Requirement Analysis Document

Carrier Link

Portfolio Project

V 1.0.0

01-31-2025

Table of Contents

1. Introduction

1.1 Purpose of The Project

The purpose of this project is to build and/or buy a system that streamlines the Portfolio process for students and faculty at Minnesota State University, Mankato (MNSU). Students will be able to easily upload evidence and essays and be able share their portfolios with potential employers. Faculty will be able to access the student portfolios for grading and accreditation purposes. This portfolio system will be used by the Computer Science and the Twin Cities Engineering program by Fall 2025.

1.2 Scope

The scope of this project is to have a fully operational system by the end of the semester. The minimum requirements are that the system is usable by students and faculty and successfully stores portfolio data for a minimum of 6 years. The system should be easy to use and as cost-effective as possible. A stretch goal is to make the application usable for other departments or programs at MNSU.

1.3 Objectives

- Determine the needs of the client and formulate a requirements analysis document
- Conduct research to identify existing portfolio systems that may be suitable

- Compare Build and Buy options to determine which option, or combination of options, meets the client requirements best
- If Build option is chosen, create a full-stack system that utilizes existing university resources when possible
- If Buy option is chosen, create a financial purchasing plan that justifies the effectiveness of the chosen system and includes training documentation for the purchased system

2. Deliverables

2.1 Requirements Analysis Document

Work Type: Needs Assessment and Requirement Development

Activities: Work with Becky Bates and Rob Sleezer to understand the exact details of the current problem that the new portfolio system is supposed to solve. The summary of the current problem can be found in the next section of this document. The specific technical and functional requirements will also be outlined in this document.

Resources: Clients, Students, MNSU library resources, and other departments at MNSU

Tech Skills: Software requirement elicitation and analysis

Priority: High

2.2 Cost-Benefit Comparison

Work Type: Solution Research

Activities: Research potential off-the-shelf systems, or systems that could be partially implemented with university resources. Compare the pros and cons of using an existing system or creating one from scratch. Create a recommendation report for purchasing and implementing an existing portfolio system, or for building a new solution.

Resources: MNSU library resources, online research and potential vendor information

Tech Skills: Technical comparison and economic analysis

Priority: High

2.3 Portfolio System Proof of Principle

Work Type: Implementation

Activities: Create a basic prototype version of the portfolio system to test if the solution will be viable. If purchasing a system, we would need to obtain licenses for the software and customize it for a trail run. If building a system, we would develop a simple system with functional main features to test. Work with students and faculty to get feedback on prototype system.

Resources: Requirements Analysis Document, vendors, development tools (specific tools not determined yet)

Tech Skills: Architectural design, coding, and purchasing if needed

Priority: High

2.4 Proof of Principle Performance Report

Work Type: Assessment

Activities: Evaluate feedback gathered from prototype system testing, then create a detailed report that assesses if the system meets the minimum requirements, if it's easy for students and faculty to use, and outlines any needed fixes or improvements.

Resources: Proof of principle system and user testing feedback

Tech Skills: Software testing and reporting

Priority: High

2.5 Refined System

Work Type: Improvement

Activities: Fix and improve the initial system based on feedback received from students and faculty. Updates will include resolving any known bugs or usability issues, adding necessary features that were missing in the first iteration, and improving existing features until the system meets the project and client requirements.

Resources: Initial proof of principle system, proof of principle report, and client feedback

Tech Skills: Software development

Priority: Medium

3. The Current Problem

3.1 Current System

The Computer Science program at MNSU currently uses an electronic system for the organization and submission of portfolios. There is no clear standard for how each student organizes or stores the evidence for their portfolios. The portfolios are turned in for grading on D2L, the university's online grading platform.

The Twin Cities Engineering program has used a mix of different methods for portfolio organization, including physical binders full of essays and print outs, compressed electronic files with unorganized evidence (similar to Computer Science program), and organized documents that contain links to unorganized evidence.

These systems are unclear, can be unorganized, and can make the portfolio creation process difficult for students and complicated for faculty to grade. *The creation or implementation of a new system will alleviate these current issues.*

3.2 Constituents

As mentioned in the previous section, students utilize the current portfolio systems to update their academic portfolios throughout their time spent in their respective programs. Students can also show their portfolios to potential employers, though that is not currently a straightforward process.

Faculty use the systems to view and grade student portfolios. For final portfolio submissions, faculty often need to download multiple gigabytes worth of student evidence for grading purposes.

3.3 Scope and Scale

This project is focused primarily on solving the portfolio problem for the two previously mentioned programs (Computer Science and Twin Cities Engineering), but the solution could be applied to other programs within the university if they choose. For example, the Honors Program asks students to create their own Wix webpages. If the system we create this semester works better than their current system, it is possible that the Honors program will adopt our solution in the future.

