

# Class Scheduler

DESIGN DOCUMENT

TEAM NUMBER: **SDDEC22-09**

CLIENT: **VICKY THORLAND-OSTER**

TEAM MEMBERS:

**ZACHARY BUNCH- TEAM LEADER AND BACKEND DEVELOPER**

**CHRIS HORVATICH - FRONTEND DEVELOPER**

**CONNOR GAECKE - BACKEND DEVELOPER**

**CHARLES MULDERINK - FRONTEND DEVELOPER**

TEAM EMAIL: [SDDEC22-09@IASTATE.EDU](mailto:SDDEC22-09@IASTATE.EDU)

TEAMWEBSITE: [HTTP://SDDEC22-09.SD.ECE.IASTATE.EDU](http://SDDEC22-09.SD.ECE.IASTATE.EDU)

REVISED: **Apr 24, 2022** /VERSION 1

# Executive Summary

## Development Standards & Practices Used

We used Figma to get an overview of the GUI for the Class Scheduler.

We will be using PyCharm to do most of the coding. For the front end, we will be using PyQt5 and the tools that are related to that python binding that is implemented as a python plugin.

An engineering standard that we decided to use to keep our code clean was PEP 8 which is a python coding standard. This is to make sure that all of our code looks uniform no matter who is working on the project.

## Summary of Requirements

1. Be a document that can be changed with the schedule of classes. If it could be connected to the Schedule of Classes on ISU's website—that would be fabulous. If not, it should be duplicated from the fall before or the spring before.
2. It will need to have the labs associated with the specific rooms.
3. Have them “group” related (so Cpr E 530, 531, 532, 533, 534, 535, 536, 537...show up as one color or some demarcation so that I know I am not putting something on top of another class in that area).
4. It needs to have the times of day that MWF classes are scheduled and the TR scheduled times.
5. Needs to make sure that it doesn't list it twice if it is cross-listed with another.
6. The ability to add Com S, Phys, and Math courses would be sweet

## Applicable Courses from Iowa State University Curriculum

In CprE 185 (Intro to Computer Engineering and Problem Solving) we had our introduction to computer engineering, we started to learn how to code using basic C.

In COMS 227 (Intro to Object-oriented Programming) we had our introduction to object-oriented programming, this class was a very important stepping-off point for what we are currently doing. We learned Java programming which is close to the programming language we are currently using which is python.

After 227 we took COMS 228 (Intro to Data Structures) which was a more in-depth look into object-oriented programming.

The class COMS 309 (Software Development Practices) took what we had learned in 227 and 228 and was able to put it all together. We learned about the different design approach methods which were waterfall and agile, we also learned about how to work with a front and backend with a web-based application.

Finally, in COMS 311 we would learn how to implement algorithms in order to get software/applications to output what we needed them to output. In our project, we will be using such algorithms to generate a schedule of classes.

### New Skills/Knowledge acquired that was not taught in courses

We decided to go with python programming language for our project because it was something that we were not taught in-depth in any of our courses at ISU. This project is going to be closely related to our COMS 309 software development course but instead of using Java, we will be coding the project in python.

## Table of Contents

1	Team	6
1.1	TEAM MEMBERS	6
1.2	REQUIRED SKILL SETS FOR YOUR PROJECT	6
1.3	SKILL SETS COVERED BY THE TEAM	6
1.4	PROJECT MANAGEMENT STYLE ADOPTED BY THE TEAM	6
1.5	INITIAL PROJECT MANAGEMENT ROLES	6
2	Introduction	7
2.1	PROBLEM STATEMENT	7
2.2	REQUIREMENTS & CONSTRAINTS	7
2.3	ENGINEERING STANDARDS	8
2.4	INTENDED USERS AND USES	8
3	Project Plan	9
3.1	Project Management/Tracking Procedures	9
3.2	Task Decomposition	9
3.3	Project Proposed Milestones, Metrics, and Evaluation Criteria	9
3.4	Project Timeline/Schedule	10
3.5	Risks And Risk Management/Mitigation	10
3.6	Personnel Effort Requirements	11
3.7	Other Resource Requirements	11
4	Design	12
4.1	Design Context	12
4.1.1	Broader Context	12
4.1.2	User Needs	12
4.1.3	Prior Work/Solutions	13
4.1.4	Technical Complexity	13
4.2	Design Exploration	13
4.2.1	Design Decisions	13
4.2.2	Ideation	13
4.2.3	Decision-Making and Trade-Off	14

