

# Circuit Drawing and Simulation Website

## PROJECT PLAN

Team Number: SDMAY19-44

Client: Professor Andrew Bolstad

Adviser: Professor Andrew Bolstad

### Team Members/Roles:

Joseph Veal: Back-End Code Leader

Alexandra Sutton: Meeting Facilitator & Scribe

Lucas Maring: Report Manager &

Keegan McCarthy: Team Leader

Cassandra Plata: Front-End Code Leader

Tyler Schurk: Back-End Code Leader

Team Email: [sdmay19-44@iastate.edu](mailto:sdmay19-44@iastate.edu)

Team Website: <https://sdmay19-44.sd.ece.iastate.edu>

Revised: December 2<sup>nd</sup>, 2018 / Version: 3.0

# Table of Contents

---

<b>1 Introductory Material</b>	3
1.1 Acknowledgement	3
1.2 Problem Statement	3
1.3 Operating Environment	4
1.4 Intended Users and Intended Uses	4
1.5 Assumptions and Limitations	5
1.6 Expected End Product and Other Deliverables	5
<b>2 Proposed Approach and Statement of Work</b>	6
2.1 Objective of the Task	6
2.2 Functional Requirements	6
2.3 Constraints Considerations	7
2.4 Previous Work And Literature	8
2.5 Proposed Design	9
2.6 Technology Considerations	11
2.7 Safety Considerations	12
2.8 Task Approach	12
2.9 Possible Risks And Risk Management	14
2.10 Project Proposed Milestones and Evaluation Criteria	15
2.11 Project Tracking Procedures	15
2.12 Expected Results and Validation	16
2.13 Test Plan	17
<b>3 Project Timeline, Estimated Resources, and Challenges</b>	18
3.1 Project Timeline	18
3.2 Feasibility Assessment	22
3.3 Personnel Effort Requirements	22
3.4 Other Resource Requirements	20
3.5 Financial Requirements	24
<b>4 Closure Materials</b>	24
4.1 Conclusion	24
4.2 References	25

## List of Figures

---

Figure 1: Drawn Prototype of Web Application

Figure 2: Relationship Between Django and Supporting Languages

Figure 3: Process Diagram of Work to be Accomplished

## List of Tables

---

Table 1: Testing Application

Table 2: Project Timeline

Table 3: Visual Project Timeline and Deadlines

Table 4: The Estimated Time Required to Complete Various Components of the Project

## List of Definitions

---

**CSS:** Cascading Style Sheets; a style language used for changing the style of HTML files

**Django:** A Python-based Web Framework

**GUI:** Graphical User Interface; the visual software component that users view

**HTML:** Hypertext Markup Language; is the standard markup language for creating web pages and applications

**PNG:** Portable Network Graphics; the file format that the circuit schematics will be saved in

**Python:** an interpreted language that is nice for scripting because it is an open source language that has a wide range of modules for developers to pick from. Python can be used as a server side language as well; and because it interacts with servers so well, it can be used to manage the backend of a web dev application

## 1 Introductory Material

---

### 1.1 Acknowledgement

We would like to thank our client, Professor Andrew Bolstad, for providing us with this project. We all hope to create a website application that will be an educational tool for students and develop our own programming skills in doing so.

### 1.2 Problem Statement

Professor Andrew Bolstad teaches Electrical Engineering courses for Mechanical Engineers; however, he is unhappy with his lecture notes concerning circuit diagrams. He is currently using PowerPoint to create circuit diagrams for his lectures, but he found that it is too time

consuming and difficult to create the circuits that he wants. Rather than using pre-made circuit components for his notes, he is using the shapes found in PowerPoint to build his circuits. He wants his circuit components to have thick borders, so they are clearly visible from the grid background, and he wants the wires to stay connected to circuit components when the components are moved around. PowerPoint, however, is unable to function in this way. Furthermore, our client is unable to save the circuits that he creates, in order to use the circuit diagram for other class notes or for homework; and so he is continuously building new circuit diagrams. This has been a very time consuming and labor-intensive task for creating his class lectures, so Professor Bolstad is looking for a solution - a website that is intuitive and easy to use to make visually appealing circuit diagrams. He wants a circuit diagram that is neat, easy to read, easy to comprehend, and has labels for each circuit component. In addition, he wants to save these circuit diagrams for later use, efficiently create any circuit he needs in a timely manner, and possibly simulate the circuits for educational practices.

Although there are currently circuit drawing websites available that Professor Bolstad can use, no one website has everything he needs. Therefore, we will create a website application that combines several features of various websites in order to meet every need of Professor Bolstad and his students. We will create visually appealing circuit drawings using Python and JavaScript in order to create a website that can be updated and altered to tailor Professor Bolstad's needs. Our website will include easy-to-use rotate and delete features, it will have a "components" tab that will be an educational tool for the user to learn about various components, it will have a "help" tab that explains the program, and the user will have the ability to move the circuit around while maintaining wired connections.

This website application will not only be a tool for students inside Professor Bolstad's class to use, but it will be a tool for any students willing to learn about circuit theory. We hope to create both an efficient circuit drawing website and an educational resource that will teach students the fundamentals of circuit theory and application.

### 1.3 Operating Environment

The operating environment for this project is the server/database that we plan to use. This database will securely store all of the data needed for this project, and so it will not be exposed to harsh conditions or security threats.

Furthermore, this website will primarily be used by our client, Professor Bolstad, but it will also have the potential to be used by any students either inside or outside of the class who are interested in learning more about circuit design and simulation. Any outside user will not be able to alter any of the code associated with the project, and so the user will not be able to harm the website in any way.

### 1.4 Intended Users and Intended Uses

This project intends to provide a free way to draw, design and simulate circuits. This project is to be designed with academic users in mind, but it can be used by anyone who wishes to draw or simulate circuits. The goal is to provide users with a visually pleasing,

downloadable schematic that will be free to use for the public. The Simulation App will be web based so that the application is easily accessible for all. This will also remove the need for users to download and install a software application to be able to use the Simulation app. Users will be able to work with our Simulation app on any device that has internet connection capabilities.

