app as it would be easy to design an app that overwhelms the user with information on careers and planning details.

5. The way users learn the system:

21. Forgiveness (UDP p. 104)

- Does our app truly avoid errors and minimize the negative consequences of errors when they do occur (i.e. is this done through good affordances, reversibility of actions, safety nets, confirmations, warnings, help functions)?

- What additional forgiveness measures would you suggest?

Alex W:

- Our app clearly marks input features mainly through the use of bold highlighted text and icons that signify the radio button and drop-down selectors in the menus of the My Plan function. Also when adding action items to their timeline, a user can tell if the required education/skills/experience has already been accounted for on their timeline. These are all examples of good affordances.

- Furthermore, reversibility of actions is fully allowed when one pins and unpins a career, and edits/deletes action and submission items.

- User error is pretty much effectively minimized through our app's design, the only problems that could arise is if a user does not take the time to fully read through the text related to menu selections. However in that case the results of their slip would become apparent as their would be gaps or unexpected components to their career timeline. Reversibility of actions could be further improved, if the "add action item" and "submission item" control functions would be separated, since they could be edited separately then (see above "Mapping" heuristic).

Kun Xie:

- We used enough explanation text to let user know what is what to avoid unwanted actions. Also, there is always a "x" or close button to let user jump out from unintended actions.

Xin Li:

- Our app is design in the way to maximize forgiveness, so users will be fine using it.

Ryan D

- Our app has excellent forgiveness incorporated into the design which is especially highlighted in the Explore task - when users pin a career, they can immediately unpin a mistakenly pinned career without even moving their cursor or viewport. Although we were

not able to finish the Edit feature in the Create Plan for the prototype, it is implied through our design document that the ability to edit and delete Action Items added to the plan's timeline would allow for a high level of forgiveness.

22. Garbage In-Garbage Out (UDP p. 112)

- Does our app limit problems of input (especially concerning problems of type) and promote correct input?

- Any suggested improvements or additional measures?

Alex W:

- This heuristic ties directly to the above "Forgiveness" heuristic. The primary way our app limits problems of type (especially in the initial My Plan menu, as well as action item, and submission item menus) is through highlighting to improve the visibility of these choices. And they generally work, if the user is making their career choices carefully.

- Some suggestions for improvement could include a function to increase the font size, though it may be superfluous in the face of browser zoom capabilities.

Xin Li:

- The input the app asked is detailed and not ambiguous.

Kun Xie:

- As described in above sections, we use explanation text with bold or highlight fonts to guide user what to input

- Can add an input example with results.

Ryan D

- Our application uses clear and numbered instructions to help guide the user through the app.

- One area to improve is to give the user a preview of what they can achieve with the Create Plan feature to help them better plan properly.

23. Performance Load (UDP p. 178)

- Does our app truly ease the cognitive load of searching for, planning, and applying for careers? During this evaluation, was there any point where you had to think about (or even write about) information related to the app, externally?

- Any suggested improvements?

Alex W:

- The app definitely does ease the cognitive load as far as searching for, planning, and applying for careers. There was not a point where I was writing anything down (when testing a mock career choice) as I was assuming the app notifies me in some fashion (via email?) when submission items and action items require attention.

- As a suggestion, I recommend the addition of an “export” function, where I can save career information (especially information stored in the My Plan timeline) outside of the web app.

Xin Li:

- Our app eases the cognitive load for most of the functions.

Kun Xie:

- Our app does ease the cognitive load for our basic functions

- An “Upload” function to save wanted information to google drive or icloud

Ryan D

- Our app does a good job of helping the user ease their cognitive load of the overwhelming number of career options that may exist for them and also, how to go about planning their academic path during their years at their educational institution. I never felt the need to utilize resources outside of the app to facilitate in my goals while using the app.

24. Progressive Disclosure (UDP p. 188)

- Does our app’s progressive disclosure implementation, reduce information overload?

- Any suggested improvements?

Alex W:

- Our app makes good use of progressive disclosure, which can be seen especially in the “Explore” careers function (e.g. career results, initially are not returned with an overwhelming amount of background information).

- A possible suggestion I was going to make, was to implement progressive disclosure on the career timeline action and submission item menus (i.e. have the menu grow after each category of menu selection is completed), as to not overwhelm the user with options, and minimize erroneous input. Though this could be undesirable, especially if it makes editing action and submission items annoying (i.e. a user would have to go through the progressive disclosure sequence, each time they edit an item).

