WEB VISION

An AI assisted Web Design Mockup to HTML generator

Submitted By

Ajay Kumar - 17103253
Tanish Johar - 17103252
Prayas Arora - 17103249

Under the supervision of:

Dr. Indu Chawla

NOVEMBER 2019

Submitted in partial fulfillment of the Degree of
Bachelor of Technology
in
Computer Science Engineering

Department of CSE/IT

Jaypee Institute of Information Technology University, Noida

TABLE OF CONTENTS

DECLARATION

I/We hereby declare that this submission is my/our own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which has been accepted for the award of any other degree or diploma from a university or other institute of higher learning, except where due acknowledgment has been made in the text.

Ajay Kumar	Tanish Johar	Prayas Arora
17103253	17103252	17103249

CERTIFICATE

This is to certify that the work titled **Web Vision** submitted by **Ajay Kumar** , **Tanish Johar and Prayas Arora** in partial fulfillment for the award of degree of Bachelor of Technology of Jaypee Institute of Information Technology, Noida has been carried out under my supervision. This work has not been submitted partially or wholly to any other University or Institute for the award of this or any other degree or diploma.

Signature of Supervisor ……………………..
Name of Supervisor Dr. Indu Chawla
Designation Senior Professor
Date ……………………..

ACKNOWLEDGEMENT

We would like to express our special thanks of gratitude to our mentor Dr.Indu Chawla, who gave us a golden opportunity to do this wonderful project on the topic Web Vision, which also helped us in doing a lot of research and we came to know about so many new things. We are extremely thankful for this opportunity.

Signature of the Student ……………………..
Name of Student ……………………..
Enrollment Number ……………………..
Date ……………………..

INTRODUCTION

Websites are what has made the Internet so vast and openly accessible for everybody. In this technologically advanced era, websites have emerged as the new area to do business. A good website with a good user interface is what attracts customers and make them try their products. And, for the same reason, web design UI becomes one of the most important parts of website building workflow.

Today, the work of developing web UIs or any GUI in general is divided into two roles, first a UI/UX designer and the other Front-end Developer. A UI/UX designer is given the responsibility to develop attractive UI designs for the website either hand-drawn or using any UI developing software such as Adobe Xd. Further, these designs are passed on to a Front-end developer who produces code for the drawn design as to be rendered to the user.

This workflow if studied properly, clearly suggests that it consumes an extra time, that is, when a designer already has a drawn design for the website, why is there a need for a developer to write code for the whole design, taking up a whole lot of time and effort. A better workflow or rather, a more technically updated approach would be to generate a front-end code for a given web design. Implementing GUI code is, however, time-consuming and prevent developers from dedicating the majority of their time implementing the actual functionality and logic of the software they are building. Moreover, the computer languages used to implement such GUIs are specific to each target runtime system; thus resulting in tedious and repetitive work when the software being built is expected to run on multiple platforms using native technologies.

Why not make machines handle this problem? The hottest topic of this era, Artificial Intelligence is the key to this problem. Today, when we have the best computing power and data to train a Deep Learning model which could generate UI codes for an input image.

PROBLEM STATEMENT

This project aims to develop websites or HTML code to be more specific, for a hand-drawn UI image or a UI developing tool generated image. As part of the project, we intend to develop a production ready Web App which can take these images and produce required HTML code, saving a developer’s time. Also, a RestAPI has also been developed as part of the same project, to enable enthusiastic developers to use our model cross-platform and integrate in their systems.

The project also implements existing papers on the same topic and we have put on our own method and the results have also been compared.

REAL WORLD APPLICATIONS

Everyone now a days is finding solutions to achieve optimal results by investing less time and cost. First of all, hiring a UI/UX developer and a front-end developer both for the job is an expensive step and also very time consuming for the organization. Further, the activity of web designing itself takes double time by making a design by UI designer at first and then developing code for the same by the front-end developer whereas, developers could put in more of their time to optimize the website and handling the backend.

In such a scenario, this project comes out very handy for not only developers but also to an individual or the organization. Also, this could help many backend developers who have less knowledge about front-end designing and wish to publish their websites. Such developers could just draw in their design they wish to incorporate in their website and just use the web app and their website would built automatically.

Also, the RestAPI as part of the project could also be used by further developers to incorporate in their systems and extend this project to another level.