There are very little options for drawing and simulating circuits for free. There are many web based tools that will simulate circuits for free, but none of the tools available can meet the drawing specifications our client needs. Users of the Simulator Application will need the tool to create a PNG of the schematic they designed, since no other simulation or drawing tool possesses that functionality. These were the observations of our client. Because our client will be using these circuit schematics in his lectures, he needs an elegant way to integrate them into PowerPoint. The criteria for this project was designed with our client in mind; however, the end result of the project will provide convenience for all users.

## 1.5 Assumptions and Limitations

Assumptions:

- Django and Python will be relatively easy to learn and use, making for a simplistic end user result
- Server space will be provided by ISU ECpE for user login
- People will be able to access the web app from anywhere (not just ISU students)

Limitations:

- Learning a brand new coding language for each group member slows down completion of the project
- The ability of the group implementing features that can be used in the simulation app, such as having the wires staying connected while the circuit is moved around
- The ability to complete each task within the timeframes set out

Note: We will discover more limitations as the project moves along, and have more assumptions as well

## 1.6 Expected End Product and Other Deliverables

The expected end product and other deliverables will be the following:

**Prototype I Completion Date: December 15th 2018**

Prototype I is intended to be a test to show how things will be displayed and overall functionality. Prototype I will consist of a functional GUI with the following capabilities.

- Drag and Drop component features
- Functional website layout (HTML/CSS side of project)

- Front End displayable GUI (Just on a localhost, rather than a server)

### **Prototype II Completion Date: February 16th 2019**

Prototype II will be an add-on to the first prototype. Prototype I will be a finished final product for the circuit drawing. Prototype I will include all functionality except the circuit simulation. The following capabilities will be added after prototype I:

- Running a basic website for testing purposes
- Easy to use circuit drawing
- Picture Exporting into a PNG file
- Hotkeys
- Wires move with components

### **Final Product Completion Date: April 20<sup>th</sup> 2019**

The final product will include an immersive circuit drawing website with basic simulation capabilities as a stretch goal. It will run seamlessly with the following:

- User circuit saving interface
- Pre-drawn basic circuits
- Tutorial interface

## **2 Proposed Approach and Statement of Work**

---

### **2.1 Objective of the Task**

Our client, Professor Andrew Bolstad, has requested a solution to his need of creating a free, user friendly, graphically appealing circuit drawing and simulating application. Our first stage of the task is to create an application that has circuit drawing capabilities. The second stage, which is also a stretch goal of the project, is to add a simulating functionality to the application. Because our client would like to easily save the circuit and use it in his lecture slides, the drawing functionality should not only be intuitive, but the end result should have a reasonably attractive graphic look to it as well.

### **2.2 Functional Requirements**

Our website will function as both a circuit drawing tool and a simulator, therefore, the functional requirements will be split in two categories. Initially, a user should be able to place components, connect them, and then create an image for placement elsewhere. The end result should be a visually appealing, user-created circuit that can be transferred into reports or presentations. A stretch goal will be the implementation of a circuit simulator that will be limited initially and fleshed out as time goes on. In addition, a component

information tab and the simulator will be a part of the learning tool aspect of the website for students to understand circuit behavior.

## Drawing

**FR.1:** The website allows for placement of circuit elements, such as wires, voltage sources, resistors, capacitors, and other similar components.

**FR.2:** Circuit elements should stay connected with a wire if they share the same node and are dragged around the window.

**FR.3:** Give users the ability to save created circuits for use later, which will require the implementation of user accounts.

**FR.4:** The schematic G.U.I. should display the basic component information next to its element, i.e.  $1k\Omega$  next to a resistor.

**FR.5:** A user can generate a picture of the created schematic and have it placed into the computer's clipboard for pasting on other documents.

## Simulation

**FR.6:** Simulating displays information such as voltage/current at nodes and components.

**FR.7:** The user can attach a scope element to view a signal at a node.

**FR.8:** Equations for components should be displayed when requested in order to expand on the simulation results

## 2.3 Constraints Considerations

A project of this caliber most often includes functional requirements and nonfunctional requirements. In many cases these requirements are given by the user. Some examples of these types of requirements are protocols and security standards. This project has given our team lots of freedom to choose protocols and procedures that we must adhere to. On the other hand, since we are doing a web development project, there are certain protocols that we must follow in order to have our application operational.

One of the protocols that is required for our project is the HTTP protocol. This is a standard web protocol that allows for web development languages to be sent as data over the internet. The HTTP protocol will help us with verification. HTTP has a variety of status codes that will allow us to debug issues by reading the server log. Another protocol our team needs to keep in mind is the IP protocol. This protocol will be important when we are testing and setting up the server. The IP protocol assigns our server an address, and allows other machines to make HTTP requests to it. Once we migrate our server we will have a permanent IP address, but while we are in testing we will take advantage of one of the localhost ports for our HTTP requests. These protocols are more important for networking than for web development; however, they must be followed if our project is to work

correctly. To automatically handle these protocols, we have investigated web development techniques that will help us meet our nonfunctional requirements.

Our group did extensive research to determine which coding languages and frameworks we should utilize to develop the best version of our project. Our conclusion was that we would use Python and the Django framework to manage the backend, and JavaScript to create a GUI for our front end. Using a framework will allow us to control the database and any routing that the web server we will be using. The management of routing will allow us to adhere to the standards of the IP protocol and help us achieve our nonfunctional and functional requirements. Simply choosing standard web development languages will allow us to use the HTTP protocol correctly.

There is a possibility that our team will need to follow a security protocol. A simple and very effective protocol that will allow us to protect our website is the OAuth protocol. If our group determines that this protocol is needed, signing into the website will be required to access the content. In order to sign into the website, the user will need to present a valid username and password. The Django framework will also allow us to use modules to perform security functionality and the framework does have a OAuth module available.

## 2.4 Previous Work and Literature

While there are several circuit drawing websites currently available, there is not one website that fits all of Professor Bolstad's needs. Regardless, we researched similar products to draw inspiration of potential components we could add to our project.

