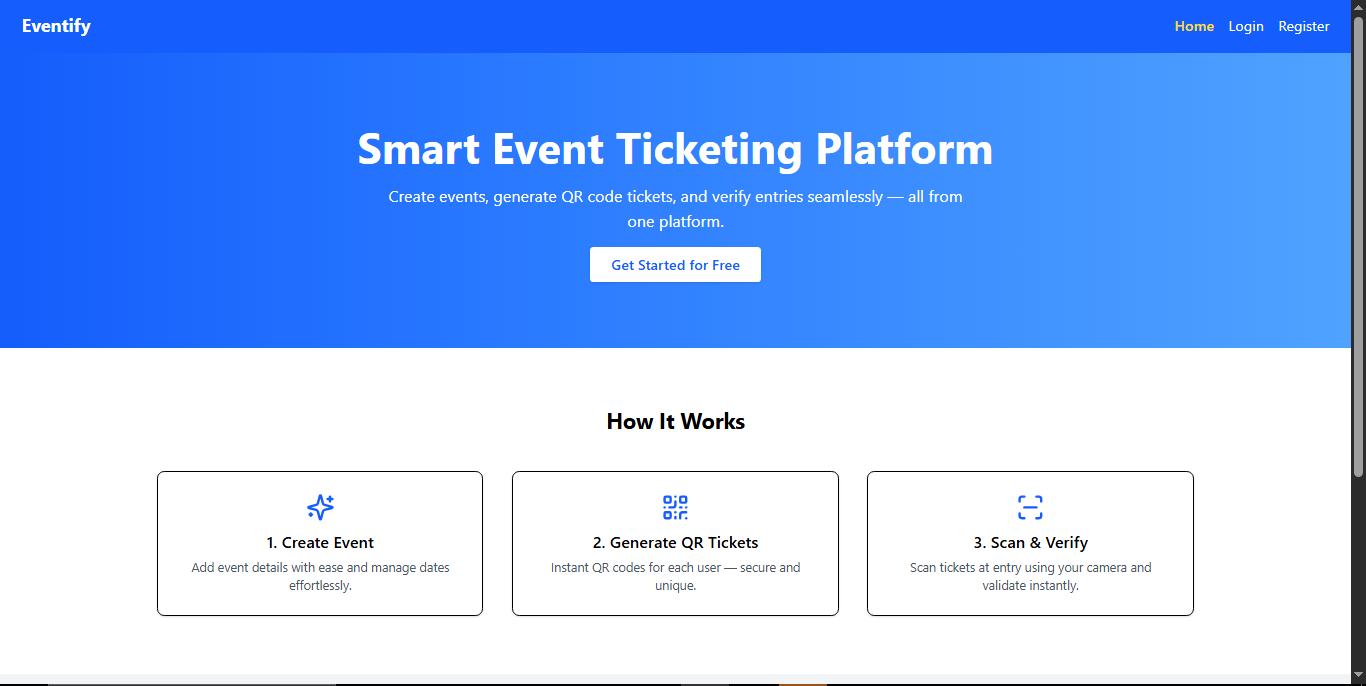
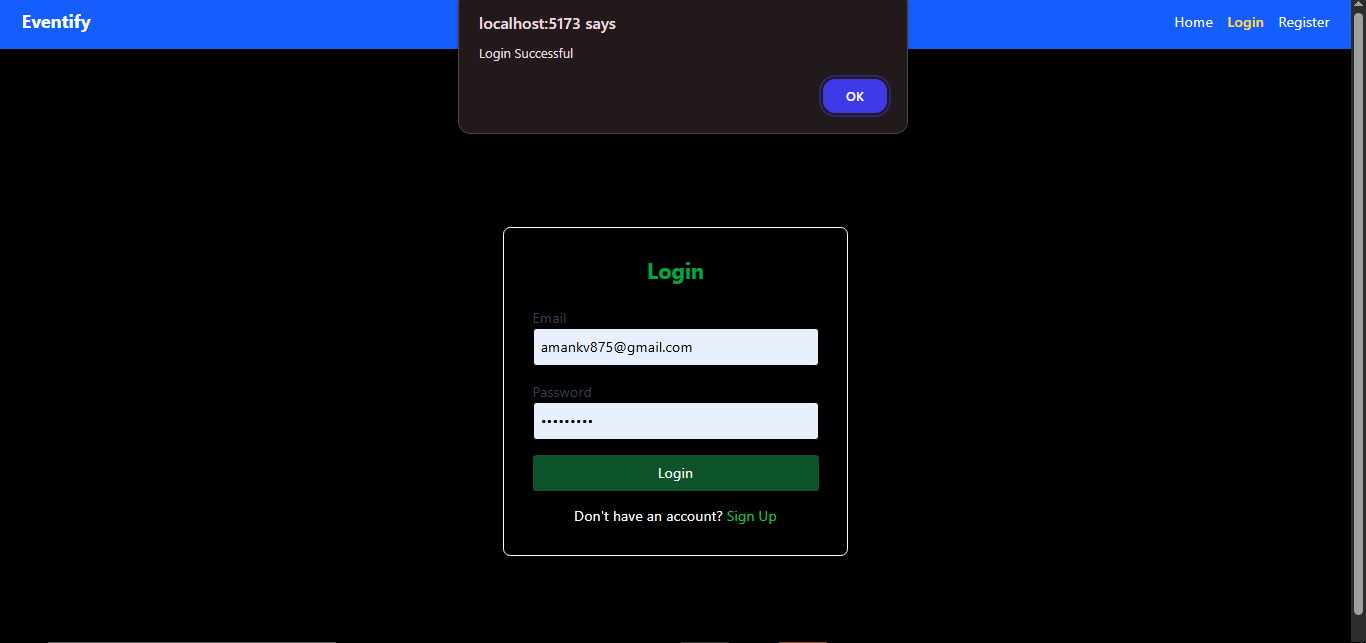
**Event Ticketing Platform - Step-by-Step Full System Flow**



