

## ECODers

Project: *Colorado Plateau Cooperative Ecosystem Studies Unit (CPCESU) Project Management System*

---

# Requirements Specification

---

**Overview:** The purpose of this document is to list out the requirements and other criteria that will determine if we have successfully delivered a successful product that solves out client's problem

**Team Members:**

Colton Nunley  
Joseph Remy, Jr.  
Jasque Saydyk  
Ana Paula Chaves Steinmacher - *Mentor*

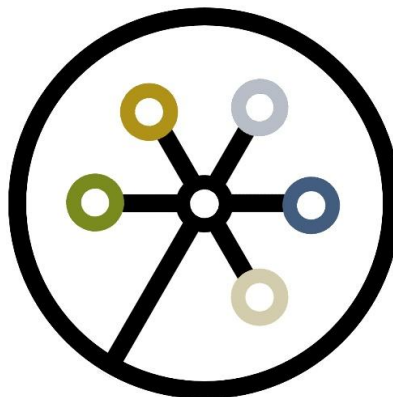
**Clients:**

Dr. Todd Chaudhry  
Laurie Thom  
Rebecca Dyball - *Intern*

Northern Arizona University  
School of Informatics, Computing, and Cyber Systems

Cooperative Ecosystem Studies Unit  
Colorado Plateau

November 26, 2018



**ECODers**



# Table of Contents

<b>Table of Contents</b>	<b>3</b>
<b>1. Introduction</b>	<b>5</b>
<b>2. Problem Statement</b>	<b>7</b>
<b>3. Solution Vision - Summit</b>	<b>10</b>
<b>4. Project Requirements</b>	<b>11</b>
User Requirements	11
Functional Requirements	12
Non-functional Requirements	23
<b>5. Process for Requirements Modification</b>	<b>27</b>
<b>6. Potential Risks</b>	<b>28</b>
<b>7. Project Plan</b>	<b>30</b>
<b>8. Conclusion</b>	<b>33</b>
<b>9. Appendices</b>	<b>35</b>
Glossary	35
Attachments	35



# 1. Introduction

Preserving nature and our past requires a wide variety of work and information to identify the problems, develop solutions, and gain insight into how our world used to be and how it is changing. This is done with a wide variety of studies. From investigating archeology sites to conducting zoological surveys, all of these projects help preserve the nature and history of the American Southwest, allowing us to learn more about the impacts we have on our fragile ecosystem.

The Colorado Plateau Cooperative Ecosystem Studies Unit (CPCESU) is a government agency which organizes federal agencies and Native, state, and local governments with non-governmental organizations and universities to complete a variety of these conservation projects. To do this, the CPCESU gets dozens of project proposals, hundreds of modifications, and millions of dollars each year to setup, organize, track, and archive these projects. Since their founding, they have used an off-the-cuff approach to project management using email, Microsoft Excel, a Microsoft Access database, and a shared file system drive.

Due to the ad hoc nature of the forms and data storage, the CPCESU staff are having to spend up to half of their time at work to track these projects. If the number of projects were to double, the CPCESU staff would find themselves overwhelmed and have no time to spend on their other work for the organization. In addition to this, there is no defined structure for data entry into the database, leading to partially completed rows and a lack of consistent, vital information the organization needs to renew its charter. There is no way for project leaders and organizations to modify their contracts to extend the timeline and allocate more funds without emailing or calling the CPCESU, and the CPCESU staff need to be able to search their dataset and export statistics they need to report to inquiring organizations, NAU, and the CPCESU Director. Lastly, certain parts of CPCESU projects depend on budget spreadsheets, approval documents, final reports and more, which are not directly linked to projects and need to be found manually in a massive file system.

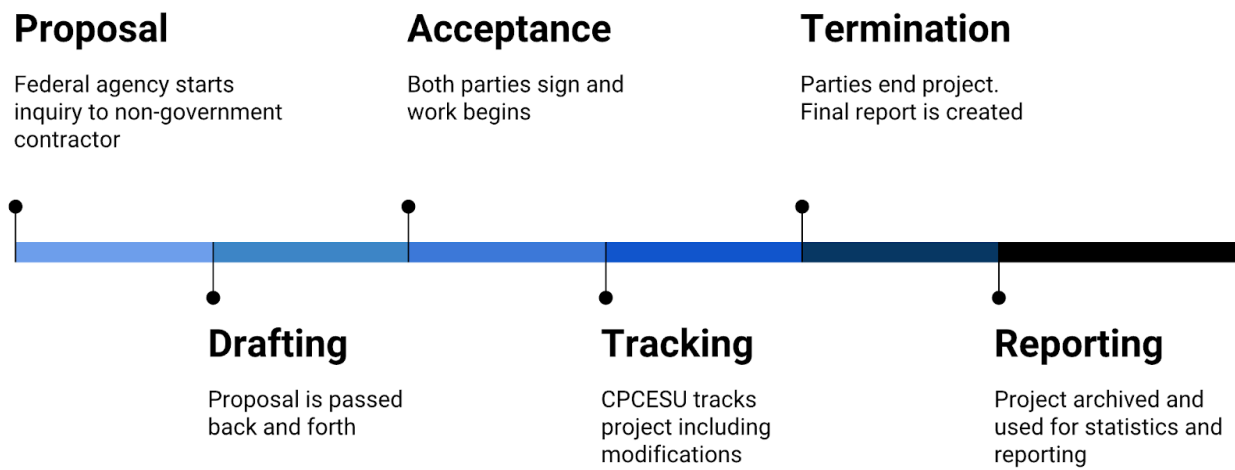
ECODers is a senior Computer Science capstone group formed under the School of Informatics, Computing, and Cyber Systems at Northern Arizona University. Team members include Joseph Remy, Jr. (Team Lead), Colton Nunley, and Jasque Saydyk and the Team Mentor is Ana Paula Chaves Steinmacher. We were tasked by Dr. Todd

Chaudhry and Laurie Thom with developing a website and database solution for the CPCEU with three essential components: a project management system for the CPCEU; a website for showcasing public projects and information; and a system allowing organizations to develop, submit, and track projects.