Kun Xie:

- The “View checklist” is a good example of progressive disclosure. We don’t present the checklist directly to occupy space, but we allow user to check their progress anytime by

clicking view checklist button.

Xin Li:

- Yes. Especially the checklist feature, it isolate all the unnecessary information and reduce the information overload.

Ryan D:

- Yes, our app does a great job at this by initially only providing enough information to peek the user's interest enough to decide if they want to navigate to the next area or deeper within the existing element.

25. Readability (UDP p. 198)

- Does our app truly use basic english and minimize jargon/complex words? What is the Fry's Readability Graph score?

- Any suggested improvements?

Alex W:

- Our app does use basic english and minimal jargon when instructing a user on its functions. However, some complex words (and related jargon), do appear in career full descriptions. This is tied to specific occupations (e.g. computer science job descriptions do contain industry jargon). Becuase of this, our Fry's Readability Graph score (acquired here: http://www.readabilityformulas.com/free-fry-graph-test.php) is quite high (almost US Grade 15, which indicates post-secondar reading levels).

- Suggestions for improving the clarity and simplicity of the writing, could come from a targeted high school focus group (if we had the time and ethics approval).

Kun Xie:

- Yes, our app uses natural english and minimal jargon to make sure our functionalities are easy to understand.

- Maybe a language translation feature is useful for more diverse usage.

Xin Li:

- Yes. The language is natural and maybe multi-language support for user interface would be better.

Ryan D:

- Our app uses minimal jargon and relies on well-known terms and phrases that are commonly used to describe jobs, careers, industries and education-related concepts.

6. The general usability of the system:

26. Confirmation (UDP p. 54)

- Does our app use confirmation mechanisms for user inputs?

- Any suggested improvements beyond the implemented two-step process?

Alex W:

- Our app does include “two-step operation” for users to confirm their inputs. This is implemented primarily in selecting initial options for the career plan, as well as options in the action and submission item menus, in the form of a confirmation button at the bottom of the menu.

- This could be further improved by making a confirmation pop-up (e.g. stating “are you sure you made the right selections”) appear when selecting options for the “add action item” menu.

Xin Li:

- Our app used confirmation alert boxes to ask user for confirmation

Kun Xie:

- Yes, in my plan section, we have popup window to make conformation for user input

- Can make confirmation in edit qualification stage or “unpin” function.

Ryan D:

- Our app utilizes confirmation notification pop ups to aid in confirmation quite effectively and elegantly.

27. Control (UDP p. 64)

- Has our approach to control (i.e. a one size fits all approach) been properly reasoned, and does it accommodate possible future expansion of functionality for the app (e.g. advanced functions)?

Alex W:

- Our app’s core functions are simplified to the point where it offers the same level of control for beginner, intermediate, and experienced users. I had the idea in mind, that our app is in certain regards akin to an ATM machine as far as frequency of interaction with a user. As (in my mind) users will utilize the app quite frequently at the start, but once they select, and finalize a career plan, they will not interact with the app often, until they apply for jobs with it. So there is not much room for advanced control functionality.

- An idea for advanced controls, would be if the app was opened up for the use of high-school guidance counselors. In that case a guidance counselor mode could be set up, which would include macro level controls (e.g. to oversee and guide student plans).

Kun Xie:

- Our app is design with easy and reasonable control, which allows targeted users to use it without problem.

- For advanced control, maybe it is a good idea to design different contents and functionality display for high school and university student.

Xin Li:

- I am of the same thoughts, regarding the one size fits all control scheme.

Ryan D:

- Our app does a great job of designing a one-size fits all design. It might benefit from having future functionality to be designed for specifically highschool students versus university students versus graduate students versus academic/career counsellors etc.

28. Errors (UDP p. 82)

- Does our app actually minimize (or even eliminate) the possibility of user error? Did you make an error or forsee a potential error, while testing out the app?

- Any suggested improvements?

Alex W:

- Overall the app prevented me from making input errors. However my bias (as part of the app's design team) makes it harder for me to make slips, since due to familiarity (i.e. I cannot put myself in the shoes of a first time user).

- I will reiterate my above suggestion, of increasing the font size (to improve visibility) for the options in the Education, Skills, and Experience categories, when adding action items.

Kun Xie:

- The way our app designed actually minimize the error cases, user are always in a “safe” situation.

- Can add some tooltip or help in job search or timeline part just in case if anything goes wrong.

Xin Li:

- I did not make any errors, when testing the app. I think it sufficiently prevents errors, if the user is taking the information they are entering into the app, seriously.