Create Event → User Enrolls → Ticket Generated → QR Scanned → User Verified

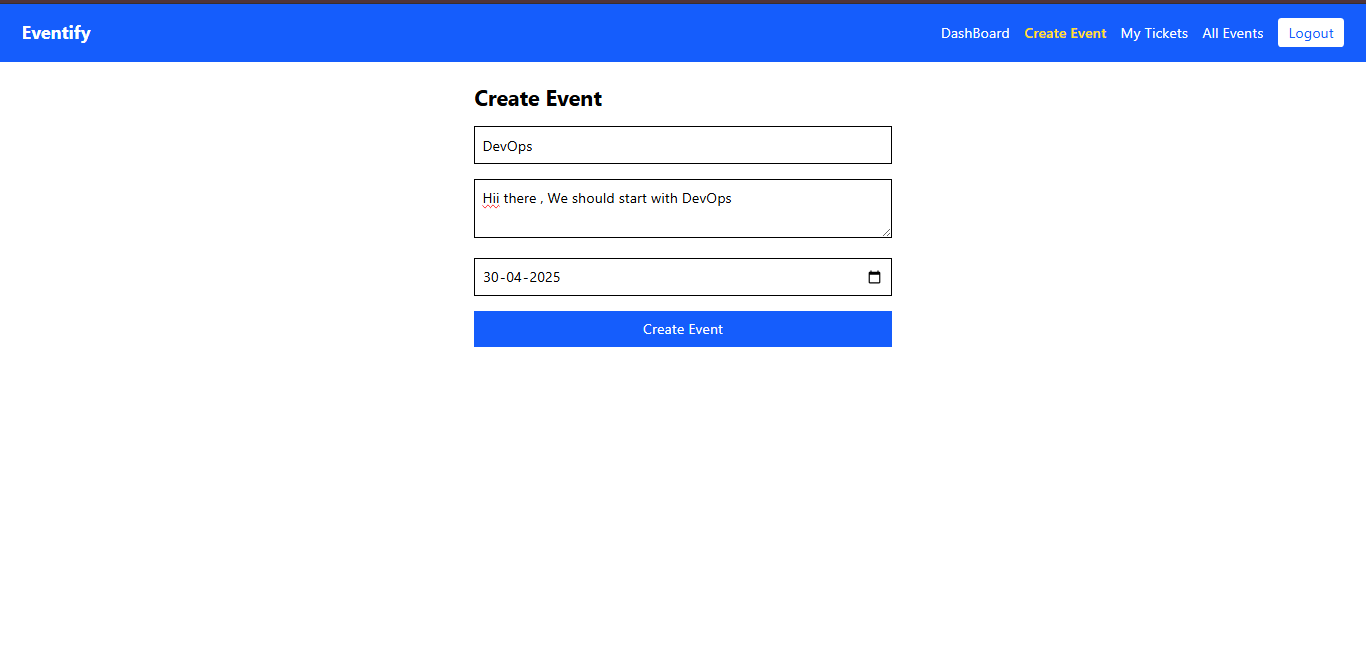
1. User Registration / Login

