Python mini project report

Title-Certificate Butler

Mark Lopes-9913

Vivian Ludrick-9914

Introduction:

The project aims to develop a certificate generation system utilizing Vite + Svelte for frontend and Django with SQLite for backend. It begins with requirement gathering and technology stack selection, followed by frontend and backend development phases covering UI design, API implementation, data processing, image generation, zip file creation, and frontend-backend integration. The system will streamline certificate management for competitions, including features for form submission, history access, and efficient response handling.

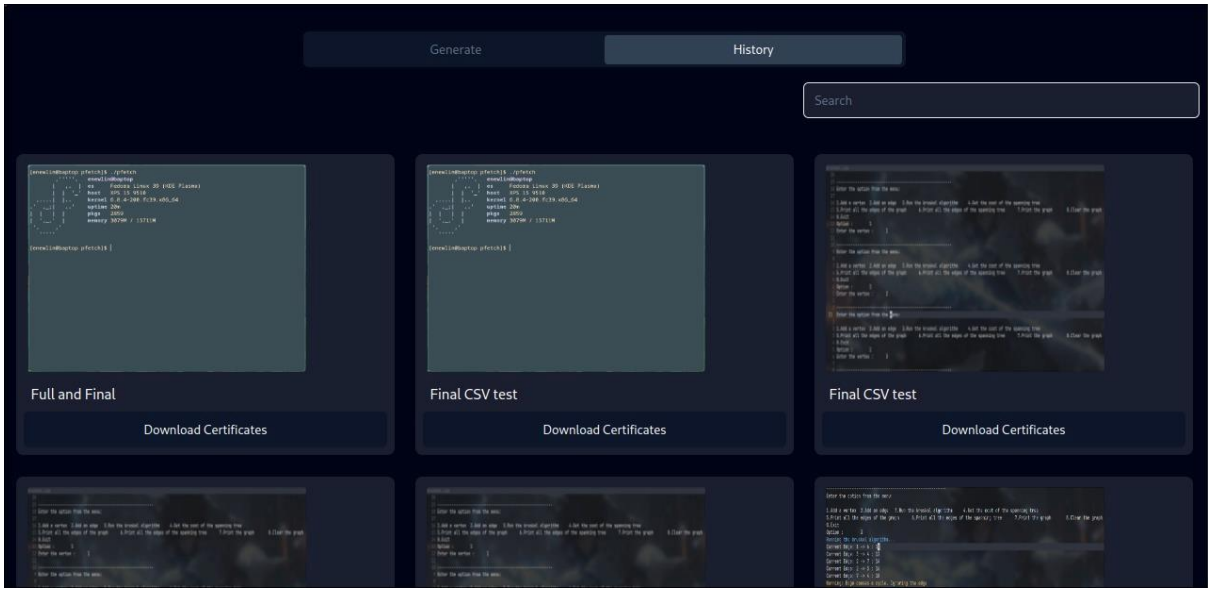
Methodology:

