SPE MAJOR PROJECT

YOVO-AGEVERIFY

Rakshit Gupta (IMT2019516)
Abhi Jain(IMT2019501)
Vishal Varma(IMT2019094)

Github Repo:-
https://github.com/rakshit-g/SPE-YOVO
DockerHub Profile:
https://hub.docker.com/r/abhijjainn/yovo-backend-image
https://hub.docker.com/r/abhijjainn/yovo-frontend-image

Abstract

The YOVO approach limits access to age-restricted products and contents. We are using several techniques together which leaves negligible chances of false identification.

The various features of the project are:
The first one is ID card verification - This feature requires the user to upload a photo of his identification card (Aadhar card). The card details are extracted and it uses the Aadhar Api/Verhoeff algo to verify whether the user is above 18 or not.

The second one is Automated facial age estimation - Using this feature, users can verify themselves by capturing a real-time selfie. The selfie is then sent to an ML model which predicts if the user is above 18 or not. The model implemented is quick and accurate.

The final feature is the age token - This feature is a “verify once use anywhere” feature in which on successful verification a user's details are stored that can be used directly on other applications without the hassle of going through the verification process again. Once verified, we will provide a token that will be linked to the user's phone number. The user can verify their age using this token and an OTP which will be sent to his phone number.

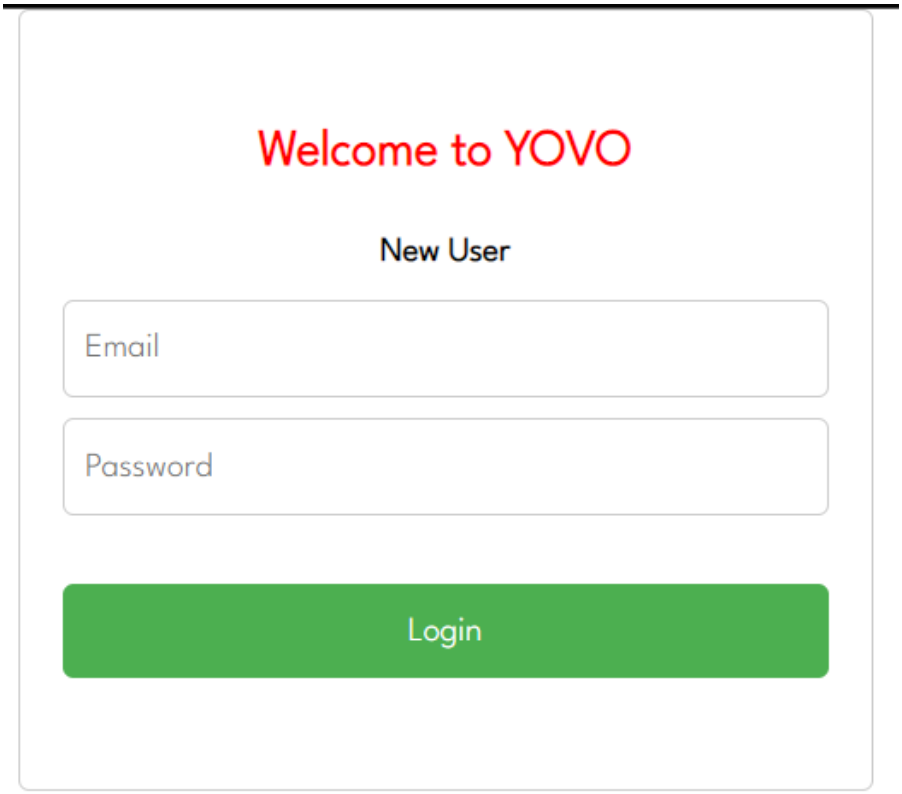
In the backend, we have a machine learning model for predicting the user's age in real-time. There is also a model to extract the information from the aadhar card of the user. This model uses OCR (Optical Character Recognition) technique for extracting the data. There is also a call to Aadhar Api/Verhoeff algo which is used for verifying the aadhar details of the user. We also maintain a database that keeps track of the age tokens provided to the users.

The front-end workflow is as follows. The applications which require age verification, redirect their users to our webpage where users are provided with three choices for verifying their age, namely - ID verification, and real-time age detection. Upon verification, users get an age token which can be used in other applications

