Senior Design Server/Client Development for Project Matching [Phase 2]

Design Document

SD '23 - Team #18
Clients: Jacob Grundmeier, Akhilesh Tyagi
Advisor: Akhilesh Tyagi

Haylee Lawrence	Lead Presenter, Minute Keeper, Testing, Document Editor
MyTien Kien	Team Organization, Client Interaction
Sanjana Amatya	Individual Component Design, Report Manager, Assignment Tracker
Alec Elsbernd	Lead Researcher, Floating Help

Team Email: sdmay23-18@iastate.edu
Team Website: https://sdmay23-18.sd.ece.iastate.edu/

Executive Summary

Development Standards & Practices Used

Since most of our product is software-based, we will be using the common software practices used in industry. This includes testing early and testing often. This will help us catch errors earlier and easier as time continues. Along with that, we plan to code efficiently so we can save time and resources in the future. This will follow with code reviews from everyone in the group and possibly from our advisor to ensure our code is up-to-date and as efficient as possible. Lastly, documentation and/or commenting code will help everyone tremendously in the present and future.

The following are Engineering Standards that apply to our project.

- IEEE/ISO/IEC 26531-2015
    - This standard provides the requirements for managing content used in the software development lifecycle. We will consult this standard to ensure we manage all content throughout the product lifecycle and the user and service management documentation process.
    - https://standards.ieee.org/ieee/26531/5753/

- IEEE P2887
    - IEEE P2887 is not a standard but a recommended practice that will help us implement a Zero Trust Security (ZTS) architecture. A ZTS architecture will help us ensure our project's confidentiality, integrity, and availability.
    - ZTS is an effective way to ensure comprehensive security for a project by asserting that no user, network, or application should be trusted by default.
    - https://standards.ieee.org/ieee/2887/10278/

- ISO/IEC/IEEE 16085:2021
    - This standard outlines risk management processes that we can implement in our project to reduce/mitigate risks and better handle risks as they occur.
    - https://www.iso.org/standard/74371.html

- WCAG Version 2.1 and ADA Regulations
    - These standards and regulations will guide us to help us to make a product that is accessible to all users.

Summary of Requirements

Functional

- Website: Our overall product will be hosted on a fully functional website that includes pages such as preferences, team matching, project proposals, and more.
- Database: A database is required as we will be storing tons of information and transferring from frontend to backend, such as names, preferences, projects, etc.
- Team Formation: The main part of our project is how the teams will form based on preferences given from both the clients and students.

Nonfunctional

- We want our website to be easily accessible and easy to understand for everyone who comes across it.
- We will use the Auctions algorithm to help form our teams efficiently and accurately.
- Security is a big requirement as we will deal with FERPA, and information collected from all our users.

Applicable Courses from Iowa State University Curriculum

Below is a brief list of courses taken at Iowa State University that will help our team develop our project.

- COM S 228: Introduction to Data Structures
- COM S 309: Software Development Practices
- COM S 311: Introduction to Design and Analysis of Algorithms
- COM S 319: Construction of User Interfaces
- COM S 327: Advanced Programming Techniques
- COM S 363: Introduction to Database Management Systems
- COM S 409: Software Requirements Engineering

New Skills and Knowledge

There will be several learning curves throughout this project, including gaining knowledge about something we have not taken a course for. We will also continuously learn new things about the other tools we will use to help build our product. Below is a list of what we will be learning.

- **Figma**: A collaborative web application for interface design
- **Lavarel**: a backend framework that provides all of the features needed to build modern web applications, such as routing, validation, caching, queues, file storage, and more
- **Auctions Algorithm**: an algorithm that operates like an auction whereby unassigned persons bid simultaneously for objects thereby raising their prices, solves the classic assignment problem

Table of Contents

List of Figures and Definitions

1 Team

1.1 TEAM MEMBERS

- **Haylee Lawrence** (Software Engineering)
- **MyTien Kien** (Software Engineering)
- **Sanjana Amatya** (Software Engineering)
- **Alec Elsbernd** (Software Engineering)

1.2 REQUIRED SKILL SETS FOR YOUR PROJECT

- Frontend Development
- Backend Development
- Database Development
- CICD knowledge
- Project management skills
- Client interaction skills
- Teamwork skills
- Agile experience

1.3 SKILL SETS COVERED BY THE TEAM

- **Frontend Development**
    - Haylee Lawrence, MyTien Kien, Alec Elsbernd, Sanjana Amatya
- **Backend Development**
    - Haylee Lawrence, MyTien Kien, Alec Elsbernd, Sanjana Amatya
- **Database Development**
    - Haylee Lawrence, MyTien Kien, Alec Elsbernd, Sanjana Amatya
- **CI/CD Knowledge**
    - Haylee Lawrence, MyTien Kien, Alec Elsbernd, Sanjana Amatya
- **Project Management Skills**
    - Haylee Lawrence, MyTien Kien, Alec Elsbernd, Sanjana Amatya
- **Client Interaction Skills**
    - Haylee Lawrence, MyTien Kien, Alec Elsbernd, Sanjana Amatya
- **Teamwork Skills**
    - Haylee Lawrence, MyTien Kien, Alec Elsbernd, Sanjana Amatya
- **Agile Experience**
    - Haylee Lawrence, MyTien Kien, Alec Elsbernd, Sanjana Amatya

1.4 PROJECT MANAGEMENT STYLE ADOPTED BY THE TEAM

Our group plans on adopting an Agile project management style because it is more applicable to our needs, with

- Short-term deadlines
- The ability to incorporate changes at any time into the project
- Take stakeholders' feedback into account throughout the project
- Potential to overlap work between teammates

Each individual on our team has more experience working in an Agile environment and is more comfortable using it. Using Agile, we can work on different aspects of the project simultaneously and combine them at the end of each sprint. The end of the sprint will include a meeting discussing the parts we have been working on and what we like and don't like so far, including what we could improve on before we continue onto the next sprint. We will also be considering the stakeholders' feedback as the project continues; this is best possible in an agile environment.

1.5 INITIAL PROJECT MANAGEMENT ROLES

- **Haylee Lawrence** (Lead Presenter, Minute Keeper, Testing, Document Editor)
- **MyTien Kien** (Team Organization, Client Interaction)
- **Sanjana Amatya** (Individual Component Design, Report Manager, Assignment Tracker)
- **Alec Elsbernd** (Lead Researcher, Floating Help)

2 Introduction

2.1 PROBLEM STATEMENT

Our project is what will begin the Senior Design lifecycle for the class. Our main focus is to create an easier process for all users involved, specifically improving the project management system.

Some problems include us thinking about:

1. How might we allow for group selection for professors/TAs/admins so that they need to do as little work as possible along with retrieving maximal results?
2. How might we allow for easy preferences input for Senior Design students so they can quickly and easily get the right project for them?
3. How might we allow Clients to easily submit project proposals?
4. How might we allow Senior Design instructors to easily evaluate project submissions?
5. How might we streamline Faculty/Industry panel signups?

6. How might we allow for easy project evaluation (of design documents and grades) for Industry review panelists and ABET Evaluators so they can perform project evaluations?

This project exists because there is room to automate the Senior Design project system. We want everyone included in this process to be satisfied with the result. Although several students are satisfied with their current project and team, there's always room for improvement. Students and clients are most affected by the problem statements above. However, Instructors/TAs, Faculty, and ABET Accreditors are also stakeholders to consider.

This problem only occurs within the Senior Design classroom/course. We aim to solve this problem as we want students to enjoy what they're doing throughout the project and have them end their college careers by creating a working project that they can present to the faculty. Our solution will be to create a web-hosted system that future Senior Design students can use to accurately assign projects based on their preferences.

2.2 INTENDED USERS AND USES

There will be several users using our product. These include but are not limited to: students, clients, IT managers, Senior Design professors, teaching assistants, faculty advisors, and ABET evaluators. Below is a quick description of how a user may interact with the product and who they are.

Students

Key Characteristics

A student using this product will be a senior in electrical, software, computer, or cybersecurity engineering. This student would be taking the senior design course. Their goals would likely include ending their senior year with a great project and graduating. Depending on their preferences, the student would like a great group to work with and a project they are at least interested in.

Needs Related to the Project

The student needs a way to state project/group/working preferences because this is how the system would match students to specific projects.

How They Might Use or Benefit From the Product

Students will fill out their preference forms. Based on their answers, our product will help to accurately pick out the best project based on their needs and others. The goal is for the group assignments to be just as effective, if not more effective, than group assignments done by hand.

Clients

Key Characteristics

A client using this product will be either from the industry or a faculty member. Their goal would be to find student help with a specific project. They would also be willing to help and teach the students over the year.

Needs Related to the Project

The client would need to fill out the senior design project proposal form because the students would need to know what the project is about and what is needed out of the project.

How They Might Use or Benefit From the Product

The system would result in a group of students interested in the client's project, giving the client a team to work with to help get their project idea up and running.

IT Managers

Key Characteristics

The IT manager work for Iowa State University. They are responsible for each Senior Design group's servers and other technical needs. This person would have a good knowledge of what technologies are needed for Senior Design. They will work with the advisors/faculty to see what the students need for their year-long project.