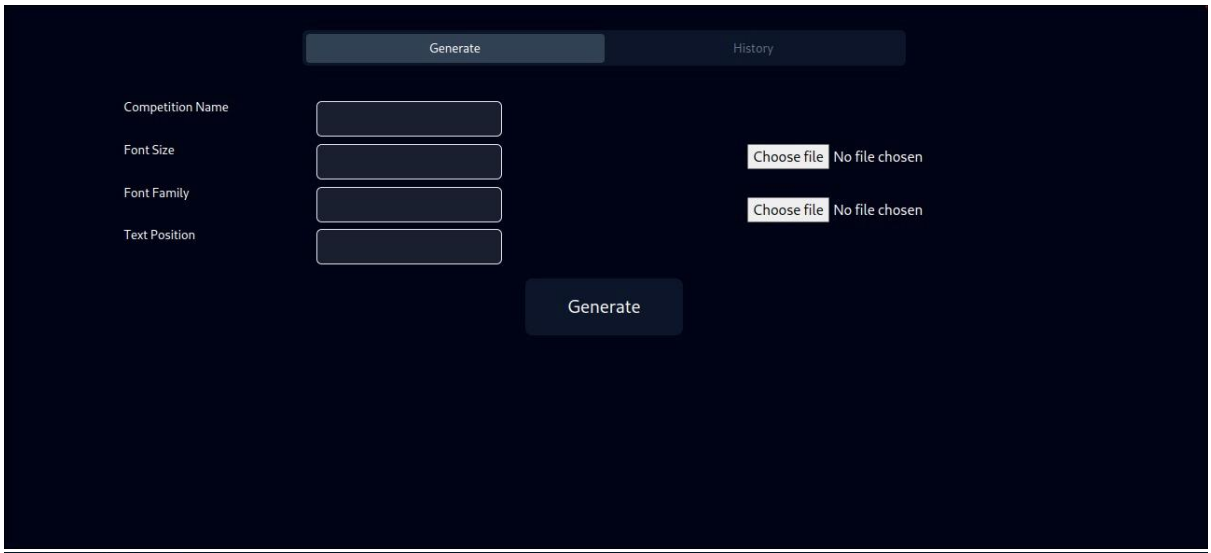
1. Project Planning:
    - Requirement Gathering: Understand the user needs and define the features required for the application.
    - Technology Stack Selection: Choose Vite + Svelte for frontend development and Django with SQLite for backend development based on project requirements and familiarity.
2. Frontend Development:
    - Environment Setup: Set up the development environment for Vite + Svelte.
    - UI/UX Design: Design the user interface, including the homepage, form for certificate generation, and history tab.
    - Form Implementation: Create the form to capture user inputs for certificate generation.
3. Backend Development:
    - Environment Setup: Set up Django and SQLite for backend development.

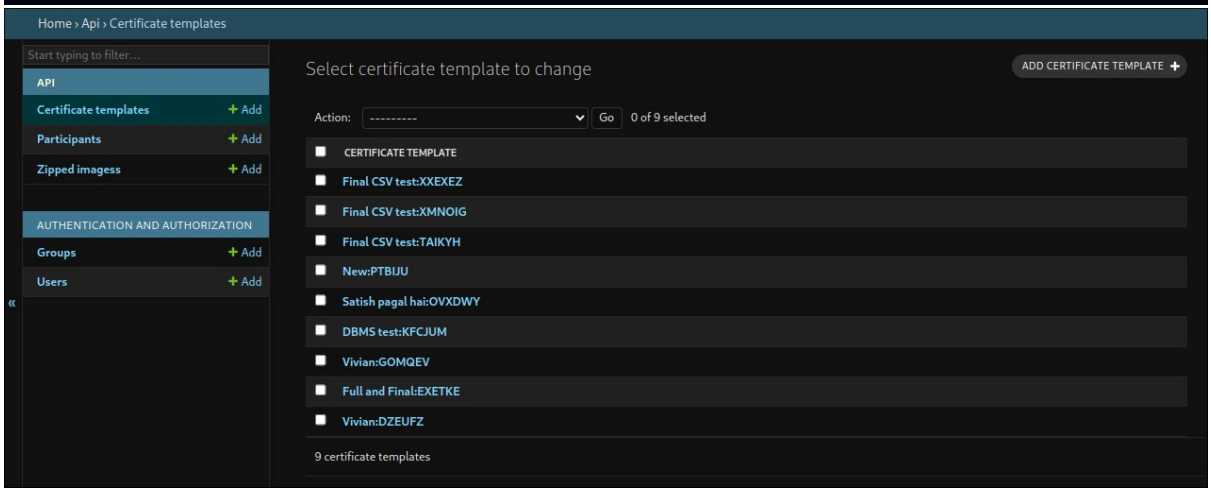
- Database Design: Define the database schema with tables: Certificate_templates, Participants, User, and Zipped_file.
- API Development: Implement APIs to handle incoming data from the frontend, including storing certificate templates, participant details, and generating zip files.
- Integration with Pillow: Integrate the Pillow library for image generation and manipulation.

