



E-CERTIFICATE GENERATOR

GUIDE: JISSMOL JOSE

MEMBERS

**MOHAMMED
FARHAN
UNT22CS040**

**MOHAMMED
NOWFAL
UNT22CS042**

**RADHESYAM
RAGAV
UNT22CS051**

**SABAH KJ
UNT22CS053**

TABLE OF CONTENT

01

ABSTRACT

02

INTRODUCTION

03

MODULES

04

CONCLUSION

1. ABSTRACT

The "E-Certificate Generator" project will help organizations create certificates quickly and easily for events and programs. Users will upload participant details through a CSV file, add images like logos or signatures, and select a template. The system will automatically generate certificates in bulk. This project will be done to reduce the time and effort needed for manual certificate creation. It will focus on solving problems like inaccuracy and inefficiency in handling large amounts of data. These findings will be important as they will provide an easy-to-use and reliable way to create professional certificates, saving time and ensuring accuracy.

2. INTRODUCTION

- Organizations and institutions often need to generate large volumes of certificates for **events**, **courses**, or **achievements** which can be a **time-consuming** and **error-prone** process if done manually.
- To address this challenge, we are developing an **E-Certificate Generator** that automates **the creation of certificates in bulk**.
- The system will utilize **CSV files** for recipient data, **customizable templates**, and images to generate personalized certificates in **PDF format**.
- This project aims to simplify and streamline the certificate generation process, reducing **manual effort**, ensuring **accuracy**, and saving **valuable time** for users.

3. MODULES

3.1

User Interface:

The frontend will be built using HTML, CSS, and JavaScript. It will let users upload CSV files, images, and choose templates.

3. MODULES

3.1

User Interface:

The frontend will be built using HTML, CSS, and JavaScript. It will let users upload CSV files, images, and choose templates.

3.2

Template

Management :

The system will include predefined templates for certificate designs. It will allow users to place images like logos and signatures on the templates before integrating CSV data.

3. MODULES

3.1

User Interface:

The frontend will be built using HTML, CSS, and JavaScript. It will let users upload CSV files, images, and choose templates.

3.2

Template

Management :

The system will include predefined templates for certificate designs. It will allow users to place images like logos and signatures on the templates before integrating CSV data.

3.3

CSV Processing:

The system will read and process the CSV file using Fast-CSV. It will extract recipient data for creating certificates.

3. MODULES

3.1

User Interface:

The frontend will be built using HTML, CSS, and JavaScript. It will let users upload CSV files, images, and choose templates.

3.2

Template

Management :

The system will include predefined templates for certificate designs. It will allow users to place images like logos and signatures on the templates before integrating CSV data.

3.3

CSV Processing:

The system will read and process the CSV file using Fast-CSV. It will extract recipient data for creating certificates.

3.4

Certificate Generation:

The system will create certificates by combining the selected template (with integrated images) and recipient data to make personalized certificates.

3. MODULES

3.1

User Interface:
The frontend will be built using HTML, CSS, and JavaScript. It will let users upload CSV files, images, and choose templates.

3.2

Template Management :
The system will include predefined templates for certificate designs. It will allow users to place images like logos and signatures on the templates before integrating CSV data.

3.3

CSV Processing:
The system will read and process the CSV file using Fast-CSV. It will extract recipient data for creating certificates.

3.4

Certificate Generation:
The system will create certificates by combining the selected template (with integrated images) and recipient data to make personalized certificates.

3.5

Download:
The system will let users download the created certificates as a ZIP file or individual PDFs.

3. MODULES

3.1

User Interface:
The frontend will be built using HTML, CSS, and JavaScript. It will let users upload CSV files, images, and choose templates.

3.2

Template Management :
The system will include predefined templates for certificate designs. It will allow users to place images like logos and signatures on the templates before integrating CSV data.

3.3

CSV Processing:
The system will read and process the CSV file using Fast-CSV. It will extract recipient data for creating certificates.