For example, Falstad is an excellent simulation and circuit drawing website, but the circuits are not visually appealing for lecture slides (*Circuit Simulator Applet*). The background of the circuit simulator is black while all of the components are brightly colored; and this is not what our client is looking for. He wants to place the circuits that he builds into his PowerPoint slides and he wants the images of the circuits to be bright, easy to read, and easy to understand. On the other hand, this website has several default hotkeys that allow the user to quickly draw circuits and all circuits can be saved and shared with others. Professor Bolstad wants to save images of his circuits, without using a screenshot or snipping tool, so this will be one feature that we will include in our circuit drawing website.

Furthermore, Partsim (SPICE) is a circuit drawing website that has a wide library of components and it is intuitive to use, but the software is not free ("Pspice User Forum"). Professor Bolstad's students, therefore, would not be able to benefit from this educational program because of this cost barrier. Our goal is to create a free circuit drawing website that will be available to anyone who is interested in using it. However, unlike Falstad, this site has a white background and circuit components that are easy to see. Therefore, this circuit display will be very similar to the circuit display that we will be creating.

Another example of a circuit drawing website is Circuit Diagram (*Circuit Diagram Web Editor*). This website allows users to download and save circuits, simulate the circuit, and view an image of the available components in a separate "components" tab. Although this website has good features, none of the features are exceptionally great. For example, the component image for the resistor is simply a block with a label above it. Because this is an atypical design for a resistor, students will not benefit from this visual because this block is

not commonly seen. Therefore, we plan to implement commonly seen component visuals so students will be able to more effectively follow our client's class notes. Furthermore, we decided to create an informational "components" tab that will teach students about various components and supply associated equations so students will be able to perform calculations by hand. This will enhance this circuit drawing website because it will become an educational tool for students who want to learn more about circuit theory.

In addition, the website CircuitLab is an intuitive circuit drawing website, but it is not easy to use (*CircuitLab - Editing "Unnamed Circuit"*). It is not easy to rotate, move, and delete components from the circuit, which defeats the intention of an efficient circuit drawing website. Rather than simply creating circuits right from the start, the student would need to learn how to use this website first. This website does, however, have an informational "help" tab that is useful for first time website visitors to use when using the website for the first time. Therefore, although we plan to create a circuit drawing website that is very intuitive to use, we will also create a "help" tab that explains the program to someone who may have difficulty. We will also include more efficient rotate, move, and delete features into our website, in order to eliminate potential confusion when building circuits.

## 2.5 Proposed Design

The project consists of implementing two major components: the ability to create a visually appealing circuit and, later on, the ability to simulate the circuits that have been drawn. The second part, simulation, is a stretch goal that will hopefully be completed later in the second semester. Implementing these two components will consist of a backend structure that will be managed by the framework Django, which will allow us to use Python. The front-end will be coded in HTML and JavaScript, allowing us to design our web app in a basic, but user-friendly manner.

Possible design alternatives could include using a different framework called Spring, which uses Java as its language, if we are unsuccessful integrating the HTML files into a Django template. This would change the procedure of our project, and it may not even be possible to integrate the two. Other alternative ideas that we have discussed, regarding additional components to add to our main project, would be adding more complex electrical components such as diodes, transistors, etc.

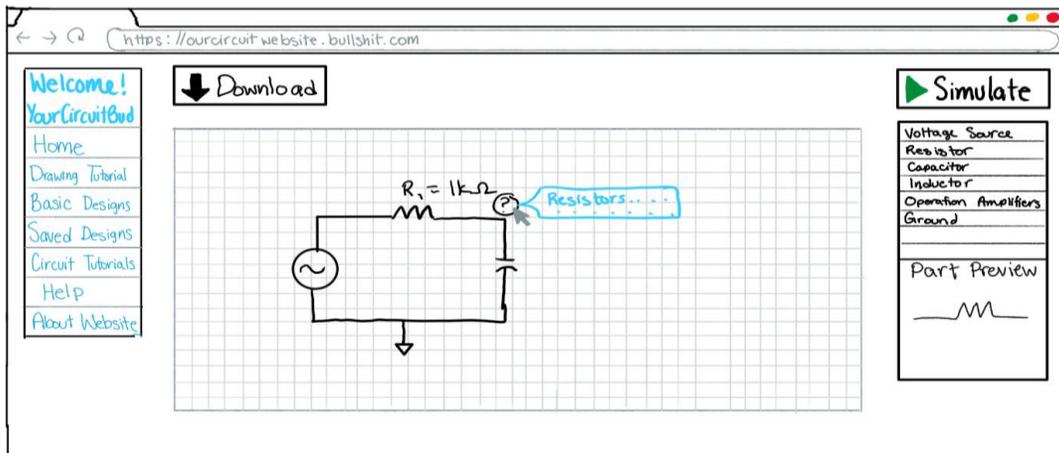


Figure 1: Drawn Prototype of Web Application

The figure above is a drawn prototype of what we expect our web application to look like. The left column will consist of buttons and tabs consisting of tutorials, examples, and a help tab. In the middle is the “drawing grid” which is where all of the components will be dragged and dropped into a DrawSpace to create the picture our client needs. Whenever a component is dropped into the grid, there will be an option to give that component a name and a value like shown above. A possible goal that we may include will be the question mark next to the component, giving information about that component. On top of the drawing grid is the download button that allows the user to save the image of the drawing space to wherever the user requests. Lastly, we have the simulation button and the components on the right. The simulation will be a stretch goal towards the end of the second semester where it will simulate the circuit and give output values the user may want. The components will be listed on the right in a fashion to be decided later. A preview of the component selected will be shown as it is above.

One possible weakness we have identified in our prototype is its visual appeal. Our client is looking for basic, but graphically appealing circuits and so the circuits will meet his requirements. However, it seemed to be difficult to make the look of the entire website very visually appealing because it will simply be a simple circuit drawing and simulation application. This is something that may be a stretch goal for us to make the website look much more appealing and detailed.

Another potential weakness with our design, is the fact that people have different sized monitors. When we first started creating the website, everything in the website would appear as how it was programmed to appear because the monitors we were using were quite large. Although if a user tries to use our website with a different sized monitor, then the drawing space may be partially covered by some of the buttons on either side. This is something we need to do more research on and determine if the layout needs to change.