Ryan D:

- It was a bit difficult to evaluate this criterion due to our high-level of familiarity of th app since we designed it but I felt that prevention of errors were implemented in the design fairly well.

29. Fitts’ Law (UDP p. 98)

- Does our app properly account for this principle? While testing the app did you have trouble navigating a the cursor over to a desired control or selection? Did you have trouble clicking on a desired control or selection?

- Any suggested improvements?

Alex W:

- The use of buttons (e.g. for “Explore”, “Pin This”, “Add Action Item”, “Add to Timeline”) were mostly easy to navigate to via cursor (though this could depend on screen resolution).

- Making the buttons (perhaps 25%) larger, and assuring that they scale to screen resolution would help to avoid the pitfalls of Fitts' Law.

Kun Xie:

- Yes, we followed Fitts' law during our app implementation, and our buttons and links work fine and are easy to navigate.

Xin Li:

- I never had any trouble clicking on any button or toggle, there were of sufficient size.

Ryan D:

- The buttons and interactive components had a very high-contrasting colour versus the background and surrounding elements so it made selecting these interactive elements very easy.

30. Wayfinding (UDP p. 260)

- Does our app's design, truly orientate a user and facilitate route monitoring, and destination recognition.

- Any suggested improvements or concepts for a better wayfinding system?

Alex W:

- At no point during the evaluation of our app was I lost as per what functional stage I was in. The navigation bar was helpful in this regard.

Kun Xie:

- The navigation bar, button icons and explanation text can guide user easily to find information and functionality, so it is hard to get lost during usage.

- Add some animation on homepage before user inputs to tell user what will he/she be able to do with each button or tabs. (like: a small hand shows up pointing every function)

Xin Li:

- Yes, it accomplishes this fine.

Ryan D:

- Our 3 tab system greatly incorporates Wayfinding at a very effective level where the order of the tabs from left to right imply a very intuitive direction of workflow. Scrolling from top to bottom to go through the search results was also as natural as one could design the Explore task of the application along with the Create Plan's timeline.

Blank Page for Marking

Visual Breakdown of Executive Summary

CS 4474 Team 4

Start

Task 1: Searching for & Choosing from Careers/Jobs that are related to what user is studying

Direction of Task-flow over Time

Step 1	***Input by User:***	*Inputs area they are studying or degree they are pursuing*
	Output by App:	App displays 2 things in return: 1) All Career Choices related to previous input —> *Each choice is click-able* 2) A summary of all the career choices, industries, average salaries, etc as an overview of what their degree/area of study yields
Step 2	***Input by User:***	*User clicks on a Career Choice they want to know more information about*
	Output by App:	App displays a description of the selected career, the industry the career is in, and other relevant career information such as salary averages and range, usual benefits and available job projections —> *User can select a career choice as a "Career They Are Interested In"*
Step 3	***Input by User:***	*Select a listed Career Choice as a "Career They are Interested In"*
	Output by App:	App will add it to their list of careers they are interested in —> *the app then allows the user to create a plan for this career choice*
Step 4	***Input by User:***	*Selects "Create a Plan" for a chosen career*
	Output by App:	App will guide user to create a plan that will result in user becoming fully qualified for applying for chosen job by the time they graduate **(Task 2 + 3)**

Task Finished

Task 2: Creating a Plan that will result in user becoming fully qualified for applying for chosen job by the time they graduate

+

Task 3: Tracking user's progress of acquiring all qualifications in time for graduation

Start

Direction of Task-flow over Time

Task 2 Step 1	**Output by App:**	App requires user to fill out the following information before guiding them to create a plan: • Year they've started or will start their degree/program • Expected graduation date (can automatically generate this info if user inputs how long the degree/program is • A Yes or No input of whether their program offers internship options or not • A Yes or No input on if they'll be doing their degree part-time or full-time • A Yes or No input on if they have summer breaks
	Input by User:	*Fills out relevant information listed above and submits their information*
Step 2	**Output by App:**	App displays the following: • Generate List of Required Qualifications • Breaks down list of Required Qualifications in to categories and may offer various details of information on each item
	Input by User:	• *User can edit/add to generated list of required qualifications* • *User adds items from list of **Qualification Items** on to **Timeline** to plan when to acquire each Qualification Item by (in the form of an Action)*
Step 3	**Output by App:**	App highlights in red or whatever colour to indicate remaining items that still need to be planned —> user completes planning everything else
	Input by User:	Time passes by and user's point in time in the real word starts to approach a "Take Action" deadline as saved in the Plan's Timeline
Task 3 Step 1	**Output by App:**	App sends notification to user via email to notify them of a Required Action that's part of their plan
	Input by User:	*After further passage of time and user acquires a Qualification Item, user checks it off*
	Output by App:	App updates itself to show newly acquired Qualification Item.
Step 2	***Input by User:***	*Time passes by and user's point in time in the real word starts to approach Graduation. User clicks on "Apply for Jobs"*
	Output by App:	App shows search results + links from Monster, LinkedIn + other Job Search Platforms with available developer APIs —> User applies to desired postings