4.3	Proposed Design	15
4.3.1	Design Visual and Description	15
4.3.2	Functionality	16
4.3.3	Areas of Concern and Development	16
4.4	Technology Considerations	16
4.5	Design Analysis	17
4.6	Design Plan	17
5	Testing	17
5.1	Unit Testing	17
5.2	Interface Testing	18
5.3	Integration Testing	19
5.4	System Testing	19
5.5	Regression Testing	19
5.6	Acceptance Testing	19
5.7	Security Testing (if applicable)	19
5.8	Results	19
6	Implementation	20
7	Professionalism	20
7.1	Areas of Responsibility	20
7.2	Project Specific Professional Responsibility Areas	22
7.3	Most Applicable Professional Responsibility Area	23
8	Closing Material	23
8.1	Discussion	23
8.2	Conclusion	23
8.3	References	24
8.4	Appendices	24
8.4.1	Team Contract	24

# 1 Team

## 1.1 TEAM MEMBERS

Zachary Bunch, Connor Gaecke, Chris Horvatich, and Charles Mulderink

## 1.2 REQUIRED SKILL SETS FOR YOUR PROJECT

- Programming in Python
- Algorithm Design
- GUI Design using Python
- Backend Database with available classes

## 1.3 SKILL SETS COVERED BY THE TEAM

- Programming in Python
  - Charles Mulderink
  - Chris Horvatich
  - Zachary Bunch
- Algorithm Design
  - Connor Gaecke
  - Charles Mulderink
  - Chris Horvatich
- GUI Design using Python
  - Charles Mulderink
  - Chris Horvatich
- Backend Database with available classes
  - Zachary Bunch
  - Chris Horvatich

## 1.4 PROJECT MANAGEMENT STYLE ADOPTED BY THE TEAM

We will be using the Agile Methodology.

## 1.5 INITIAL PROJECT MANAGEMENT ROLES

Zachary Bunch- Team Leader and Back-End Development

Connor Gaecke- Back-End Development

Chris Horvatich- Front End GUI Development

Charles Mulderink- Front End GUI development

## 2 Introduction

### 2.1 PROBLEM STATEMENT

We are trying to make a more efficient and effective way of scheduling the classes within the ECpE department of ISU. Our goal is to make an application that would allow the user to easily generate a class schedule without having to do it by hand. The current method of creating a schedule is the client entering all of the information into an Excel spreadsheet, and manually checking to make sure there are no conflicts within the generated schedule. Our application will automate this process by implementing various algorithms that will check for conflicts between the courses in the schedule it generates. The application will also have the courses grouped in a way that makes it easier for the user to see how the various courses are being distributed within the generated schedule.

### 2.2 REQUIREMENTS & CONSTRAINTS

- IDE to work in (PyCharm)
- GUI editor thinking about using is PyQt5
- Work within schedule of classes (Constraint)
- UI will have a schedule of classes and show what time are available and times that are not.
- Green and Red showing availability

Requirements and constraints from Client

1. Be a document which can be changed with the schedule of classes. If it could be connected to the Schedule of Classes on ISU's website—that would be fabulous. If not, it should be duplicated from the fall before or the spring before.
2. It will need to have the labs associated with the specific rooms.
3. Have them "group" related (so Cpr E 530, 531, 532, 533, 534, 535, 536, 537...show up as one color or some demarcation so that I know I am not putting something on top of another class in that area).
4. It needs to have the times of day that MWF classes are scheduled and the TR scheduled times.
5. Needs to make sure that it doesn't list it twice if it is cross-listed with another.
6. The ability to add Com S, Phys, and Math courses would be nice.

## 2.3 ENGINEERING STANDARDS

What Engineering standards are likely to apply to your project? Some standards might be built into your requirements (Use 802.11 ac wifi standard) and many others might fall out of design. For each standard listed, also provide a brief justification

PEP 8 – python coding standard to keep our notes and code unified.

Due to the nature of our project being all software and not network enabled we do not have many technical standards we are required to follow.

## 2.4 INTENDED USERS AND USES

Who benefits from the results of your project? Who cares that it exists? How will they use it? Enumerating as many “use cases” as possible also helps you make sure that your requirements are complete (each use case may give rise to its own set of requirements).

The primary person benefiting from this project is the advisor who is generating the schedule of courses for the ECpE department. The main benefits of this project is that it will decrease the amount of time needed to generate the schedule and it will create a more efficient schedule. Students will also benefit from this project since it might be able to reduce the chance of scheduling conflicts that arise for students. Professors will also benefit from this application since it will allow them to see where there are possible time slots to add courses into the schedule. This can decrease the amount of time that professors spend trying to see where they can add new courses or change existing courses within the schedule.

# 3 Project Plan

## 3.1 PROJECT MANAGEMENT/TRACKING PROCEDURES

We are adopting agile. This fits our goals because for our project we can start with our base code and update it accordingly to accommodate feedback from the client. Being able to present earlier versions of the project to the client will allow us to get a better understanding of what they would like to see changed going forward. This will help reduce the likelihood of last-minute changes to the project and reduce the amount of last-minute debugging errors since we would be testing early and often.