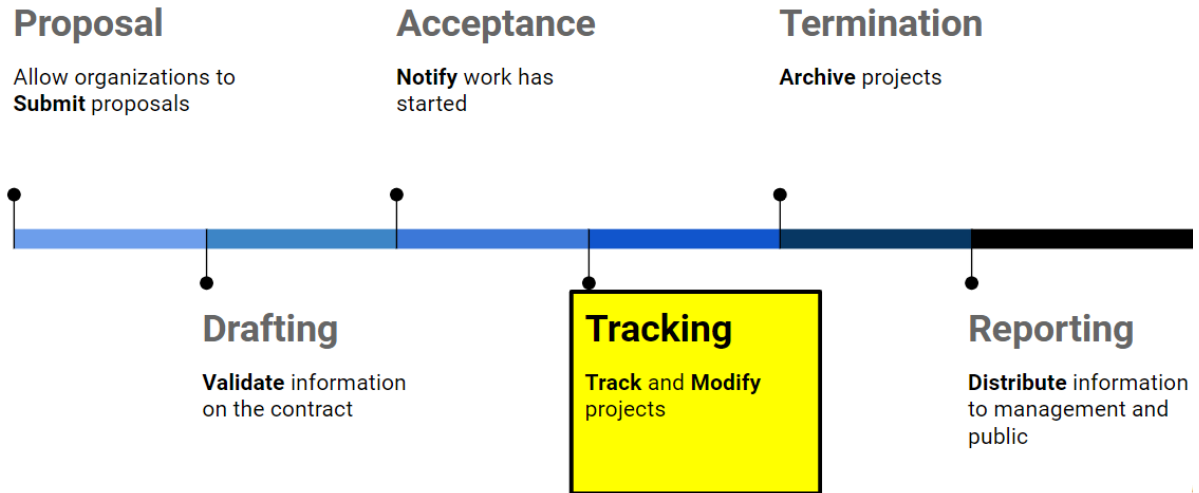
This software requirements specification begins by detailing out our client's problem and our proposed solution to this problem. After this, we detail out every requirement for the project, beginning with the high level requirements working down to the low level requirements for the system. Each requirement has a description as to what it entails and some criteria to verify when the requirement has been fulfilled. This is followed by a list of the potential risks with the project, their likelihood, and how we will go about mitigating them. Lastly, we note our project plan for the remainder of the project, detailing our expected milestones for the project.

## 2. Problem Statement

The CPCESU currently uses a combination of various products to create, track, and close projects within their organization. They use email and phone calls to communicate, create, and draft various projects, an Excel spreadsheet to track projects through their various phases, a Microsoft Access database to archive the projects and keep track of modifications to projects, and a shared file system to store the files of the CPCESU as they draft and archive the final reports of the projects.



As for the CPCESUs workflow, every project begins as a proposal from an organization, typically some form of government entity. The project is then assigned a Principal Investigator, who negotiates with the organization and the CPCESU as they draft the Task Agreement for the project, which is a contract for how the project is conducted, funded, and reported. When the Principal Investigator and the organization both sign the contract, the project is then accepted, and enters the tracking phase. During the tracking phase, projects are modified by organizations, which could be anything with the project, but is mainly funding, personel, or time related. When a final report for a project has been completed, defunded, or the project has lasted for more than five years, then the project is closed, entering the termination phase as it is archived into the CPCESU database. While archived, the CPCESU may need to access the project's data again to report on it, give it to a requesting agency, or to distribute some select information to the public.



The issue with the CPCESUs workflow is with the tracking phase, as they receive hundreds of modifications for projects each year that modifying the budget, time, or anything else regarding the projects, that they are unable to track effectively. They already spend up to half their time at work tracking these projects, but with their work being scattered across several programs, there is no one system to make these modifications. If there has been a mistake, they cannot easily look at the history of a modification to correct, which has steadily lead to the problem of a database that is never up to date with the current projects, which fixing would only consume more and more time to do.

This has lead to the current situation of having to recharter the organization with NAU using inaccurate budget numbers from this outdated database, requiring them to understate their numbers. This puts their organization current form in jeopardy, as NAU wants them to pay them 50% of what the project brings in, rather than the 17% they currently give, which they justify with large throughput of projects, high value for the 17%, and providing work to students.

The CPCESU also has some other minor issues that need to be handled by any solution that is delivered. They need a mechanism to allow organizations to submit projects, as organizations will sometimes begin projects without the CPCESU's knowledge. This leads to situations where the CPCESU is made aware of the project when it is completed, which may be due to the time cost of communicating with the CPCESU, which we seek to reduce.