Tasks Finished

Task 2 **Step 2**

Check-List / Progress Tracking

Potential **Required** + **Suggested** "Qualifications"

Required:

Education:

- *Degree(s) & Certificates/Training*

Skills:

- *Technical Skills*
- *Non-Technical Skills*

Experience:

- *Unpaid experience (volunteer)*
- *Paid experience (internship)*
- *Portfolio*
- *References*

Suggested / Extra-curricular:

- *Clubs & Organizations*
- *Volunteer Work*
- *Awards & Accomplishments*
- *Personal Accomplishments*

***Add** each "**Qualification Item**" from **Check-List** on to **Timeline** to plan when to acquire each one by*

Mock Example of a 3 Year Program's Timeline
+ Modelled Plan of When to Acquire Mock Qualifications by

You Are Here

Year 1 — 2 Semesters — Summer

Take Action:
Apply for unpaid or paid work experience

View Details

Year 2 — 2 Semesters — Summer

Take Action:
- Apply for paid work experience
- Assemble a portfolio
- Apply for certificate / training

View Details

Click on "View Details"

View Details

Take Action by: Jan-April, Yr 1
Apply for unpaid or paid work experience

Purpose is to Acquire:
- paid/unpaid work experience
- technical + non-technical skills

Complete Goal by:
End of summer, Yr 1

Year 3 — 2 Semesters

Graduation

Apply for Jobs

Part of **Task 2** **Step 2+3** & **Task 3** **Step 1**

End point of application has been reached and all user's goals have been achieved.

User fills out Online Job Posting from search results

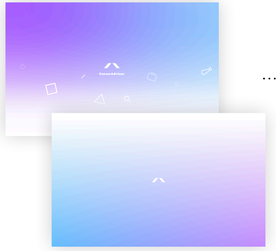
Task 3 **Step 2**

Click on "Apply for Jobs"

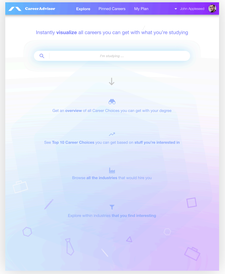
CS 4474 Team 4
Breakdown of Tasks

CS 4474 Team 4 Navigational Map

Task 1

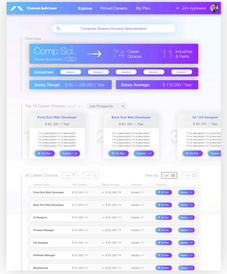


Landing Page + Loading Screen



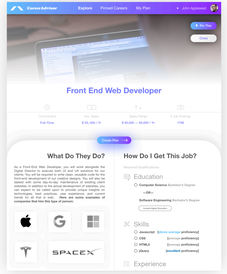
Search Bar (Home Page)
The first screen thee user sees has a search bar and explanation of how they can use this app to serve their goals.

Input
User has input their degree / program



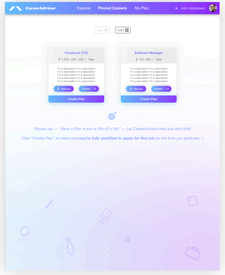
Search Results
User can see an overview of the relevant careers information relating to their inputted degree. They can see the top 10 careers and all career choices at the bottom. Users are able to **pin the careers** they are interested in or click **explore** to learn more about the career. In addition, they can select different view modes to browse career results.