We are utilizing Trello and Git to organize our goals and objectives for this project. These tools will help us stay organized and allow us to easily break large tasks into smaller and more manageable tasks. This will help reduce the complexity of the required tasks for the



team to implement and allow us to have a clear path to implement the features within the project.

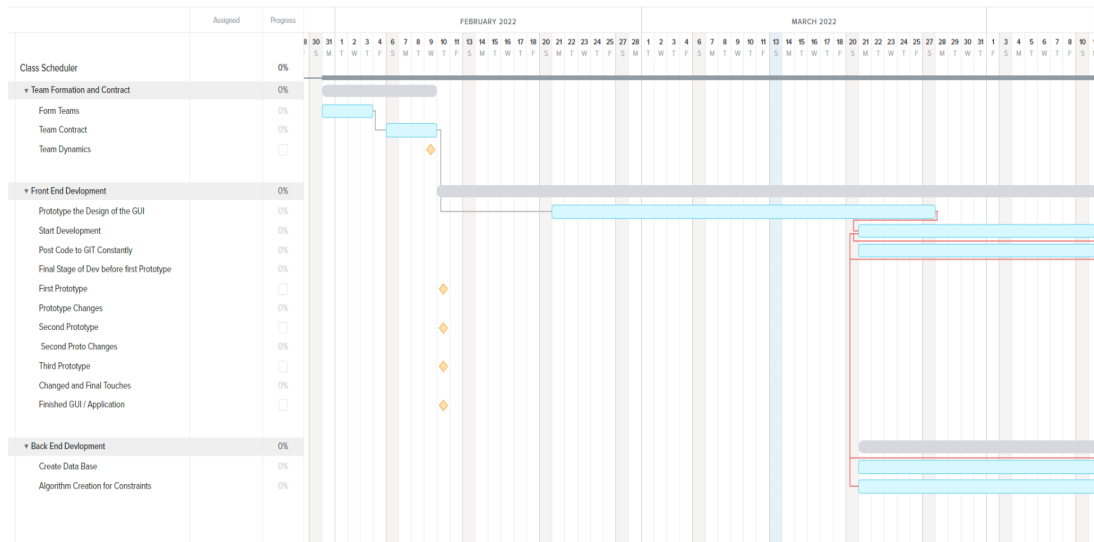
### 3.2 TASK DECOMPOSITION

- Front End:
  - UI/UX design for application
    - Time-table that outputs schedule that algorithm produces
      - Time-table is modifiable by user
    - “Lock” certain classes into specific time slots
      - Algorithm should be reran afterwards at user discretion
- Back End:
  - Create a proper database for storing application information
    - Classes and Labs
    - Corresponding Rooms
    - Class Prerequisites and Corequisites
  - Algorithm that schedules classes based on:
    - Class Prerequisites and Corequisites
    - Predefined times/rooms for certain classes
  - Create output packet that can be interpreted by front end application

### 3.3 PROJECT PROPOSED MILESTONES, METRICS, AND EVALUATION CRITERIA

1. For the frontend, functional GUI that has boxes that are able to be dragged and dropped (not connected to backend yet). For the back end, implement a database structure to store the various classes and labs.
2. For the frontend, functional GUI that is able to interact with the backend. For the backend, implement the CRUD functionality needed by the frontend.
3. For the frontend, implement the time table part of the GUI. For the backend, implement the scheduling algorithm that will generate the output for the class schedules.
4. For the frontend, add remaining features like locking the classes. For the backend, implement remaining features and refine the scheduling algorithm.

### 3.4 PROJECT TIMELINE/SCHEDULE



### 3.5 RISKS AND RISK MANAGEMENT/MITIGATION

Risks:	Risk Probability:	Risk Mitigation:
The framework we choose does not implement all of the capabilities that we need.	0.25	
The algorithm does not create the desired output.	0.5	Using the Agile methodology, we will test the algorithm early and often to make sure there are no last minute issues with the functionality of

		the algorithm.
The project expands beyond the original scope for the project and adds more features than we originally anticipated.	0.5	Using the Agile methodology will allow us to present several versions of the project to the client so we can add any additional features into the project earlier on and not at the last minute.
The GUI does not meet the clients expectations.	0.5	Using the Agile methodology will allow us to present several versions of the project to the client and allow us to see what they are looking for to improve the GUI.
Poor code quality that delays the debugging phase of the project.	0.2	

### 3.6 PERSONNEL EFFORT REQUIREMENTS

<i>Front End Development</i>	<i>100h</i>
<i>Prototype Design</i>	<i>10h</i>
<i>Coding for Frontend / UI</i>	<i>80h</i>
<i>Research on Python Coding GUI</i>	<i>10h</i>
<i>Back End Development</i>	<i>90h</i>
<i>Algorithms for Creating Schedule</i>	<i>60h</i>
<i>Database Development</i>	<i>30h</i>

### 3.7 OTHER RESOURCE REQUIREMENTS

The main resources we will need during the project is documentation on how to implement various tools within the framework to implement the functionality required for the project. Beyond that, we do not need any other resources for our project.