Stacks Used

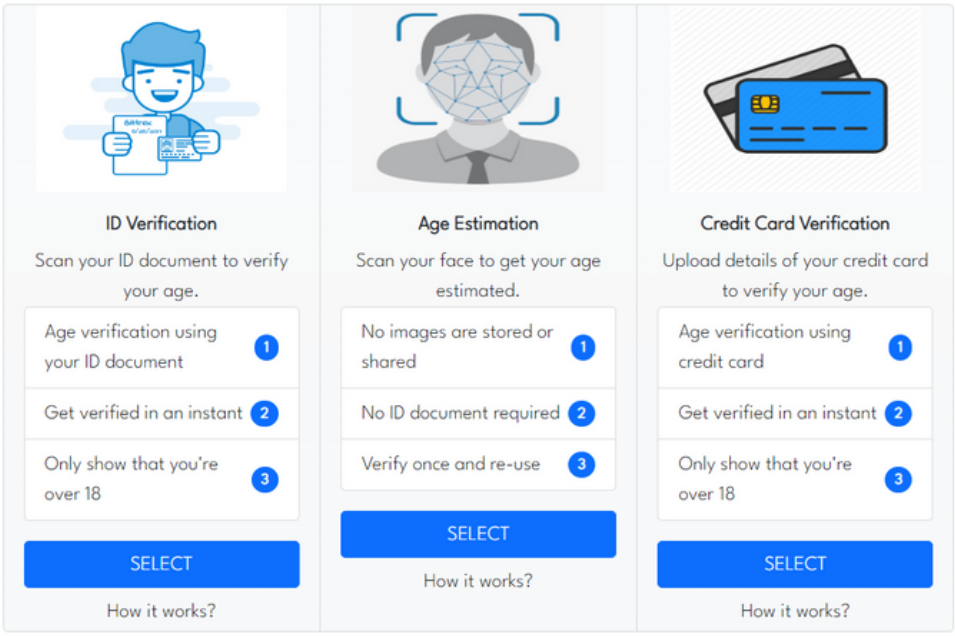
Frontend- React
Backend- Python
Database- Mongo DB
Containerization- Docker
VCS- Git