Needs Related to the Project

The IT managers need a way to edit/view the faculty advisor and client database because they will need to manage the behind-the-scenes of the project (something that only IT managers have access to).

The IT managers need to create a GitLab website and other resources to manage and maintain the site because the code will be part of Iowa State repositories that students don't have access to.

How They Might Use or Benefit From the Product

The product is there to select and create groups for Senior Design. Once that process is done, the IT managers are assigned to create the group's email, website, etc. Many IT managers will be a part of a team to help with further technical needs as time progresses.

Senior Design Instructors

Key Characteristics

These Senior Design Instructors are faculty members here at Iowa State. Their role in this course is to teach students skills that may aid them in their senior design project. They are also responsible for the day-to-day grades in the class.

Needs Related to the Project

Senior design instructors need access to all submitted projects because they will need to go through them and approve the viable projects.

Senior design instructors will need a way to submit student lists to the database and ensure that it matches the roster because they will need to ensure that all Senior Design students receive a project and are correctly matched to their preferred groups/project.

How They Might Use or Benefit From the Product

As the current process stands, the senior design instructors are the ones that are doing the manual work of assigning projects. They will benefit greatly from this project because it will significantly reduce the time and effort they must put into determining and distributing project assignments. This product will also create an easy-to-access collection of all project submissions to more easily determine which projects will be viable.

Faculty Advisors

Key Characteristics

Faculty advisors are ISU faculty members assigned to advise and supervise one or more senior design teams. They will be knowledgeable about the project that the student teams are working on and will be able to provide ample guidance to students throughout the year.

Needs Related to the Project

The Faculty Advisors need a way to access their assigned senior design groups because they will need to determine which group(s) have been assigned to their project(s).

How They Might Use or Benefit From the Product

The faculty advisors will benefit from this product because they will be assigned groups interested in their project. This tool will allow easier access to their project groups' contact info.

Faculty & Industry Review Board Members

Key Characteristics

Faculty and Industry Review Board members will be ISU Faculty or Industry Members who volunteer to sit on a Review Board to evaluate Senior Design Projects. The Faculty Review Board will evaluate projects at the end of SD 491, and the Industry Board will evaluate projects at the end of SD 491. Both parties have significant knowledge of various industries and the Project Lifecycle.

Needs Related to the Project

The Faculty and Industry Review Members need a way to sign up for timeslots, during which they will evaluate projects. They also need a way to access information regarding the teams they are evaluating (e.g., team number(s), access to the team website, and design document(s)).

How They Might Use or Benefit From the Product

The Review Board Members will benefit from this product because it will consolidate the time signup and team information into one place.

ABET Evaluators

Key Characteristics

ABET evaluators using this product goal would be to create groups that meet diversity requirements and have the appropriate skills and interests to complete the project while creating a group with a good learning experience.

Needs Related to the Project

ABET Evaluators need to ensure that their constraints are being taken into account when creating teams because it is their goal to see diversity and inclusion in the programs they direct.

How They Might Use or Benefit From the Product

ABET Evaluators benefit from this product because, if done correctly, they will have an efficient algorithm to ensure that these diversity and inclusion standards are being held in this program.

2.3 REQUIREMENTS & CONSTRAINTS

2.3.1 Functional Requirements

1. *Web-Based Application:* This project requires a front-end interface for students, clients, and faculty members to interact with. The front end will be where students set their preferences for groups. After team formation, this interface will also be how users can see the groups and project information.

2. *Database:* Our product will need to have a backend database to store the data received from the users. The database is essential in creating a way to ensure that users can choose their preferred group while also reaching ABET standards for diversity and inclusion.

3. *Team Formation:* One major aspect of our application will be how it can form teams. It will consider the students' preferences on which project they want to work on, who they want to work with, and what skills they excel in. It will also consider ABET standards for

diversity and inclusion to create teams with which students and administrators are content.

2.3.2 Non-Functional Requirements

1. *Usability and Humanity:* This product will be usable by Students, Clients, IT managers, Senior design instructors, Faculty advisors, and ABET evaluators and can be used on the first attempt with basic knowledge about websites and how the senior design class operates. The product will create a server/client system for the senior design course to organize and accurately assign projects to teams and clients based on preferences.

2. *Performance:* The product will collect data from the seniors in 491 using a preferences form that will assist students in choosing and differentiating what they would like to do in terms of their project or what they're looking for overall. This product will be accessible on both the web and mobile devices. The product will use the auctions algorithm to compare a student's preferences in each column of the database and assign projects.

3. *Security:* The product's data and functionality can only be accessed by authorized users and employees of Iowa State University. Specific data may be sent to clients in case of special requests. The system should protect data in the product's database from corruption and unauthorized/ accidental disclosure. Data that has been printed as a hard copy should be properly disposed of. The product will also retain all records of data processed out of the system to ensure the safety of its users.

4. *Cultural and Political:* The product will use English as its default language. It will not use any offensive wording, icons, or pictures that could displease someone.

2.3.3 User Interface and Experience Requirements

We want the users to have an easy yet quick experience when choosing their preferences or submitting project proposals. Our project will result in students inserting their information in a way that both sides (students and clients) can easily understand. The preferences form will allow users to easily choose and differentiate what they would like to do in terms of their project or what they're looking for overall. Everything will be accessible on both the web and mobile devices. Below is a brief list of what every user may need regarding user interface ad experience requirements.

User	Requirements/Needs
Student	Picking project preferences, access to GitLab, website
Client	Project proposals
Senior Design Instructors and TAs	Grading system, access to GitLab, approving projects website, running project matching

Faculty Advisors	Grading system, access to GitLab, website, assigned to project proposals
IT Managers	Creating GitLab, website, servers
Faculty & Industry Review Board Members	Signing up for review times, viewing team information and websites
ABET Evaluators	Reviewing project proposals, reviewing project presentations

Constraints

1. *Time:* The main constraint for this project is time. We are given limited time as we have only two semesters to fully plan and develop this project. If given more time, we would likely include more features and make a more well-rounded product.

2. *Technology Used & Tools:* Since our project is starting in phase 2, we will need to build our project off of last semester's work to maximize efficiency. For whichever portions we reuse, we will need to emulate their technology as well as possible for an easy transition. Similarly, we are limited by which technologies we are familiar with. Thus, we will need to sacrifice reusability to maximize coding efficiency.

3. *Cost:* We are not given any money to help build our site, so we are constrained to using only free resources. Tools and frameworks such as our coding languages, database, and security scanners will all need to be free. We also need to be mindful of the cost of IT managers maintaining the site. To limit the amount of money the University needs to pay them to maintain our site, we will make our project fault-tolerant, easy to use, and as safe as possible.

2.4 ENGINEERING STANDARDS

The following are Engineering Standards that apply to our project.

- IEEE/ISO/IEC 26531-2015
    - This standard provides the requirements for managing content used in the software development lifecycle. We will consult this standard to ensure we manage all content throughout the product lifecycle and the user and service management documentation process.
    - https://standards.ieee.org/ieee/26531/5753/
- IEEE P2887
    - IEEE P2887 is not a standard but a recommended practice that will help us implement a Zero Trust Security (ZTS) architecture. A ZTS architecture will help us ensure our project's confidentiality, integrity, and availability as the data within it.

    - ZTS is an effective way to ensure comprehensive security for a project by asserting that no user, network, or application should be trusted by default.
    - https://standards.ieee.org/ieee/2887/10278/
- ISO/IEC/IEEE 16085:2021
    - This standard outlines risk management processes that we can implement in our project to reduce/mitigate risks and better handle risks as they occur.
    - https://www.iso.org/standard/74371.html
- WCAG Version 2.1 and ADA Regulations
    - These standards and regulations will guide us to help us to make a product that is accessible to all users.

3 Project Plan

3.1 PROJECT MANAGEMENT/TRACKING PROCEDURES

Below is a list of the team's pros and cons to compare the two project management styles we were considering.

Waterfall

- Provides a concrete plan from start to finish
- Saves time by establishing all requirements from the beginning
- Does not allow for flexibility or change
- Can cause problems if Stakeholders disagree on something after the project has been executed.

Agile

- Short-term deadlines
- Can incorporate changes at any time into the project
- Takes stakeholders' feedback into account throughout the project
- potential overlap of work between teammates
- Increases risk of miscommunication

Our group plans on adopting an agile project management style because it is more applicable towards our needs. We picked an agile style over a waterfall management style because each individual on our team has more experience working in an agile environment and is more comfortable using it. With the use of agile, we can work on different aspects of the project simultaneously and combine them at the end of each sprint. The end of the sprint will include a meeting discussing the parts we have been working on and what we like and don't like so far, including what we could improve on before we continue onto the next sprint. We also like considering the stakeholders' feedback as the project continues; this is best possible in an agile environment.

We plan on using Notion to track progress throughout this and the next semester. Notion is a project management and note-taking application that helps groups coordinate deadlines,

capture ideas, and stay on task with their built-in calendar. It can be used as a writing repository and has a database included in the software. Its extreme flexibility works perfectly for our team as we adopt an agile project management style.