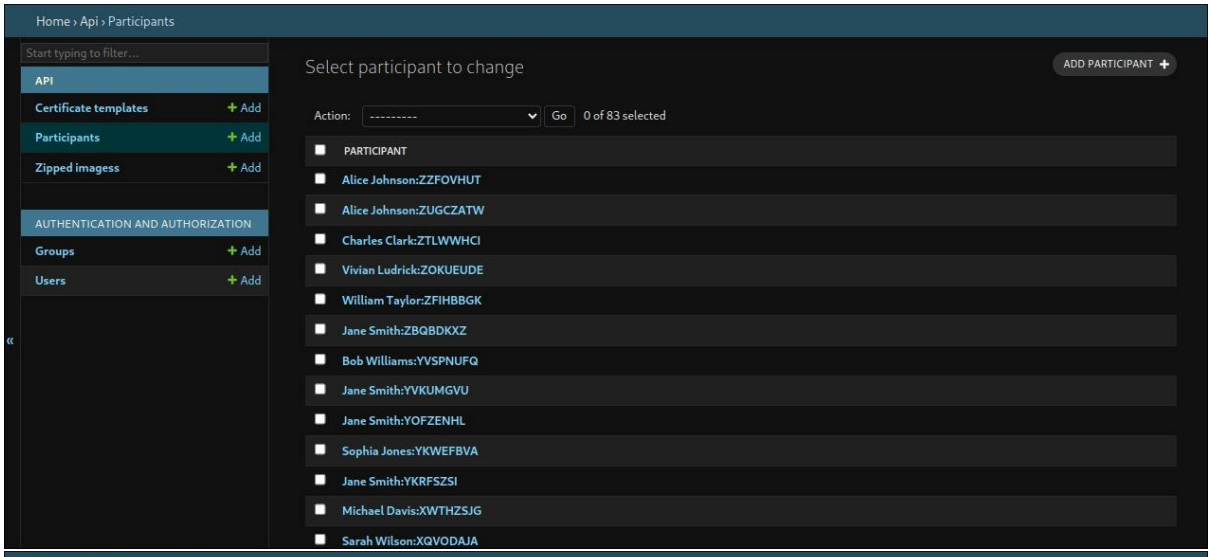
4. Data Processing:
   - CSV Parsing: Develop functionality to parse the uploaded CSV file to extract participant details.
   - Code Generation: Implement logic to generate unique codes for certificate templates and associate participants with the generated template code.
5. Image Generation:
   - Certificate Rendering: Use Pillow library to render participant details on the certificate templates.
   - Image Storage: Store the generated images in the storage and create public links for access.
6. Zip File Creation:
   - Certificate Compilation: Compile individual certificate images into a zip file for each competition.
   - Storage and Linking: Store the generated zip files and link them to the corresponding competition IDs.
7. Frontend-Backend Integration:
   - API Integration: Connect the frontend with the backend through APIs to facilitate data exchange.
   - Response Handling: Implement logic to handle responses from the backend, particularly the public links for downloading zip files.
8. History Access:
   - Implement a history tab on the homepage.Allow users to view and download previously generated certificates for past competitions.