Login Page

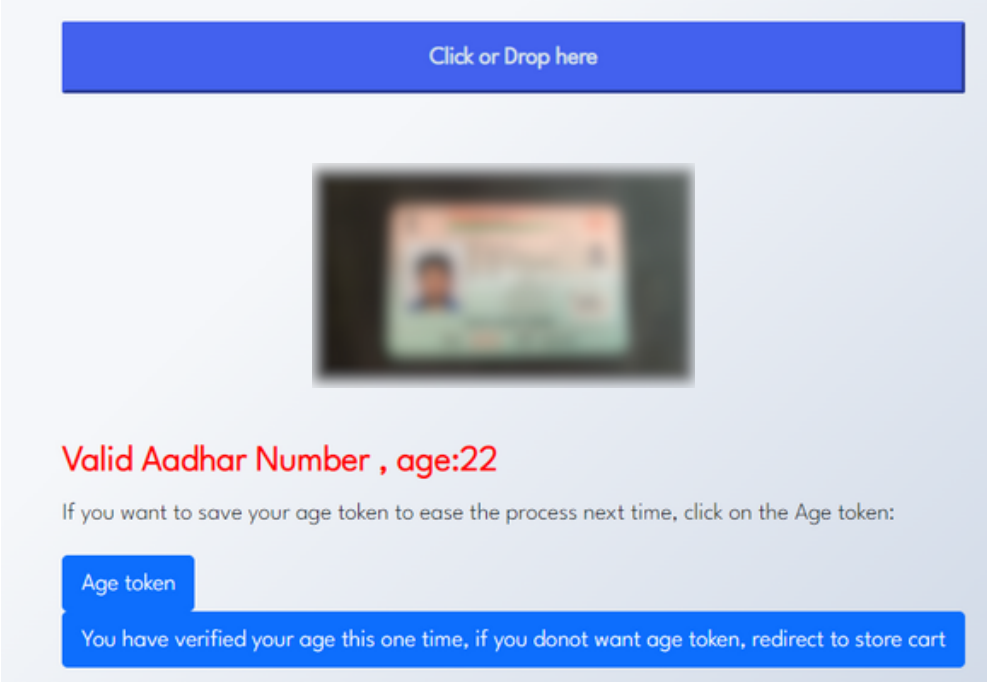


When the user clicks on new user, Ageveriification page appears

Age-Verification



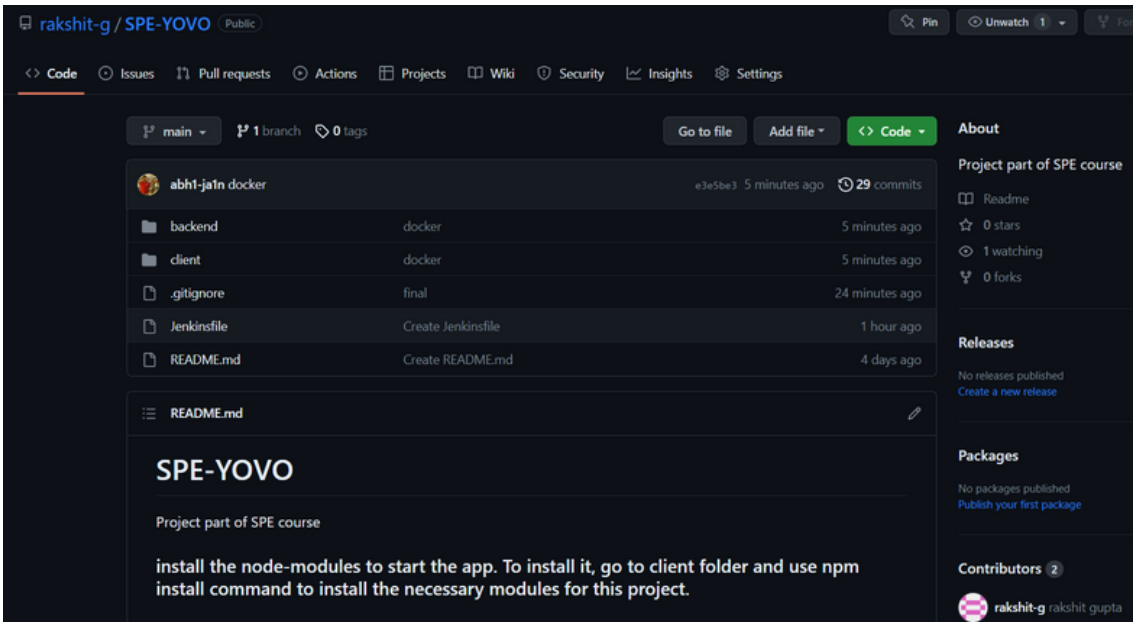
The user has three steps to verify his/her age. For verification once, the user can choose any of the three methods. For saving his age token for future verification, the user can verfify their age only by ID verification and once verified, the user can opt the option to save his/her details so that the user can verify their age in the future directly by logging in with email and password.