They also need to validate the information coming into their organization, so ensure the database stays correct and doesn't contain common mistakes. As their database has been survived through the decades, it has slowly been filled with more and more mistakes and design flaws that need to be corrected. Small issues like not upper

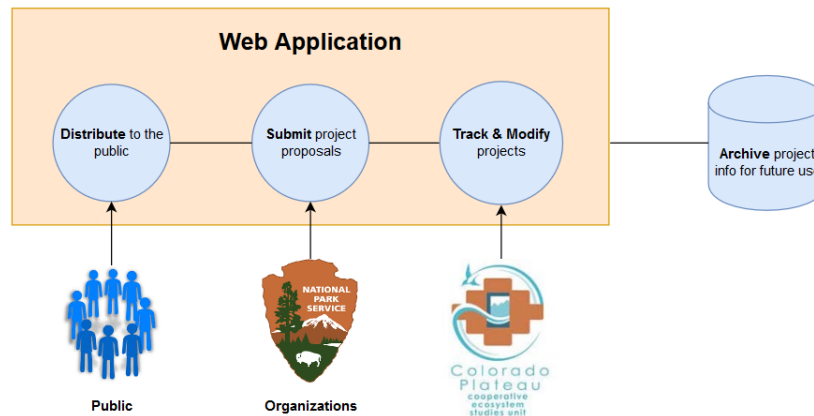


casing names and mistyping in an organization's name then makes it impossible to search and find information in the database.

Lastly, they need a way to distribute select information to the public and organizations. They currently distribute information about their projects with links to PDFs on their NAU website, which is not searchable or maintainable in the long run. In addition to this, various organizations call the CPCEU requesting a list of projects they have done with them, or information regarding previous projects are pulled up in a meeting. They currently fulfil these requests by running premade queries on their database and exporting the results to Excel.

### 3. Solution Vision - *Summit*

Figure 1: Solution Diagram



Our solution to the CPCEUs problem, nicknamed Summit, will be a web application with three primary sections: a public facing front end that distributes projects and informs the public about the CPCEU, a section for organizations to submit project proposals and modifications to projects, and a backend for the CPCEU to manage, track, and modify their projects. All of this is archived into a robust, easy to access, and maintainable database.

The overall functionality of our solution will have a very heavy focus in supporting our client's workflow. We want them to be able to hop on our web portal, query a database for project information, and get quick and easy to read results. This also involves creating a seamless transition between project details without losing information. Finally, storing data from previous projects for analysis and statistics that will be used to support future projects is also an important aspect to our vision. The production version of the website will be hosted using NAU ITS, which will be in a Red Hat Enterprise Linux 7 virtual machine. The website itself will be built using Django v1.11, which is the latest long term supported version of the framework till 2020. The database will be using PostgreSQL 7 and Unicorn will be used as the WSGI HTTP server for the website.

Our solution will fix the client's problem by providing each stakeholder to this website a unique way of accessing the website that addresses their needs. Most importantly, it will decrease the workload with regards to finding and analyzing project data.

## 4. Project Requirements

To deliver a product that matches the solution vision, we have developed various requirements. There are various different types of requirements:

- User Requirements
- Functional and Non-functional
- Environmental
- Performance
- Constraints

### User Requirements

Before starting our technical requirements, we created the following user requirements. They are written as user stories to help quickly and succinctly convey the broader requirements of this solution.

**UR.01** - As a CPCEUSU Staffer, I want to track projects so that I understand, at a glance, what the project status is.

**UR.02** - As a CPCEUSU Staffer, I want to be notified to review and approve of project modifications.

**UR.03** - As a CPCEUSU Staffer, I want to ensure correct data so that common errors are not entered into the database.

**UR.04** - As a CPCEUSU Staffer, I want to autofill from a document so that I don't have to type in all of the data.

**UR.05** - As a CPCEUSU Staffer, I want to generate and export data from searches.

**UR.06** - As an organization, I was to submit project proposals so that it can be drafted by the CPCEUSU.

**UR.07** - As an organization, I want to be able to make modifications to a project so that it stays updated with the CPCEUSU.

**UR.08** - As a student, I want to be able to search through CPCEU projects so that I can find research I am interested in.

**UR.09** - As a person with disabilities, I want to be able to use the CPCEU website so that I can get involved with the CPCEU organization.

From these user requirements, we have created the following functional and non-functional requirements, which are of way of conveying how our solution should be built and simplifies tracking of progress. Certain requirements have a higher level of abstraction and can include lower-level requirements. Overall, all of our requirements fall into five hemispheres of requirements or domains.

1. Creating a core, functional project management system
2. Developing user permissions and functionality for organizations and partners
3. Expand the application with public-facing pages
4. Building an intuitive, universally accessible website
5. Expand on additional capabilities for data entry and project tracking

By organizing our functional requirements into domains, it makes it easier to classify and prioritize our work and how we should create a timeline for our project. Each domain contains functional high-level requirements and nested lower-level requirements, constraints, and so on. In the following sections, we will be addressing each requirement, its type, and how it is important to our project and to our client.

## Functional Requirements

In this section, we will cover our core functional requirements and have various layers of granularity to further specify the technical aspects of the solution that shall be implemented.

### Domain 1: Creating a core, functional project management system

Domain 1 is about creating the core of the project and the overall minimum viable product: a project management system that works for only the CPCEU staff.