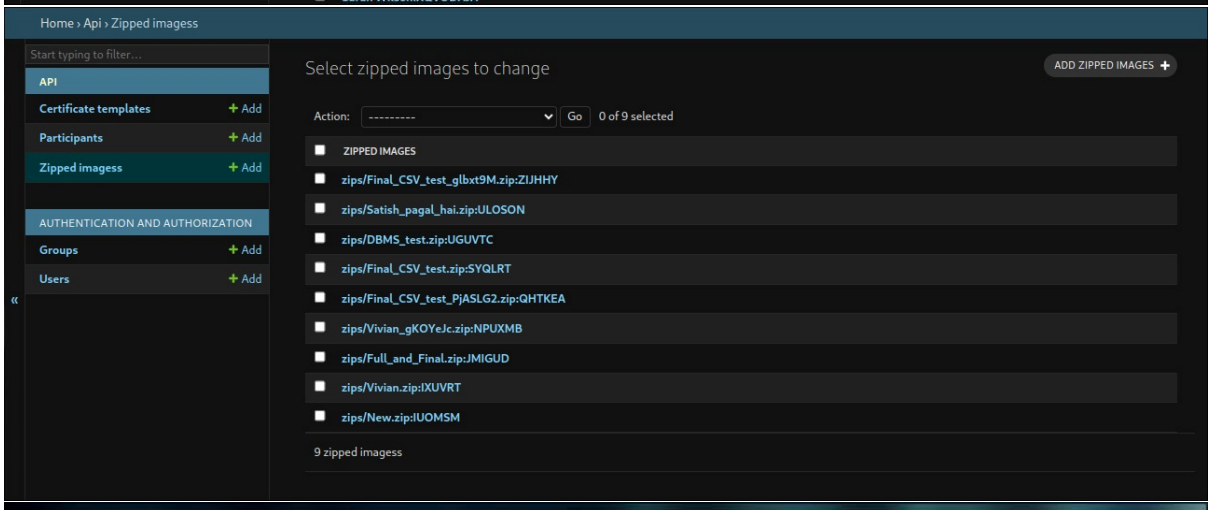
Result:

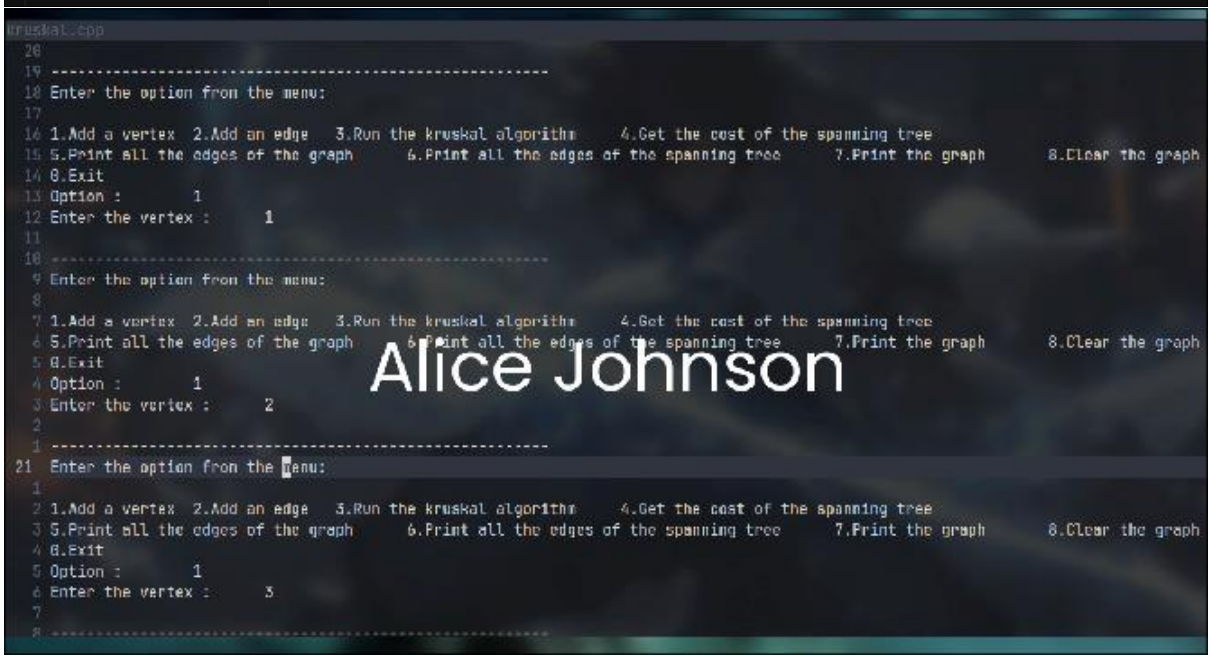

Generate
History
Search
Full and Final
Download Certificates
Final CSV test
Download Certificates
Final CSV test
Download Certificates