## 4 Design

### 4.1 DESIGN CONTEXT

#### 4.1.1 Broader Context

Area	Description	Examples
Public health, safety, and welfare	How does your project affect the general well-being of various stakeholder groups? These groups may be direct users or may be indirectly affected (e.g., solution is implemented in their communities)	The health, safety, and welfare of the public should not be affected by our application. If anything it will save the public time and reduce scheduling conflicts for students.
Global, cultural, and social	How well does your project reflect the values, practices, and aims of the cultural groups it affects? Groups may include but are not limited to specific communities, nations, professions, workplaces, and ethnic cultures.	Our application will reflect the practices of the students, advisors, and professors that are involved with classes or scheduling in the EcpE department.
Environmental	What environmental impact might your project have? This can include indirect effects, such as deforestation or unsustainable practices related to materials manufacture or procurement.	Our application should not have much impact on the environment. If anything it might save more paper by making it all self-contained within the application.
Economic	What economic impact might your project have? This can include the financial viability of your product within your team or company, cost to consumers, or broader economic effects on communities, markets, nations, and other groups.	Our application should directly reduce the time that is spent with scheduling for the EcpE department. In turn, this will help the college save money.

#### 4.1.2 User Needs

List each of your user groups. For each user group, list a needs statement in the form of:

Advisors need a way to schedule classes without having to do it by hand while taking into consideration all of the constraints that go along with scheduling classes.

Students need a way to be able to make sure that they are taking the appropriate classes each semester so that they will be able to graduate on time.

Professors need a way to quickly see if they are able to change the time around for one of their classes, or add new experimental classes to the schedule.

#### 4.1.3 Prior Work/Solutions

Something that is as specific as our project has not been done before, products that exist in the market that may be similar to our application would be something like an appointment scheduler. We are creating a product that is specifically for use within the ISU EcpE department and will be developed with only that in mind. Our application will not be used outside of the department, the source code may possibly be used to help another department with scheduling.

#### 4.1.4 Technical Complexity

1. The design of our project contains sufficient technical complexity, the use of python and frameworks within python to complete our application will give us new technical skills that may not have been taught in-depth during our course work.
2. Our application will have a front and back end, forward-facing the user will see a GUI and all of the constraints and automation will happen on the backend.
3. The main engineering principle that is used would be software engineering because we are developing an application but we will also be taking into consideration cyber security. Only specific users should be able to modify the database or constraints for the scheduling of classes.

### 4.2 DESIGN EXPLORATION

#### 4.2.1 Design Decisions

We have decided to use the Python language to create both the front-end graphics interface and back-end algorithm and database. Currently, we are planning to utilize the Kivy framework, which is a Python framework, to help build the front-end of the application. We are also not planning on using a web server for the database.

#### 4.2.2 Ideation

One of the design decisions we have made is how the classes and class requirements are stored in memory. Here are five ideas we thought through to solve this problem.

Hard code the classes- this was our original idea but we quickly realized this wouldn't solve the clients needs sufficiently as the classes requirements couldn't be easily updated with the changing curriculum.

Have a single profile that the program stores - This would have been easy to implement as the program would not need to write the data into a transferable location and all the classes specifications could be easily managed but it is not ideal because the profile could not be easily moved or copied which would make iterating on it difficult.

Have the user input a formatted Excel spreadsheet with the classes- This option would have been easy for us to implement but it would have created a worse user experience for the client. The whole goal of this project is to reduce the amount of work the client has to do by hand, so this solution does not seem like it really achieves that goal since the client would still be entering data into Excel.

Create a database and have it run on a server- This option would have simplified our application but it would have created a worse user experience as well. We cannot expect our client to maintain and debug a server with a database application running on it, so this option would lead to potential problems in the future when the database application needs to be updated. It could also lead to issues where the server might be down and the client isn't able to bring the server online and that would make the application useless. That is why we decided against this solution.

Storing the profiles as JSON files - This is currently our plan for saving the class list. This approach is good because the user can copy and iterate the class list without interfering with previous work. The user can have multiple profiles with multiple setups which will make planning future semesters easier. The user can also make edits without saving them to the file so experimentation comes at no cost of losing work.