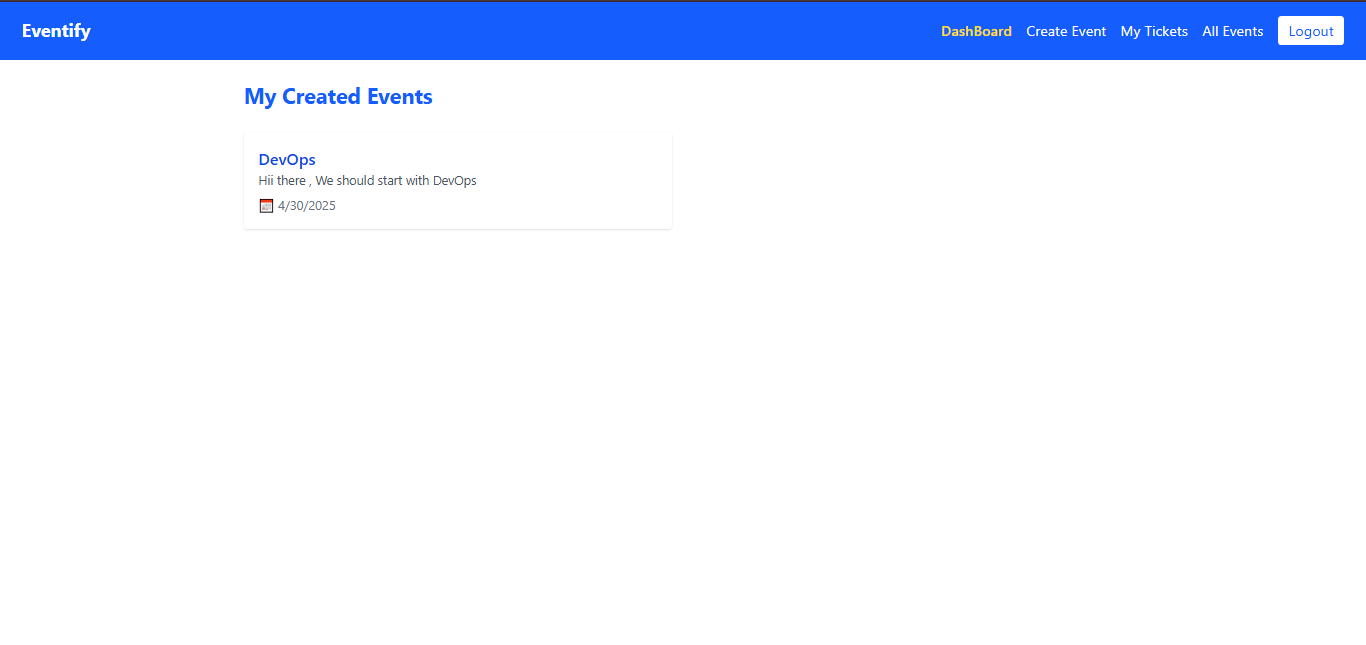
* User registers or logs in.
* JWT token is received and stored in localStorage.
* Token is attached in headers for all protected API calls.