WORK APPROACH

1. Finding the Deep Learning solution for the problem

1st Approach - Image Classification and code generation

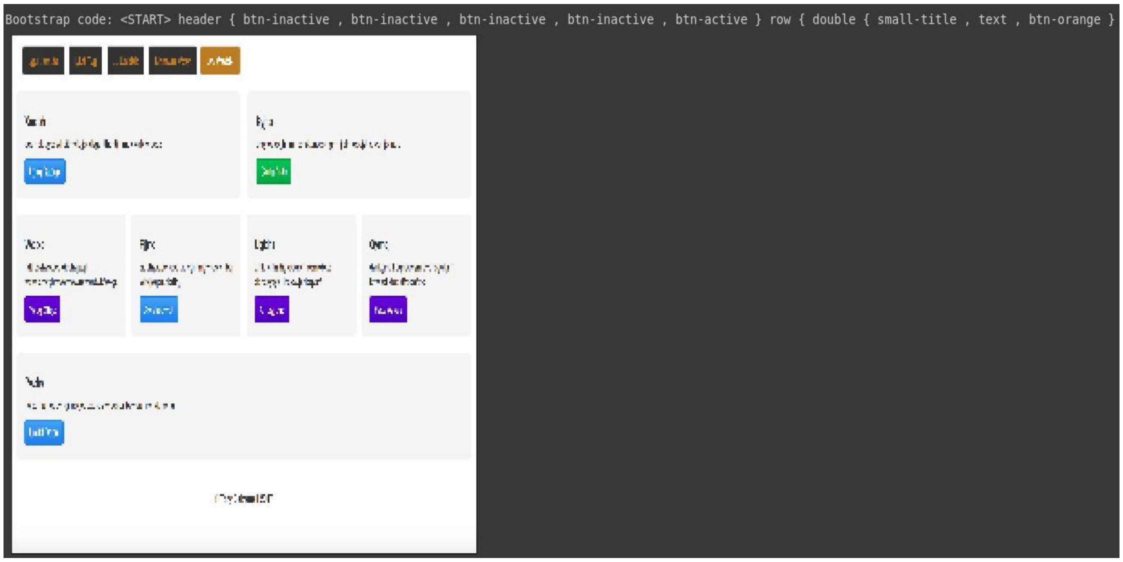
The problem clearly is a branch of Image Classification problem because it clearly intends to take in web design images and generate HTML code for the same.

Initially, we mapped the problem to an Image classification cum text generation problem, since, the model should take in an image and output an HTML code which can be treated as text generation.

DATASET

We used the openly available rather, the only dataset available on GUI and their respective code outputs, which is provided as part of the Pix2Code research paper.

The dataset has 100k web UI images with their corresponding DSL codes. Since, the dataset includes UIs only generated through software and no hand drawn UIs, the model has problem completely understanding an unknown hand drawn image.



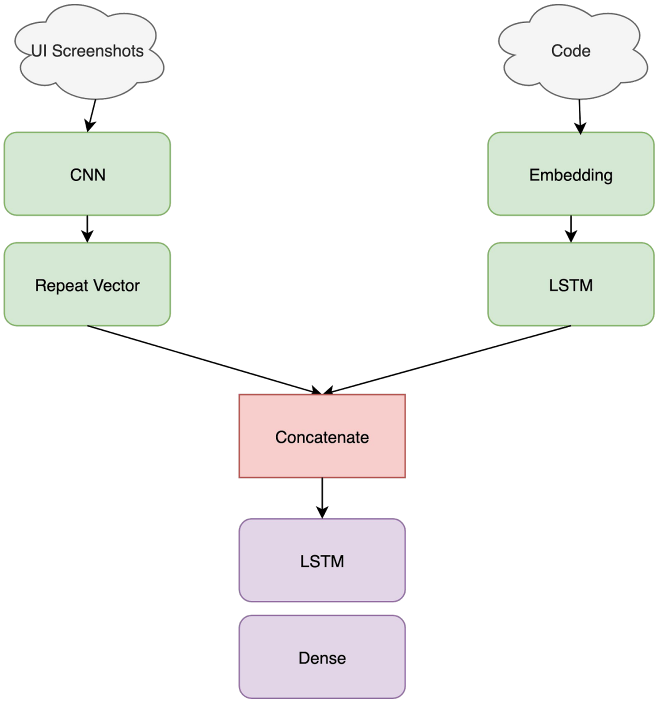
```
Bootstrap code: <START> header { btn-inactive , btn-inactive , btn-inactive , btn-inactive , btn-active } row { double { small-title , text , btn-orange }
```

This is an image from the same dataset. It clearly is a set of preprocessed Web UI images and are labelled by their corresponding DSL language.