3.2 TASK DECOMPOSITION

We split tasks into six categories: Design, Backend, Frontend, Database, Algorithm, and Testing. For each category, more tasks needed to be done for our project to succeed. More specific details are in the *Personnel Effort Requirements* section.

Design

- Determine the project's architecture
- Finalize design documents

Backend

- Review Phase 1's backend component
- Determine additional languages/frameworks
- High-level design of controllers and classes needed
    - Determine which controllers and classes needed
    - Code the controllers and classes
- Implement SSO backend

Frontend

- Determine languages/frameworks
- Determine necessary user roles
    - Which users will see specific components/pages
    - Which users can see certain databases
- Wireframe site in Figma
    - What pages will we need
    - Color and font themes
- Create pages based off of wireframe
    - General Pages
        - Dashboard
        - Login (SSO)
        - Profile
            - Logout
            - User information
    - User Pages
        - Student
            - Preference Input
            - Review Sign-up
            - Profile (Student Specific)
                - Project Information
                - Teammates

        - Client
            - Project submission form
            - Previous Submissions (and status)
            - Specific Project information
                - Group number
                - Group Information
        - SD Instructor
            - Preference form editor
            - Student csv input/editor
            - Project Review
            - Project Selection
                - Run algorithm
                - View results
        - Faculty or Industry Review Board
            - Sign up for time slots
            - See links to project sites
        - ABET
            - View project selection results
                - Current and previous
            - See links to project sites
- Connect with backend
- Finalize pages based on Testing and User feedback

Database

- Evaluate previous project's database design's to see what can be reused/improved upon
- Create/modify tables
    - Students
    - Groups
    - Projects
    - Faculty Advisors
    - Senior Design Instructors
    - ABET users
- Connect database to backend

Algorithm

- Research algorithms
- Determine input parameters and weights
- Design and code the algorithm in the backend

Testing

- Frontend
    - UI testing to verify
        - Rendering
        - Button/input functionality

    - Test connectivity between frontend and backend
- Backend
    - Test connectivity between frontend and backend
    - Test connectivity between backend and database
    - Test correctness of backend controllers, classes, and methods
- Algorithm
    - Test its ability to create teams that fit the ABET standards
    - Test its ability to create teams that maximize student and client satisfaction
- Security
    - Test security of Database
    - Test security of SSO

3.3 PROJECT PROPOSED MILESTONES, METRICS, AND EVALUATION CRITERIA

Below is a rough draft of what we plan to accomplish for our project. The main goal we want here is to match projects to clients/students with 95% satisfaction. We'll evaluate everything by going through a testing stage with each milestone and checking if they work correctly and accurately. We will thoroughly research how to implement each milestone and ask questions when we can. The database is already handled, so we don't need to do much with that. The project matching algorithm will hopefully be implemented where accuracy is up to at least 90%.

1. **Web Application**
    a. Homepage
    b. Project proposals list
    c. Login for all types of users
    d. Web server setup
    e. Database setup
    f. Connecting database and server
    g. Connecting server and client

2. **Project Proposals**
    a. Client project proposal form
    b. Client form edit view
    c. Client information in database

3. **Project Review**
    a. View submitted project proposals
    b. Accept or reject proposals
    c. Parse project information into the database based on results

4. **Project Selection**
    a. Student preference form
    b. View project proposal list

c. Student preference selection into database
d. Email students, clients, and advisors with matched projects

5. **Project Team Generation**
    a. Research classic assignment algorithms
    b. Design algorithm
    c. Testing

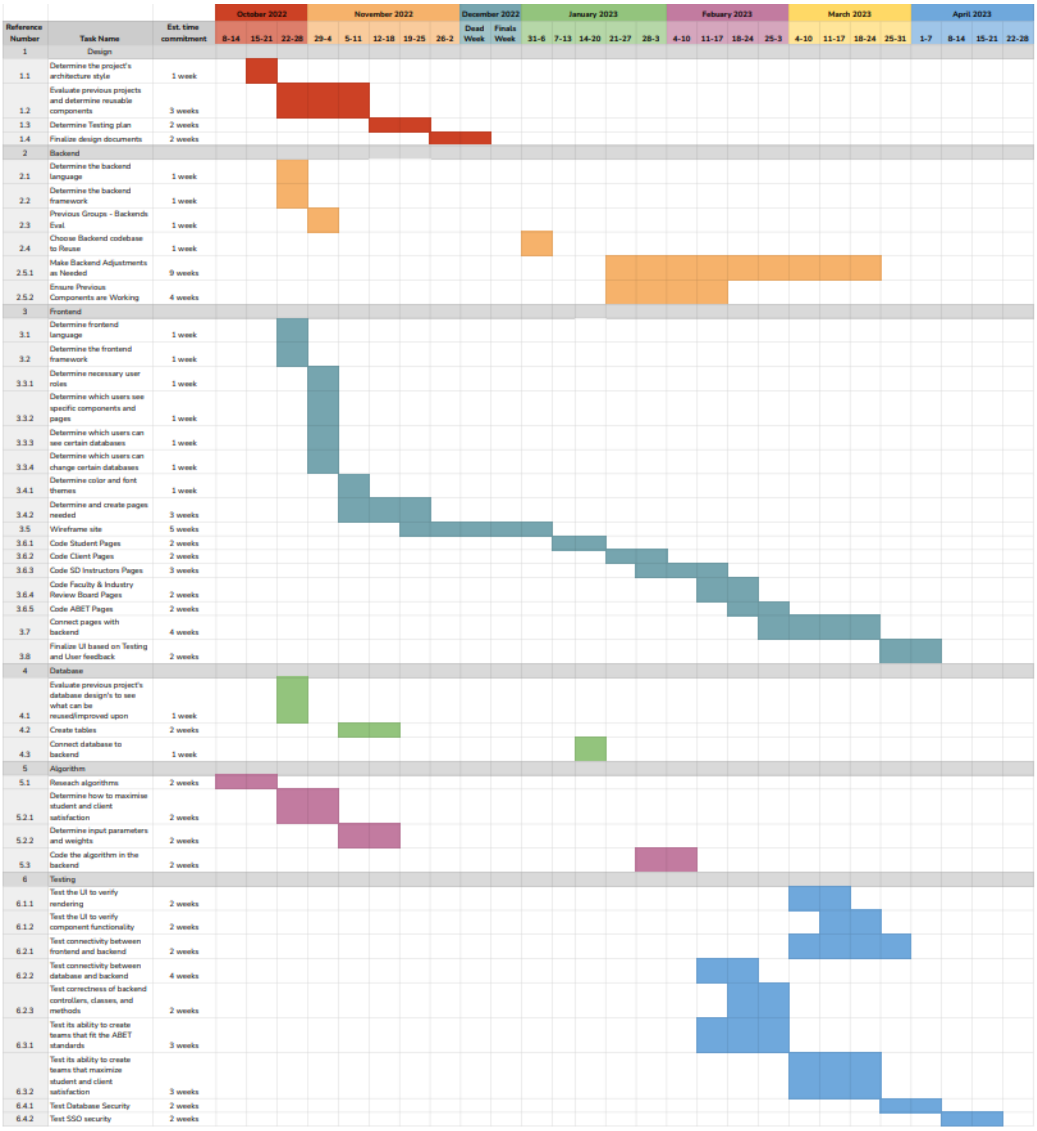
6. **Project Preferences Form**
    a. Create preference criteria
    b. Create form
    c. Insert data into the database

7. **Design Document**
    a. Completed best to ability and finalized after multiple iterations and ready to be implemented

3.4 PROJECT TIMELINE/SCHEDULE

Project Gantt Chart

A PDF version is also linked on our Project Site:
https://sdmay23-18.sd.ece.iastate.edu/docs/Project%20Gantt%20Chart.pdf



3.5 RISKS AND RISK MANAGEMENT/MITIGATION

Risks	Risk Probability	Mitigation
Issues with getting the desired results with our algorithm.	0.25	To mitigate this risk, we can add more team members and hours to the task. Right now, we are committing quite a bit

		of resources so the probability is low
Not reaching the desired deadline for completion due to not allocating enough time each week to the project	0.1	As a whole, our team has been very good about reaching deadlines, but to make sure we stay on track we meet weekly together to go over our assignments. We also will try to have deadlines that aren't last minute to allow for some problems.
Significant Software and/or Hardware problems with the backend/database	0.4	This problem has a higher risk due to it being somewhat random. To mitigate this risk, the entire team is committed to working extra weekly hours if we encounter problems. Another way we are planning to mitigate this cause is by planning to have certain troubles and accounting for them in our scheduling.
Problems with getting the desired final UI/frontend	0.1	Although, in our experience, these problems are less frequent than the backend, we must still allocate time in our schedule to deal with them if they arise. Also, our team is committed to putting in more hours if needed.