We, as a team, plan to accomplish these tasks by splitting up into three smaller teams of two people. These smaller teams will tackle separate parts of the project so we can complete more tasks in a shorter period of time. An example of this separation would be one team working on the HTML and CSS side of the web application, another team working on the

Python and Django side, and the final team working on the JavaScript part to complete the integration of everything.

## 2.6 Technology Considerations

Our team will need to use many different software tools and coding languages to develop our simulation app. Our team has done considerable research to determine which tools will allow us to provide a product with the best functionality. The following paragraphs describe the tools that will be used to develop our simulation app, the languages required for development, and hosting options for the project.

To manage our code, our team plans to use Git. Git will allow us to easily control versions of our project and a code manager will have to approve changes before they are made to our main repository. Git is a commonly used and free tool that can be downloaded on any operating system. Each team member will have a Git repository on their computer that will interact with an online repository on GitLab.

Web development traditionally involves the use of several languages. These languages are each designed to handle specific tasks such as SQL for databases, HTML for basic website layout, and CSS for website design and color. Our team has decided to remove the need for some of these languages by using a framework. A framework will allow our team to manage all of the backend code with one language. This will remove the need for PHP, SQL, and some of the JavaScript we were planning to use. The framework our team has selected is called Django. Django allows us to manage the backend of the website using Python. We have chosen to use Django because our team has limited experience with object oriented programming. This makes other popular frameworks, such as Swing, difficult for our team to use. Django gives us the ability to get a version of our website running quickly. This will give us more time to work on the GUI features, which will be our most important deliverable. The Django ORM also allows us to control the database without the need to use SQL, an additional programming language. The backend will control which page is currently loaded and the data that flows through it. It will also allow us to handle any necessary background processing as well.

Our team is investigating many options for implementing drag and drop features for our GUI. Some of the options we have considered include using the HTML div tags with a draggable attribute and a DrawSpace area within a section of the webpage. Each of these options have positives and negative parts. The HTML div tags are very useful for dragging a component into the target area; however, the control of component placement is lacking for this option. The DrawSpace gives us plenty of control over the placement of components, but it does not allow for components to be dragged in from another section. We are currently working to see if we can use one of these tools to contain our GUI.

In addition, HTML is an important tool in any web development project. HTML allows a developer to design the basic layout of a website. It has specific tags that will be used to determine the section of a website for a piece of text. It will also be used to add links and many other basic functions of a website. In our case, HTML will exist inside of a Django file known as a template. The file will be rendered in the browser of the requester once the Django view has finished any backend processing.

Our team felt the best option to manage our backend was Python because Python is an easy to learn language. Python is an interpreted language which makes it difficult to debug at times; however, it does not require a compiler which will be advantageous in the future. This is an advantage to us because it allows Python to function as a scripting language. Python also has a wide range of modules that our team can use to efficiently run backend functionality.

Our team has realized that unforeseen obstacles will require us to make decisions we cannot account for in this document. We address these issues by holding weekly meetings to allow for group input on decisions. Our team has pondered decisions such as creating data messages and labels and decided to delay these decisions in order to ensure that the choice we make is compatible with our current developments. We made the choice to delay some decisions because they are dependent on parts of the design that are subject to change.

## 2.7 Safety Considerations

Because our project is a website application, there are no safety considerations to mention.

## 2.8 Task Approach

In order to begin this project, we met with Professor Bolstad and created a list of deliverables. The deliverables involved creating a web based application that included visually appealing and easy to read circuit schematics, the ability to download schematics, and several drag and drop features. Our team then began meeting to determine how we could create a product that would meet Professor Bolstad's needs. After extensive research into what web development tools would be best for the task, we determined that we would have to use certain web development tools such as JavaScript. Once we determined what tools we needed to complete the project, our team broke down the project into sub-components and assigned team members to work on those parts. The team then decided what coding tools would be used to complete each part. The following paragraph describes how we plan to use each tool to complete each task:

Arguably, the most important part of the project is the graphic user interface. After completing our research, we determined the only effective way to create a drag and drop interface was to use JavaScript and open-source JavaScript drawing libraries. JavaScript is one of the main web development languages and is ideal for this application. JavaScript can be easily integrated into HTML, which is another web development language that we will utilize to organize the location of components and text on our website. The JavaScript is embedded in its own section within the index.html file. By organizing the JavaScript functions in the file, we can create HTML containers and define their characteristics by calling these JavaScript functions. For example, a <div> container can be created, given the ability of "draggable," and then it can call original JavaScript functions for "onclick" and "ondrag". This will give the ability that, when a button is pressed, creates a <div> containing an image that can be dragged around the space.

Furthermore, in order to manage all of our web pages, we chose to use a Python framework to simplify routing. We are currently implementing Django which allows us to manage the URL patterns within the project. Because we will have several tabs that the user will be able to view within the project, Django allows us to manage these requests to the user. The relationship between our code and the Django framework is shown below in Figure 2.

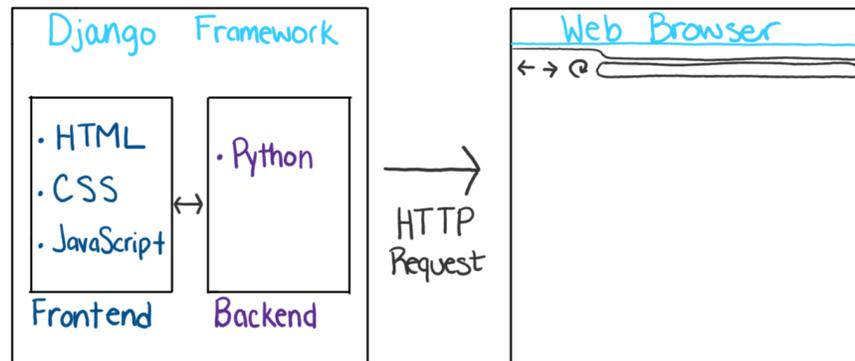


Figure 2: Relationship Between Django and Supporting Languages

Moreover, each team has been working to get sub-components of the project completed and unit tested. We will then meet with Professor Bolstad to see if the sub-components need any revision or if they are ready to be implemented into our final product. Once we have components that can deliver our end functionality, we can begin integrating them into the prototype. We will also have a second round of revisions with Professor Bolstad to ensure that our project meets his expectations. Ultimately, we will be able to move into the implementation of the project and publish our website.