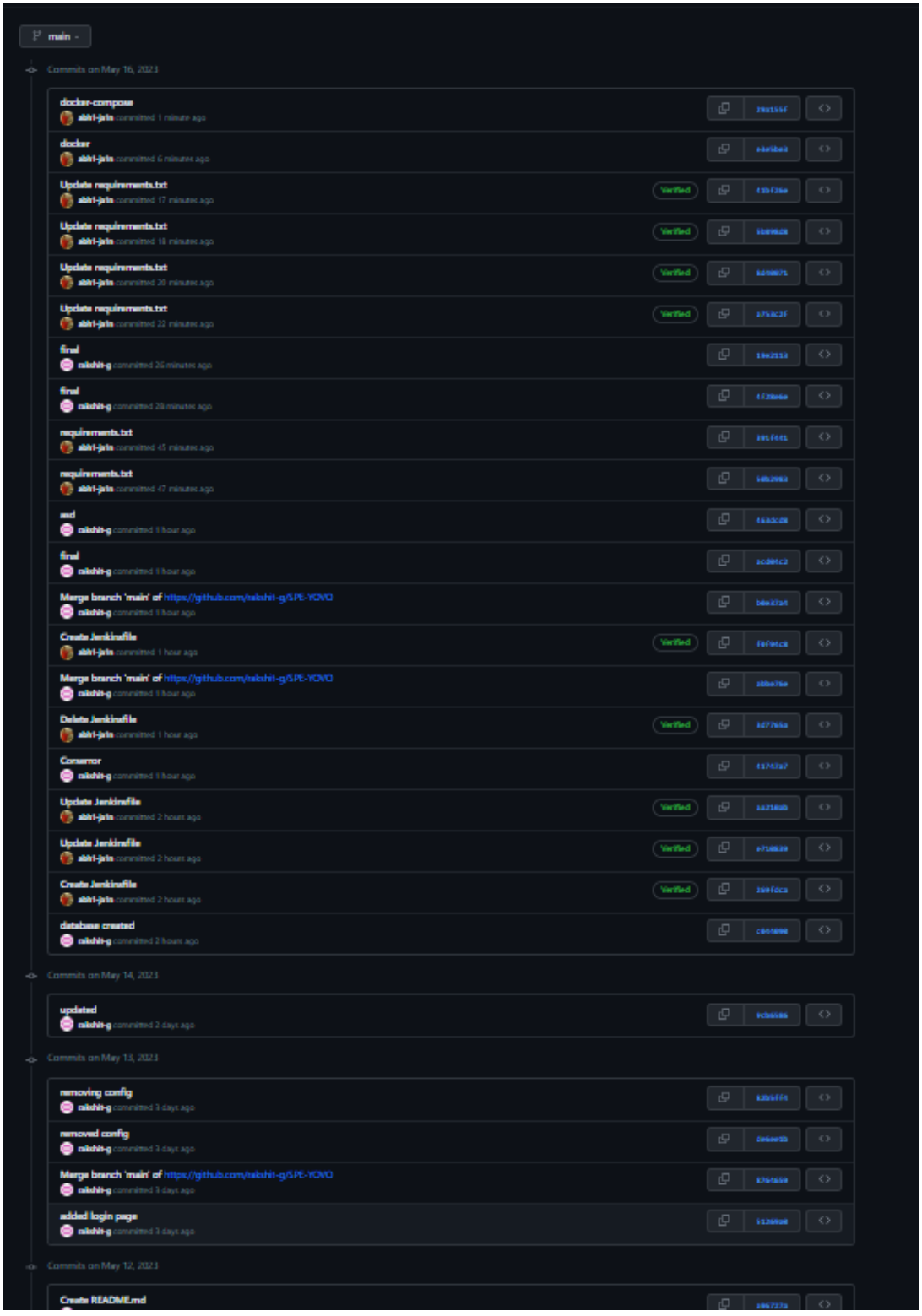


DEVOPS

Git

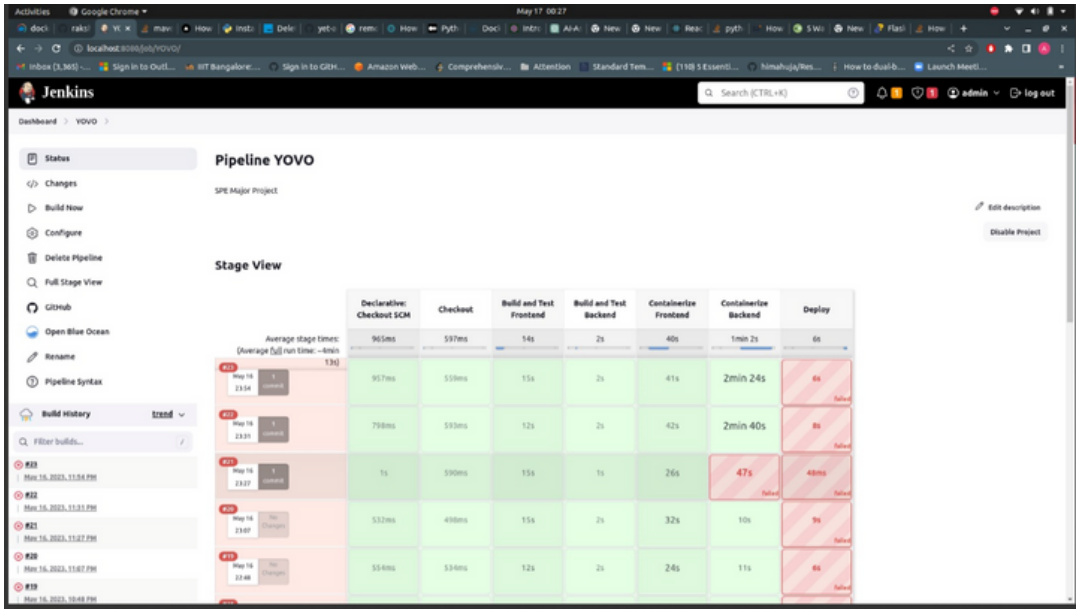
The series of commands are executed to push the local code to the remote repository of GitHub.
git init
 git add.
 git commit -m “Commit Message”
git remote add origin
https://github.com/rakshit-g/SPE-YOVO.git
git push -u origin main
For Contributors
git clone https://github.com/rakshit-g/SPE-YOVO
git add
git commit -m “Commit Message”
git remote add origin
https://github.com/rakshit-g/SPE-YOVO
git push -u origin main





