For your **Alumni Association Platform**, here’s a structured API plan based on common features such a platform might need:

## ****1. Authentication APIs (User Management)****

* POST /auth/register → Register a new user (alumni, admin, faculty, etc.)
* POST /auth/login → Authenticate user and return JWT token
* POST /auth/logout → Log out the user
* POST /auth/refresh-token → Refresh authentication token
* GET /auth/profile → Get the logged-in user’s profile
* PUT /auth/profile → Update the logged-in user’s profile
* PUT /auth/change-password → Change password

## ****2. Alumni Management APIs****

* GET /alumni → Get a list of registered alumni
* GET /alumni/:id → Get alumni details by ID
* PUT /alumni/:id → Update alumni details
* DELETE /alumni/:id → Remove an alumni profile (Admin only)
* GET /alumni/search?q=name → Search alumni by name, batch, or department

## ****3. Events Management APIs****

* POST /events → Create a new event (Admin only)
* GET /events → Get all events
* GET /events/:id → Get event details
* PUT /events/:id → Update event details (Admin only)
* DELETE /events/:id → Delete an event (Admin only)
* POST /events/:id/register → Alumni register for an event

## ****4. Posts & Announcements APIs****

* POST /posts → Create a new post/announcement
* GET /posts → Get all posts
* GET /posts/:id → Get a single post
* PUT /posts/:id → Edit a post
* DELETE /posts/:id → Delete a post
* POST /posts/:id/comment → Add a comment on a post

## ****5. Job & Internship Board APIs****

* POST /jobs → Post a job/internship opportunity
* GET /jobs → Get all jobs
* GET /jobs/:id → Get job details
* PUT /jobs/:id → Edit a job (Admin/Job Poster)
* DELETE /jobs/:id → Delete a job

## ****6. Mentorship & Networking APIs****

* POST /mentorship → Alumni can offer mentorship
* GET /mentorship → Get available mentors
* POST /mentorship/request → Request mentorship
* GET /mentorship/:id → Get mentor profile

## ****7. Donations & Fundraising APIs (If applicable)****

* POST /donations → Make a donation
* GET /donations → Get donation records (Admin only)
* GET /donations/:id → Get donation details
* POST /fundraisers → Create a new fundraising campaign
* GET /fundraisers → Get all fundraisers

## ****8. Messaging & Notifications APIs****

* POST /messages → Send a message to another user
* GET /messages/:userId → Get messages with a specific user
* GET /notifications → Get user notifications

### ****Backend Considerations****

* Use **JWT for authentication**
* Implement **role-based access control (RBAC)** for admins and alumni
* Use **pagination** for large datasets (alumni list, posts, events)

Would you like me to help with API structuring using **Node.js (Express) or Django REST Framework**?