1. Create an Event

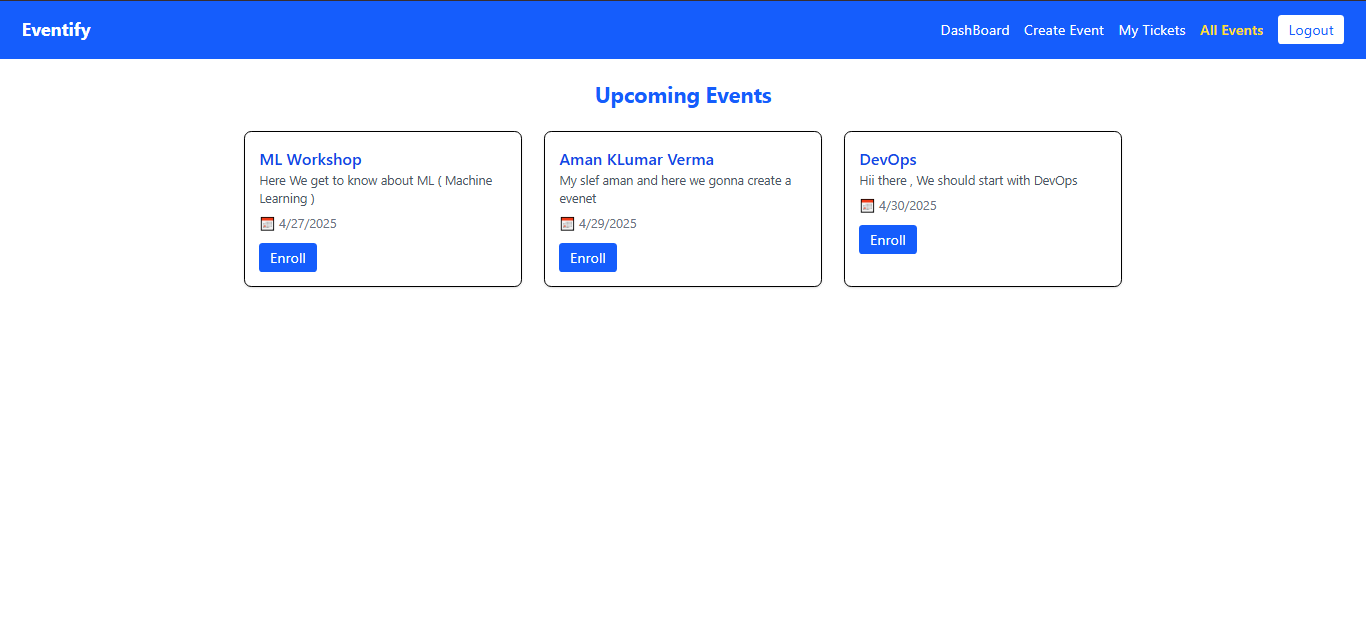
* Logged-in user navigates to the Create Event page.
* Fills in event details: Title, Description, Date.
* API Call: POST /api/events (token required).
* Event is saved in MongoDB with reference to the creator.





1. View All Events

* Home page (All Events page) fetches events.
* API Call: GET /api/events/all
* Events are displayed with Enroll buttons.