3.4 Root Cause

The Computer Science and Twin Cities Engineering programs have been small enough for the current rudimentary systems to work well enough. As both programs continue to grow, a more sophisticated system is required. The Computer Science degree program may be up for accreditation in the near future as well, so faculty need an effective and organized method of storing 6 years' worth of student work.

4. Research

4.1 Research Overview

Part of the project is understanding the significance of a student portfolio system and how it can address existing challenges. The research aims to analyze current approaches to portfolio management, identify the best practices, and evaluate potential solutions. A critical aspect of this research is the buy vs. build decision, which will consider cost, technical feasibility, scalability, and long-term sustainability of the system.

4.2 Existing Portfolio Systems

Objective: Identify and analyze portfolio systems currently used by academic institutions to determine their suitability for MNSU.

Activities:

- Research portfolio management tools available for educational institutions.
- Compare features, usability, and scalability of existing solutions.
- Analyze how existing systems align with the specific needs of MNSU programs.

4.3 Build vs. Buy Analysis

Objective: Evaluate the pros and cons of building a custom system versus purchasing an off-the-shelf solution.

Activities:

- Analyze financial, technical, and operational implications of both approaches.
- Consider university infrastructure and available resources for system development.
- Identify constraints such as budget, timeline, and staffing requirements.

Comparison Criteria:

- **Cost:** Development vs. licensing and maintenance fees.
- **Customization:** Ability to tailor features to meet MNSU's needs.
- **Maintenance:** Long-term support and updates.
- **Compatibility** with existing university systems (e.g., D2L).
- **Scalability:** Usability across multiple departments.

Deliverable:

- A recommendation report detailing feasibility, costs, and implementation plans.

5. Data Storage

5.1 Data Overview

This project will require data storage capabilities for various types of portfolio artifacts. The primary document artifact type will be a .pdf file, but support for showcasing code from various file types, videos, images, and PowerPoint files. These files will be required to be hosted for a sliding window period of at least 6 years, due to requirements surrounding accreditation.

In addition to hosting these files, our client has indicated a desire to have a tagging system. This system will be built out for both Twin Cities Engineering and the MSU Computer Science program and contain two types of tags. The first kind of tag will be a tag relating to the goal areas, or outcomes, that both programs have for their portfolios, and the second type of tag will be an assignment type tag. These tags will be used for data science purposes by the client.

Discussion has been made about utilizing potential third-party file storage solutions. This solution could work, but a proposed buy solution would require high security, and the ability to have faculty oversee files hosted by students. This solution would also need to be cost-effective over spinning up the program's own CDN, or one on university servers.

5.2 Database Requirements

This database will need to be structured in a way that keeps track of the individual users uploading the files. Users will see the files in an ordered list, with options to sort by tag types. Faculty will be able to see all students within their program and be able to see the files they have uploaded.

The clients have also expressed a desire for the ability for faculty to also have views that allow them to sort all documents from students in a project snapshot and get statistics on types of assignments being represented in certain goal areas. This view will be a separate tab area than the general view where faculty see all of their students and their represented files.

6. Web application

6.1 Application Overview

The web application for the student portfolio system will have an interactive interface for students and faculty to manage, submit and review portfolios. The system aims to provide an intuitive, user-friendly platform that enhances the organization and accessibility of portfolio evidence while ensuring compliance with accreditation requirements.

The project is still in the decision-making phase for a buy vs. build approach, the web application's structure will depend on the final decision. Regardless of the choice, the system must support multi-user access, efficient data management, and seamless integration with university resources.

6.2 Application Requirements

- **User Authentication & Roles:** Secure login system for students to create and manage their portfolios. Faculty can view and grade portfolios, while employers can access public student portfolios.

- **Portfolio Management:** Only students can upload and manage portfolio artifacts (documents, code, videos), which must be stored throughout their studies, with options to label and sort files.

- **Integration with University System:** possible integration with D2L Brightspace for grading and feedback.

- **Long Term Data Storage:** Portfolios will be securely stored for at least six years

- **User Experience & Accessibility:** A clear and simple design to help students and faculty navigate the system easily

6.3 Application Technologies

The application technology depends on whether we build or buy the system.

Option 1: Build
If the system is developed, the following technologies may be used:

- **User interface:** Next.js (React) for structured and easy to use design.

- **Backend System:** Node.js with Express or Django (Python) to handle data storage and communication between different parts of the system.

- **Database:** PostgreSQL or MongoDB to store student portfolios.

- **File Storage:** AWS S3, Google Cloud Storage, or university-managed servers for long-term storage.

- **Hosting:** AWS, Azure, or university servers to ensure security and accessibility

Option 2: Buy

If third party is purchased it may include:

- A platform such as Wix, Google Sites or Mahara
- Integration with University Systems for authentication and seamless access.
- Training & Setup to configure the system and assist faculty and students in using it effectively.

7. Client Approval

Client	**Signature**	**Date**
Client	**Signature**	**Date**