#### 4.2.3 Decision-Making and Trade-Off

Idea:	Pros:	Cons:
Hard-coding the classes	<ul style="list-style-type: none"> <li>All class data is stored in a single file</li> </ul>	<ul style="list-style-type: none"> <li>Has to be updated manually</li> </ul>
Single Profile of classes	<ul style="list-style-type: none"> <li>All class data is stored in a single area</li> </ul>	<ul style="list-style-type: none"> <li>Has to constantly edit a single profile with no way to access earlier versions</li> </ul>
Storing profiles as JSON	<ul style="list-style-type: none"> <li>Multiple class setups</li> <li>Can make edits without editing any files</li> </ul>	<ul style="list-style-type: none"> <li>Many profiles will take up more local storage</li> </ul>

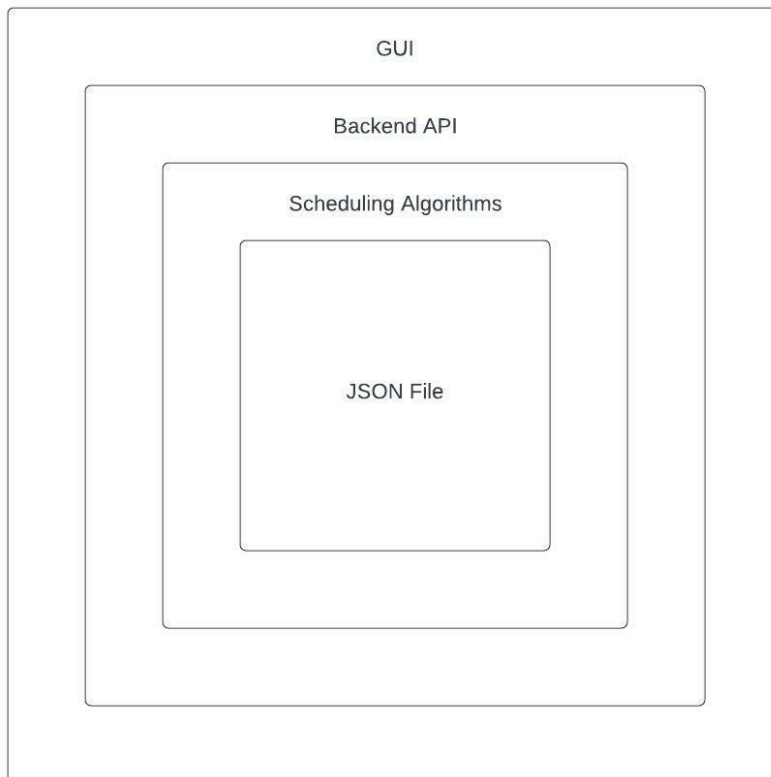
Web Server	<ul style="list-style-type: none"> <li>• Can use database tools/programs to make edits</li> </ul>	<ul style="list-style-type: none"> <li>• Needs Internet Connection to run</li> <li>• User has to maintain web server</li> </ul>
Excel Spreadsheet	<ul style="list-style-type: none"> <li>• Ease of access to class data</li> </ul>	<ul style="list-style-type: none"> <li>• Excel spreadsheets are already being used and we are trying to move away from them</li> </ul>

We decided to go with storing class profiles as JSON files. We chose this option because it was the most flexible while providing the most benefits to the user.

### 4.3 PROPOSED DESIGN

Discuss what you have done so far – what have you tried/implemented/tested?

#### 4.3.1 Design Visual and Description



This image represents the overall layout of our application in terms of layers. In the innermost layer “JSON File”, this is the file that will act as the database for the application and will be the farthest layer from the user. The next layer up is “Scheduling Algorithms” which will use the information contained within the JSON File to create the class schedule. The next layer up is “Backend API”, which includes all of the CRUD functionality for the database, so it can directly interact with the JSON file, and it includes the APIs to interact with the scheduling algorithms to generate the class schedules. The uppermost layer is the “GUI”, which is the only layer that the user will have direct access to. This layer will handle all of the features regarding displaying the schedule, displaying scheduling conflicts, and any other feature within the GUI.

### 4.3.2 Functionality

Our primary goal for this project is to create an easy-to-use desktop application that requires the user to have a minimal understanding of the application to be able to use it. We decided to use a stand-alone application since it would allow us to wrap all of the layers in the previous image into one executable for the client to run. We did not want them to have to worry about starting up a database on their computer or trying to get them to connect to a database hosted on a remote server. That is why we decided to bundle all of the required components into one application and not split them out between true frontend and backend applications. We also anticipated that this application might be passed on to future advisors and wanted to make the process of sending the application as easy as possible for our client. That also factored into our decision of making this a standalone application instead of multiple moving parts.

### 4.3.3 Areas of Concern and Development

Our current design is not easily collaborative and has no way in-app, of having people submit input to the head advisor scheduling the classes. This is not new, the current system in place also has no such system, and input is made through emails or other communications. This could be solved by making the program a web app instead of a portable executable but that is something the client said they did not want. Plans going forward on this is to talk to our client about whether they want collaboration and input to be something that can be done through our solution and if it is we may need to expand the scope of our project and build an input solution that can be used by more than one user.