MODEL ARCHITECTURE

Studying the problem, it was evident that the problem can be mapped to a modified image classification with sequence prediction. CNNs are state-of-the art model for image classification and RNNs are state-of-the art models for sequence prediction.

Hence, the model is a combination of CNNs and RNNs, more specifically LSTMs. CNN and LSTM take the image and language DSL as inputs and their outputs are concatenated to be sent to a third LSTM model which generates the language tokens.



The model can be broadly split into two parts - Encoder and Decoder

ENCODER

We took the word embeddings and ran them through an LSTM and return a sequence of markup features. Finally after applying a Time-distributed dense layer, we receive a 3 markup feature.

In parallel, we prepare the images. We take all the mini image features and flatten them to transform them into one long list. The information is not changed, just reorganized.

Next, to extract higher features, we apply dense layer and to concat the image feature with the markup feature, we apply a repeat vector layer which copies the image features 3 times since, our markup feature is of 3.
These image features are then concatenated to the markup features.

DECODER

Here we use the combined image-markup features to predict the next tag.
And, after passing through the dense layer with softmax, we get the final probabilities of the vocabulary.

2nd Approach - Object Detection

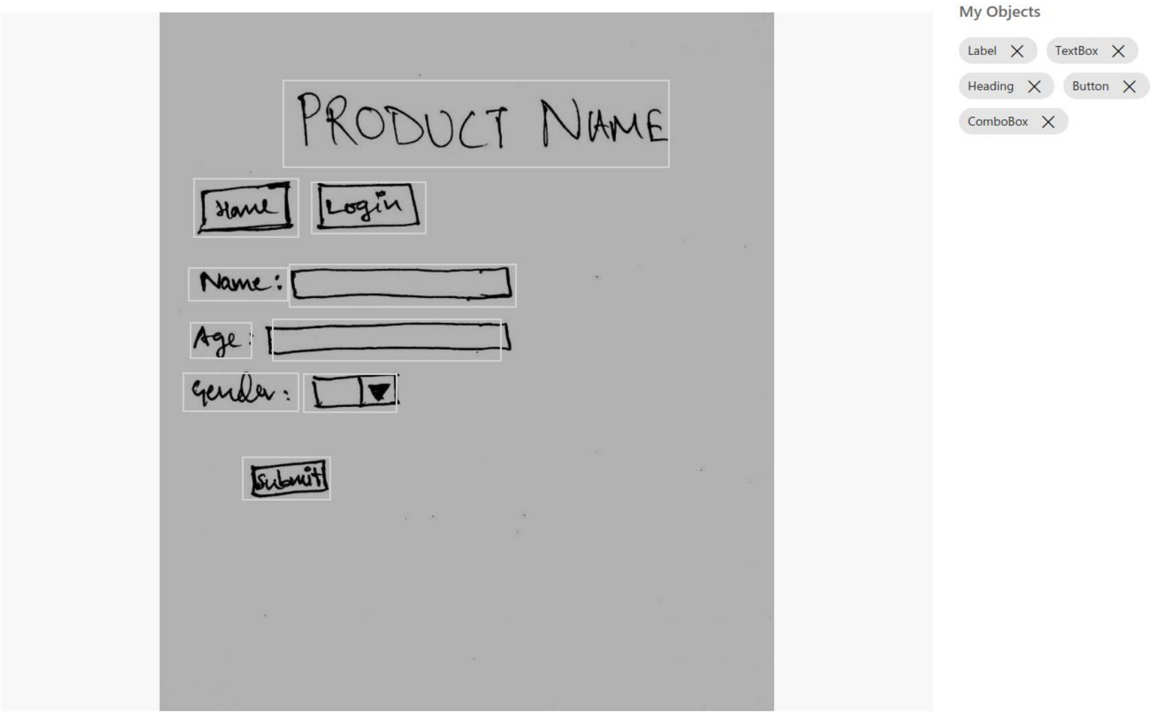
The above model after being trained for 50 epochs didn’t provide satisfying results. And, then we thought of taking a different approach to the whole problem.
Why not perceive the whole problem from the perspective of a human?
As a human being, a front-developer when seeing a design mockup, looks to the web components being used and recall their positions and implement them in the code. Keeping that in my mind, we thought why not make our model learn to identify what web components are and extract their positions from the image being inputted.
This, clearly can be thought of as an object detection problem because each web component can be treated as an object and the machine can be made to learn these objects and recognize in a given image , and, further, extract their positions from the same image.