3.6 PERSONNEL EFFORT REQUIREMENTS

Task	**Explanation/Description**	**Estimated time it will take**
Design	This task involves the first stage of designing the project structure and skeleton.	Determine project's architecture style - 1 hour Evaluate previous projects and determine reusable components - 4 hours Finalize design documents - ~5 hours Total hours = ~10 hours
Frontend	This task will be defining and implementing services and components for users and working on all viewable features of the web	Determine framework and language - 1 hour Define fonts and themes - 2 hours

	application.	Design wireframe - ~3 hours Design pages to be created - ~1 hour Determining user roles and how much access each individual is allowed - 2 hours Define role-specific components, services, and resources - 2 hours Determine which users can change the database - 2 hours Defining user roles - 2 hours Creating UI mockups - ~7 hours Implementing roles and permissions for users - ~15 hours Implementing student, client, instructor, faculty and ABET pages - 10 hours Connect pages to backend - 8 hours Implement UI mockups - ~15 hours Total hours = ~ 70 hours
Backend	This task is focused on the implementation of the backend of the web application. This includes creating interactions between the frontend and the database and utilizing API endpoints.	Defining classes and controllers for the project - 3 hours Deploying skeleton application on server - 5 hours API route design - 5 hours Model design - 10 hours Implementing controllers for each class - ~15 hours API endpoints - ~ 8 hours Implement SSO backend - ~20+ hours Total hours = ~67 hours

Database	This task is solely focused on designing and implementing the database and connecting it to the backend.	Evaluate previous project's database to figure out what we can use and improve on - 2 hours Create tables for the users ● Students - 2 hours ● Groups - 2 hours ● Projects - 2 hours ● Faculty Advisors - 2 hours ● Senior Design - 2 hours ● Instructors - 2 hours ● ABET users - 2 hours Connect the database to the backend - 25 hours Total hours = 32 hours
Algorithm	This task is determining and implementing an algorithm that will match students to an assigned group based on their preferences. We will be designing and implementing this algorithm into the application's backend.	Research algorithms - 4 hours Determine input parameters and weights - 5 hours Design algorithm - 8 hours Implement algorithm for the backend - 20+ hours Total hours = ~37 hrs
Testing	Testing the frontend, backend, how well the algorithm works with the backend, and how secure the web application is.	Test user permissions and components - ~15 + ~15 hours Test UI to verify rendering - 10 hours Test connectivity between frontend and backend - 10 hours Test connectivity between database and backend - 10 hours Test ability to create teams of ABET standard - 15 hours Test database security - 8 hours Test SSO security - 8 hours Total hours = ~91 hours

3.7 OTHER RESOURCE REQUIREMENTS

- A machine, such as a computer, capable of running an IDE and connecting to the internet/WiFi
- A web server to host the web application and connect to the database
- Opinions of senior design students, faculty, clients, and how satisfied they are with their project and team, and what would make it better
- Research on the frontend, backend, and web development

4 Design

4.1 DESIGN CONTEXT

4.1.1 Broader Context

Below is a table showing the communities that are affected by our project. Since our project is more suited to clients and students, we focus heavily on them. Overall, our project won't be affected much environmentally or economically.

Area	Example
Public health, safety, and welfare	In our project, the welfare and health of students will be acknowledged by putting them in groups they prefer and are comfortable with. This includes matching technology stacks with someone interested, matching a person with people they prefer to work with, and so on. Clients will also have a team to work on a project they've wanted help on for a while. This relieves stress and creates a fun project for the students and clients.
Global, cultural, and social	Our project here will create teams that mix different professions, such as software, electrical, cyber, or computer engineering majors. Additionally, we aim to create diverse teams with diverse technology stacks to finish projects created for different communities. Doing so will help the students and clients reflect on the values and practices of certain

	global, cultural, and social groups. With this project in place, students can spend more time connecting with their professors and advisors.
Environmental	Since our project is mainly built for the senior design class, we plan on making the code as efficient as possible.
Economic	The project itself isn't too hard to maintain financially, as we only need a few things to help complete it. We plan on making this project last for future years. With this project in place, faculty and admins will work less. Thus fewer hours are billed to the university.

4.1.2 Prior Work/Solutions

Currently, to solve this problem, faculty are doing most of the work manually and getting their input from students through google surveys. This is a strenuous and time-consuming task, so the need for our product comes from.

Team 3 and team 44 from the previous year have worked on a project trying to reach the same goal as our team. Through review, we found some areas we like and will build on and some we can see ways to improve on. Team 3 used Laravel and Vue.js, team 44 used a Larval Backend and Angular for the frontend, and team 44 used angular components and services. We feel they did a good job with their database components but lacked an in-depth algorithm.

4.1.3 Technical Complexity

Below, we show that this project is of sufficient technical complexity. Our design consists of multiple components that will depend on each other for the project to work, as well as the preferences of both the client and student.

The project scope is spread across the senior design class year pipeline. Our project will be used throughout the academic year, consisting of several things, the main feature being the project matching process. This process will include making a classic assignment algorithm entirely from scratch, working with creating databases, and learning new tools. Below is a list of the tools, systems, and components we will create or work with.

- Making a classic assignment algorithm that will match several different preferences from both students and clients to create groups that satisfy everyone
- Creating the preferences and which ones hold priority over the other
- Connecting server, database, algorithms, and web application so communication and interaction between all are possible

- Looking over Phase 1 groups and picking a group to follow off of for Phase 2
- Learning the tools that Phase 1 groups used so we can easily connect and start Phase 2
- Possible Canvas Student or OKTA integration - this is for logging in as a user
- Several factors apply to our project, such as the incoming project proposals that are supposed to be approved or denied
    - Possibly collecting information and opinions from students and clients to see what other ways we can improve this process
    - Functionality for faculty/professors/clients/evaluators

4.2 DESIGN EXPLORATION

4.2.1 Design Decisions

We will have to make the following Design Decisions:

1. Which frontend language to use
2. Which backend language to use
3. Which database to use
4. Which algorithm to use

Determining which languages, databases, and algorithms we use will rely on various criteria including our prior experience with these languages, their ability to interact productively, if previous teams were able to implement them effectively, and if they fit our goals.

4.2.2 Ideation

One of the key decisions that we found integral to the success of our project was which database to use for our web application. To make this decision and use the best database for our project, we considered a multitude of factors. We created a Weighted Decision Matrix to determine which languages to use for frontend, backend, and database. Some factors we considered were our team's previous experience with the language, the ability to be easily implemented with other components, the previous team's usage of said database, and if this is the best database to help us achieve project goals.

4.2.3 Decision-Making and Trade-Off

After considering the advantages and disadvantages of each frontend and backend framework, our team decided to use a weighted decision matrix system to find a framework that best works for us and the project goals. One of our most weighted criteria was how easily the framework could be implemented and connected to the database. We based this on how much previous experience we had using each framework and how well they could interact with each other. Calculating the total amount of points based on the weight, we were able to settle on React as

our frontend framework, Laravel as the backend framework, and MySQL as the database we will be implementing for this project.

Backend Decision Matrix

Criteria	Weight (1-5)	Spring Boot	Laravel	Flask
Our familiarity with the language	3	3	1	0
How easily does this framework work with a database/frontend	4	4	4	3
Have the other teams used them (helps reusability)	2	0	2	0
Will this language help achieve our goal product	3	3	5	3
Total	12	22	24	18

Frontend Decision Matrix

Criteria	Weight (1-5)	React.js	Angular	Ember.js	Vue.js
Team's previous experience with framework	3	5	2	0	0
Ability to be easily implemented with other components	4	4	4	3	4
Previous team's usage	2	0	2	0	2
Is this the	3	4	4	3	4

best language to use to achieve project goals?					
Total	12	25	24	18	22

Database Decision Matrix

Criteria	Weight (1-5)	MongoDB	MySQL	Maria.DB	Oracle	Microsoft Database Server
Team's previous experience with framework	3	0	5	0	2	1
Ability to be easily implemented with other components	4	4	5	4	4	4
Previous team's usage	2	0	2	0	0	0
Is this the best database to use to achieve project goals?	3	4	5	4	4	3
Total	12	20	29	20	22	20

4.3 PROPOSED DESIGN

4.3.1 Overview

Our project will be a web application that will have several features. First of all, on the application clients/instructors will be able to submit the accepted projects into the system. This will include a description of the project, the desired number of students, and their majors. The students will be able to fill out a form with their personal information and desired preferences for a team. Once this form is filled out their information will be entered into the database. Then at the due date for teams, the application will go through the database and create teams based on an algorithm. This algorithm will match the students to the project that matches their preferences best and from there the final teams will be released.

4.3.2 Detailed Design and Visual(s)

Below is a Block Diagram detailing the architecture of our web application. It is made up of a Frontend, Backend, and MySQL Server Database. The Frontend, Backend, and Database sections below explain each piece in more detail.

