Card Payment System

You are working as a Fullstack developer for a Payment Network company. You are given a new project that requires you to create a card registration and payment for a payment gateway with the given requirements.

Backend Functionalities

A REST API will be developed in Django using Django Rest Framework.

Cards Model

Field	Data Type	Present in Request Body
id	UUID	No
name	String	No
number	String	No
expiry	String	No
cvv	Number	No
pin	Number	No

Payment Model

Field	Data Type	Present in Request Body
id	UUID	Yes
card_id	String	Yes
date	Datetime	Yes
amount	Number	Yes

API Routes/Methods

Route	Method	Description
/cards	POST	Register a new Card
/verify	POST	Verify card information (without PIN)
/pay	POST	Payment with PIN
/payments	GET	Payment History

Notes:

- Use only SQLite database. And make proper tables/models and relations accordingly.
- Do not remove any existing files.

Frontend Components

Angular application will be developed with the mentioned components/pages and proper routing.

Register Cards Page

- Create a Card Registration Page on the /register route.
- It will contain a name and pin input and a generate button. On clicking Generate a card will be generated with a random 16 digit number, 3 digit cvv, and an expiry month/year of 5 years from the current date.
- These generated card details will be displayed to the user and will be sent to API via a POST request on /register endpoint.

Payment Page

- Create Payment Page on the /pay route.
- It will contain a reactive form that will take input of card details and payment amount. While the user is typing card details should be verified from the /verify endpoint on API expect for the PIN.
- Once the user clicks the Pay button, a POST request should be done on the /pay route with the form data, and a success message should be displayed.

Payment History Page

- Create a Payments History Page on the /payments route.
- It will display all the previous Payments with Name, Amount, Date/Time and card number with only last 4 digits visible and rest as X.
- All the payments will be in a scrollable list format with the payments sorted from latest to oldest.

Notes

- Do not remove any existing dependency or make changes in a file other than the ones mentioned above.
- Do not remove any **data-testid** attributes.
- To test your progress, you can hit the **npm test** command.