We could have easily setup our own custom object detection model using tensorflow and keras. But, when today we have so many pre trained models, why to invest time in building our own. So, we decided to use Microsoft Azure’s Custom Vision API.

DATASET

We built our own small dataset of almost 150 images to train this model. The 150 hand-drawn web design images were first uploaded to the Custom Vision web handle and each image was tagged using their tagging tool and the following classes were used:-

a) Button
b) CheckBox
c) ComboBox
d) Heading
e) Image
f) Label
g) Link
h) Paragraph
i) RadioButton
j) TextBox



Above is an image of the dataset created in Custom Vision API.

TRAINING

The tagged dataset is trained on the pretrained Custom Vision model of Microsoft Azure and a prediction endpoint is created which can be used for predictions.

2. Flask API

The trained models had to be publicly usable by users, so a user friendly app was necessary. Hence, we used Flask to build a dynamic Web App which takes in images from users and serves them with the output HTML file.

Apart from the Web App, we also built a RestAPI service for developers willing to extend this project with a better UI.

3. Generated web page enhancement toolkit

Since,this project is in its initial stage, the produced HTML code might not be exactly what they need. There may be some minor variations.
So, taking into consideration that the users of this app will be completely unfamiliar with front end programming, we made an enhancement toolkit with the flask app that can be used by anyone easily.

Now, any person without any knowledge of front-end programming can build a website easily.

RESULTS

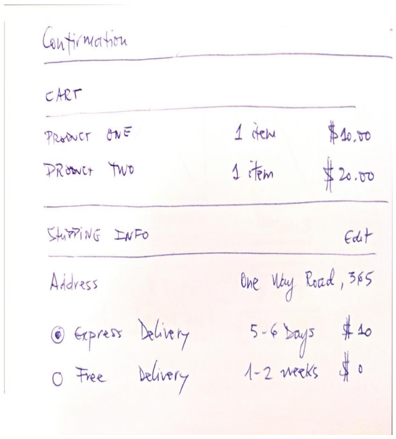
The project has been divided into two versions for evaluating :-

1) **Version - 1** : CNN and LSTM model
2) **Version - 2** : Web Component Detection Model

Although version-1 due to computing power constraints, didn't perform well, still it gives good accuracy on the validation set images. Version - 1 has good accuracy on images related to it’s dataset that are software generated web design mockup images. But, fails generating code for a hand-drawn images.

Version-2 works very well in detecting web components within a design mockup image due to having learnt the lower level features already. It marks each component inside a bounding box and tag them with their respective classes. This bounding box data is taken into the Web App and real HTML components corresponding to the class labels is generated.

Below are some comparisons with order as :- Input Image , Version-1 generated output , Version-2 generated output.



Confirmation

CART

PRODUCT ONE 1 item $10.00

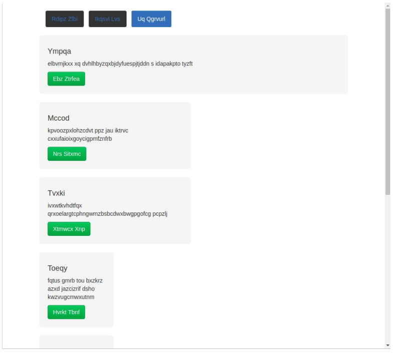
PRODUCT TWO 1 item $20.00

SHIPPING INFO Edit

Address One Way Road, 365

◉ Express Delivery 5-6 Days $10

○ Free Delivery 1-2 weeks $0



Ympqa

elbvmjkxx xq dvhlhbyzqxbjdyfuespjtjddn s idapakpto tyzft

Mccod

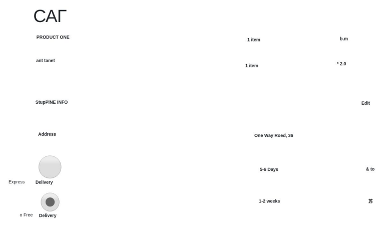
kpvoozpxlohzcdvt ppz jau iktrvc cxxufaioixgoycigpmfznfrb

