

GONZAGA UNIVERSITY
School of Engineering and Applied Science
Center for Engineering Design and Entrepreneurship

Graphical Document Generator from Datasets

Project Overview Plan Section 01

Release:
Draft v0.1

PROJECT PLAN DRAFT STAGE DOCUMENT
September 10, 2024

The Knights of Saint Gall



**Joshua Venable, Joseph Holdnak,
Sebastian Matthews, Raja Sori**

1 Project Overview

1.1 Project Summary

Currently there are no ways that a client may convert websites into flow diagrams and instruction sets that may be used to automatically create automation tests. The purpose of this project is to help customers visualize their website such that they may see how one of their users might interact with the site. The project will also allow clients to ensure that they have tested all necessary components during the development lifecycle, as in conjunction with <CS25-16 link>, it will enable automation testing of the website.

This project will create a standardized instruction set that can be sent to <CS25-16 link> so that their team may create automation testing tools for the customer's website and data. A pipeline will be created from this project to the aforementioned, allowing for a rapid development cycle in which this project intakes data from the customer, transforms it into an instruction set, and sends it to the partner project who will create automation tests from the instructions. Another goal of this project is to create a simple front-end interface that will also generate a data-rich flow diagram chart of the data that can be used for client discussions, design mockups, and design or code reviews.

1.2 Project Objectives

TBD (business objectives)

1.3 Project Stakeholders

- This Team
 - A stakeholder of the project would be the team designing it, as the developers need to implement certain items in the backend/frontend that would prove useful for future iterations as well as maintaining the process after handoff of the deliverables.
- Team CS25-16
 - The partner project working concurrently with this team to create the full working end-to-end service. They are expecting certain deliverables, primarily the instruction set, as they ingest the instructions sent to them in order to create automation tests.
- D2B Including Pete Messina and Dwight Brayton
 - The sponsor and advisor of our group are large stakeholders who are defining the business needs of the project, as well as what are the critical functionalities that must be integrated.
- DAB – Ben McDonald
 - The DAB of this team is a stakeholder, who will be helping in reviewing the project as it progresses and helps the team as it forms the process for working on the project.

TBD (we've received no response from our sponsor)

1.4 Project Deliverables

This project will have a few key deliverables:

- Data-rich graphical flow diagram generator
 - This flow diagram generator will follow the [ISO 5807](#) standard
 - Each element within the diagram will include data/metadata associated with it
- Simple User Interface that will display a generated flow diagram
 - <TBD software/tools available to visualize>
- Instruction set generator that can be sent to <CS25-16 link>
 - From incoming data/website, create an instruction set that fully encapsulates actions that can be taken, including clickable links, buttons, etc.
 - Includes associated metadata and <TBD>

1.5 Project Scope

Outside the scope of this project is the creation of a simple UI that will be used for testing the project. This project is also not scoped to the automatic creation of tests from an instruction set that will give English language steps and the creation of a report that shows code coverage from said tests. The creation of the instruction set, the flow diagram, and the interface to visualize the flow diagram are all a part of the scope of this project, and the pipeline handing off the instruction set to <CS25-16> will be a shared responsibility between teams.