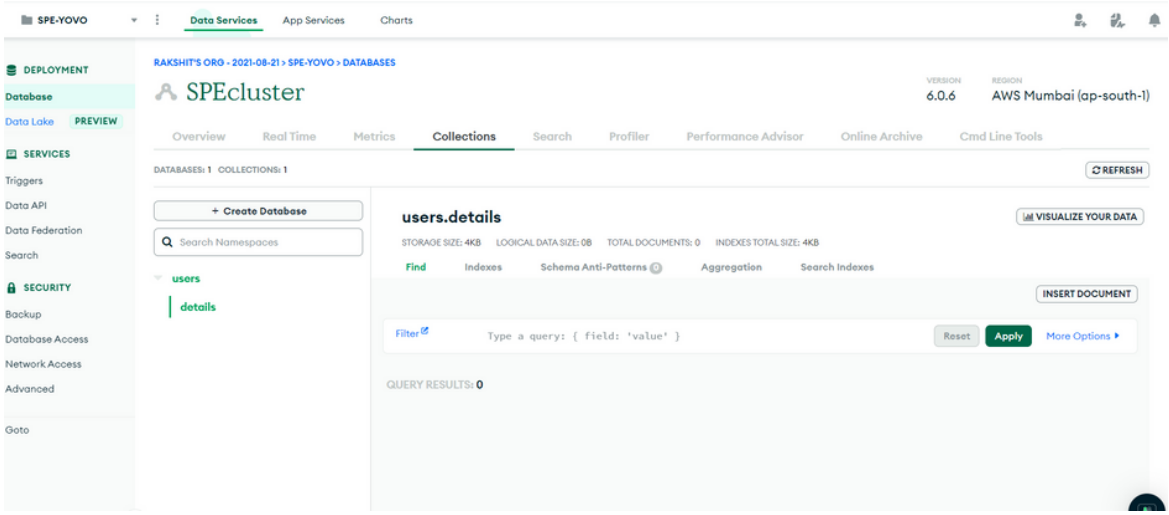
Commit history

Jenkins

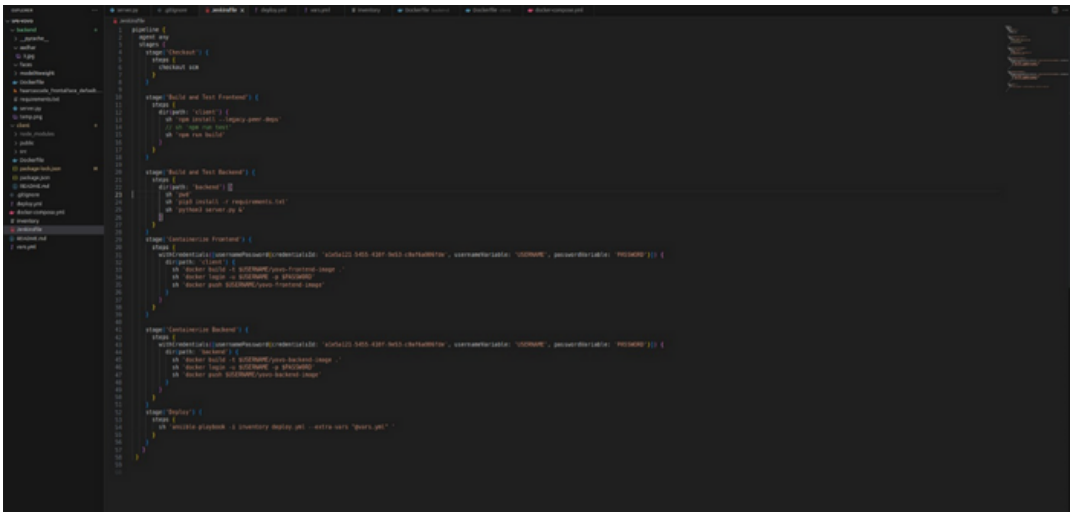


Jenkins Pipeline

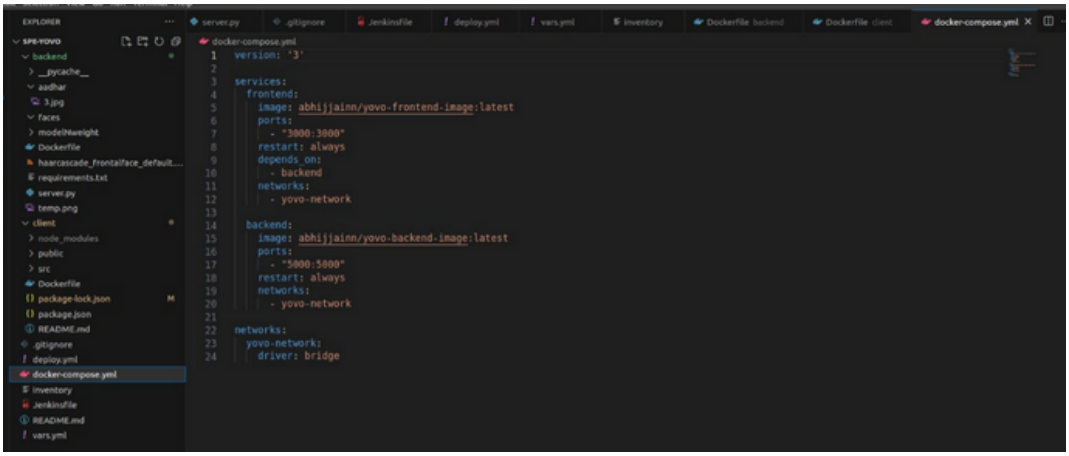
MongoDB



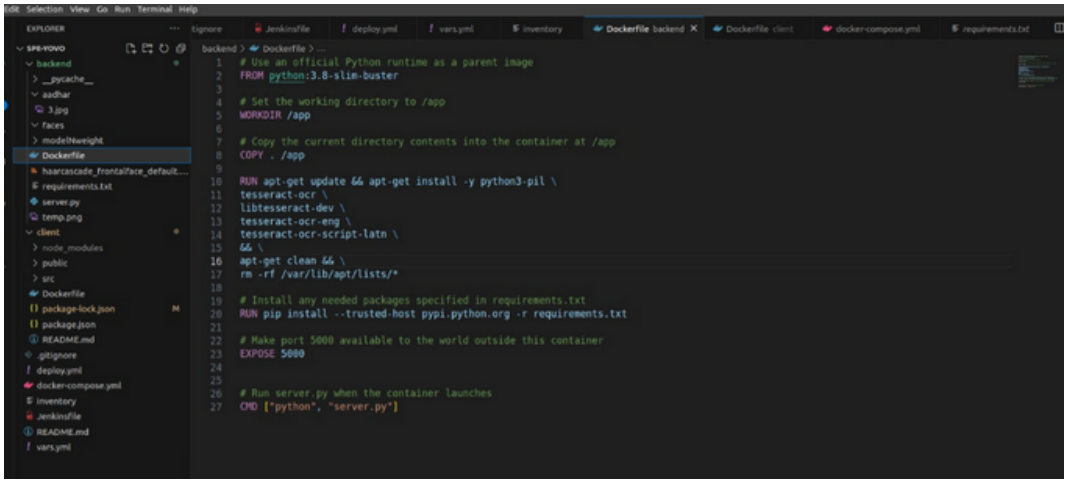
MongoDb database



Jenkins Pipeline



Docker compose file

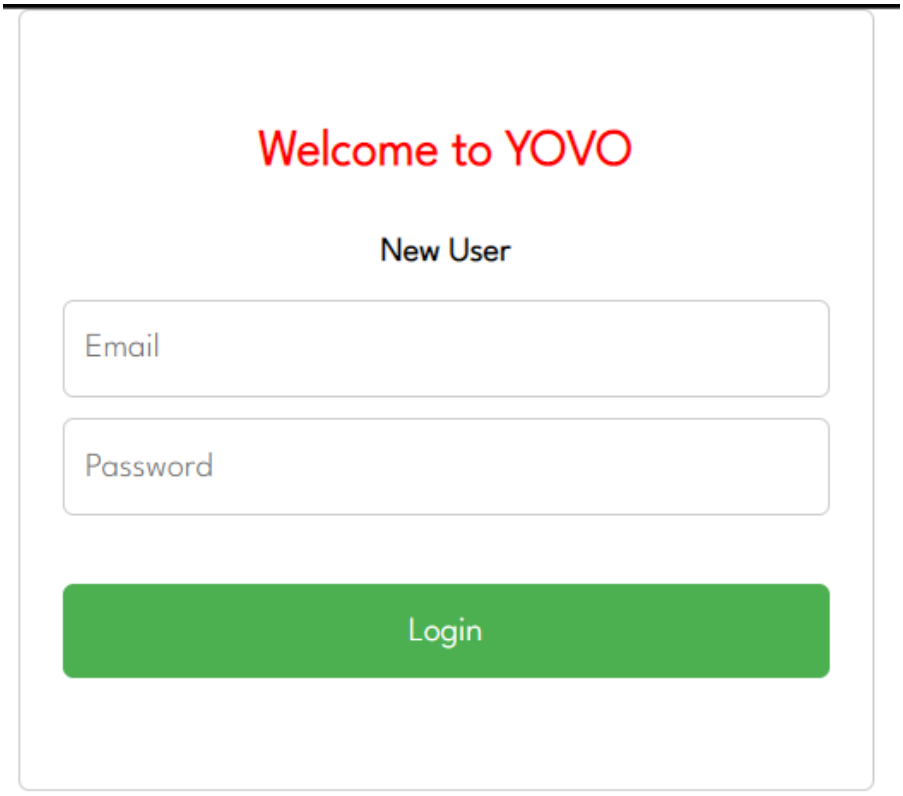