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

**F1.01** - All of the projects shall be listed out to be viewed

**F1.02** - Projects shall be marked so that they can be identified through five of their phases, Proposal, Drafting, Acceptance, Tracking, and Termination

**F1.03** - Projects shall have fields so that the following fields can be identified:

- Project Title
- Partner
- Type
- Park/Unit
- FA Office
- Discipline
- Draft Stage
- Budget
  - Overall figure
  - Attachment of budget (optional)
- Tentative Start Date
- Actual Start Date
- End Date
- Date Project Initially Received
- Sensitive Data?
- Point of Contact
- Principal Investigator
- Student Support
- ATR
- P Number
- Initial Amount
- Date Review Comments Sent
- Date TA Approved
- Date Executed

**F1.04** - Projects shall include the optional fields:

- Agreements Tech Representative
  - Project lead - defaults to Point of Contact if empty

**F1.05** - Projects shall be sorted by a variety of ways, including:

- Project Status
- Project Title
- Federal Agency
- Partner
- Type
- Start Date
- (Expected) End Date
- Draft Stage
- Budget

**F1.06** - Projects should allow users to upload attachments to justify work**F1.06.01** - This may include the following document types

- Supporting Documents - .pdf, .doc, .docx, .jpg, .png, .gif, .tif
- Budgets and Personnel Allotment - .csv, .xls, .xlsx

**F1.07** - At a minimum, the project dashboard for CPCEU shall include the following fields for each active project:

- Project Status
  - Location on global workflow
- Point of Contact
- Principal Investigator
- Budget (including any modifications)
- Indictication of Modification Status
  - View will say whether or not a project has modifications pending or requesting review

**F2.00** - Modifications to a project shall be tracked**F2.01** - When a modification is submitted by the organization, a CP Staffer shall be notified of the modification**F2.01.01** - There shall be a button for them to click on**F2.01.02** - This button will redirect to a custom view where the modification can be looked at by the appointed CP Staffer(s)**F2.02** - The CP Staff shall approve or deny the modification

**F2.02.01** - When approved, the modification shall be logged

See **SRS: F2.03** for logging procedures

**F2.02.02** - When denied, the CP Staffer shall note a reason from a field or write a custom message.

Field entries may include:

- Inaccurate or missing data
- Spelling and grammar changes
- Realistic corrections to time, money, and/or personnel

**F2.03** - The result of this modification, approval or denial, shall be logged

**F2.03.01** The system shall automatically keep track of modifications, regardless of approval or denial, to show revisions.

**F2.03.02** The system shall automatically email the federal Point of Contact and the partner's Principal Investigator and any other contacts listed out in the Task Agreement

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

**F3.01** - Project termination can be started by the federal agency or the partner organization

**F3.01.01** - Terms for closing a project by a federal agency:

- Removing time, money, and/or personnel
  - Time is up without extension
  - Money is deallocated
  - No one assigned to project
- Project is incomplete
- Partner does not conduct final report

**F3.01.02** - Terms for closing a project by a partner:

- Project completion with deliverable and supporting materials

**F3.02** - Project termination shall be the same general process including:

- Email notification shall be sent to all parties and Points of Contact (federal agency, partner organization, and CP Staffer assigned to project)
- Project termination may be started by either party or CP Staffer

- All parties shall approve project closure before it is formally terminated

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

**F4.01** - All user input fields shall have client and server-side verification

**F4.02** - The following fields shall have dropdown, search, or auto-fill capabilities to limit user input errors:

- Partners
  - Principal Investigator
- Type

**F4.03** - The following fields shall have the appropriate user field and verifications based on its type:

- Expected Start Date - Date
- Expected End Date - Date
- Budget - Number
- User constraints on all forms
  - Not all text boxes, but the best constraints are using dropdowns and autofill

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

**F5.01** - CP Staffers may print or save the project details page as a PDF.

**F5.02** - CP Staffers shall be able to download project attachments.

## Domain 2: Developing user permissions and functionality for organizations and partners

Domain 2 involves extending the functionality of the project management system to allow for federal agencies and other partners to access the system. This includes segmenting the user base into groups and limit access based on a lowest-level permission system and hierarchy.

**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



**F6.01** - The solution shall include, at a minimum, the following user groups:

**F6.01.01** - CPCESU - Top-level management group

- Administrators - Todd and Laurie
- Staff - Becca
- Read-only

**F6.01.02** - Federal Agencies

- Admins - primary Point of Contacts
- Staff
- Read-only

**F6.01.03** - Non-governmental Partners

- Admins - primary Point of Contacts
- Staff
- Read-only

**F6.01.04** - Public - Anonymous access

- No users or groups

**F6.02** - The CPCESU administrators shall be allowed to add, modify, or remove the groups and users of the system

**F6.02.01** - The CPCESU administrators shall follow the following requirements for modifying user groups:

- CP administrators shall be able to add, modify, and remove users and groups through the admin dashboard built-into the web application
- Partners and federal agencies may have an application process separate built-into the web application to apply

**F6.03** - Federal agencies and partners shall be allowed to add, modify, and remove users from their own group

**F6.03.01** - Group administrators shall be allowed to invite new users, modify users and group details, and change user permissions

**F6.04** - The public (anonymous users) shall have no access to the system except for public facing pages as defined by **SRS: F8**

**F7.00** Each type of user shall have permissions and privileges based on their user group and status

**F7.01** - Users should be allowed to view/query/search database

**F7.01.01** - CPCESU shall be allowed to view/query/search including

- The database for specified projects
- Fiscal year, federal agency, and/or partner
- Past, present, proposed, and cancelled projects
- Any sensitive information relating to each project