## 4.4 TECHNOLOGY CONSIDERATIONS

Since our project is a self-contained application, there are little to no considerations to be made in regard to technology. The only consideration that we have to make is the language that we are developing our application, but over the last few months, we have concluded that we made the correct decision in choosing to develop our application with Python.



#### 4.5 DESIGN ANALYSIS

We are not at a point in our project's development where we can get any substantial results. So as of now, our design does not have full functionality.

#### 4.6 DESIGN PLAN

Describe a design plan with respect to use-cases within the context of requirements, modules in your design (dependency/concurrency of modules through a module diagram, interfaces, architectural overview), module constraints tied to requirements.

Since our front and back end are modular they can be worked on separately. Our first priority is to create a functional prototype for the front end of our application. Once we have an environment that a user can interact with on a meaningful level we will start to finish our work on the back end. The back end of our application is mostly designing a scheduling algorithm, which we will also be working on at the same time as creating the front end prototype.

### 5 Testing




Testing is an extremely important component of most projects, whether it involves a circuit, a process, a power system, or software.

The testing plan should connect the requirements and the design to the adoption test strategy and instruments. In this overarching introduction, given an overview of the testing strategy. Emphasize any unique challenges to testing for your system/design.

#### 5.1 UNIT TESTING

The units we will be testing are the various methods and functions that are implemented while we code. We believe that taking a granular approach like that will allow us to catch errors earlier on in the process and avoid last-minute debugging errors. Part of our approach will be to create a running document that lists the various methods and functions that are implemented, and within that documentation, it will include a section on how the methods and functions are expected to work. The programmer will also include what test cases they tested their methods and functions with and they will include the results of the tests.

We have not narrowed down the testing tool that we would like to use but it will most likely be one of these.

	Free software (MIT License)	Stand alone, allows compact test suites.	Unit Testing	Special and simple class fixture for making testing easier.
	Free software (MIT License)	Part of Python standard library.	Unit Testing	Fast test collection and flexible test execution.
	Free software (MIT License)	Part of Python standard library.	Unit Testing	Python Interactive Shell for the command prompt and inclusive application.

## 5.2 INTERFACE TESTING

The primary interfaces we will be testing are the APIs that are implemented by the backend. These APIs are responsible for communicating with the database and the scheduling algorithm. These APIs are critical for implementing CRUD functionality within the application.

One interface that will be tested is the interface responsible for converting the JSON file into the application's memory and converting the application's memory back to the JSON file once the application is closed. This interface will be responsible for turning python dictionaries into JSON objects that can be stored within the JSON file and then will also be responsible for converting the JSON objects back into python dictionaries. This interface will be tested by the backend team using several test cases that ensure the required functionality is implemented and that data is not being corrupted during the process of converting it between python dictionaries and JSON objects. The primary goal when testing this interface is to ensure that CRUD functionality has been properly implemented and to make sure that the changes are persistent or ignored depending on how the application is terminated.

Another interface that will be tested is the interface responsible for implementing the CRUD functionality for the application. This interface is responsible for creating, reading, updating, and deleting courses within the database. This interface will be tested using various test cases that ensure the interface is working properly. The goal is to make sure that the various functionalities for the database are working properly and the changes are persistent. Additionally, we want to ensure that none of the data entered by the user is being corrupted by the methods and functions within the interface.

Again we will have the ability to use unit testing here in order to make sure that the implementation of the JSON files is correct.

### 5.3 INTEGRATION TESTING

Our application will have multiple critical integration paths, the first of which is the path that allows for the integration of the algorithms that work on the back in order to show what times are available for specific classes on the front. We will also have an integration path that relates to how we update and modify the JSON files from the front end.

For integration testing, we will be using the application in order to see if the algorithms in place are affecting the outcomes of when specific classes can be scheduled. We may try to use the Robot testing framework.

### 5.4 SYSTEM TESTING

For our system-level testing strategy we are planning on creating multiple unit tests, potentially repurposing ones we have already made, and hooking them together to create a test for full system functionality. We will need to create a unit test that will simulate some user inputs to run the program, and then each step of our application can be tested in sequence and checked for expected outputs.

### 5.5 REGRESSION TESTING

By dividing the project in a large scale between the front end and the back end it makes it easy for the project to be very modular. Buttons on the front end can correlate to methods on the backend. This modularity makes the project resilient to breaking when new functionality is introduced.

### 5.6 ACCEPTANCE TESTING

We plan on demonstrating the project to the client and talking through what requirements we have successfully satisfied and what requirements need more work in the future. Doing this repeatedly will result in all of the requirements being satisfied and the client being happy.

### 5.7 SECURITY TESTING (IF APPLICABLE)

N/A

### 5.8 RESULTS

What are the results of your testing? How do they ensure compliance with the requirements? Include figures and tables to explain your testing process better. A summary narrative concluding that your design is as intended is useful.