Below is a diagram of the basic process we will use when creating our Circuit Drawing Website:

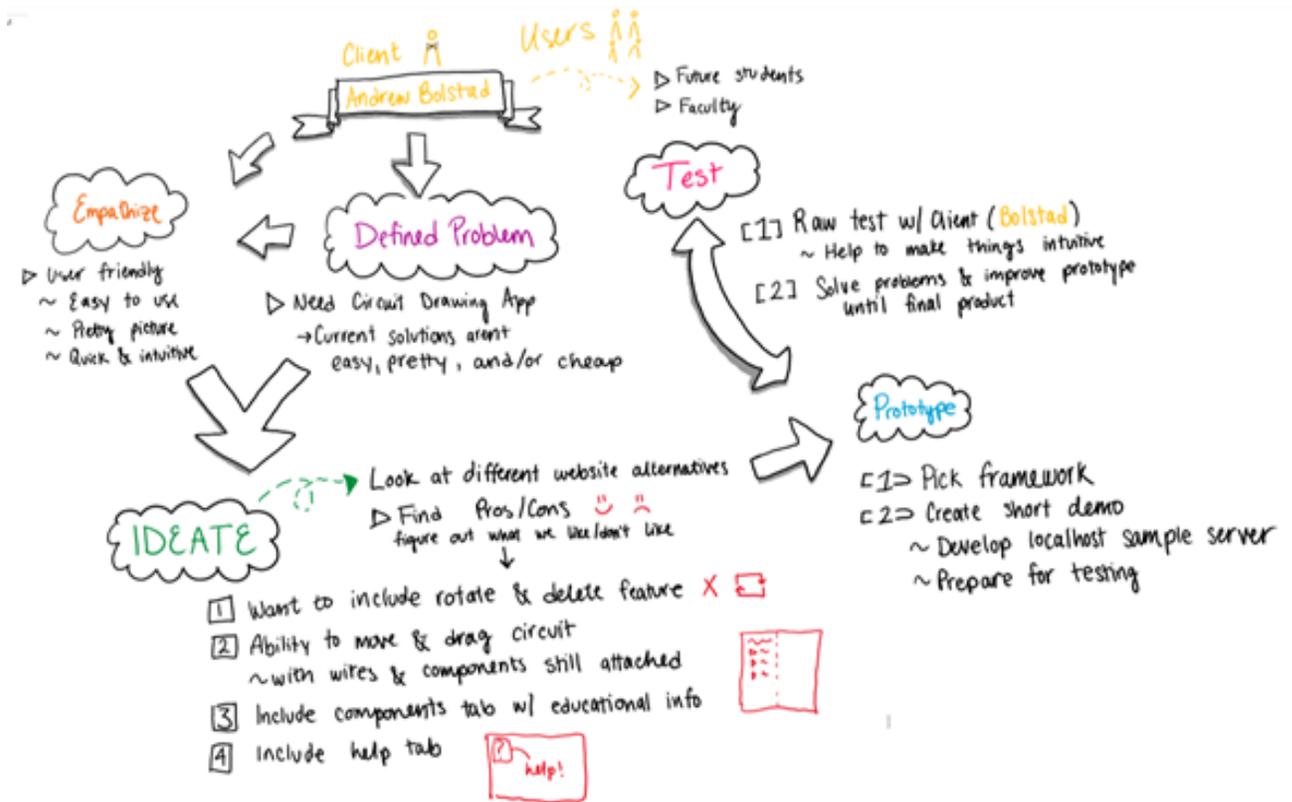


Figure 3: Process Diagram of Work to be Accomplished

## 2.9 Possible Risks and Risk Management

A big concern that will hinder our plan is the fact that we are creating a web application with coding languages Python, JavaScript, and HTML. As a team consisting of all Electrical Engineering majors, we do not have a lot of experience using these languages. As a result, our project team will need to put in additional time learning these languages on our own time. While we are confident that we can design the circuit drawing web application using these languages, we will need to plan for the time needed to both learn the languages and complete the project.

Another concern regarding costs is we will need the funds for server space to run the website. Currently, we have no funding from our client and so this will need to be discussed in more detail. We will also need to know how long the website will be running for as server space cost is mainly determined by amount of memory used and for how long it will be active. The web application is intended to be used for a minimum of a couple years past our graduation.

Furthermore, the long term feasibility of achieving a fully functional circuit simulator is certainly a stretch goal. Proper circuit simulation will be a lengthy process and will be

dependent on the progress of the circuit drawing functionality. The circuit drawing functionality is the most important part of the project and will be completed in its entirety before working on simulation. It is concerning that the website application will be incomplete if incapable of simulation; however, this might be something another senior design team can complete in the future.

## 2.10 Project Proposed Milestones and Evaluation Criteria

The main key milestones in our project are the following

- Displayable components on GUI
- Drag and Drop component feature
- Easy to use wiring attached to components
- Hotkeys for common circuit components
- Image Exporting to PNG file
- Testing website
- Basic Simulation
- Help page
- Educational components tab
- Tutorial
- Component naming
- Final Product

## 2.11 Project Tracking Procedures

Our plan is to use a variety of methods to share the progress of the project within the team and to our client. We will meet every week, sometimes more often, to share what everyone has achieved. This will help us create the weekly reports which are placed on the website. These will demonstrate what the group has accomplished for each week which includes overall accomplishments, current issues, and contributions of each individual. Part of the project tracking will be the creation of smaller groups within the team with the goal of providing assistance when necessary and keep members on task. Groups will be formed depending on roles as multiple modules will be constructed simultaneously. A project of this scope requires using different online tools to share information between group members. This information will cover both the technical aspects of the site and the non-technical portions of the project.