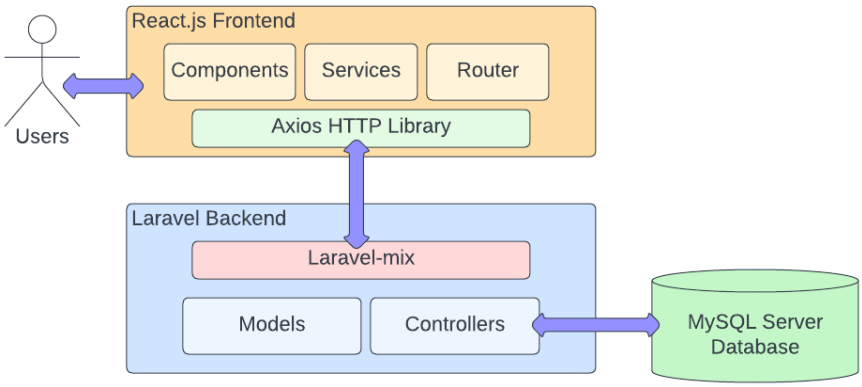


Figure 2: React and Laravel Project Block Diagram

Frontend:

- For our frontend UI design, we will utilize a similar layout as the ISU theme provided by the University, so that our design is usable for those familiar with the ISU sites. We will not be following the ISU theme exactly, however, as we do not want to be limited to their design. We will use a UI library such as Material UI for a majority if not all of our visual components.
- For our frontend architecture, we will utilize a Layered approach with Angular components and services. We also will utilize the React Router capabilities to allow for the Web app to have multiple pages. Similarly to previous group 44, we will use resource services to facilitate API Calls between the Frontend and the Backend.

Backend:

- Our Laravel project will be similar to any other Laravel application; it will contain packages for our Models, Repository, and Controllers. Similarly to SDMay22_44's group, we will use our Repository as our primary way of accessing the MySQL database. We will structure our Database similarly to Group 44's because they got their Database fully functioning during Phase 1.

Database Design:

- Our general Database design is shown below. We will have tables for all of our users, for projects, preferences, and for groups. SD Instructors will be able to update the Student's table with a CSV. The inputted CSV will be compared to the existing table, and will only add new Students or Delete students not present in the CSV. It should be noted that the Preferences values are initial, and are likely to change. Foreign keys are highlighted in orange and Primary keys are bolded and highlighted.

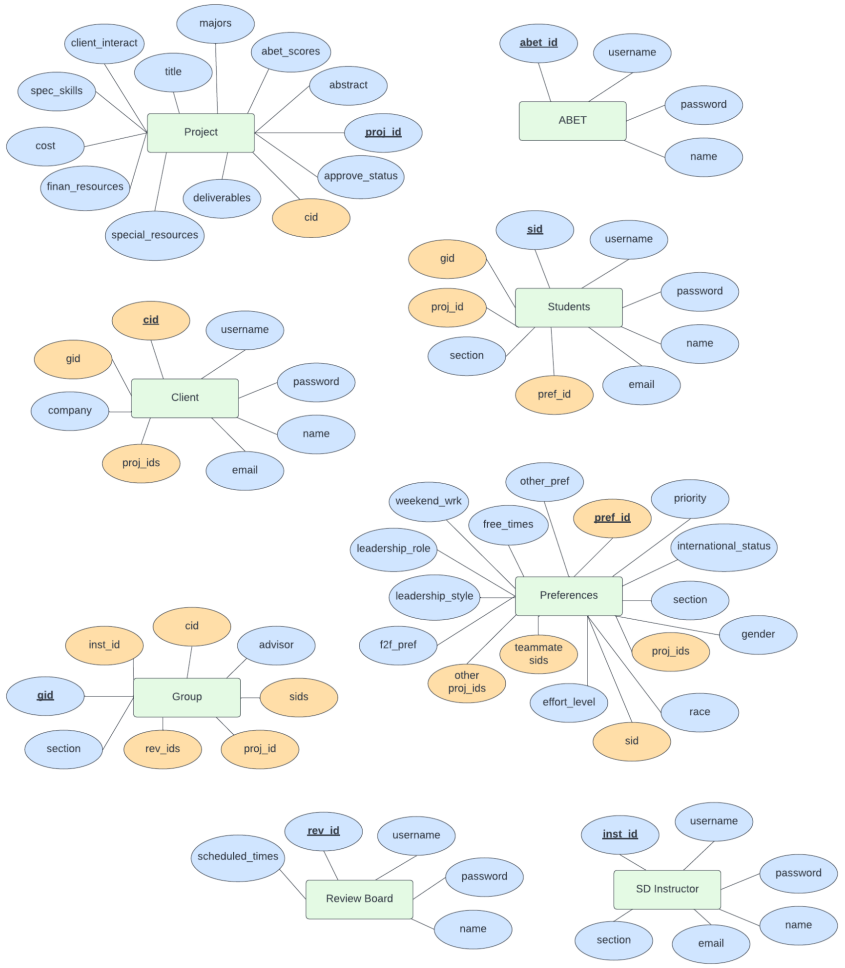


Figure 3: Product Database Diagram

Algorithm:

- Our algorithm is what will be choosing our teams and project matchup. We will base our algorithm on the Auctions algorithm, a solution to the classic assignment problem. The

classic assignment problem is about matching persons to objects based on certain criteria, more information here (1).

2. Assignment by Means of an Auction

Consider N persons wishing to divide among themselves N objects. We number persons and objects as 1, 2, ..., N. For each person i there is a nonempty subset A(i) of objects that can be assigned to i. An *assignment* S is a (possibly empty) set of person-object pairs (i,j) such that $j \in A(i)$ for all $(i,j) \in S$, for each person i there is at most one pair $(i,j) \in S$, and for each object j there is at most one pair $(i,j) \in S$. In the context of a given assignment S, we say that person i is *assigned* if there exists an object j such that $(i,j) \in S$; otherwise we say that i is *unassigned*. We use similar terminology for objects. A *complete assignment* is an assignment containing N pairs (*i.e.* every person is assigned to a distinct object). There is a given integer value a_{ij} that a person i associates with an object $j \in A(i)$. We want to find a complete assignment that maximizes

$$\sum_{(i,j) \in S} a_{ij}$$

over all complete assignments S. We call this the *primal assignment problem* and note its well-known equivalence to a linear programming (linear network flow) problem as shown in Fig. 4.

Figure 4: Classical Assignment Problem Explanation Excerpt (1)

- The main idea of the Auctions algorithm is to treat it as an auction. We will introduce a variable, price, that represents the cost the person must pay when matched up with the corresponding project. The “bidding phase” is explained below, cited from a research paper by Dimitri P. Bertsekas.

Bidding Phase: For each person i that is unassigned under the assignment S:

Compute the "current value" of each object $j \in A(i)$ given by

$$v_{ij} = a_{ij} - p_j \qquad (5)$$

Find a "best" object j* having maximum value

$$v_{ij^*} = \max_{j \in A(i)} v_{ij},$$

and find the best value offered by objects other than j*

$$w_{ij^*} = \max_{j \in A(i),\, j \neq j^*} \{a_{ij} - p_j\}. \qquad (6)$$

(If j* is the only object in A(i) we define w_{ij^*} to be $-\infty$, or, for computational purposes, a number that is much smaller than v_{ij^*}.)

Compute the "bid" of person i for object j* given by

$$b_{ij^*} = p_{j^*} + v_{ij^*} - w_{ij^*} + \varepsilon = a_{ij^*} - w_{ij^*} + \varepsilon \qquad (7)$$

[We characterize this situation by saying that person i bid for object j*, and that object j* received a bid from person i. The algorithm works if the bid has any value between $p_{j^*} + \varepsilon$ and $p_{j^*} + v_{ij^*} - w_{ij^*} + \varepsilon$, but it tends to work fastest for the maximal choice (7).]

Figure 5: Bidding Phase Excerpt (1)

- The project will be assigned to the person with the highest “bid.” These bids will be based on the student's preferences and the majors/tech stack needed for the project. More information will be provided on how this bidding system works later on.

4.3.3 Functionality

The main users here would be the client and the student. Overall, the student and the product will be the ones who will initiate a reaction/response. Below, you can see the basic functionality between our product and its users.