The results of our testing strategies will help us locate any bugs that are present in our current design. Since we are employing the Agile methodology, we are able to produce many versions of our product. With the testing framework we plan on using, we will be able to debug and develop the next version of our design faster and more efficiently.

## 6 Implementation

Describe any (preliminary) implementation plan for the next semester for your proposed design in 3.3. If your project has inseparable activities between design and implementation, you can list them either in the Design section or this section.

Next semester we will continue to utilize the agile methodology. We will increase the pace of our agile cycle and conduct sprints more frequently. Next semester we will focus more on client feedback and making the product suitable to solve the issues the client is facing.

## 7 Professionalism

### 7.1 AREAS OF RESPONSIBILITY

IEEE Code of ethics

Area of responsibility	Corresponding IEEE code of ethics rules	How they differ
Work Competence	6. to maintain and improve our technical competence and to undertake technological tasks for others only if qualified by training or experience, or after full disclosure of pertinent limitations;	These rules are mostly the same but they differ in the way of the IEEE code adding the stipulation of allowing work outside the engineer's competency if their limitations are disclosed.
Financial Responsibility	3. to be honest and realistic in stating claims or estimates based on available data; 4. to reject bribery in all its forms; 9. to avoid injuring others, their property, reputation, or employment by false or malicious action;	The NSPE rule in this area is much more specific to the issue of unfaithful action to a client but the IEEE code lays out specific things that an engineer should not do that would fall into this category like rejecting bribery.

Communication Honesty	3. to be honest and realistic in stating claims or estimates based on available data;	These rules are very similar with no notable difference.
Health, Safety, Well-Being	1. to accept responsibility in making decisions consistent with the safety, health, and welfare of the public, and to disclose promptly factors that might endanger the public or the environment;	The IEEE rule has the added clause of disclosing danger factors but besides that these rules are very similar.
Property Ownership	2. to avoid real or perceived conflicts of interest whenever possible, and to disclose them to affected parties when they do exist; 9. to avoid injuring others, their property, reputation, or employment by false or malicious action; 10. to assist colleagues and co-workers in their professional development and to support them in following this code of ethics.	For this NSPE has a much clearer rule for this as IEEE only mentions a few specific things to not do but doesn't have a rule that's broad to cover everything that NSPE's rule does for this aspect.
Sustainability	1. to accept responsibility in making decisions consistent with the safety, health, and welfare of the public, and to disclose promptly factors that might endanger the public or the environment;	NSPE does not have a great rule that clearly pertains to this. It does say to hold public health and welfare in a high regard but nothing specifically about sustainability.
Social Responsibility	7. to seek, accept, and offer honest criticism of technical work, to acknowledge	The NSPE rule for this is really a catch all that doesn't really say anything because it could be interpreted many different

	and correct errors, and to credit properly the contributions of others; 8. to treat fairly all persons and to not engage in acts of discrimination based on race, religion, gender, disability, age, national origin, sexual orientation, gender identity, or gender expression; 5. to improve the understanding of technology; its appropriate application, and potential consequences;	ways. The IEEE rules for this are more specific.
--	--	--

## 7.2 PROJECT SPECIFIC PROFESSIONAL RESPONSIBILITY AREAS

For each of the professional responsibility areas in Table 1, discuss whether it applies in your project’s professional context. Why yes or why not? How well is your team performing (High, Medium, Low, N/A) in each of the seven areas of professional responsibility, again in the context of your project. Justify.

Area of responsibility	Applies to us?	How are we performing?
Work Competence	Yes, this applies universally to all projects.	High, we are performing well. It's a bit early to tell but we are trying hard.
Financial Responsibility	No, as it's a software project we really have no budget	N/A
Communication Honesty	Yes, we maintain communication with the client and do the weekly reports	High, we haven't lied about anything yet.
Health, Safety, Well-Being	No, our app poses no health risk.	N/A
Property Ownership	No, we meet with the client remotely.	N/A

Sustainability	No, we're building software that does not impact the environment.	N/A
Social Responsibility	Yes, the product will have a positive effect for our client.	High, we are building an app to make life easier for the person who schedules engineering classes at Iowa State and we plan on it being helpful.

### 7.3 MOST APPLICABLE PROFESSIONAL RESPONSIBILITY AREA

Social responsibility. Our product will help the Iowa state community by being able to efficiently plan classes so students have the most options for classes and faculty have the benefit of not needing to do this manually. Our client (Vicky) will be retiring soon and she wants an application that can automate the scheduling of EcpE classes the same way she does it manually. We hope that this Class Scheduler will help the next advisor that has to schedule the EcpE classes.

## 8 Closing Material

### 8.1 DISCUSSION

Our product currently does not meet all the requirements that it will by the end of next semester. Key features still need to be implemented such as saving and opening schedules, class sort, comprehensive UI and more. This semester was primarily focused on developing a Design Document that would guide us towards implementing the final product in the next semester. We believe that we have developed a path for success in developing the final deliverable.