Docker backend file

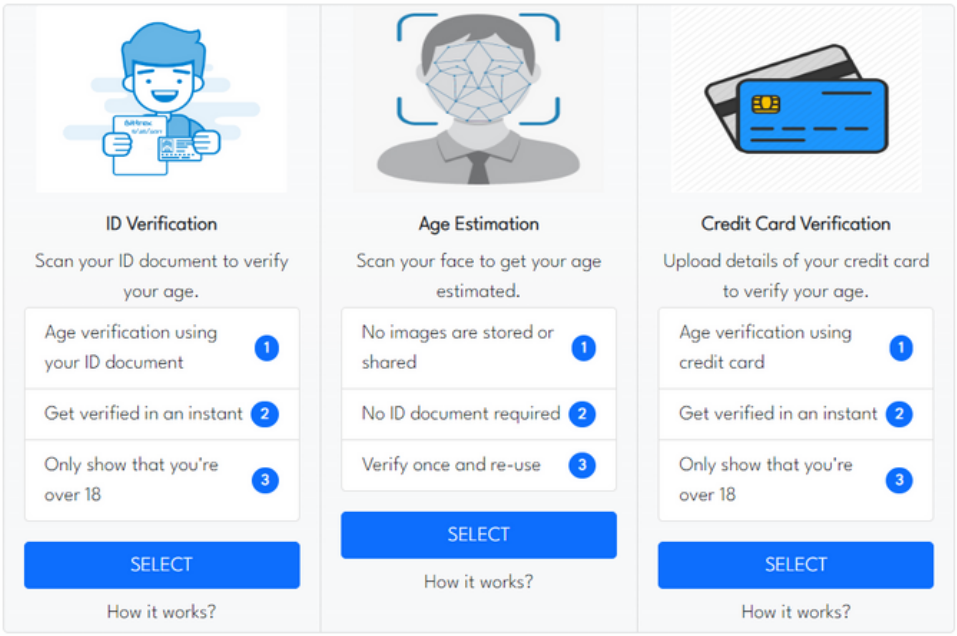
API table

No	Usecase	Api	Request Method
1	Login Page	/login	POST, GET
2	Aadhar Card verification	/aadhar	POST, GET
3	Sign up page using otp	/otp	POST,GEt
4	Automated Age estimation using real time face	/age	POST,GET
5	Credit card verification page	/credit	POST, GET