## **Document Sharing**

Google Drive will be our document sharing application. Docs lets multiple users participate in the creation of a document and make changes that appear immediately to all. Our reports and other documents will be shared through this to keep all members on track with each other's written work.

## **Project Managing**

GitLab will be our project managing tool used to share the code created, issues, and activity with regards to the technical side. GitLab is a repository manager that is widely used within software development. A group member will be able to post their code which other users can use within their local server. Issue tracking is also an important feature to track problems that the team is facing / working on. Team members will be made aware of the technical progress of the project through the use of this tool.

## **Communication**

GroupMe is a phone and web application that allows for text communication between a group. We will use it to facilitate meetings, discuss progress, and use it for other non-technical parts of the project that do not fit with the nature of Gitlab.

## **2.12 Expected Results and Validation**

The desired goal is an easily accessible website with circuit drawing and simulation features. A user should be able to navigate and place components to create a circuit for presentation purposes. Simulating the circuit should return values for current and voltage in order to analyze the effects that different components have on the circuit. As part of the analysis portion of the site, equations will be available to explain the results of the simulation.

As website elements are created, it is important that testing is done individually to verify they function as specified in section 2.2. Using multiple programmers can cause conflicts between created files which requires our team to spend time confirming that these files can be implemented together without causing failures on the website. The use of GitLab is important for this since we can push and pull code on the application to verify it functions with new modules. This tool also allows us to keep changes to the website separate, until we want to push it to the server. Creating a testing plan will provide a uniform guideline for programmers to follow with the idea of limiting failures and ensuring proper functionality.

Front-end testing will be focused on making sure that visual elements are displayed correctly within the drawing space in accordance with the functional requirements. Our client will be using our website to create circuits for use in university presentations; therefore, requiring a certain visual standard. The created visuals should help with both the professor's lectures and the student's learning. Testing visual elements requires us to go through the same process as a user to create a circuit and checking that each functional requirement can be correctly used.

Back-end testing involves verifying that the simulation results were calculated correctly. We can do this by comparing reported values to simulation results from more complicated software, such as SPICE. The back-end portion also includes **FR.5**, picture generation. Testing this will be similar to the front-end where we will test its ability to generate images prior to integrating it into the application.

## 2.13 Test Plan

The following features will need to have unit tests to ensure each function has the following capabilities. Some of these tests will be user and client tested, while others are function tests that simply need to meet certain expectations.

Function	Goal for testing	Testing Procedure
Graphics	<u>User tested</u> - Client agrees that the graphical design is appealing for what he wants	<ul style="list-style-type: none"> <li>• Ask Professor Bolstad for approval of graphical design</li> </ul>
Drawing	<p><u>User Tested</u> - Any new user can easily understand how to use the tools to create a circuit schematic. This will include not only our client's approval, but students who are currently or previously enrolled in a first level basic circuits class or higher.</p> <p><u>Drag and Drop</u> - User will be able to pull up any common circuit component from a components database and drag and drop it into their circuit</p> <p><u>Wires</u> - User would be able to draw wires from the end of one component to the next.</p> <p><u>Moving components</u> - When moving components around, the user's wires won't detach from the components' end node.</p>	<ul style="list-style-type: none"> <li>• Use mouse to click on any of the named components to the right side of the drawing space</li> <li>• Hold mouse over and release mouse click in the drawing space</li> <li>• Upon release, confirm chosen component image appears in drawspace</li> <li>• Components within drawing space are clicked on and dragged around to change the position within the drawing space</li> <li>• Click component or use "w" hotkey to be able to draw and create component wires within the drawspace</li> <li>• Allow random peer outside of this team to create their own circuit, ask about website's intuitiveness and feedback on improvements</li> </ul>

Saving Circuit	Final circuit drawing will export to a .png file format.	<ul style="list-style-type: none"> <li>• Click “Download” button</li> <li>• Confirm that downloaded .png file image matches what is in draw space</li> </ul>
Simulation	Once a final circuit is complete, the application has the ability to simulate with features that allow the user to be able to probe voltages at specific nodes.	<ul style="list-style-type: none"> <li>• Create simple 2 resistor in series circuit</li> <li>• Choose nodes to probe voltages</li> <li>• Click Simulate button and check that voltages are correct</li> <li>• Click “Basic Designs” tab to the left side of draw space.</li> <li>• Test all basic designs</li> <li>• Again, click Simulate button, and check that probed voltages are correct</li> </ul>

Table 1: Testing Application

## 3 Project Timeline, Estimated Resources, and Challenges

### 3.1 Project Timeline

Circuit Drawing Web App, Group 44			
1st Semester			DURATION (days)
START DATE	END DATE	DESCRIPTION TITLE	
9/23/18	9/29/18	Complete Project Plan version #1; Determine Coding Languages to be used	7

9/30/18	10/6/18	Meet with Andrew Bolstad; Learn basics of Python, Javascript, HTML, and CSS	7
10/7/18	10/27/18	Learn Python, Javascript, HTML, and CSS	21
10/28/18	11/10/18	Prototype of the website, using HTML and CSS; Backend development	14
10/28/18	11/10/18	Backend development Djanjo	14
11/11/18	12/1/18	Front end GUI development. Circuit component drawing, drag and drop	21
12/2/18	12/15/18	Complete prototype I, Testing with Professor Bolstad and random users; Gain feedback and make improvements to Prototype I	14
<b>2nd Semester</b>			<b>DURATION</b>
<b>START DATE</b>	<b>END DATE</b>	<b>DESCRIPTION TITLE</b>	<b>(days)</b>
1/14/18	1/19/18	Begin work on Prototype II, Regroup with team and Balstad	6
1/20/18	1/26/18	Prototype II begin working on easy to use circuit drawing functionality; wires move with components; hotkeys; picture exporting	7
1/27/18	2/16/18	Make progress and Complete wires moving with components functionality	21
1/28/18	2/16/18	Circuit drawing hotkey functionality	21
1/28/18	2/16/18	Work on back end user authentication	21
2/17/18	2/23/18	Bring all progress together into Prototype II	7