**F7.01.02** - Feds shall be allowed to view/query/search by

- Their past, present, proposed, and cancelled projects
- Fiscal year and/or partner

**F7.01.04** - Non-government partners shall be allowed to view/query/search by

- Their past, present, proposed, and cancelled projects
- Fiscal year and/or federal agency

**F7.01.04** - Public users shall be allowed to view/query/search by

- General, non-sensitive project information as outlined in **SRS: F8**

**F7.02** - Only the CPCESU user group shall be allowed to export data from all projects and user groups

- Export via PostgreSQL web command tool
- Export as CSV / Excel file via the web application

**F7.03** - Only the CPCESU administrators and federal agencies administrators and staff shall start a project proposal

**F7.03.01** - CPCESU shall enforce a standard form for all federal agencies using the required and optional fields from **SRS: F1.03** and **SRS: F1.04**

**F7.03.02** - Federal agency administrators and staff shall have access to this form to start a new task agreement

**F7.03.03** - The designated partner and its administrators and appointed staff members (usually Principal Investigator(s)) shall have access to

this project proposal once assigned by a CPCEU administrator or designated staff member

### Domain 3: Expand the application with public-facing pages

For Domain 3, we extend the user groups to the anonymous access or the public front-end. This will allow users to access publicly accessible information without having to register or log in. This also creates an embeddable project search page.

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

**F8.01** Homepage shall contain information pertaining to the CPCEU based on their current website at <http://cpcesu.nau.edu>

**F8.01.01** - Homepage shall contain the following:

- Summary of CPCEU
- Contact information
- Link to current NAU website as noted in **SRS: F8.01**
- Link to project search

**F8.01.02** - Homepage may contain the following:

- Highlighted projects
  - Based on most recent start time
  - May be CPCEU administrator or staff selected

**F8.02** - The project search page shall allow the public to search through the CPCEU project management system without logging in

**F8.02.01** - Each project on the search page shall contain the following information as outlined in Figure 01 below:

Figure 01: An example of what the public can view

Task Agreement	P17AC00181
Project Title	African American Homesteaders Historic Resource Study for HOME and NICO (HRS)
Partner Institution	University of Nebraska-Lincoln
Federal Agency	National Park Service
Project Start Year	2017
Project Status	Active

**F8.02.02** - The search criteria for this page shall contain the following fields as outlined in Figure 02 below:

Figure 02: An example of what the public can search and filter on

**F8.02.03** The current CPESU Wordpress site shall be allowed to embed this search page on its site

Domain 4: Building an intuitive, universally accessible website

Domain 4 involves the general appearance and responsiveness of the website. Primarily, this encompasses the requirements for software accessibility.

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

**F9.01** The solution shall adhere to the following accessibility standards:

- NAU’s Web Accessibility Standards

- W3C Web Accessibility Standards

### Domain 5: Expand on additional capabilities for data entry and project tracking

Lastly, Domain 5 involves requirements that may be optional to the project, but can ultimately extend the solution's functionality and use. This involves things such as data importing, customizable workflows set by federal agencies, and so on.

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

**F10.01** - This system may allow for both new and older projects to be uploaded.

**F10.02** - This system may allow for scanning and optical character recognition for the following use cases:

- Creating a new project
- Archiving an old project that is already terminated
- Submitting a modification or request to terminate an existing project

**F11.00** - The solution may encompass a more detailed project details page for the project search capability explained in **SRS: F8.02**

**F11.01** - The projects on the search page shall have a clickable button that will direct the user to a public project details page

**F11.02** - The public project details page shall include, at a minimum, the following details, unless the field is marked as sensitive:

Figure 03: An example of what the public can view on the project details page after searching

<b>Project Title:</b> African American Homesteaders Historic Resource Study for HOME and NICO (HRS)	<b>Location(s) of Project Activity:</b> Homestead National Monument and Nicodemus National Historical Site
<b>Task Agreement:</b> P17AC00181	<b>Partner Institution</b> University of Nebraska-Lincoln
<b>Project Status:</b> Active	<b>Federal Agency:</b> National Park Service
<b>Project Start Year:</b> 2017	<b>Student Involvement:</b> Unknown
<b>Project Funds</b> \$175,224.00	
<b>Principal Investigator(s)</b> Richard Edwards	<b>Federal Agency Contact</b> Robert Marcell
<b>Project Abstract</b> This project will explore the history, circumstances, and outcomes of African American homesteading experiences in the Great Plains. It provides new scholarship on this little known history by focusing on three historic venues of African American homesteading: (i) DeWitty, Neb.; (ii) Nicodemus, Kan.; (iii) Empire, WY.; and it will also include, if possible, a fourth focus, (iv) isolated African American homesteaders (such as the Shores family in Custer County, Neb.) elsewhere in the Great Plains, if a sufficient number of such homesteaders can be identified. The purpose of the research will be to identify who the homesteaders and their families were, what land claims they made, how successful they were in making claims and gaining ownership, and other information about their educational, cultural, social, recreational, and other life experiences as homesteaders. An effort will be made to document the specific landscapes occupied by African American homesteaders. This project will also benefit the interpretive programs at both Homestead National Monument of America (HOME) and Nicodemus National Historic Site (NICO) in addition to filling a gap in the scholarship of this important national homesteading story.	
<b>Project Modification 1</b> Year: 2017 Task Modification Number: 0001 Funds: \$0.00	
<b>Project Modification 2</b> Year: 2018 Task Modification Number: 0002 Funds: \$66,770.00	

**F12.00** The solution may allow for each federal agency to control its own fields when creating a new project, adding a modification, or having a partner close the project

**F12.01** - The default forms used by all parties shall be the ones set by the CP administrators to assure that required CPCESU information is obtained.