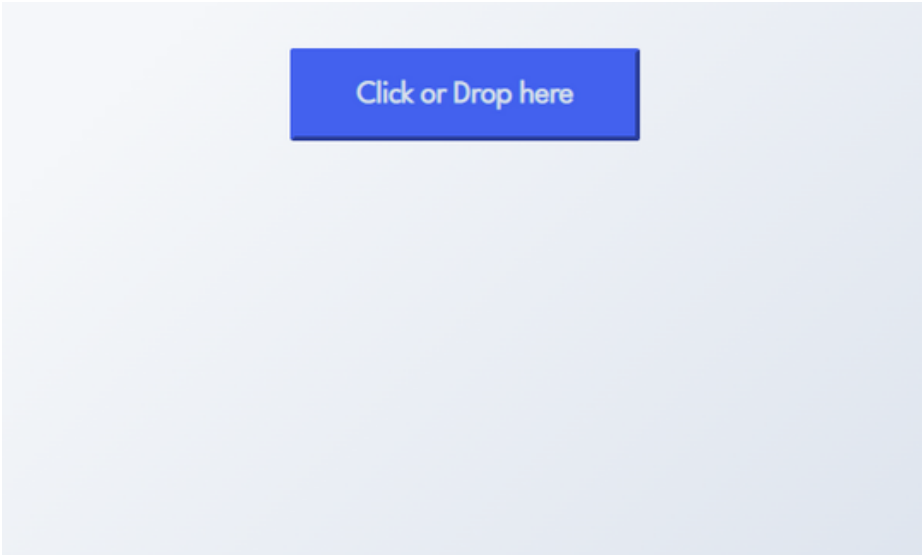
YOVO-ageverify SNAPSHOTS



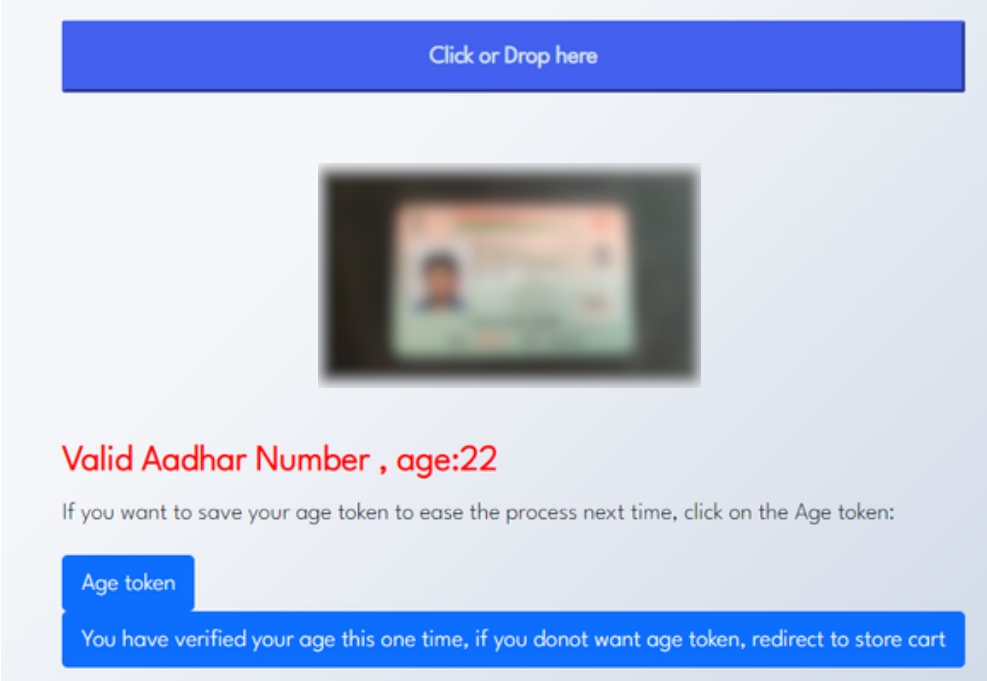
Login page



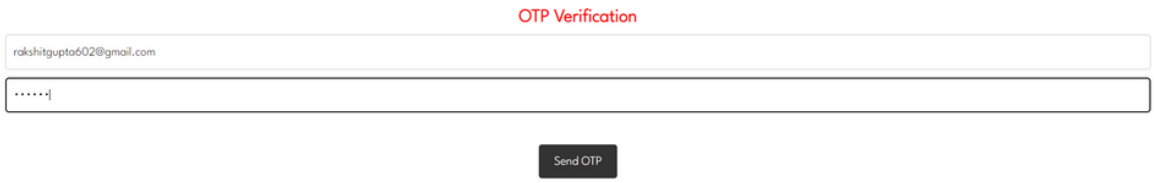
Verification methods



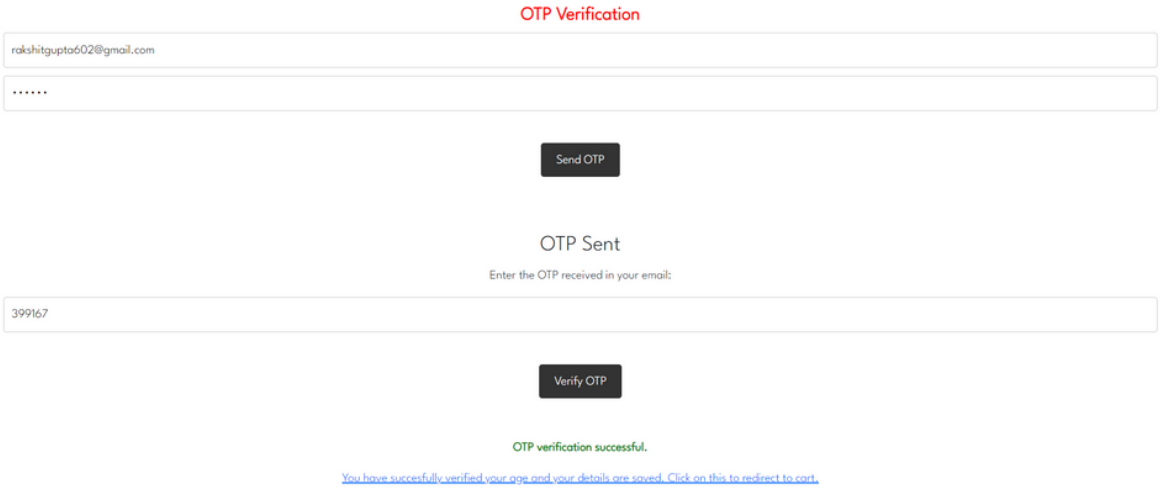
Aadhar upload page



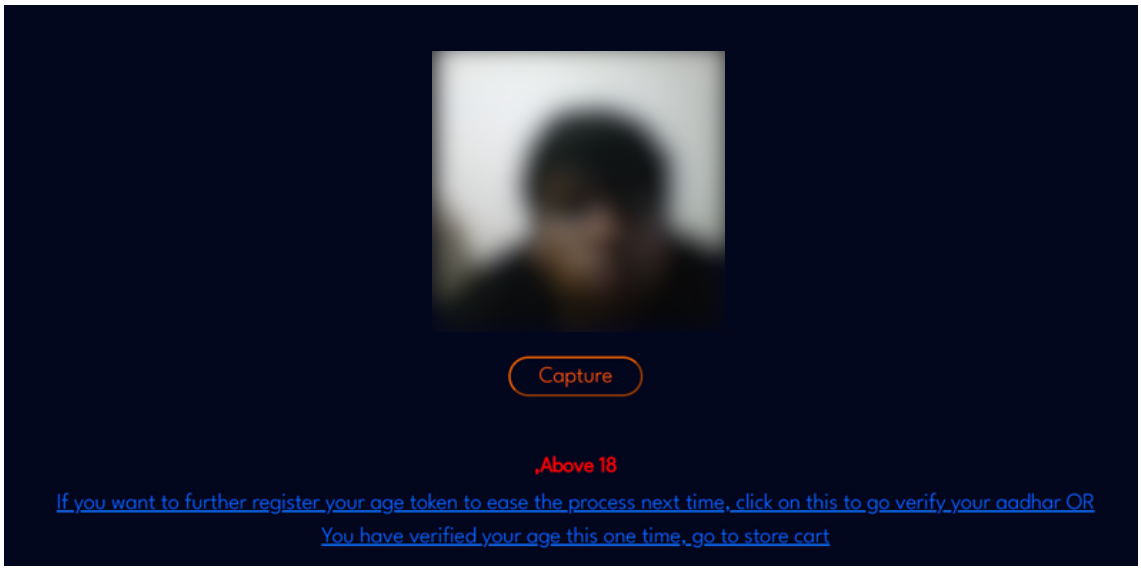
after verfication of adhar



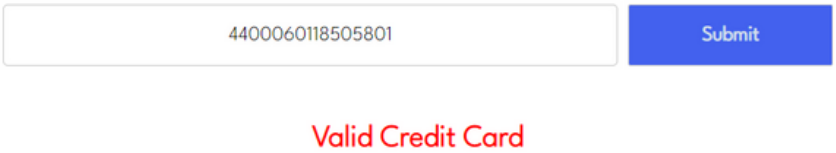
sign up using otp verification



Otp verified



real-time age verification using face



credit card verification page

7. **Monitoring:**

Monitoring ELK stack is a collection of three open-source products — Elasticsearch, Logstash, and Kibana. ELK stack provides centralized logging in order to identify problems with servers or applications.
E stands for ElasticSearch: used for storing logs L stands for LogStash : used for both shipping as well as processing and storing logs K stands for Kibana: is a visualization tool (a web interface) which is hosted through Nginx or Apache.