Navigation
User has clicked on **"Explore"** for Front End Web Developer

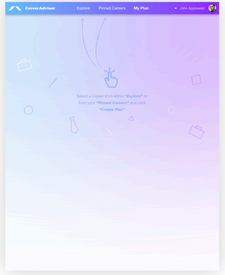


Exploring a Career
When the user clicks the "Explore" button, they will be able to see the detailed information of the selected career. The user can **Pin This** to their Pinned Careers for future planning, **Go Back** to view other careers from the previous search results or click on **Create Plan.**

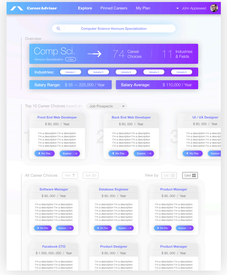
Navigation:
User has clicked on "Pinned Careers"



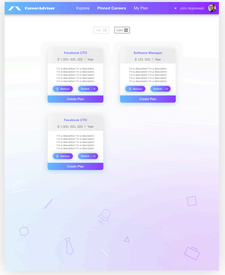
Navigation:
User has clicked on "Create Careers"



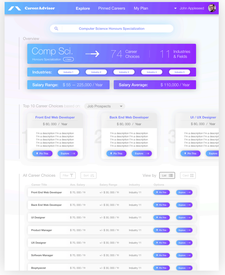
View-Mode Change:
User has changed to "Card View"



Navigation:
User has clicked "Pin This"



Navigation:
User has clicked on "Go Back"

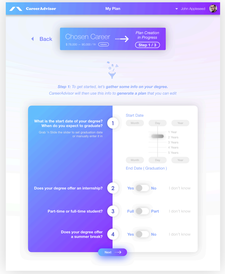


Pinned Careers
In the Pinned Careers tab, users can see all the careers they have "pinned" before, where they can choose to **delete** a unwanted career, **explore** the career in detail again or start to **create a plan** for the career. Again, the user can view the careers in both list and card mode.

Navigation
User has clicked on **"Create Plan"** for Front End Web Developer

Navigation
User has clicked on **"Create Plan"** for Front End Web Developer

Task 2



Create Plan (Step 1)
In **Step 1** of the Create Plan tab, the user will input their specific details of their program or degree. Then, the user can click **next** to proceed to Step 2 or **cancel** to go back.

Input
User has input their program information and clicked **next**

Create Plan (Step 2)
In **Step 2** of the Create Plan tab, the user will see the detailed qualification requirements of the selected career. The user will be able to **edit** the list to add specific requirements that they know they need to accomplish in order to be fully qualified for that career. Or, the user can skip this step to the **next** step by accepting the auto generated list.

Navigation
User has clicked **next**

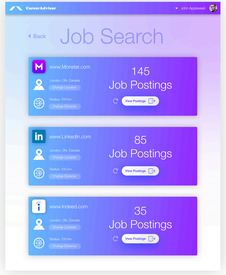
Next PDF Slide

Task 2 + 3

Create Plan (Step 3)
In **Step 3** of the Create Plan tab, the user will finalize their plan by planning on when to acquire all the necessary qualifications for their chosen career. The interface will display a Timeline to allow them to visually plan their goals.

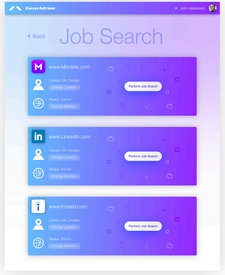
Since Step 3 will serve as the main complex task aided by CareerAdvisor, the detailed interactions of Step 3 are quite large. Thus, it is described in the next PDF slide.

Task 3



End point of application has been reached and all user's goals have been achieved.
CareerAdvisor will display the number of job postings and provide a link to the search results that will open in the user's browser **(View Postings)**. From there, it is up to the user to fill out applications and submit them to their desired companies.

Input
User has filled in location and distance for search and clicked Perform Job Search



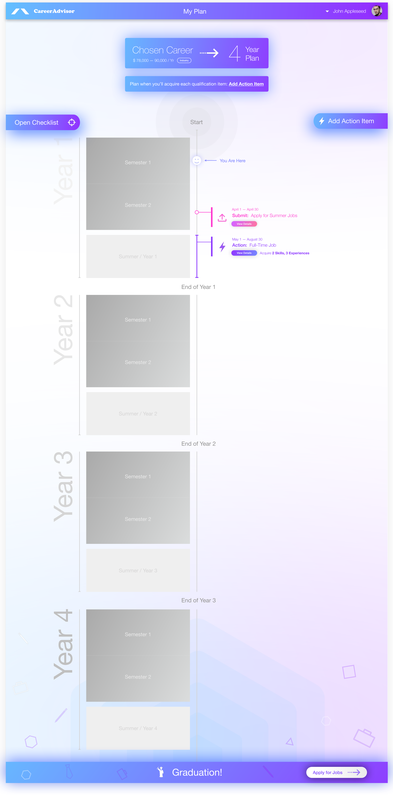
Job Search
Searching and applying for jobs is the final goal of the user and can be done once they are nearing graduation. Job search platforms that have search APIs available will be hooked in to CareerAdvisor (such as Monster, Indeed, etc). The user can then input search parameters (other than their career) and **Perform a Job Search** or go **back.**