### 8.2 CONCLUSION

Currently, we have been able to design an outline for our final deliverable and created a design document to help guide us towards developing a quality final product. Our primary goal is to create an application that allows our client to more efficiently create the schedule of courses for the ECPE department. Within that goal are several sub goals that include making the course scheduler automated, having a graphical user interface that creates a more positive experience for the user, and to reduce the chance of scheduling conflicts. The primary thing that has constrained us from achieving these

goals is that we are very inexperienced with Python and the GUI frameworks that work with Python. There is nothing that can be done in future design iterations since using Python was the best design choice we could make when it came to selecting a programming language. Going forward in future implementation iterations, we will have a better understanding of Python and the frameworks that work with Python. This will allow us to more effectively work on the project and develop a better final product.

### 8.3 REFERENCES

N/A

### 8.4 APPENDICES

Any additional information that would be helpful to the evaluation of your design document.

PEP 8 Style Guide: <https://peps.python.org/pep-0008/>

#### 8.4.1 Team Contract

##### **Team Members:**

- 1) Zachary Bunch
- 2) Connor Gaecke
- 3) Chris Horvatich
- 4) Charles Mulderink

##### **Team Procedures**

1. Day, time, and location (face-to-face or virtual) for regular team meetings:
  - a. **Wednesdays at 4 pm on Discord**
2. Preferred method of communication updates, reminders, issues, and scheduling (e.g., e-mail, phone, app, face-to-face):
  - a. **Discord**
3. Decision-making policy (e.g., consensus, majority vote):
  - a. **Majority vote**
4. Procedures for record keeping (i.e., who will keep meeting minutes, how will minutes be shared/archived):
  - a. **The team leader will keep track of the meeting minutes.**

##### **Participation Expectations**

1. Expected individual attendance, punctuality, and participation at all team meetings:



- a. **Yes, everyone is expected to be present and punctual, unless a notice is given in advance.**
2. Expected level of responsibility for fulfilling team assignments, timelines, and deadlines:
  - a. **Everyone is responsible for fulfilling their responsibilities. If a deadline can not be met the student will inform the group and we will jump in to help on that task.**
3. Expected level of communication with other team members:
  - a. **As needed and weekly meetings.**
4. Expected level of commitment to team decisions and tasks:
  - a. **Equal commitment among team members.**

### **Leadership**

1. Leadership roles for each team member (e.g., team organization, client interaction, individual component design, testing, etc.):
  - a. **Chris Horvatic - Front end, testing, implementation**
  - b. **Connor Gaecke - Back End, design and testing**
  - c. **Charles Mulderink - Front end, interface design, prototyping**
  - d. **Zachary Bunch- Team Leader, Back End, Client Interaction**
2. Strategies for supporting and guiding the work of all team members:
  - a. **Constructive criticism**
3. Strategies for recognizing the contributions of all team members:
  - a. **Using git, meetings where team members demonstrate what they have worked on.**

### **Collaboration and Inclusion**

1. Describe the skills, expertise, and unique perspectives each team member brings to the team.
  - a. **Charles Mulderink - python knowledge, worked as a platform engineer.**
  - b. **Chris Horvatic - A bit of python, CprE 309 gave me software expertise**
  - c. **Connor Gaecke - Programming Knowledge, 309 group project experience**
  - d. **Zachary Bunch- Python knowledge, 309 group project experience**
2. Strategies for encouraging and support contributions and ideas from all team members:
  - a. **Grading and being a good team member/human**
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?)
  - a. **Bring issues up to the team leader, and they will resolve the issue between the team members.**

## Goal-Setting, Planning, and Execution

1. Team goals for this semester:
  - a. **Get the design/prototype ready for the class scheduler and start to implement.**
2. Strategies for planning and assigning individual and team work:
  - a. **We will assign individual and team work using Trello.**
3. Strategies for keeping on task:
  - a. **We will assign due dates on the Trello cards.**

## Consequences for Not Adhering to Team Contract

1. How will you handle infractions of any of the obligations of this team contract?
  - a. **Bring it up to the team and come to a solution as a team**
2. What will your team do if the infractions continue?
  - a. **Talk to our faculty advisor about the problem**

\*\*\*\*\*  
\*\*\*

a) *I participated in formulating the standards, roles, and procedures as stated in this contract.*

b) *I understand that I am obligated to 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) <u>Zachary Bunch</u>     | DATE <u>2/09/2022</u>  |
| 2) <u>Connor Gaecke</u>     | DATE <u>2/09/2022</u>  |
| 3) <u>Chris Horvatich</u>   | DATE <u>02/09/2022</u> |
| 4) <u>Charles Mulderink</u> | DATE <u>02/09/2022</u> |