2/24/18	3/9/18	Test Prototype II, gain feedback from Professor Balstad and peers	14
3/10/18	3/23/18	Implement Feedback	14
3/24/18	4/6/18	Design and make pre drawn circuit library for easy to use access	14
3/24/18	4/6/18	Implement help tab and tutorial throughout the website	14
4/7/18	4/13/18	Bring everything together for testing phase round 3	7
4/14/18	5/11/18	Finalize website application based on feedback	21

Table 2: Project Timeline

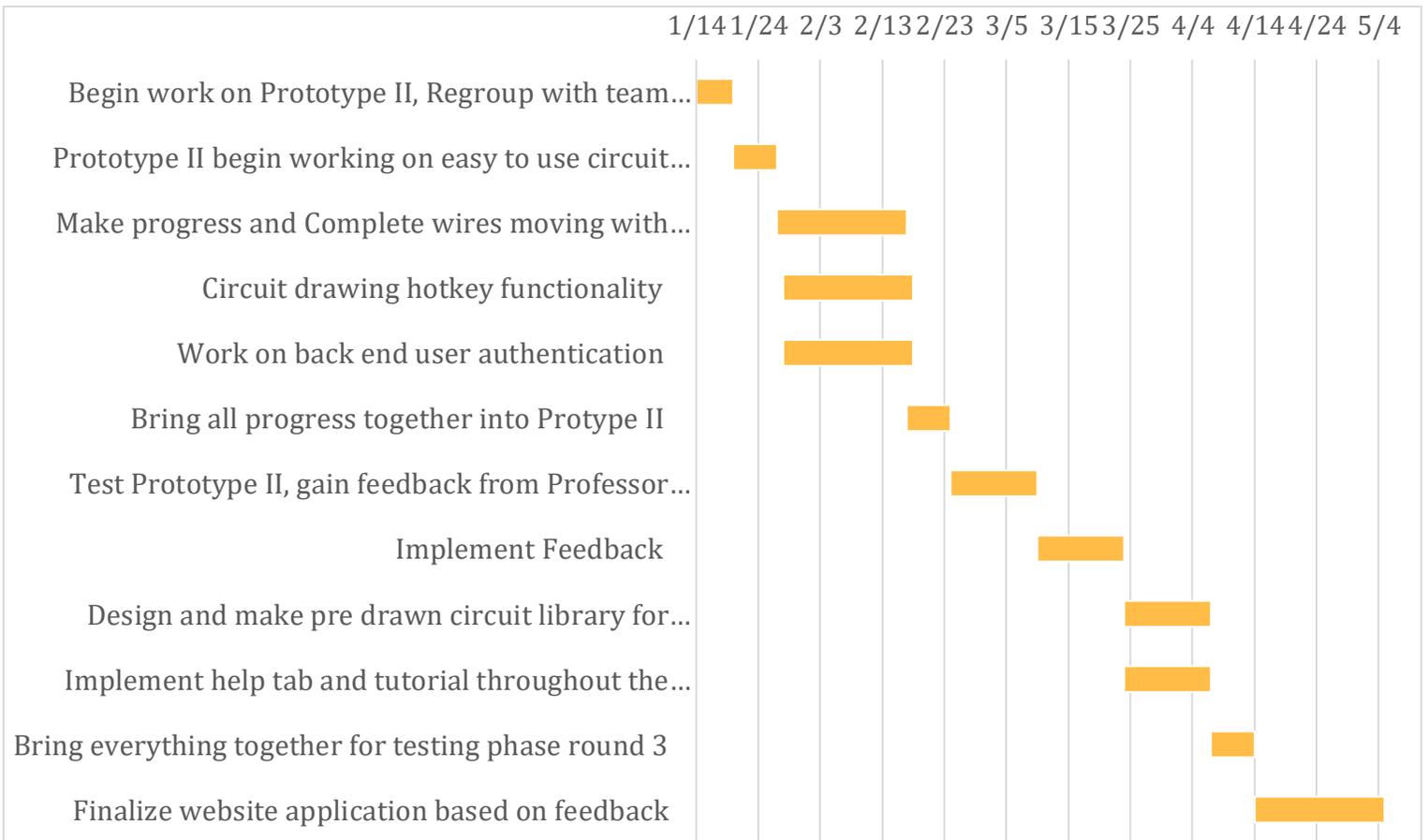
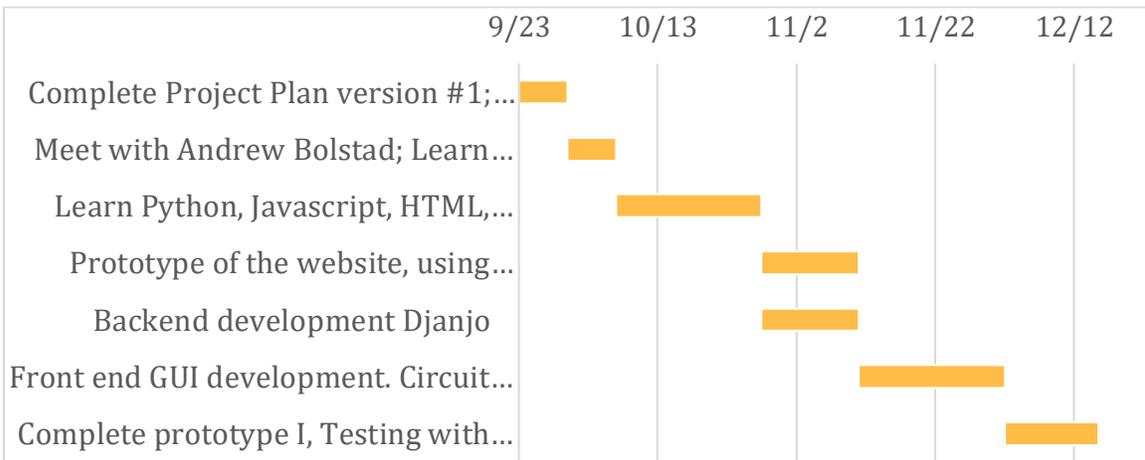


Table 3: Visual Project Timeline and Deadlines

### 3.2 Feasibility Assessment

The project will include a web application with the ability to drag and drop circuit components onto a grid, allowing for a nice picture to be saved for lecture slides. The project will also include the ability to simulate the circuits that have been created on screen. There exist challenges in the fact that our group is not very experienced in web app design, but we are all willing to learn and complete this project. We must take care in designing our application to meet the end user's requirements of being able to drag and drop components, keep the wires connected while moving components, and ensure that the end picture of the circuit looks good.

