

This project is a student frontend for viewing academic results in a certificate format. It consists of two main pages: the front page where students enter their enrollment number and the result page where their results are displayed in a certificate format, ready for printing.

Features

1. **Enrollment Number Input:** Allows students to enter their enrollment number.
2. **Result Display:** Shows the student's result in a detailed certificate format.
3. **Printing:** Provides an option to print the result certificate.
4. **Backend Integration:** Fetches result data from the backend server based on the enrollment number.

Project Structure

java

```
studentfrontend/  
├── public/  
│   └── index.html  
├── src/  
│   ├── assets/  
│   │   └── images/  
│   │       ├── Ct_logo.png  
│   │       └── CT_University_logo.png  
│   ├── components/  
│   │   ├── Certificate.js  
│   │   ├── FrontPage.js  
│   │   └── ResultPage.js  
│   ├── App.css  
│   ├── App.js  
│   └── index.js  
├── .gitignore  
├── package.json  
└── README.md
```

Installation and Setup

1. Clone the Repository:

sh

```
git clone <repository-url>  
cd studentfrontend
```

2. Install Dependencies:

sh

```
npm install
```

3. Run the Application:

```
sh
```

```
npm start
```

4. Backend Server:

Ensure the backend server is running at `http://localhost:4000`.

Components

1. `App.js`

The main component that manages the state and renders either the `FrontPage` or `ResultPage` based on the current state.

2. `FrontPage.js`

Handles the input of the enrollment number and transitions to the `ResultPage`. It includes:

- An input field for the enrollment number.
- A search button to submit the enrollment number.

3. `ResultPage.js`

Displays the result in a certificate format fetched from the backend. It includes:

- The `Certificate` component to display the result.
- A back button to return to the `FrontPage`.
- A print button to print the certificate.

4. `Certificate.js`

Displays the detailed marks certificate (DMC) with:

- Enrollment number
- Student name
- College
- Course
- Semester
- Exam month/year
- Subject details (subject codes, subjects, and GPA)
- Status
- Percentage

Data Flow and Block Diagram

Data Flow

1. Enrollment Number Input:

- The student enters their enrollment number on the `FrontPage`.

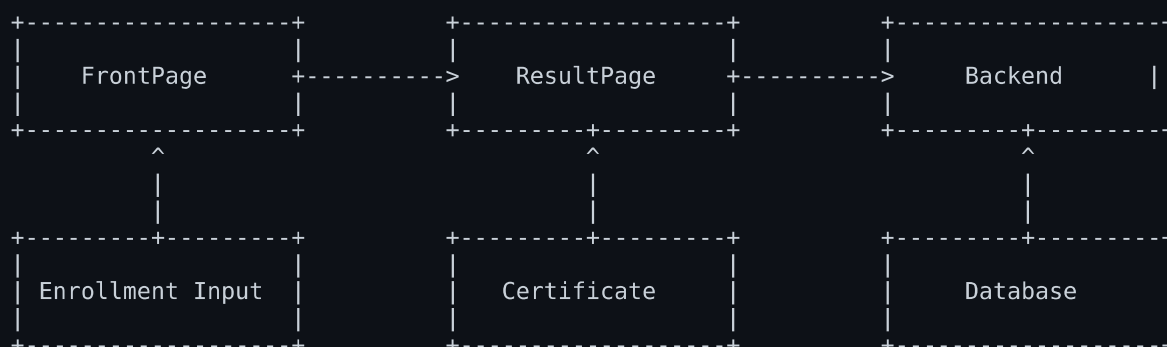
- Upon clicking "Search", the enrollment number is stored in the state and the view switches to `ResultPage`.

2. Result Display:

- `ResultPage` fetches the result data from the backend server using the enrollment number.
- The fetched data is displayed in the `Certificate` component.
- The student can print the certificate or go back to the `FrontPage`.

Block Diagram

plaintext



Explanation:

1. FrontPage:

- Handles input of the enrollment number.
- Transitions to `ResultPage` on successful input.

2. Enrollment Input:

- Component for entering the enrollment number.

3. ResultPage:

- Fetches and displays the result data.
- Provides options to go back to the `FrontPage` or print the certificate.

4. Certificate:

- Displays the detailed marks certificate with all relevant details.

5. Backend:

- Stores and manages the data.
- Interacts with a database to retrieve the data based on the enrollment number.

Running the Project

1. Start the Frontend:

```
sh
```

```
npm start
```

2. Ensure the Backend Server is Running:

The backend server should be running at `http://localhost:4000`.