Project Description

(*Use additional pages as needed for this report.*)

PART I.

1. General Information

Project Title: Full Stack Dev AI

Submitted by (insert names of team members): Ian Straatman, Lyndon Renaud, Hong Qian Zhao

Submitted to (supervisor name): Kalyani Selvarajah

Date submitted: Friday November 5th, 2021

2. Project Overview (*Describe the project & its purpose*)

The purpose of this project is to make it possible for people who have no web development experience to create basic websites/webforms from a hand-drawn photo. This can also be used to make it quicker and easier for people with experience in web development through automation. Designing a web form with the corresponding backend code is a repetitive task that can take a lot of time. Using computer vision and machine learning, we aim to build a system that can understand hand-drawn web forms and generate the corresponding backend code.

We are training a model to recognize hand-drawn form elements like a textbox or button. We will also be using OCR functionality to extract hand-written text labels associated with form elements. By identifying form elements and their associated labels, we are able to generate a REST API containing the code necessary to process the form's data. We use a JSON object to store the identified form elements, where each key is a form label mapping to a form input type. This JSON is then sent to a REST API for backend and frontend code generation.

A large portion of the project will involve training a machine learning model with data our team will create. The data will first consist of individual form elements to train the model in binary classification. From there, we will create data with multiple form elements in one photo to train the model in multi classification. Our hope is to create an efficient and accurate model that will allow developers and designers to be more productive.

3. Deliverables (*Describe all products to be produced*)

- AI capable of recognizing elements of a website from hand drawn photos.
- REST API for producing the HTML code corresponding to requests sent.
- OCR to read form element text.
- Image segmentation to identify form elements.
- Binary classification to classify each form element.
- Multi classification to classify all form elements.

4. Requirements (*Describe the required resources e.g. hw/sw, technical knowledge and skills etc*)

Software Requirements:

- Python, Flask
- OpenCV for Python
- Tensorflow
- MS Sketch2Code

Knowledge & Skills:

- Python
- Javascript
- AI and machine learning
- HTML
- REST APIs

5. Constraints

- Time period given.
- Beginner knowledge of AI.
- Experience in machine learning.
- Interference from other projects.
- Complexity of the entire project.
- Cannot train the whole component within the given time period.

PART II.

1. Risk analysis *(There is no fixed format, as it will vary depending on the nature of the project. For some general guidelines please see the file under the **ResourcesàRisks** link on BB)*

Complexity

- Network adaptability and possible network failures, as well as possible reading problems for handwriting.
- Due to technical issues, as well as adjustments to the network and possible leased servers, it may happen that the project needs to change the plan.
- Medium risk. Because we may have technical limitations.
- Use simpler technology, or replace the original solution to read handwritten content.
- Not the best solution, but this solution can help us.

Performance of AI

- The learning of AI may be biased. Just like the text written by different users will affect the learning of ai. How to make AI read customer input to know that it is a picture instead of text or something.
- This may seriously affect the operation of AI and the entire program. The cost is very high after a problem occurs.
- High risk. Everyone's handwriting may be different.
- Regular maintenance and inspection by personnel. Add more complete user feedback and reports. And you can ask the user to indicate the area of the picture before handwriting to tell the AI where the picture appears.
- Excellent solution. This solution can perfectly solve the problems of AI. Customer feedback can also help our program improve better.

Poor training dataset

- Different training data appears, or the collected data does not meet the requirements of the test.
- No pre existing data sets, which means we need to create all data ourselves
- Need data sets for each different element individually as well as combined, this leads to a larger number of data.
- Medium cost. AI can't meet customer requirements well.
- Medium risk. This usually happens when training AI.
- We can conduct more manual screening of training materials by personnel.
- Not the best solution. Although it is not a very good solution, it solves the problem of training materials very effectively.

Interference from other projects

- During the work there may be other teams producing the same work. Or another project may appear to affect the speed.
- Low cost. We do not need to complete this project very quickly. And there are a lot of similar software in the market, which will not affect sales too much.
- Low risk. It will not have a big impact on our work.
- We only need to allocate a portion of the time to make this project.
- Nice solution. This is not a very important risk, but we need to know and consider.

Platforms becoming obsolete

- The platform may appear outdated, or the user's new system may not work well.
- High cost. After a problem occurs, it may affect the completion of the entire project and the utilization of Customer.
- Low risk. Most companies do not use the latest platforms, and generally use older or long-term platforms.
- We can look for a platform that can provide long-term technical support during the production time, or a platform that most companies are using.
- Not a big problem. And a good solution. This is also the solution of most companies now.

Team member leaving

- People may leave their jobs for various reasons, wage disputes or physical conditions.
- Medium cost. The replacement of personnel may extend the time of the project, and any new team members need to spend time to familiarize themselves with the progress of the project.
- Medium risk. It doesn't happen very often and needs attention.
- We can make more complete notes and introductions to the project. Help new team members to get familiar with our project better.
- Medium solution. Not a very good solution, but the departure of team members does have an impact on the speed of project completion.

Poor task allocation

- The tasks that may arise are not part of the team members are good at. And the length of time the team members can work on this project at different times.
- Medium cost. It may appear that the specified project cannot be completed within the specified time.
- Low risk. Although it may happen, it is easy to avoid.

- We can specify different work items from the beginning, and let the team members choose what they are good at. If there is a problem with the length of work, team members can work together to complete faster and exchange opinions and plans.
- Excellent solution. Almost all problems in this area can be avoided.

Image Segmentation - train different models

- Maybe AI cannot read the picture or read the wrong picture.
- High cost. It may affect the user experience and the operation of the program.
- High risk. Because AI is difficult to train to recognize different pictures, it is a technical problem.
- We can ask users to indicate the area of the picture and the segmentation where the picture appears at the very beginning.
- Excellent solution. This aspect can avoid the huge problems of AI, and directly avoid the possible problems.

PART III.

1. Work Breakdown Structure (WBS) *(There is no fixed format, as it will vary depending on the nature of the project. For some general guidelines please see the file under the* ***Resourcesà Work Breakdown Structure (WBS)*** *link on BB)*

Machine learning model

1. Research HTML form elements.
2. Create a training dataset.
3. Image segmentation of form element text for OCR.
4. Binary classification to identify a single type of form component in an image (text box, button, etc.).
5. Multi classification to identify all types of form components in an image.

REST API

1. Research how REST APIs work.
2. Research how REST APIs receive and process data.

Program automation

1. Use the machine learning model to generate a JSON object for the given hand drawn picture.
2. Generate REST API for the given JSON object.