

CS Capstone Design

Alpha Prototype Demo Grading Sheet (100 pts)

TEAM: ECoders

Overview: The purpose of the Alpha Prototype Demo is to clearly demonstrate the extent to which all core user flows envisioned for the product are supported by the current implementation. The flow of the demo is very natural: you simply introduce each of the major usage scenarios, and then follow through each of them, just as an end-user would in using the product. Grading is based on how completely the current product supports all key functional aspects within a coherent, realistic user flow. Interface refinement, clunkiness, and aesthetics should be ignored for now; the focus is simply on functional ability to complete the user flow.

This template is fleshed out by the team, approved by the team mentor, and brought to demo as a grading sheet.

Overview of major product requirements

Outlined below are the major, must-have functional requirements. These are taken from the Requirements Specification document, version 1.3. These are the high-level requirements we are considering for this demonstration.

F1.00 Projects (also named Task Agreements) shall be tracked through their six phases:
Proposal, Drafting, Acceptance, Tracking, Termination, Reporting

F2.00 Modifications to a project shall be tracked

F3.00 Projects shall be terminated based on the following requirements.

F4.00 Data coming into and out of the system shall be validated for data quality and assurance

F5.00 Data from projects shall have the ability to be download and/or exported

F6.00 The web application shall be made up of user groups to provide access to the system, customized functionality, and the separation of permissions

F8.00 The public shall be allowed to view the following information as explained below:

F9.00 The website and any of its software shall implement software accessibility for all users, regardless of disability, impairment, etc.

F10.00 Our web application shall make it possible to import data directly from user uploaded documents as specified below:

Note: these are the nine functional requirements we would like to demonstrate and have implemented, but not yet finalized for this Alpha Prototype Demo.

Overview of major product use cases

Based on the Requirements document and subsequent development discussions with your client and mentor, briefly describe each of the key use cases for your product:

UC1: CPCESU Staff - Create Two Projects: Scratch vs. Existing Users/Groups

Requirements Satisfied: **F1, F2, F3, F4, F6, F9**

- CPCESU Staff will have the ability to create new projects while also having the functionality to update projects with existing federal agencies, partners, etc.

UC2: CPCESU Staff - Create Project from PDF and Optical Character Recognition

Requirements Satisfied: **F1, F2, F3, F4, F6, F9, F10**

- CPCESU Staff can upload PDFs and automatically create new projects with the respective PDF information.

UC3: CPCESU Staff - Modify Project in “Executed” State

Requirements Satisfied: **F1, F2, F4, F6, F9**

- CPCESU employees have the ability to add/delete modifications to current projects.

UC4: CPCESU Staff and Anonymous User - Different Site Views

Requirements Satisfied: **F5, F6, F8, F9**

- All user groups will have varying permissions and views, tailored to who they are in the tiered system. For example, public / anonymous users have the least access and permission to data whereas CPCESU employees have access to everything in the system.

UC5: CPCESU Staff - Notifications

Requirements Satisfied: **F1, F2, F3, F4, F6**

- CPCESU Staff will be given a notification system to alert staff members of pending project modifications or project initializations. Outside organizations can request modifications to their respective projects, which notifies CPCESU Staff.

User Flows: Detailed walk-through for each use case:

In this section, we outline the demonstrations of each use case that we have prepared, giving a step-by-step outline of the user flow that would be followed by a real user for that use case.

Use case 1: CPCESU Staff - Create Two Projects: Scratch vs. Existing Users/Groups

User Flow: Step by step overview of user interactions with product

1. Login with given CPCESU credentials
2. Navigate to Create Project tab (Projects → Create Project)
3. Fill out required project fields on Project Form (Create Project → Project Form)

- a. Enter/Don't Enter values to test form validation
 - b. Submit
4. Verify Details
5. Repeat steps 1-5 for creation of second project
 - a. Use the existing Federal Agency, Partner, etc. profiles you have created for the first project for the second projects.
6. Check Project tab for project dashboard (Projects → Project List)

Evaluation and Comments:

✓ Convincingly demoed each of listed challenges?