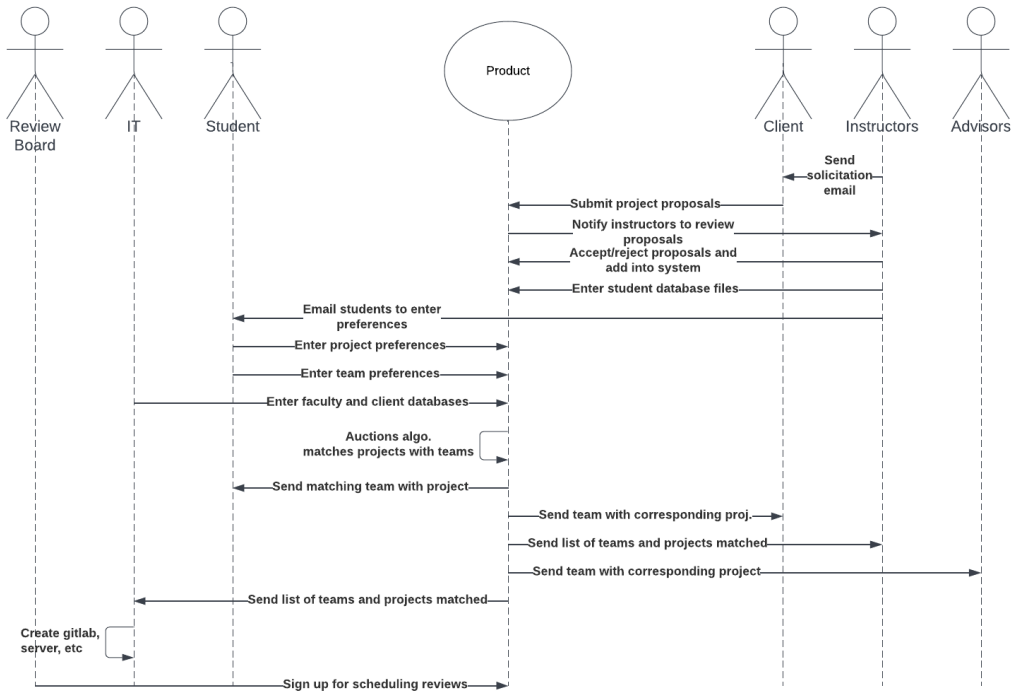


Figure 6: Product Functionality Diagram

4.3.4 Areas of Concern and Development

We have a few main concerns about our project. First would be the algorithm we are implementing. We are basing our algorithm on the Auctions algorithm, a solution we found to the classic assignment problem through research. After tons of research, we see that the Auctions algorithm is the most efficient and is a winner against its competitors. As none of us are familiar with this algorithm, it'll take a while to perfect it within our project. One problem is that no matter what algorithm we choose, we can't always find a solution for everyone. In that case, there may be students that are left out since it's impossible to match everyone with their top preferences. With that, the solution would be to have the faculty/instructors manually insert the remaining student(s) into a group based on their other preferences or interests.

Another concern we have is FERPA. With the amount of information we need for the project, for example, student name, major, and preferences, we're not sure what we can or cannot access. This goes for the clients as well. To fix this issue, we will be taking steps to ensure that FERPA is obeyed throughout our project, such as taking a course or talking to an expert.

Last but not least is following the ABET policy of diversity. Since we are using an algorithm to help sort the teams and projects, we can't quite ensure that the teams will be "diverse." We may be confused about what diverse means in this context, but the solution is to seek advice from our advisors and clients. We will also most likely talk to experts that are familiar with ABET.

4.4 TECHNOLOGY CONSIDERATIONS

To come to a final decision on each of these different technology options, we created a weighted decision matrix. This can be found in section 4.2.3

Laravel: The main strengths of Laravel are that many of the group members have experience using it, and it will integrate easily with each of the frameworks we will be using. The previous team also used Laravel, so we will be able to build off their work. We also looked into using Springboot and Flask but concluded that Laravel would make the most sense as a group.

React.js: React's main strengths are that many of the group members have experience using it, and it will integrate easily with each of the frameworks we will be using. We also looked into using Angular, Ember.js, and Vue.js but figured since none of us had prior experience with them, it would be more time-consuming to learn a new framework.

MySQL: Overall, MySQL was the clear choice for our database. It had a perfect score in all of our categories, and for a good reason. We all have extensive knowledge using it, and can easily be implemented with the rest of our frameworks and Iowa State since they already run some MySQL databases. It was also what previous teams used, so we will be able to take a lot from the previous projects to help us. Although no other options were even close, we looked at MongoDB, Maria.DB, Oracle, and Microsoft Database Server.

4.5 DESIGN ANALYSIS

Taking the previous sections of this Design Document into account, we have constructed a detailed outline of the tasks we plan to undertake. Utilizing a weighted decision matrix, we have decided to use a Layered approach with React.js components and services for our Frontend framework, Laravel for our backend framework, and MySQL for our database. We wanted to find a framework suitable for each team member based on their previous technical skills and experience. So far, our proposed design has been working as expected.

For the visual aspect of the web application, our team has determined that we will be following the same ISU layout and themes previous teams have utilized for this project. This is just so that our application is consistent with other ISU websites. We will be creating wireframes for each section/ page created to make better UI decisions and for documentation purposes.

5 Testing

5.1 UNIT TESTING

Frontend

- For each component or service, we will create a test file that will ensure coverage of edge cases and normal functionality.
- All tests will be coded using Jest.
- Our testing will cover proper rendering, the functionality of inputs and outputs, and proper communication with the Backend and Database.

Backend

- Test files will be generated for every controller to test for the proper functionality of methods, API endpoints, and Database interactions.
- All tests will be coded using JUnit and Mockito.

Database

- The JUnit and Mockito tests in the Backend will also ensure correct communication between the Database and the Backend.

Algorithm

- A Test file for our Algorithm will be generated as a part of Backend Unit Testing.
- JUnit tests will ensure our Algorithm works as expected.
- We will also model Client and Student Satisfaction – or some other input — to ensure we are allowing for optimal project matching.

5.2 INTERFACE TESTING

As described earlier, we want a website that introduces the senior design lifecycle for senior design students, specifically the part that focuses on project matchmaking. As with any other website, we plan on focusing on two specific components: user experience and functionality. User experience is an essential part of websites. We want all users of our product to have a simple and stress-free time when choosing their preferences and browsing proposals. To test this, we chose to do on-site testing involving moderation of previous or present senior design students. As the website creators, we will evaluate how the students will get to point A from point B on the website, ask questions about their experience, and so on. This will give us an idea of how to improve and work with what we have.

We will be testing the functionality of many things. This will include mostly the functionality of the website as a whole and the back-to-back transfer of data. We will use manual testing and Selenium (automated testing) for the functionality. Selenium is a well-known tool used for testing automation on web applications. Additionally, we are familiar with the tool, which will help

achieve a reliable and great product. For data transfer, we plan on using Postman to test if the results transfer correctly and quickly.

5.3 INTEGRATION TESTING

Connection between Backend and Database
To test the connection between the backend and the database, we will use Postman. Postman allows us to pass test data into the database and see if these two components communicate correctly. This will ensure we can save our data into the database through the server.

Connection between Frontend and Backend
Testing the integration between the frontend and backend will be different than testing the Backend and Database. We can use Postman to some extent, but much of the effort will be manual. We will have to go through and test that calls to the backend from the application work correctly and that we get the correct information from the backend. Also, we will need to test that data entered into the web application is properly entered into the backend.

5.4 SYSTEM TESTING:

The tests above must be successful before thoroughly testing everything. Once we start system testing, we will ensure everything is working as expected. This will include continuous testing with “dummy” or fake data. We will repeatedly test and fix the problems until everything functions as expected. We will use fake data in several tests for the main parts of our product, such as: running demos on the algorithm, project editing, viewing, submission, preferences input, and project selection.

5.5 REGRESSION TESTING

To ensure that our system is running correctly when new functionality is added, we will be retesting using the techniques described in the previous sections. For the new addition to be officially integrated into production or the main branch, it must pass all other tests. This will also mean that we may have to change our tests to include a new section if we change anything drastically.

The most critical features we need to ensure don’t break are the algorithm and how it connects to the system and the overall communication between all the separate components of the system. If the algorithm were to break, we would be losing one of the main features of our system since this is a system designed to create teams using the algorithm. It could also be very hard to see if our algorithm is broken, which is why good unit tests are so important. Regarding communication between all components, if one section of the system is not working, then the system's usefulness is gone.

5.6 ACCEPTANCE TESTING

Our team will create tests based on our unit cases to ensure we meet all functional and user requirements. We want to ensure our product also corresponds to the business requirements and can be utilized by the end user. We plan on doing this through Alpha and Beta Testing our project using Atlassian's Jira Software, a popular agile development software with UAT testing frameworks.

Alpha Testing provides feedback to clients about the built-in features of the application. We plan on utilizing this by talking with our clients and stakeholders about what they believe should be on the list of requirements to be met by the final product.

Through Beta Testing, we will be getting feedback from customers or users before the web application is launched to the public. We plan on collecting data from our product after utilizing it in different situations. This will help us identify problems that might pop up before students, advisors, or professors use the product. The collected data will help our team analyze the feedback we receive so that we can make the necessary changes and adjustments needed so that our product meets all the requirements of our clients.

5.7 SECURITY TESTING

The Project Matching site will deal with many users and their data. Thus we will take significant measures to ensure that our site is as secure as possible for our Users.

5.7.1 Following Best Coding Practices

Throughout the creation of our site, every member of our group will follow the best coding practices. Including steps such as sanitizing all inputs, using a Zero-Trust approach, minimizing Extraneous Functionality, and utilizing a strict and secure Login Function. Taking these steps will ensure that we limit vulnerabilities from the beginning.

5.7.2 Testing Against Common Attacks

During the Testing Phase (see Gantt Chart above), we will test for the following common vulnerabilities:

Vulnerability	**Test**	**Solution**
SQL Injection	Injecting SQL statements such as *hacker' AND password = 'whatever' OR '1'='1'* Into all of the forms that access the Database (e.g., Login Form) to see if we can (a) Login without a password (b) View/edit/delete data	Sanitize inputs and use the Principle of Least Privilege.

