

“PH University”
Requirement Analysis - MVP Project
MERN Stack MVP Web Application

1) Functional Requirements:

a) Authentication:

i) Student:

- (1) Can login and logout securely.
- (2) Can update their password.

ii) Faculty:

- (1) Can login and logout securely.
- (2) Can update their password.

iii) Admin:

- (1) Can login and logout securely.
- (2) Can update their password.

b) Profile Management:

i) Student:

- (1) Can manage and update their profile information.
- (2) Can update certain fields only.

ii) Faculty:

- (1) Can manage and update their profile information.
- (2) Can update certain fields only.

iii) Admin:

- (1) Can manage and update their profile information.
- (2) Can update certain fields only.

c) Academic Procedures:

i) Student:

- (1) Can enroll in offered courses for a specific semester.
- (2) Can view their class schedule.
- (3) Can view their grades and results.
- (4) Can view notice boards and events.

ii) Faculty:

- (1) Can manage students' grades.
- (2) Can access students' personal and academic information.

iii) Admin:

- (1) Can manage multiple processes:
 - (a) Semesters;
 - (b) Courses;
 - (c) Offered Courses;
 - (d) Sections;
 - (e) Rooms;
 - (f) Buildings;

d) User Management:

i) Admin:

- (1) Can manage multiple types of users.
- (2) Can block/unblock users.
- (3) Can change users' passwords.

2) Data Model:

a) Academic Semester:

- i) _id
- ii) name
- iii) year
- iv) code
- v) startMonth
- vi) endMonth
- vii) createdAt
- viii) updatedAt

b) Academic Faculty:

- i) _id
- ii) name
- iii) createdAt
- iv) updatedAt

c) Academic Department:

- i) _id
- ii) name
- iii) academicFaculty
- iv) createdAt
- v) updatedAt

d) User:

- i) _id
- ii) id (generated)
- iii) password
- iv) needsPasswordChange
- v) role
- vi) status
- vii) createdAt
- viii) deletedAt
- ix) updatedAt

e) Student:

- i) _id
- ii) id (generated)
- iii) user
- iv) name
 - (1) firstName
 - (2) middleName
 - (3) lastName
- v) gender
- vi) dateOfBirth
- vii) email
- viii) contactNo
- ix) emergencyContactNo
- x) bloodGroup
- xi) presentAddress
- xii) permanentAddress
- xiii) guardian
 - (1) father
 - (a) name
 - (i) firstName
 - (ii) middleName

- (iii) lastName
 - (b) occupation
 - (c) contactNo
- (2) mother
 - (a) name
 - (i) firstName
 - (ii) middleName
 - (iii) lastName
 - (b) occupation
 - (c) contactNo
- xiv) localGuardian
 - (1) name
 - (a) firstName
 - (b) middleName
 - (c) lastName
 - (2) occupation
 - (3) contact No
 - (4) address
 - (5) relationship
- xv) profileImage
- xvi) admissionSemester
- xvii) nationality
- xviii) Religion
- xix) isDeleted

f) Faculty:

- i) _id
- ii) id (generated)
- iii) user
- iv) designation
- v) name
 - (1) firstName
 - (2) middleName
 - (3) lastName
- vi) gender
- vii) dateOfBirth
- viii) email
- ix) contactNo
- x) emergencyContactNo
- xi) presentAddress
- xii) permanentAddress
- xiii) profileImage
- xiv) academicDepartment
- xv) academicFaculty
- xvi) nationality
- xvii) religion
- xviii) isDeleted

g) Admin:

- i) _id
- ii) id (generated)
- iii) user
- iv) designation
- v) name
 - (1) firstName
 - (2) middleName
 - (3) lastName
- vi) gender

- vii) dateOfBirth
- viii) email
- ix) contactNo
- x) emergencyContactNo
- xi) presentAddress
- xii) permanentAddress
- xiii) profileImage
- xiv) managementDepartment
- xv) nationality
- xvi) religion
- xvii) isDeleted

3) API Endpoints and CRUD Operation Methods:

a) User:

i) Create different types of users:

- (1) users/create-student (POST)
- (2) users/create-faculty (POST)
- (3) users/create-admin (POST)

b) Student:

- i) **Get all students:** students (GET)
- ii) **Get a specific student:** student/:id (GET)
- iii) **Update a specific student:** student/:id (PATCH)
- iv) **Delete a specific student:** student/:id (DELETE)
- v) **Get students' profile:** students/my-profile (GET)

c) Faculty:

- i) **Get all faculties:** faculties (GET)
- ii) **Get a specific faculty:** faculty/:id (GET)
- iii) **Update a specific faculty:** faculty/:id (PATCH)
- iv) **Delete a specific faculty:** faculty/:id (DELETE)
- v) **Get faculties' profile:** faculties/my-profile (GET)

d) Admin:

- i) **Get all admins:** admins (GET)
- ii) **Get a specific admin:** admin/:id (GET)
- iii) **Update a specific admin:** admin/:id (PATCH)
- iv) **Delete a specific admin:** admin/:id (DELETE)
- v) **Get admins' profile:** admins/my-profile (GET)

e) Auth:

- i) **Login:** auth/login
- ii) **Register:** auth/register
- iii) **Refresh Token:** auth/refresh-token
- iv) **Change Password:** auth/change-password
- v) **Forgot Password:** auth/forgot-password
- vi) **Reset Password:** auth/reset-password