✓ Other evaluative comments:

Use case 2: CPCESU Staff - Create Project from PDF and Optical Character Recognition

User Flow: Step by step overview of user interactions with product

1. Open create project page
2. Put a valid Grant and Cooperative Agreement PDF into upload folder
3. Wait no more than 30 seconds
4. Form fields will be updated with some select data from the PDF
5. User will repeat these steps with a different Grant and Cooperative Agreement PDF on the same form, the form will update with the new information
6. User will upload an invalid PDF, no new data will be filled. An error will be displayed

Evaluation and Comments:

✓ Convincingly demoed each of listed challenges?

✓ Other evaluative comments:

Use case 3: CPCESU Staff - Modify Project in “Executed” State

User Flow: Step by step overview of user interactions with product

1. Make sure user is logged in as CPCESU Staff
2. Next, go to Project List (Projects → Project List)
3. Click on the “View More” button for the project to-be-edited
 - a. Project **must** be in a “executed” state
4. Click on “Add Modification” link
5. Fill in fields as necessary
 - a. Fields include:
 - i. Time (end date)
 - ii. Money (new budget OR adjusted value +/-)
 - iii. Personel (change in contact information)
6. Submit modification
7. View results in the details page
 - a. After submitting the modification, there should be an automatic redirect to the project’s details page
 - b. The details should adjust to the new information and the modification should be listed on the page with a link to view the revision

Evaluation and Comments:

✓ Convincingly demo’d each of listed challenges?

✓ Other evaluative comments:

Use Case 4: CPCESU Staff and Anonymous User - Different Site Views

User Flow: Step by step overview of user interactions with product

1. First, go to the web server’s domain name.
2. If logged in as CPCESU staff member, follow these steps:
 - a. Note the different authenticated user navigation menu (Projects should have a “Create Project” link, Notifications and Admin Site should also be visible)
 - b. Go to “Projects” then “Project List” then select a project
 - i. Click on the “View More” button

- c. On the project's details page, there should be a "Add Modification" and "Export to CSV" button
 - i. Click one to confirm that you are a CPCEU staff and have access to these views
- d. Next, logout and return to the home page.
- e. If already done Step 3, go to Step 4. Otherwise, continue to Step 3.
- 3. If not logged in (anonymous user), follow these steps:
 - a. Note the different navigation menu (Projects should only be a single link to the "Public Projects List" view, no access to notifications or the admin site).
 - b. Go to "Projects" then select a project
 - i. Click on the "View More" button
 - c. On the project's details page, there should only be a dummy "Request Information" button and the public details
 - i. There should not be things such as attachments, modifications, or revision history
 - d. Next, click on the "Login" link at the top right of the page.
 - e. Log in with CPCEU credentials.
 - f. If already completed Step 2, go to Step 4. Otherwise, continue to Step 2.
- 4. Completed user flow.

Evaluation and Comments:

✓ Convincingly demoed each of listed challenges?

✓ Other evaluative comments:

Use Case 5: CPCEU Staff - Notifications

User Flow: Step by step overview of user interactions with product

1. First, a project is created
2. Then move into the admin-site, and edit the project so it was created 30+ days in the past
3. Wait 15 to 30 seconds for the background task to activate and create a notification
4. Notification will be displayed in the notification pull down bar
5. Notification will be available on any page in the website, except the admin-site
6. When logged out, the notification will not be displayed
7. When the notification is clicked on, the user is moved to the project details page of the notification

8. When the user inspects the notification again, the notification will no longer be in the drop-down menu

Evaluation and Comments:

- ✓ Convincingly demoed each of listed challenges?

- ✓ Other evaluative comments:

Known shortcomings: Functionality still deficient/missing:

If there were challenges you listed earlier that were *not* covered by a demo, list here. This will hopefully be a short list...but better to be clear about where you are. If you have items here, you could list (if applicable) any pending plans/schedule to get this implemented.

Project Dashboard

While our client was impressed by the current project page, we admit that it is not the best for managing projects through their various states. Thus, we would like to design an elegant solution for the project management system

User Settings

A page to allow the user to change their password and make minor modifications to their account. Also include a page for People Management for CPCEUSU.

Public Searching of Projects

This will be easy to implement, as the code for has already been written as the project index. We just need to add some light logic to prevent sensitive data from being shown and to change the user group of how can see the search.

Documentation

A non crucial app, as the website will be designed to be approachable by anyone, but the expert information contained in it will hopefully make the product indispensable between administration changes in the CPCEUSU. Will also contain information regarding the code base as well.