**F12.01.01** - All custom forms set by the federal agency's administrators shall include the required CPCEU fields and use the CPCEU forms as templates

**F12.01.02** - Any conflicts between the forms shall use the CPCEU version of the field

**F12.01.02** - Federal agency forms shall not remove any of the required fields set by the CPCEU

**F12.02** - Additional fields that are set by the federal agency must be from the global type list included below:

- Text box
  - Single and multi-line
- Number
  - Can be Floats (like budgets with dollars and cents) or integers
- Date picker
- User selector
  - Based on autofill or search capabilities for the federal agency's users

**F13.00** The solution may include a Representational State Transfer (REST) endpoint to allow for public users to grab public, non-sensitive data from the CPCEU without directly interacting with the database

**F13.01** - The REST endpoint shall be read-only and rate limited by the web framework to reduce traffic where possible

**F13.02** - This functionality may include private REST endpoints for other CESUs to access and read data

## Non-functional Requirements

This section will cover the various non-functional requirements that will affect the project overall. These include environmental requirements, constraints, performance requirements, and so on.

**NF1.00** Data shall always

- Immediately update in database

- Be done live and between the CPCEU server and the institutional server to make the changes
- Shall take less than five seconds, excluding file uploads and dependent on current CPCEU server or institutional network and database usage

**NF2.00** Our web application should have security that protects sensitive information

**NF2.01** - Sensitive information includes, but is not limited to:

- Budgets
- Content of final reports
  - Local information such as archaeological sites, endangered species information, etc.

**NF2.02** - These guidelines and plans shall be outlined in the Security plan described in **SRS: NF9**

**NF3.00** The CESU and federal agencies may have constraints or specifications that our web application including, but are not limited to:

**NF3.01** - The solution shall not be run by the end users or require users to download an executable of any kind. Federal computers used by agencies do not allow external software

**NF3.02** - The software shall allow for the eventual national functionality such as a REST endpoint to allow for the sharing of data amongst the CESU network and the public

- The data may include:
  - Public project data
  - List of federal agencies and CPCEU partners
- This data shall only be read-only

**NF4.00** The deployment shall be executed with the following constraints and specifications as per NAU ITS and CPCEU:

**NF4.01** - NAU ITS shall host the production server as per the agreements between NAU, the host organization, and the CPCEU

- The project shall use NAU ITS supported tech only.



- ECODers shall adhere to providing any documentation requests as per NAU ITS's policies, code of conduct, etc. These are outlined below as **SRS: NF8** and **SRS: NF9**

**NF4.02** - NAU ITS shall provide a domain name based on CPCEU's request.

- This must be locked in for SSL certificate for secure communications between the end users and our solution.
- This will be taken care of (and paid for) by NAU ITS

**NF5.00** Auth and sensitive data should be contained with

- the need to use defaults and "best practices"
- sensitive data and should have least permissions

**NF6.00** Growth of our web application should allow for

- For upto and including approximately 100 users for now
- The ability to add additional resources if need be
- The growth strategy for this application shall be outlined in the Maintenance plan as per **SRS: NF8**

**NF7.00** Institutional database constraints set by NAU ITS shall allow for

- Database being located on a separate host
- All backing up and maintenance done by NAU ITS
- BLOBs as the file storage

**NF8.00** A Maintenance plan shall be in place before the solution is delivered to the client

- A plan is required to ensure continuous functionality of the web application
- The plan will keep in mind that ITS does nightly security updates as well as major seasonal updates

**NF9.00** A Security plan shall be created and be in place before the solution is delivered to the client

- This is required by NAU ITS to ensure protection of sensitive data in the database, software security, and so on.

**NF10.00** Our web application shall implement Web Server Gateway Interface (WSGI) to serve multiple users at once

**NF10.01** - Green Unicorn (Gunicorn) may be the WSGI technology that our web application uses

**NF10.02** - Our WSGI implementation must be NAU ITS standards and have an included segment in the maintenance plan referenced in **SRS: NF8.00**

**NF11.00** The user interface shall be responsive and use the principles of material design.

**NF11.01** The front-end shall respond within 250 ms to any user input on client side.

**NF11.02** The front-end shall follow the material design principles set out by Google at <https://material.io>. Specifically, this includes:

- Creating a unified user experience
- Customizability tailored to the users of the product
- Implement good design principles based on Don Norman's Principles of Interaction Design:
  - Visibility - the overall appearance of the user interface
  - Affordance - what the user can do with the interface, both the expected and unexpected
  - Mapping - what labels are needed or present

**NF12.00** Our web application may have functionality that will support the search for 'experts'

**NF12.01** - Users may have the ability to find experts based on fields of study which may include

- Zoologists to Anthropologists
  - Built in to the NGO partners
- Not accomplished before

# 5. Process for Requirements Modification

To be updated.

## 6. Potential Risks

During our requirements gathering and technological feasibility studies, we have determined a few key risks to this project's success and overall impact. These were documented as we went through both processes and conducted general research including reading about software engineering (namely the risks and mitigation strategies). It is important to outline the risks and ways to overcome them, because they can ultimately lead to the project not functioning or completely failing.