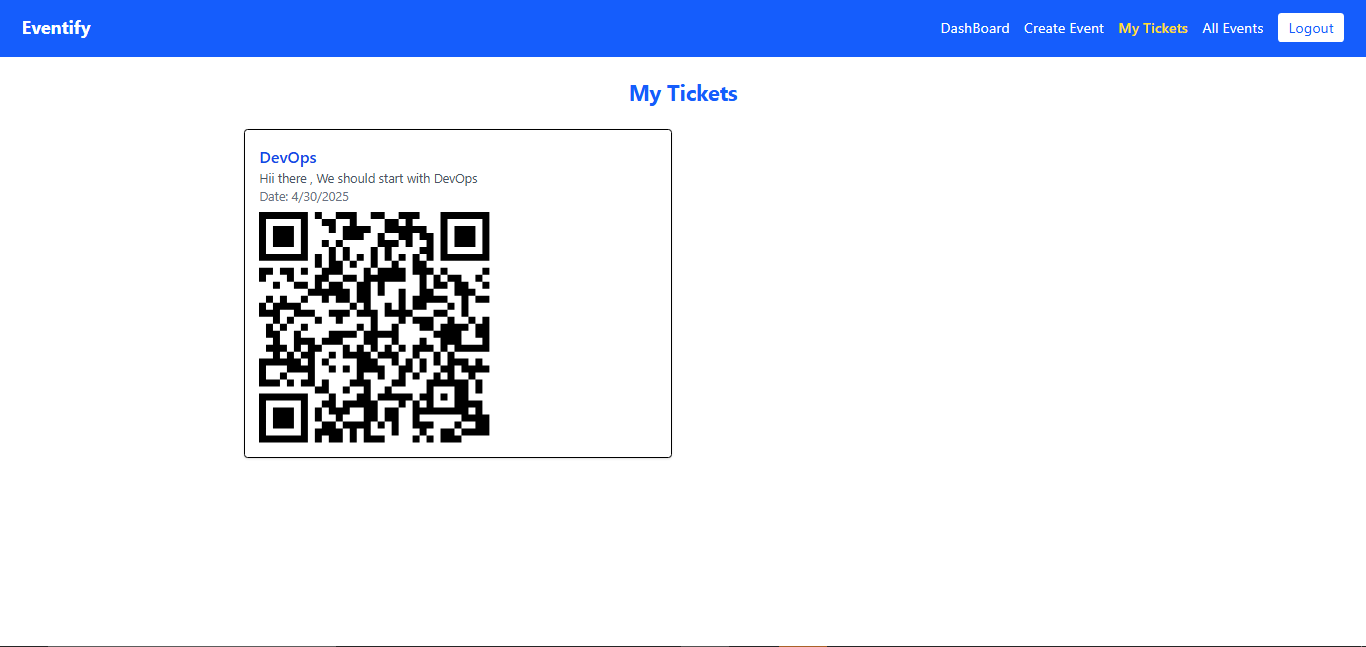


1. Enroll in an Event (Ticket Generation)

* User clicks Enroll on an event.
* API Call: POST /api/tickets/generate/:eventId (token required).
* Server checks for duplicate tickets.
* New ticket created with event, user, and unique ticketId.

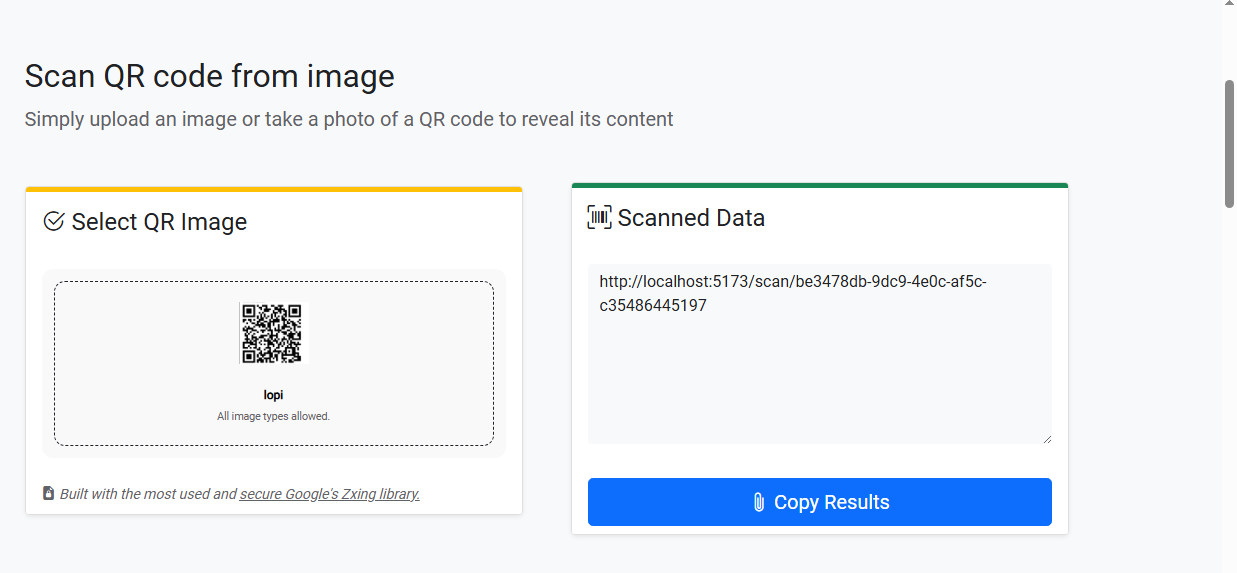
1. Display Ticket with QR Code

* User navigates to My Tickets page.
* API Call: GET /api/tickets/my (token required).
* Tickets are shown with QR codes containing ticketId.