Generate
History
Competition Name
Font Size
Font Family
Text Position
Choose file
No file chosen
Choose file
No file chosen
Generate


Home › Api › Certificate templates
Start typing to filter...
API
Certificate templates + Add
Participants + Add
Zipped imagess + Add
AUTHENTICATION AND AUTHORIZATION
Groups + Add
Users + Add
Select certificate template to change
ADD CERTIFICATE TEMPLATE +
Action: --------- Go 0 of 9 selected
CERTIFICATE TEMPLATE
Final CSV test:XXEXEZ
Final CSV test:XMNOIG
Final CSV test:TAIKYH
New:PTBIJU
Satish pagal hai:OVXDWY
DBMS test:KFCJUM
Vivian:GOMQEV
Full and Final:EXETKE
Vivian:DZEUFZ
9 certificate templates


Home › Api › Participants
Start typing to filter...
API
Certificate templates + Add
Participants + Add
Zipped imagess + Add
AUTHENTICATION AND AUTHORIZATION
Groups + Add
Users + Add
Select participant to change
ADD PARTICIPANT +
Action: --------- Go 0 of 83 selected
PARTICIPANT
Alice Johnson:ZZFOVHUT
Alice Johnson:ZUGCZATW
Charles Clark:ZTLWWHCI
Vivian Ludrick:ZOKUEUDE
William Taylor:ZFIHBBGK
Jane Smith:ZBQBDKXZ
Bob Williams:YVSPNUFQ
Jane Smith:YVKUMGVU
Jane Smith:YOFZENHL
Sophia Jones:YKWEFBVA
Jane Smith:YKRFSZSI
Michael Davis:XWTHZSJG
Sarah Wilson:XQVODAJA


Home › Api › Zipped imagess
Start typing to filter...
API
Certificate templates + Add
Participants + Add
Zipped imagess + Add
AUTHENTICATION AND AUTHORIZATION
Groups + Add
Users + Add
Select zipped images to change
ADD ZIPPED IMAGES +
Action: --------- Go 0 of 9 selected
ZIPPED IMAGES
zips/Final_CSV_test_glbxt9M.zip:ZIJHHY
zips/Satish_pagal_hai.zip:ULOSON
zips/DBMS_test.zip:UGUVTC
zips/Final_CSV_test.zip:SYQLRT
zips/Final_CSV_test_PjASLG2.zip:QHTKEA
zips/Vivian_gKOYeJc.zip:NPUXMB
zips/Full_and_Final.zip:JMIGUD
zips/Vivian.zip:IXUVRT
zips/New.zip:IUOMSM
9 zipped imagess


kruskal.cpp
Enter the option from the menu:
1.Add a vertex 2.Add an edge 3.Run the kruskal algorithm 4.Get the cost of the spanning tree
5.Print all the edges of the graph 6.Print all the edges of the spanning tree 7.Print the graph 8.Clear the graph
0.Exit
Option : 1
Enter the vertex : 1
Enter the option from the menu:
1.Add a vertex 2.Add an edge 3.Run the kruskal algorithm 4.Get the cost of the spanning tree
5.Print all the edges of the graph 6.Print all the edges of the spanning tree 7.Print the graph 8.Clear the graph
0.Exit
Option : 1
Enter the vertex : 2
Alice Johnson
Enter the option from the menu:
1.Add a vertex 2.Add an edge 3.Run the kruskal algorithm 4.Get the cost of the spanning tree
5.Print all the edges of the graph 6.Print all the edges of the spanning tree 7.Print the graph 8.Clear the graph
0.Exit
Option : 1
Enter the vertex : 3

Conclusion:

In conclusion, the certificates generation system developed offers a robust solution for successfully managing certificate for numerous competitions or occasions. By leveraging a cautiously selected era stack and following a structured development method, the device gives a person-pleasant interface, seamless data exchange between frontend and backend, and complete functionalities which includes shape submission, records get entry to, and zip report creation. With its potential to streamline certificates control processes, the project contributes to enhancing efficiency and person enjoy in managing certificate for numerous occasions.