

Versioned REST API Documentation (v1)

All endpoints are under the base path /api/v1 and use JSON over HTTPS. They require an Authorization: Bearer <token> header.

Common Response Models

Success (2xx)

```
{  
  "status": "success",  
  "data": { /* resource-specific payload */ }  
}
```

Error (4xx/5xx)

```
{  
  "status": "error",  
  "error": {  
    "code": "string", // e.g. "BadRequest", "NotFound"  
    "message": "string" // human-readable description  
  }  
}
```

1. Student Management

Create Student

- **POST** /api/v1/students
- **Headers:** Authorization, Content-Type: application/json
- **Request Body:**
 - {
 - "name": "string",
 - "email": "string",
 - "skills": ["string"],
 - "careerInterests": ["string"],
 - "discordId": "string"
 - }
- **Responses:**
 - 201 Created
 - {
 - "status": "success",
 - "data": {
 - "studentId": "string"
 - }
 - }
 - 400 Bad Request if validation fails

Retrieve Student

- **GET** /api/v1/students/{studentId}
- **Responses:**
 - 200 OK
 - {
 - "status": "success",
 - "data": {
 - "studentId": "string",
 - "name": "string",
 - "email": "string",
 - "skills": ["string"],
 - "careerInterests": ["string"],
 - "discordId": "string",
 - "cvUrl": "string"
 - }
 - }
 - }
- 404 Not Found

Update Student

- **PUT** /api/v1/students/{studentId}
- **Request Body:** (any subset of updatable fields)
 - {
 - "name": "string",
 - "skills": ["string"]
 - }
- **Responses:**
 - 200 OK (returns full student object)
 - 400 Bad Request, 404 Not Found

Delete Student

- **DELETE** /api/v1/students/{studentId}
- **Responses:**
 - 204 No Content
 - 404 Not Found

2. CV Upload & Management

Upload CV

- **POST** /api/v1/students/{studentId}/cv
- **Headers:** Content-Type: multipart/form-data
- **Form Data:** file: PDF or DOCX
- **Responses:**
 - 200 OK { "status": "success", "data": { "cvUrl": "string" } }
 - 400 Bad Request, 404 Not Found

Delete CV

- **DELETE** /api/v1/students/{studentId}/cv
- **Responses:**
 - 204 No Content
 - 404 Not Found

3. Opportunity Management

List Opportunities

- **GET** /api/v1/opportunities
- **Query Parameters (optional):** location, techStack, companyName
- **Responses:**
 - 200 OK (array of opportunity objects)

Create Opportunity

- **POST** /api/v1/opportunities
- **Request Body:**

```
{  
    "title": "string",  
    "companyName": "string",  
    "location": "string",  
    "duration": "string",  
    "techStack": ["string"],  
    "description": "string"  
}
```
- **Responses:**
 - 201 Created { "status": "success", "data": { "opportunityId": "string" } }
 - 400 Bad Request

Retrieve Opportunity

- **GET** /api/v1/opportunities/{opportunityId}
- **Responses:**
 - 200 OK
 - 404 Not Found

Update Opportunity

- **PUT** /api/v1/opportunities/{opportunityId}
- **Request Body:** (partial or full)

```
{ /* any updatable fields */ }
```
- **Responses:**
 - 200 OK (returns updated opportunity)
 - 400 Bad Request, 404 Not Found

Delete Opportunity

- **DELETE** /api/v1/opportunities/{opportunityId}
- **Responses:**
 - 204 No Content
 - 404 Not Found

4. Matching & Recommendations

Trigger Matching

- **POST** /api/v1/students/{studentId}/matches
- **Responses:**
 - 202 Accepted (processing)
 - 404 Not Found

Get Recommendations

- **GET** /api/v1/students/{studentId}/recommendations
- **Responses:**
 - 200 OK (array of opportunity summaries)
 - 404 Not Found

5. Interaction & Interest Logging

Log Interest

- **POST** /api/v1/students/{studentId}/interests
- **Request Body:**
 - {
 - "opportunityId": "string",
 - "action": "apply" | "bookmark"
 - }
- **Responses:**
 - 200 OK
 - 400 Bad Request, 404 Not Found

6. Notifications

Send Notification

- **POST** /api/v1/notifications
- **Request Body:**

- {
- "studentId": "string",
- "type": "info" | "warning" | "alert",
- "message": "string"
- }
- **Responses:**
 - 202 Accepted
 - 400 Bad Request

Retrieve Notifications

- **GET** /api/v1/students/{studentId}/notifications
- **Responses:**
 - 200 OK (array of notification objects)