Tvxki

ivxwtkvhdtfqx qrxoelargtcphngwmzbsbcdwxbwgpgofcg pcpzlj

Toeqy

fqtus gmrb tou bxzkrz azxd jazcizrif dsho kwzvugcmwxutnm



САГ

PRODUCT ONE 1 item b.m

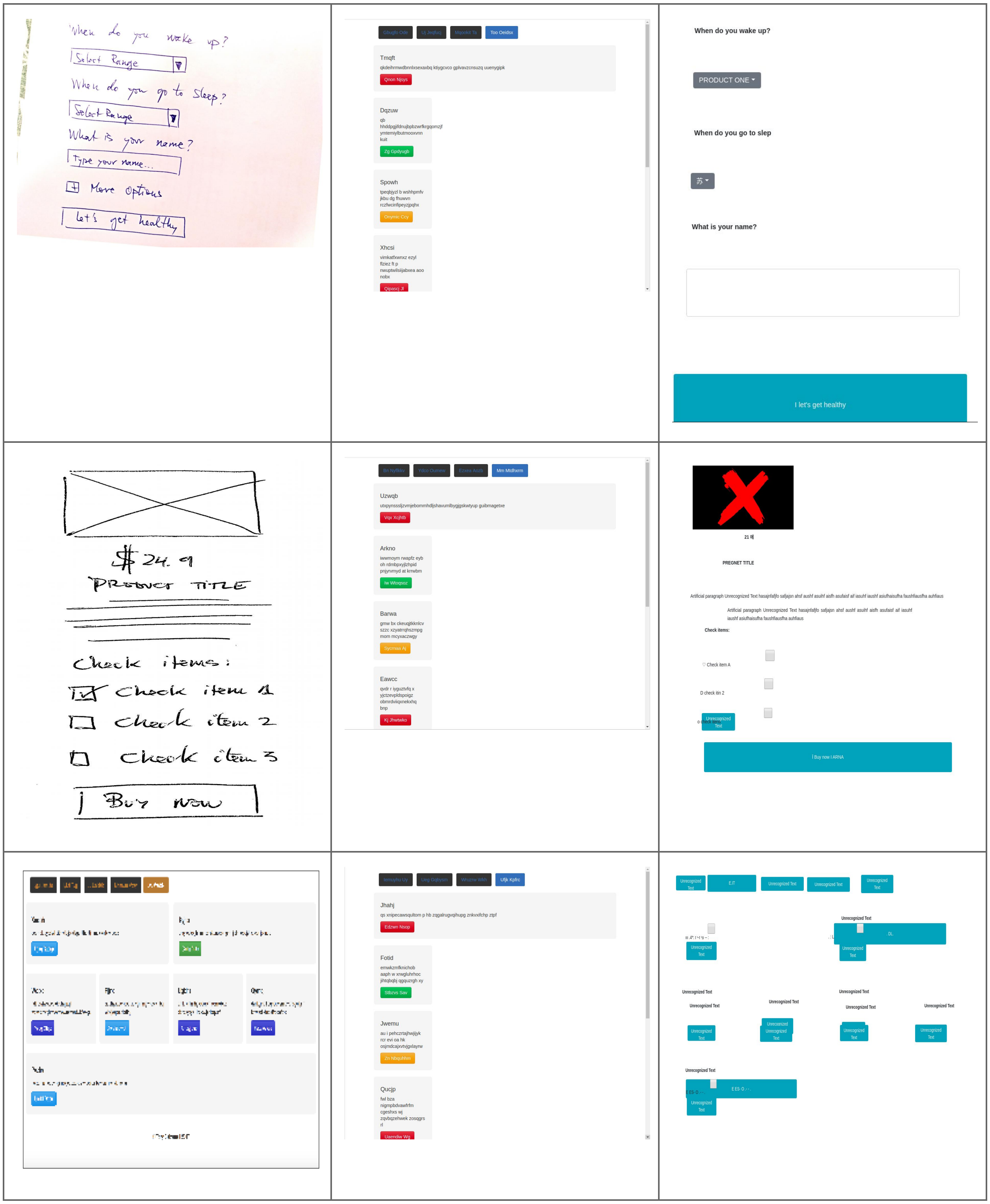
ant tanet 1 item * 2.0

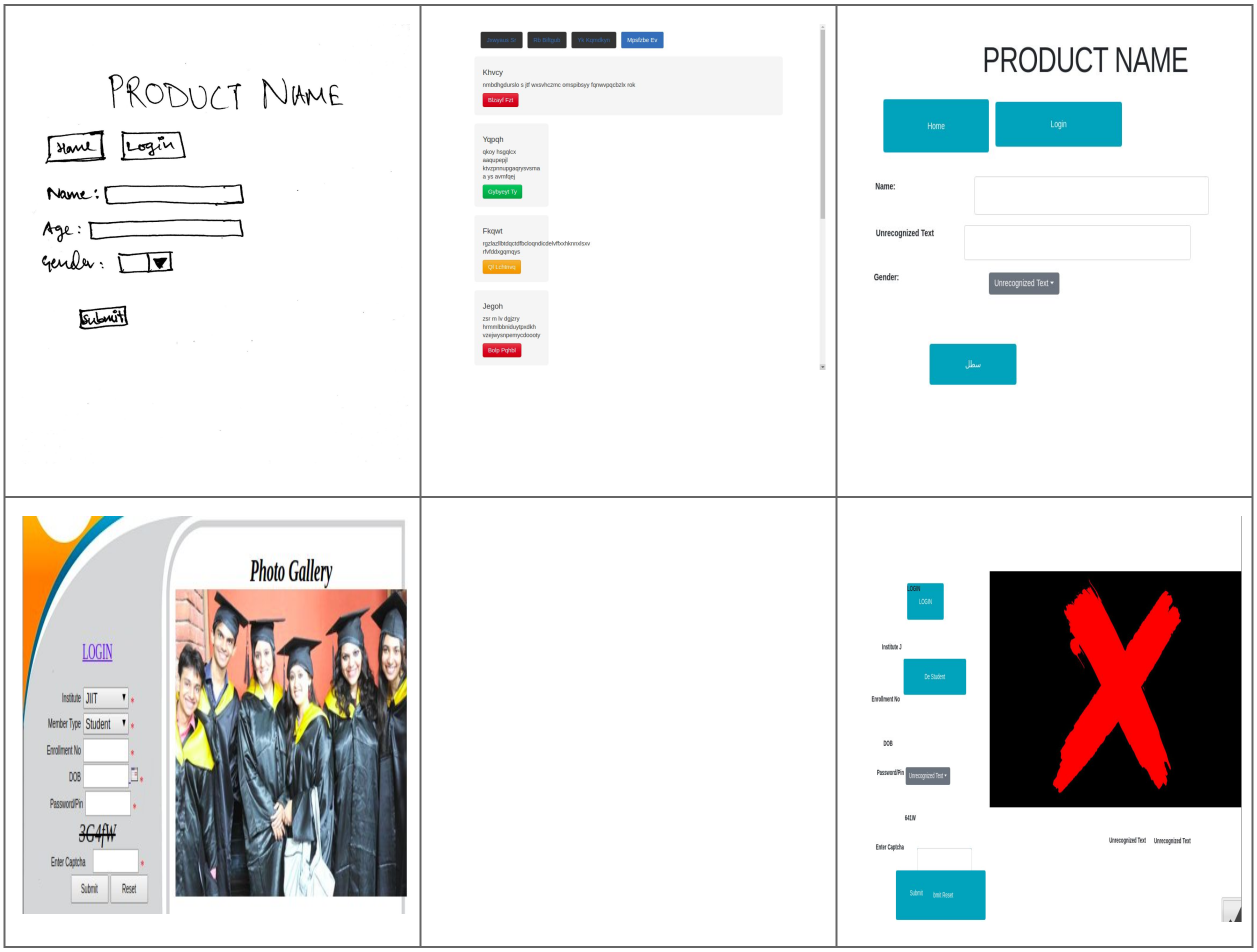
StupPiNE INFO Edit

Address One Way Roed, 36

Express Delivery 5-6 Days & to

o Free Delivery 1-2 weeks 苏


When do you wake up?
Select Range
When do you go to sleep?
Select Range
What is your name?
Type your name...
More Options
let's get healthy
When do you wake up?
PRODUCT ONE
When do you go to slep
What is your name?
I let's get healthy
$ 24.9
PRODUCT TITLE
Check items:
Check item 1
Check item 2
Check item 3
Buy Now
21 에
PREGNET TITLE
Artificial paragraph Unrecognized Text hasajnfafjfo safjajsn ahsf aushf asuihf aisfh asufaisf aif iasuhf iaushf asiufhaisufha faushfiausfha auhfiaus
Artificial paragraph Unrecognized Text hasajnfafjfo safjajsn ahsf aushf asuihf aisfh asufaisf aif iasuhf iaushf asiufhaisufha faushfiausfha auhfiaus
Check items:
Check item A
D check itin 2
o check items
Unrecognized Text
Î Buy now I ARNA
Unrecognized Text
E.IT
Unrecognized Text
Unrecognized Text
Unrecognized Text
Unrecognized Text
Unrecognized Text
Unrecognized Text
Unrecognized Text
Unrecognized Text
Unrecognized Text
Unrecognized Text
Unrecognized Text
Unrecognized Text
Unrecognized Text
Unrecognized Text
Unrecognized Text



After evaluating the models, it is evident they aren't perfect but to start, this is giving good results. With better training power and data, this project can have a great future.