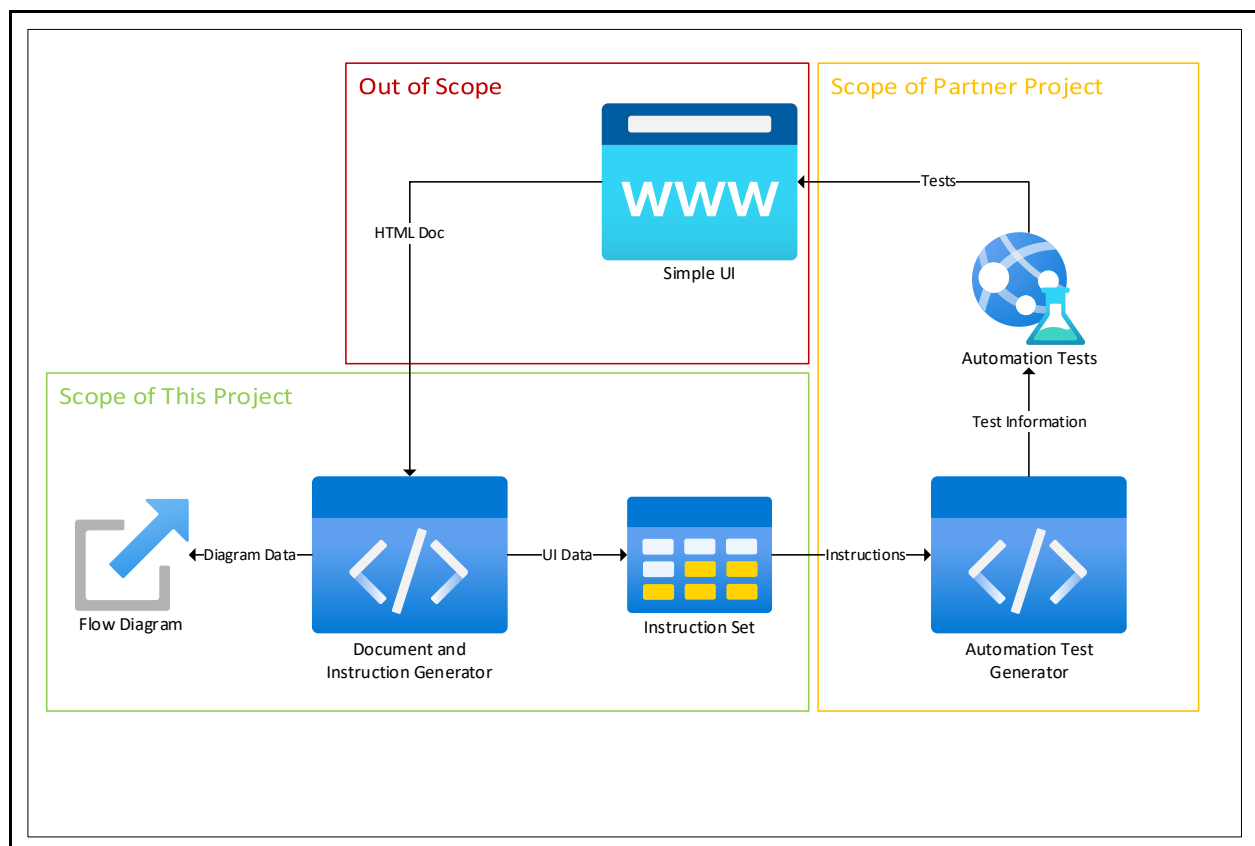


Figure 1: Context Diagram showing the scope of this project (bottom left) and its relation to the project <CS25-16> (right) and what is out of scope (top).

1.6 Related Work

Provide a description of existing systems and/or approaches that try to solve a similar problem as yours. Identify and describe the system most-closely related to the one you are planning on building, discussing both the similarities and the differences between this system and yours. Additionally, summarize the major similarities and differences of those less related, but still similar to your project. The goal of this section is to show that you have examined and understand the product landscape and have a clear idea of the needs of your project and how they are similar and different to the current systems/approaches available. Provide a link or reference (as a footnote) to each system you describe.

Draw.io¹ is a web app that allows anyone to create diagrams. The way it works is that users are able to manipulate shapes, paths, and various vectors to put together a diagram of their choice. Although both Draw.io and our software form diagrams, Draw.io was as a creative web app made for creative design. Our software is going to automatically generate diagrams from data.

Code to Flowchart² is a web app that generates flowchart diagrams. The problem it attempts to solve is to spare programmers from building diagrams by hand. This can be a time-consuming activity. The way it works is that programmers will input their source code, then an artificial intelligence model will analyze the source code. Once the analysis is complete, the web app will generate a flowchart based on each instruction of the source code. Our software is similar in that it takes instructions then generates flowchart diagrams. Where our software differs, it's going to analyze an HTML file, take user interface data, then generate a flowchart and instruction set.

JSON Studio³ is a web app that converts plain text to JSON format. It attempts to solve the issue of taking a string that already exists and manually formatting it. The way it works is that a programmer will input their plaintext, then the web app makes a request to the ChatGPT API. Once the web app receives a response from the API, the web app will take the response body (which contains the formatted JSON object) and provide it to the programmer. Our software is similar in that it returns formatted instructions. Where it differs, our software is going to create instruction sets, not convert from a format. It's also going to create diagrams.

¹ <https://draw.io>

² <https://codetoflow.com/>

³ <https://jsonstudio.io/>