There also exists the challenge of implementing the Django framework with advanced JavaScript functions. The objective is to make this integration as simple as possible; but with our limited experience, this may prove difficult. Python is an "easy-to-learn" coding language, which is why we have chosen it. There still exists a challenge of actually learning and understanding the language so we can perform the actions we want for our simulation app.

We as a team believe that we will be able to complete the drawing section of our project by the end of this semester. We will be able to complete everything necessary to have the user be able to drag and drop components into the drawing space, and be able to download the picture of said drawing space. We have been working hard and the progress is showing as we already have the drag and drop feature included in our project; however, now we need to improve the website's visuals and integrate the Python server. The simulation section of our project will be set aside for the latter part of the second semester so that we can focus on the main need of our client, the drawing aspect. Any specific features that are not completely necessary for the drawing aspect to work will be taken care of in the second semester as well.

The server space that will be needed was also a question that came up for us. We have spoken to our advisor and we will be using server space that is hosted on an ECpE department server to avoid the costs of paying for server space.

### 3.3 Personnel Effort Requirements

<b>Task</b>	<b>Description</b>	<b>Developers</b>	<b>Total Hours Needed</b>
Create Django workspace	Installation of software program and setup of the server testing component	1	1
Develop an HTML interface	This is the creation of text in the site. It prepares the site for the creation of the GUI by dividing the page into sections	2	50

Enhance features with CSS	Used to implement design changes. It can also provide some functionality within the site	2	50
Integrate backend processing to django templates	Implementation of the background processing. Eventually this will be moved into simulations. This also controls the interaction with the database. Allows for switching of web pages with ease.	3	80
Create a javascript GUI	The main requirement of the project. Create a drag and drop interface with a changeable background and	3	200
Enable JPEG image file downloads	Using backend processing to download GUI images. JPEG images may also be saved	2	20
Implement Django Authentication Module	If files are to be saved and edited later, a user authentication system will track available designs and provide needed security	2	10
Integration Testing	The last phase of testing that will ensure all of the components are working together as expected	4	60
Migrate the database	Move the database out of the testing environment and into a service that will allow for public internet interaction	3	5
Document the front end	Provide web development descriptions and a list of functionality.	6	40
Document the backend	Comment code and provide a description of each page, view and url.	6	40

Table 4: The Estimated Time Required to Complete Various Components of the Project

### 3.4 Other Resource Requirements

The main resource that will be required is a domain name. In order for this application to be on a website, we will require server space and a domain name. Server space may be

maintained by the university or by outside resources that will require a payment plan for their services.

### 3.5 Financial Requirements

This project may require funding to keep the web application up and running. In order for the site to run, the project must have server space as well as a domain name. These are relatively cheap expenses, and our team hopes to avoid these expenses by obtaining a free domain name and free server space from Iowa State. In the event that we are not able to obtain these, we will have to pay for both server space and the domain name. A domain name can be purchased on Amazon.com's platform for \$12.00. The rate Amazon has set for storage is \$.021/GB of data transferred.

The simulation app will also require a host server. There are many options available for hosting. The most common way to host is to use Amazon Web Services (AWS); however, this may not be cost efficient for this project. This project requires that the server must be able to receive multiple types of requests as opposed to just the HTTP GET request. Because more requests are needed, the cost of a website could be considerably higher. Iowa State has server space available, but we currently do not have enough information on the capabilities of the Iowa State server space.

This project will make use of free software tools to decrease the amount of funding needed. At the option of our client, we will be able to make the code for this project private. However, this will require additional funding. The funding would be used to purchase a private GitHub repository, which is also a relatively low cost.

## 4 Closure Materials

---

### 4.1 Conclusion

There exists many different circuit applications, but they are often weighed down with heavy processor requirements and expensive licensing costs. In addition, some of these applications are not free and intuitive to use. Therefore, we are creating an educational, easy to use, and free web-based application that will eliminate these issues. Our website will feature an appealing circuit drawing tool that can be used to teach students who are interested in circuit design and theory. We plan to add important formulas for circuit components and a simulation feature for the circuit so students will be able to understand the function of any circuit.

In order to make this project possible, we will divide the work accordingly so each member will learn how to utilize both Python and JavaScript. This will give each member thorough experience with software design, and it will allow everyone to follow the progress of the project. Furthermore, we will have weekly meetings in order to discuss the team's accomplishments, schedule, and issues concerning the various components to the project. This constant communication will ensure an efficient design process and it will allow us to

stay on top of any issues in order to resolve them as soon as possible. Although we are all electrical engineering students, our planning and organization skills are exactly what this project needs to solve this software design problem.

## 4.2 References

1. *Circuit Diagram Web Editor*, [www.circuit-diagram.org/editor/](http://www.circuit-diagram.org/editor/).
2. *Circuit Simulator Applet*, [www.falstad.com/circuit/](http://www.falstad.com/circuit/).
3. *CircuitLab - Editing "Unnamed Circuit"*, [www.circuitlab.com/editor/#?id=7pq5wm](http://www.circuitlab.com/editor/#?id=7pq5wm).
4. "PSpice User Forum." *Electronic Circuit Optimization & Simulation | Cadence PSpice | PSpice*, [www.pspice.com/](http://www.pspice.com/).
5. "Python 3.7.1 Documentation." *3.7.1 Documentation*, [docs.python.org/3/](https://docs.python.org/3/).
6. "Until November 1, Get PyCharm for 30% off. All Money Goes to the DSF!" *The Web Framework for Perfectionists with Deadlines | Django*, [www.djangoproject.com/](http://www.djangoproject.com/).