CHALLENGES AND CONCLUSION

Clearly, the model isn't that accurate as of now. It has an accuracy of almost 89%. The main reason being the unavailability of proper balanced dataset. The current model was just trained on a custom built 150 images dataset which is very less.

A better dataset could ensure a better accuracy.

Also, the training could take much longer and could cost very high. And, for the same reason we couldn't train it beyond using the fast training option in the Custom Vision API. We hope training it more, could make it more robust in recognizing web components more accurately.

FUTURE WORK

The project has a great potential in the future. With a better dataset and more computing abilities and training, this project could outperform anything of it's kind and build accurate websites for the input design mockups.

We already know that Generative models are what is going to be the future of AI. Generative models are the most closely related to human behaviour because they generate and learn by their faults. But, training generative models and specifically for our use case could take a very long time.

Hence, in future, the same project could be extended with a generative model.
Also, the API could be made more scalable and secure.

A reverse training mechanism can be added which can be used to retrain wrong predictions.

REFERENCES

[1] (2017, May 22). pix2code: Generating Code from Graphical User Interface .... Retrieved November 26, 2019, from https://arxiv.org/abs/1705.07962

[2] (n.d.). Custom Vision | Microsoft Azure. Retrieved November 26, 2019, from https://azure.microsoft.com/en-us/services/cognitive-services/custom-vision-service/

[3] (2016, November 7). DeepCoder: Learning to Write Programs. Retrieved November 26, 2019, from https://arxiv.org/abs/1611.01989

[4] (n.d.). Sketching Interfaces – Airbnb Design. Retrieved November 26, 2019, from https://airbnb.design/sketching-interfaces/

[5] (n.d.). Research Papers for Cognitive Service APIs – Customer .... Retrieved November 26, 2019, from https://cognitive.uservoice.com/knowledgebase/articles/1097047-research-papers-for-cognitive-service-apis

[1] (2017, May 22). pix2code: Generating Code from Graphical User Interface .... Retrieved November 26, 2019, from https://arxiv.org/abs/1705.07962

[2] (n.d.). Custom Vision | Microsoft Azure. Retrieved November 26, 2019, from https://azure.microsoft.com/en-us/services/cognitive-services/custom-vision-service/

[3] (2016, November 7). DeepCoder: Learning to Write Programs. Retrieved November 26, 2019, from https://arxiv.org/abs/1611.01989

[4] (n.d.). Sketching Interfaces – Airbnb Design. Retrieved November 26, 2019, from https://airbnb.design/sketching-interfaces/

[5] (n.d.). Research Papers for Cognitive Service APIs – Customer .... Retrieved November 26, 2019, from https://cognitive.uservoice.com/knowledgebase/articles/1097047-research-papers-for-cognitive-service-apis