Navigation
User has graduated and are ready to apply for jobs. User has clicked on **"Apply for Jobs"** in the previous screen

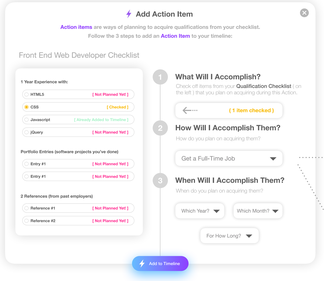
Create Plan (Step 3) — The Timeline

User can view their next 4 years and plan when to acquire Qualifications.

User clicks on "Add Action Items" as a first initial step. Follow the dotted arrows to follow user's work flow.



Popup: User is creating an **Action Item** to be added to the Timeline. User has checked off only 1 Qualification Item so far.



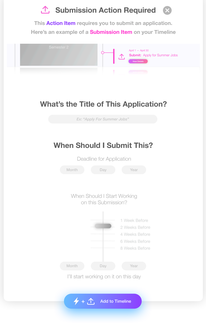
Edit

Drop-Down Menu

- Get a Full-Time Job
- Get a Part-Time Job
- Take a University Course
- Take an Online Course
- Start a Personal Project
- Start a Portfolio Entry

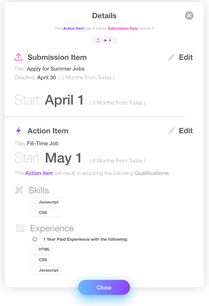
Some **Action Items** require a preceding "**Submission**" action to be performed before user can successfully perform action.

As such, user will be prompted to create a **Submission Action.**



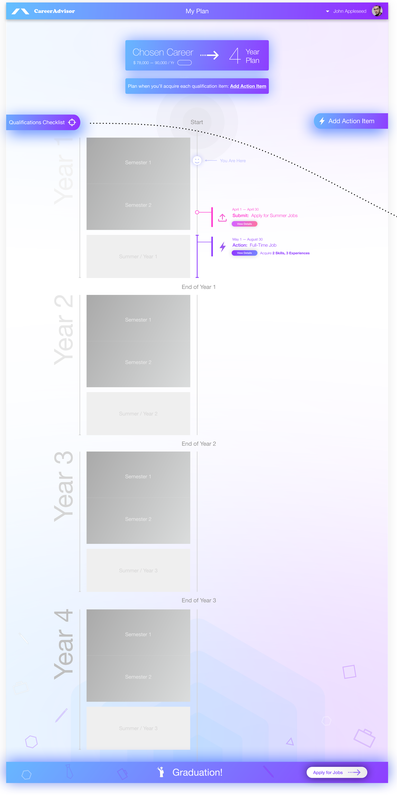
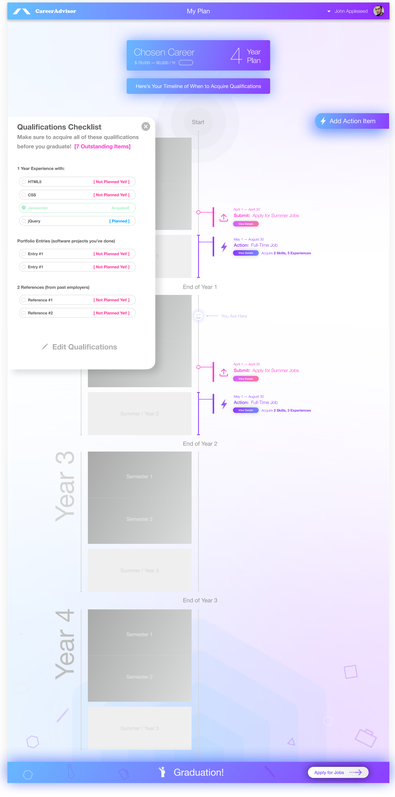
May 1 — August 30
Action: Full-Time Job
Not Started
Acquire 2 Skills, 3 Experiences

Popup: User can view item details and choose to edit item details



Action Items can be **edited** from the View Details popup

Edit



User can access a birds-eye view of their progress on the acquisition of qualifications from their Qualifications Checklist