Cross-Site Scripting	Inputting malicious HTML into inputs Ex. *“onmouseover= alert(‘hello’);”*	Sanitize and validate inputs.
Brute Forcing	Attempting to Login multiple times with common passwords.	Allow three incorrect login attempts before locking the account for five min. After nine consecutive incorrect Login attempts, IT must be contacted. Requiring Users to have complex passwords.
URL Manipulation	Attempting to access secure pages for different Users/User types by manually inputting the URL (ex. www.[site].com/User01Info)	Perform user permission checks before displaying a page.

5.7.3 Miscellaneous Security Options

Some other security options include logging and code scanning. This allows us to take measures when scanning our code for security vulnerabilities using a free code scanner. We also will enable security options within our Frontend and Backend. Enabling such options will assist us in monitoring where vulnerabilities may lie.

5.8 RESULTS

We will have four main components that will be tested: the frontend, the backend, the database, and the project's algorithm. Each component will have a set list of requirements they shall pass to guarantee a well-tested web application.

Components	**Summary**
Frontend	The user will be able to navigate to any page on the web application based on the role and privileges they are given. For the frontend components, we will conduct unit and integration tests that go through all performance requirements from a user’s perspective and ensure user inputs and all UI functionality are set up perfectly. We also want to ensure the software we create matches the business and client's goals.
Backend	In this component, we will focus more on the product's functionality and how well the features work with new code changes throughout the semester. An example is connecting to the database between the frontend and backend components and going

	through different user scenarios to ensure the data is transferred properly and securely when requested.
Database	All data in the database will be returned correctly to ensure that the connection between all components is working properly. Data return for queries should not take more than one second, and all data deleted or inserted should be put in their proper table.
Algorithm	Our team will test the Auctions algorithm by taking in input, calculating the expected output, and comparing it to the output the algorithm gives us. Our main goal is to check if it's time efficient and not faulty. This means we need to ensure that it can take in bulk amounts of data without crashing the product

Examples of what guidelines we will be following to write functional and quality tests:

1. Simple and transparent tests
2. Creating user test cases with the end product and client in mind
3. Avoiding repetitive test cases
4. Test cases will include a description of what is being tested, an explanation of how it will be tested, and the expected results.

6 Implementation

Team 18 is Phase 2 of the Senior Design Project Matching project; the previous two teams were SDMay22 Team 44 and Team 3. As there are a lot of similarities between the goals and requirements we are trying to achieve, our team decided to build off of the code they wrote for the web application and improve on the aspects they were lacking in. Team 3 used Laravel and Vue.js, team 44 used a Larval Backend and Angular for the frontend, and Angular components and services. We think that they did a good job with their database components but lacked an in-depth algorithm which is what we have decided we want to focus on the most.

After making certain Design Decisions, we have decided to use Laravel for the backend and React.js for the frontend to keep things consistent in the project and truly focus on the algorithm aspect of the project so that we can provide a fully functioning matching application. We have started Wireframing in Figma but have not started the development process yet. However, we are on track based on the outline we have written in section 3.3 (Project Milestones), and are on-track to start designing the different components of the project.

7 Professional Responsibility

7.1 AREAS OF RESPONSIBILITY

Below is a table that compares different codes of ethics to the seven areas of responsibility. We chose to compare the IEEE Computer Society ethics as it aligned with our project more than the other Codes of Ethics. Below is an explanation of how principles from the IEEE Computer Society code of ethics compare to the NSPE Canon version.

Area of Responsibility	**NSPE Canon**	**How it addresses each of the areas of seven professional responsibilities in the table. How does the IEEE, ACM, or SE code of ethics differ from the NSPE version for each area?**
Work Competence	Perform services only in areas of their competence; Avoid deceptive acts	**Public & Clients and Employer:** Acting within the interest of the public, clients, and employers. **Product:** SE's must ensure that the products and modifications meet the highest professional standards. **Judgment & Profession:** Integrity and professional judgment are a focus both in work competence and within these principles. The SE principles do not mention timeliness, however, it could be considered by Client and Employer (2).
Financial Responsibility	Act for each employer of client as faithful agents or trustees	**Management:** Ensuring product development and maintenance are done ethically, including making sure that it is affordable to its users. There are no specific mentions of financial responsibility, however, the Management principle most closely aligns with this area.
Communication Honesty	Issue public statements only in an objective and truthful manner	**Client and Employer:** It talks about acting in a manner that is in the best interests of their clients and employer, which includes reporting work truthfully and without deception. Within the principles, there is no specific focus, however on making sure that the work is done understandably.

Health, Safety, Well-Being	Hold paramount the safety, health, and welfare of the public	There are no specific mentions of the safety and health of stakeholders. However, a majority of the principles covered this area implicitly by ensuring that actions are consistent with public interest or stakeholder interest. Acting with integrity also overlaps with health, safety, and well-being.
Property Ownership	Act for each employer or client as faithful agents or trustees	Again, there are no specific mentions of respecting property, ideas, and information, however, I believe that it falls under the umbrella of multiple principles. For example: **Judgement:** Respecting property and ideas falls under maintaining integrity. **Colleagues:** Protecting the property, ideas, and information of others includes being fair and supportive of colleagues and not disrespecting them and their work.
Sustainability	N/A	**Public:** Acting within the public interest means being concerned with protecting the environment that we all live in. There are no specific mentions of the sustainability of work within the principles, however, Public most closely resembles this area.
Social Responsibility	Conduct themselves honorably, responsibly, ethically, and lawfully so as to enhance the honor, reputation, and usefulness of the profession	**Public:** We want to create a product that aligns with Public interest so that it benefits them. **Client and Employer:** Stakeholders, clients, and employers will likely be impacted by these products, so we want to make sure our product benefits them. **Management:** We want to make sure that this product that we're making is managed in such a way that is continually ethical so that it continues to benefit society and communities

7.2 PROJECT SPECIFIC PROFESSIONAL RESPONSIBILITY AREAS

Below is a table that explains the importance of each responsibility within the context of our project as well as how we are performing as a team regarding each responsibility.

Area of Responsibility	Importance within Project	How Well Team is Performing
Work Competence	**High**. The senior design course was created to allow us to demonstrate our ability to produce high-quality work that shows professional ability. Thus, we aim to create a product that presents our best professional abilities.	**High**. Each member of our group is taking our work seriously and performing as high-quality work as we can. Every single one of us is also acting with integrity and demonstrating professional competence by creating our design document.
Financial Responsibility	**Low**. There are no costs related to our project other than the cost of our advisor's time when we meet with him. We produce work that does not require new technology, hardware, or software. All costs for the Gitlab and servers will likely not change or change much as a result of our project.	**N/A**. Especially before, our product will not cost any extra money to the department or to us. We have also not started coding yet, so there are no extra costs from the energy it takes to run our site.
Communication Honesty	**Medium**. As we have only one stakeholder that we have to report work to, not reporting our work truthfully would only hurt us and our progress. This area is not low, however, because we do want to make sure that we are reporting our progress correctly and clearly to our advisor so that he may assist us with any issues we have and give us feedback.	**Medium**. We are reporting our work to our stakeholder truthfully and in an understandable way. However, I do not think that we utilize our stakeholder often enough, and I think we have to do a better job of using our advisor as a tool to get direction and advice on our product.
Health, Safety, Well-Being	**Low**. Our product will not directly impact the health or safety of our users. The only risk is that FERPA may be violated by having data leaked, however, this would not cause any harm substantially, either financially or physically to our users.	**N/A**. Our product does not directly impact stakeholders' safety, health, or well-being. As mentioned above, the information that we are storing may impact these, however, we have no database set up or information at the moment.
Property Ownership	**Medium**. We have very little room to impact the property ideas and information of our users, however, we will be storing data, so property ownership of ideas and information is something that we will want to	**High**. Within our group, we respect others' ideas and information. With our stakeholder, we respect his ideas for what the product should do, the information he's given us, and how to proceed.

	be respectful of it by securing it properly.	
Sustainability	**Low**. We are creating a web app without the need to use materials that harm the environment. We will, to the best of our ability, try to condense our codebase so that it uses fewer resources, however, I don't believe that having a large codebase for this project would significantly impact the environment.	**N/A**. Our product does not directly impact the environment or natural resources. We also currently have no code base that is using resources.
Social Responsibility	**High**. The senior design course allows us to demonstrate our ability to create a product that shows our ability to create well-done and competent work, so we want to make sure that we are creating a product that ends up benefiting our users. While the user base doesn't make up a large portion of society, we have to address many users in the Senior Design space who we are hoping to benefit from our product.	**High**. Our group is trying our best to create a design document laying the foundation for our product. Through creating a high-quality design document, we hope to produce a product that benefits our Senior Design community. Our product will directly impact society, however, perhaps our product will allow others to benefit society by having less of their time taken up with project matching.