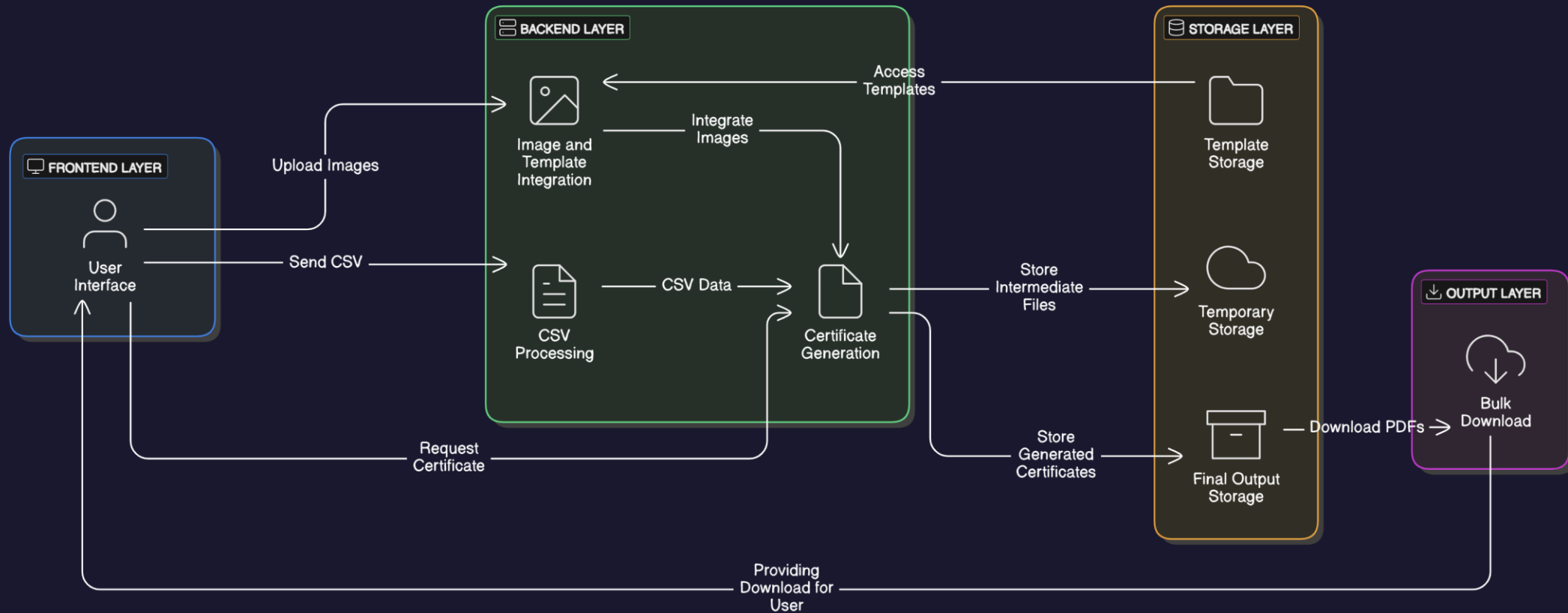
3.4

Certificate Generation:
The system will create certificates by combining the selected template (with integrated images) and recipient data to make personalized certificates.

3.5

Download:
The system will let users download the created certificates as a ZIP file or individual PDFs.

3.5. ARCHITECTURE



3.6. IMPLEMENTATION TOOLS

The following **tools** and **technologies** are used in this project:

- Frontend: **HTML** and **CSS** for designing the user interface.
- Backend: **Node.js** for managing the backend processes.
- Libraries: **Fast-CSV** for parsing CSV files and **PDF-Kit** for generating PDFs.

4. CONCLUSION

In conclusion, The "E-Certificate Generator" project will make creating certificates faster, easier, and more accurate. It will help users upload details and generate many certificates at once, solving the problems of doing it manually. This project will give organizations a simple and useful way to create professional certificates, saving time and effort. Overall, it will be a helpful tool for making certificate management easier.

THANK YOU