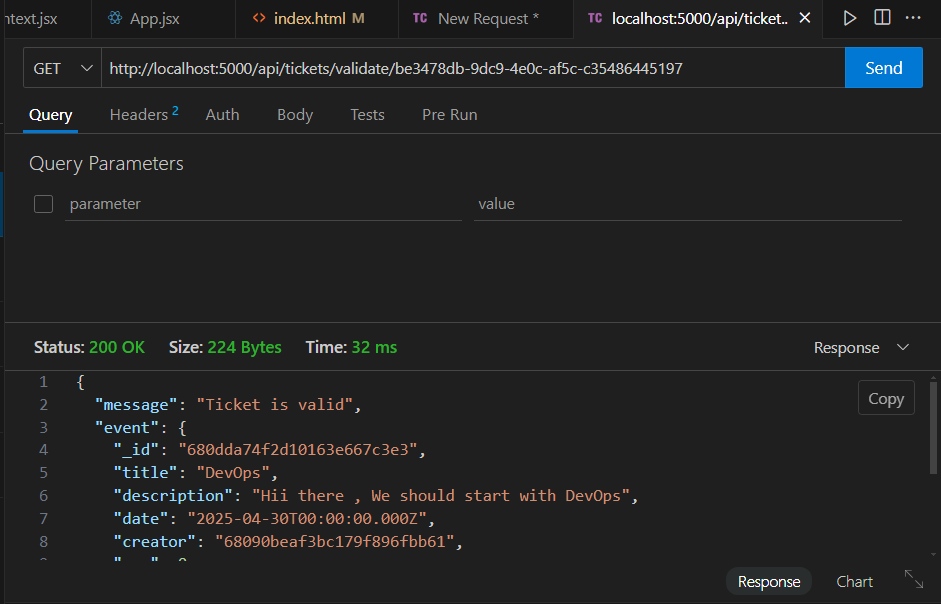
1. Scan QR Code

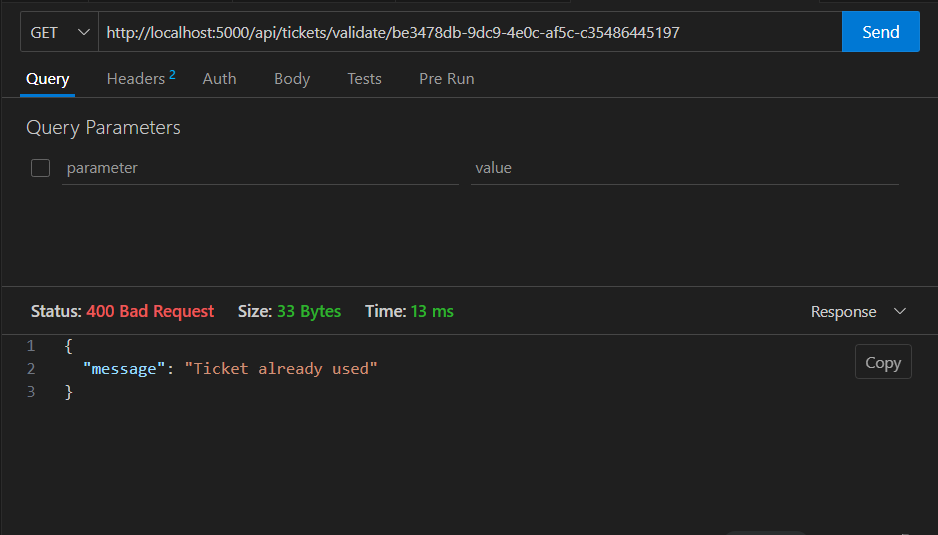
* At the event gate, QR scanner scans the QR code.
* Scanner extracts the ticketId as plain text.



1. Verify the Ticket (Validation)

* Open browser and call: <http://localhost:5000/api/tickets/validate/:ticketId>
* Server validates ticket existence and usage.
* If valid: “Ticket is valid” + event info.
* If already used or invalid: appropriate error message.





1. User Successfully Verified

* Staff sees "Ticket Verified" or "Ticket Already Used" message.
* User entry is allowed or denied accordingly.

Quick Flow Table

| **Step No.** | **Action** | **API Endpoint** |
| --- | --- | --- |
| 1 | User Login/Register | /api/auth/login or /api/auth/register |
| 2 | Create Event | POST /api/events |
| 3 | View All Events | GET /api/events/all |
| 4 | Enroll in Event (Generate Ticket) | POST /api/tickets/generate/:eventId |
| 5 | View My Tickets | GET /api/tickets/my |
| 6 | Scan Ticket QR | Read ticketId from QR code |
| 7 | Validate Ticket | GET /api/tickets/validate/:ticketId |

1. Tech Stack

* MongoDB (Database)
* Express.js (Backend Server)
* React.js (Frontend)
* Node.js (Runtime Environment)
* Tailwind CSS (Styling)
* JWT (Authentication)
* UUID (Unique Ticket IDs)