7.3 MOST APPLICABLE PROFESSIONAL RESPONSIBILITY AREA

Work Competence is the most applicable for our team. For our team, this responsibility involves working to the best of your ability, in a timely and honest manner. This would also include going through learning curves and using past experiences to make a good project/product. So far, our team has been working together to organize our project so that we can all contribute with what we have learned or want to learn. We meet weekly as a group, and with our advisor to ensure that we are on track and our ideas make sense for our project. We complete the design document assignments promptly and have done much research for our project.

8 Closing Material

8.1 DISCUSSION

Overall, this project has gone a long way in developing our professional skills. Each team member has put a substantial amount of effort into this project and that has contributed to our overall success. This document has led to learning many different aspects of project management and planning. We believe that this plan will be able to successfully create a final product that will meet the needs of our clients and the users of our product. The next step will be to learn through developing this project using the plan we have laid out in this document.

8.2 CONCLUSION

As we look forward to the next steps of our project, we are currently exactly where we planned to be. The main components of our project are the web application and the project matching algorithm. One of our main goals is the creation of a web application that not only meets the standards of our client, but also the standard of our fellow engineering students. Our other main goal is to create an algorithm that successfully creates groups that will meet the standards laid out by our client. For each of these aspects, we have finalized our plans and have started the early stages of implementing them. We look forward to the next semester, where we can continue to learn and hopefully create a final product that meets our very high expectations.

8.3 REFERENCES

(1) Bertsekas, Dimitri P. *THE AUCTION ALGORITHM: A DISTRIBUTED RELAXATION METHOD FOR THE ASSIGNMENT PROBLEM* . Laboratory for Information and Decision Systems Massachusetts Institute of Technology Cambridge, Mass. , Mar. 1987, http://www.columbia.edu/~cs2035/courses/ieor8100.F12/auction-alg.pdf.

8.4 APPENDICES

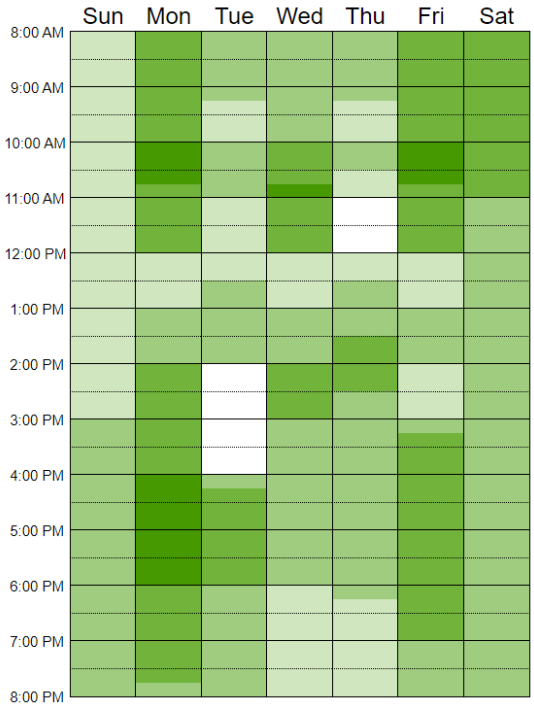
8.4.1 Team Contract

Team Members:

1) Haylee Lawrence (Software Engineering)
2) MyTien Kien (Software Engineering)
3) Sanjana Amatya (Software Engineering)
4) Alec Elsbernd (Software Engineering)

Team Procedures

1. **Day, time, and location (face-to-face or virtual) for regular team meetings:**
   Mondays at 4 pm, virtual, weekly. Below is a screenshot of when all team members can meet, with the dark green being when all team members are available.



2. **Preferred method of communication updates, reminders, issues, and scheduling (e.g., e-mail, phone, app, face-to-face):**
   We would prefer to work on project tasks as a group, face-to-face. All other communication will be done through Zoom and Discord.

3. **Decision-making policy (e.g., consensus, majority vote):**
   Depends on the situation, but we will mostly be making our decision based on consensus.

4. **Procedures for record keeping (i.e., who will keep meeting minutes, how will minutes be shared/archived):**
   Haylee Lawrence will take notes in Notion. The board will be shared with the other members.

Participation Expectations

1. **Expected individual attendance, punctuality, and participation at all team meetings:**
   We expect everyone to be punctual and attend and participate in all team meetings, including our weekly client meetings on Mondays at 10 am and our weekly team meetings on Mondays at 4 pm. If one cannot attend a meeting or will be late, they shall notify their entire team through Discord/email. All team members shall participate in these meetings when possible.

2. **Expected level of responsibility for fulfilling team assignments, timelines, and deadlines:**
   We expect everyone to, for the most part, have an equal responsibility in team assignments and to meet team deadlines

3. **Expected level of communication with other team members:**
   We want everyone to be able to communicate what they're working on and if they have any problems getting the help they need. We have a discord server for the group, so everyone is expected to respond to questions and reach out whenever necessary. If someone has prior commitments and cannot attend a meeting, they should be able to communicate with the group ahead of time.

4. **Expected level of commitment to team decisions and tasks:**
   Each team member will give as much time as necessary to complete their assigned tasks. If they have some troubles with the task, they must notify the group immediately so that their workload can be adjusted. Each team member will also complete each task to the best of their ability.

Leadership

1. **Leadership roles for each team member (e.g., team organization, client interaction, individual component design, testing, etc.):**
   Haylee Lawrence (Lead Presenter, Minute Keeper, Testing, Document Editor
   MyTien Kien (Team Organization, Client Interaction
   Sanjana Amatya (Individual Component Design, Report Manager, Assignment Tracker)
   Alec Elsbernd (Lead Researcher, Floating Help)

2. **Strategies for supporting and guiding the work of all team members:**
   Frequent communication through Discord, keeping up to date with the Notion board, making sure that we meet often and that everyone shows up. If someone has some troubles with the task, they must notify the group immediately so that their workload can be adjusted.

3. **Strategies for recognizing the contributions of all team members:**
   We will celebrate the small victories with a thumbs-up reaction on discord/party parrot.

Collaboration and Inclusion

1. **Describe the skills, expertise, and unique perspectives each team member brings to the team.**

- **Haylee Lawrence:** Project management skills, coding (Java, C/C++, Python, JS, MySQL, etc.), UI/UX design experience, experience with Client Communication, training in Agile & Design Thinking, and experience with design requirements documentation.
- **MyTien Kien:** Two software engineering internships, communication, project management, coding (Java, C/C++, C#, SQL, HTML, CSS, etc.), tools/frameworks (.NET framework, Springboot, Unity.), etc.
- **Sanjana Amatya:** Two software engineering internships, Teamwork, UI/ UX experience, people and communication skills**,** data analysis, coding (Java, C/C++, JS, MYSQL, Salesforce, etc.).
- **Alec Elsbernd:** Two software engineering internships, people and communication skills, Coding (Java, C/C++, C#, SQL, etc.).

2. **Strategies for encouraging and support contributions and ideas from all team members:**
   We will always listen to ideas that our teammates pitch. As a team, we will evaluate the fairly.

3. **Procedures for identifying and resolving collaboration or inclusion issues (e.g., how will a team member inform the team that the team environment is obstructing their opportunity or ability to contribute?)**
   Our team will most likely end up in a team meeting to discuss these issues and communicate our opinions. In this meeting, we will discuss a compromise with which everyone will hopefully agree in the end. We can refer to this team contract if needed.

Goal-Setting, Planning, and Execution

1. **Team goals for this semester:**
   - Create an effective and usable way of determining Capstone projects
   - Improve our ability to create an effective product as a team
   - Learn how to effectively communicate our progress to a panel of Faculty and Industry Professionals

2. **Strategies for planning and assigning individual and team work:**
   - Split the work into four so that everyone has a task to focus on
   - Make sure that everyone is finishing the task they've been assigned ahead of time and asking the team for help if they get stuck.
   - Starting to work on the assignment weeks before the deadline gives everyone enough time to focus on what they need to complete.

3. **Strategies for keeping on task:**
   Weekly meetings will help ensure that we are holding each accountable and moving a project forward. Each team member will be expected to stay on task and complete any work or research assigned. Each of our tasks will have clear objectives and be very focused to help keep the team on task.

Consequences for Not Adhering to Team Contract

1. **How will you handle infractions of any of the obligations of this team contract?**
   Sometimes infractions occur because of factors beyond our control. However, if a team member foresees an event or situation that will keep them from making a deadline, they shall inform the team immediately so that the other team members may compensate for this issue. If the team member commits an infraction without excuses, the team will address the infraction as a group. We will discuss what occurred and reiterate that we expect high-quality and on-time work.

2. **What will your team do if the infractions continue?**
   If infractions continue with no reasonable excuse, then the team will have an intervention involving the Professor and our Advisor.

****

a) I formulated the standards, roles, and procedures as stated in this contract.
b) I understand that I must abide by these terms and conditions.
c) I understand that if I do not abide by these terms and conditions, I will suffer the consequences as stated in this contract.

1) Haylee Lawrence DATE: 9/12/22
2) Sanjana Amatya DATE: 9/12/22
3) Alec Elsbernd DATE: 9/12/22
4) MyTien Kien DATE: 9/12/22