All of our risks are high-level and can encompass many different, low-level issues. We have also assigned a percentage of occurrence *at the worst case* (probability of happening without mitigation) and a general low, medium, or high impact rating.

### **Risk 1: Learning Curve Too High** | 25% probability | *High*

The risk here is that, as we work on the project, we may develop a solution that is so specifically tailored to some stakeholders in the project that new or existing people in the organization may find it difficult to use the software when just dropped into it. This may limit the adoption of our product into the organization, or in a worst case scenario, may lead to our solution being entirely rejected. This also has the highest probability of happening, as this problem will develop over time if we only listen to a few stakeholders during development, and by the time we realize that it is a problem, it may be too late in the development lifecycle to change the product.

We plan on overcoming this with two simple strategies: early prototyping and constant deliverables for our client to get feedback on the requirements we are implementing and, overall, to assure that we are delivering the solution they need. To generate these deliverables, we will be doing 2-week release periods for production and rapid release to our development server using Continuous Integration and Continuous Delivery methodologies.

### **Risk 2: Overlooked or Missed Requirements** | 10% probability | *Mid to High*

While we are working with technologically savvy clients, there is always an amount of risk that we may have missed a crucial requirement for the system. As we develop the requirements, we will hopefully diminish this risk, but there is always a possibility that the client mentions something that may have been a big requirement that we didn't

catch. As the project gets developed, the client will bring these additional requirements to the forefront, which we want to remain amenable towards.

To minimize this risk, we plan scheduling a buffer period into our schedule for this project to allow work on any additional requirements. In addition to this, we will include procedures at the end of this document to allow this document to be modified to include new requirements, although this may require changing or limiting certain other requirements that already exist in the document. As always, we stress that we want to give our client a product that will satisfy all of their needs in the given time we have, so anyway we can go about doing that, we will.

### **Risk 3: Security Breach** | 5% probability | *Mid to High*

Our client works with some atypical sensitive information concerning the locations of endangered species and archeological sites. This information is controlled under the Archaeological Resources Protection Act of 1979 (ARPA), which states that information on the nature and location of resources on public and Indian lands must remain confidential if it's release may harm the resources. Violating the ARPA the first time entails a fine of \$300,000 and up to one year in prison. To better protect the history of America and protect our client from violating federal law, it is imperative we build a solution that is secure against internal and external actors. While the risk of this happening is relatively low, as the CPCESU is not a large, well known organization, the impact of a security breach can, in the worst case, cost hundreds of thousands of dollars and irreparable harm the reputation of the CPCESU.

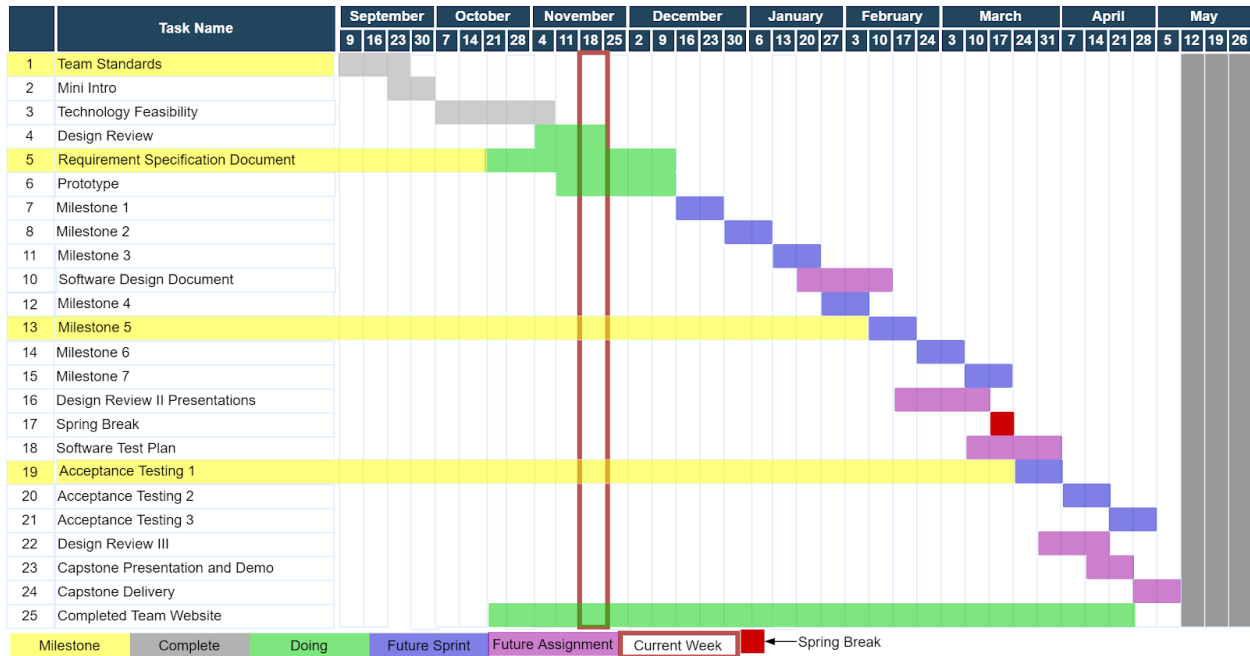
Our solution is to ingrain modern security practices into our development, to ensure that we are delivering a secure product to our client. On top of this, we plan on performing penetration testing on a clone of our website after every two sprints to determine the security flaws with our solution. The results of this penetration testing will be a report that states different ways to access the website, and the severity of the breach. Our goal is to deliver a solution that is secure against all high impact security breaches and minimizes and is able to track low and medium impact breaches as much as possible, though this may be compromised for usability in noted cases.

## 7. Project Plan

The project plan is an essential part to every software project as its primary purpose is to notice when the project is veering off-course as early on as possible, which allows for ample time to readjust the project and the final product.

To develop the CPCEU Project Management System, we plan to work in software sprints of two weeks, implementing at least a core feature every two weeks for the website. Throughout the sprint, the development server will host the website in its most current form, and at the end of the software sprint, a production release of the software is made to the production server. This will allow the client ample time to review the website and request any changes and submit and bugs during development. We will have a total of seven milestones throughout the semester up until Spring break. When Spring break is over, we should have all major features of the website completed and the product will enter into a production environment as much as possible for the client to help determine the existing flaws with the product. These last three sprints will be called the Acceptance Testing sprints, and their purpose is the get the product fully ready to be adopted by the client. At the end of the final Acceptance Testing sprint, the product is delivered to the client and they will determine if they will accept the solution or not.

A break down of the goals of each milestone are given down below. In general, each major feature of the website has a setup, complete, and verified phase. The setup phase consists of a beta or partial implementation of the feature, complete phase consists of the feature completed, and verified phase consists of getting the feature looked at by the client and adjusting it to their comments if needed.



Sprint Name	Goal
Milestone 1	CI/CD setup Overall sitemap setup Login/accounts setup Forms setup
Milestone 2	Rest API setup Public facing website complete Beta testing project management ideas
Milestone 3	Rest API integration Project management flow completed Modification system setup Forms completed
Milestone 4	Autofill forms setup Project management verified Modification system complete Data export setup Search setup
Milestone 5	Data export complete Search complete Modification system verified Autofill forms complete

**Milestone 6**      **Data export verified**  
**Search verified**  
**Autofill forms verified**

**Milestone 7**      **Buffer - Used for extra requested features and lagged schedules**



## 8. Conclusion

Planet Earth is an important part of who we are as human beings. As Carl Sagan once said, “it underscores our responsibility to deal more kindly with one another, and to preserve and cherish the pale blue dot, the only home we’ve ever known.” It is our innate responsibility to protect and preserve our tiny speck in this ever expanding universe. Solving CPCEU’s problem with their current workflow and project management system would give them the ability to work on projects themselves instead of trying to find lost data. By building Summit for the CPCEU we can support their passion for the preservation and protection out here in the southwest United States.

To review the problem, the CPCEU currently does not have a solid project management system that will allow them to store and retrieve data properly. Our clients would benefit greatly from an improved workflow and environment that is easy to use and maintain. The solution to this problem will be our project management system, Summit. Our web-based solution will provide our clients with a simple to use interface and streamlined database access, ultimately giving them the ability to effectively manage past, current, and potentially future projects. By providing data security, easy database queries, as well as a user management system with privileges, our clients can focus more of their time on important projects with quality trusted support from our project management system.

By creating this document, we have laid out the foundations of what is to come over the next few months. We will treat this document as our guide as we move into the development phase, that way our clients can know what we are working on as well as gauge our progress. By defining these requirements in a formal document, we have shined light on our path to creating a product that our client not only needs but is excited about.

As ECODers, we have come up with a plan that will allot enough time for our team to bring a good and working product to our clients in May. Throughout this process of building a plan, we came across a few key insights. One of which was that our biggest hurdle was going to be deployment of our product. We established a positive communication with NAU ITS, who will be the host of our product, and agreed on the technology that will support our system. Another key insight we established early on

was that the client wants the product to be useful for the next five years. This means that as we develop our product, we must be providing future maintainers with enough valuable information to understand the technology and functionality of our code. With all the careful thought and determination that we have been fostering over the past few months we have complete faith that we will give the CPCEU a project management system that they deserve.

## 9. Appendices

In the below sections are the glossary for looking up word and phrases and their definition in terms of this document and the domain of both ECODers and CPCEU. Following that are any auxiliary documents or sections that are referenced and/or supported in this document such as reports further detailing requirements, diagrams, processes, and so on.

### Glossary

**API** - Application Program Interface is a set of routines, protocols, and tools for building software applications

**BLOBs** - Binary Large Objects, a collection of binary data stored as a single entity in a database management system

**CPCEU** - Colorado Plateau Cooperative Ecosystem Studies Unit

**CI/CD** - Continuous Integration/Continuous Development is a development style that will allow for more frequent and reliable code changes

**ECODers** - An NAU 2018-19 computer science capstone team

**NAU ITS** - Northern Arizona University Information Technology Services

**PostgreSQL** - an object-relational database management system with an emphasis on extensibility and standards compliance

**REST** - REpresentational State Transfer is a software architectural style that defines a set of constraints to be used for creating web services

**SSL** - Secure Socket Link

**SQL** - Structured Query Language, used to communicate with a database

**WSGI** - Web Service Gateway Interface, allows for multiple users to asynchronously use a web application at once

**WordPress** - a free and open-source content management system based on PHP and MySQL

### Attachments

There are no supporting documents or attachments presently.