# ***“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) User:

| | | | | |
|---|---|---|---|---|
| _id | id (generated) | password | needs Password Change | role |
| status | isDeleted | createdAt | updatedAt | |

### b) Student:

| | | | | |
|---|---|---|---|---|
| _id | id (generated) | name | gender | dateOfBirth |
| email | contactNo | emergency ContactNo | present Address | permanent Address |
| guardian | localGuardian | profileImage | academic Department | nationality |
| religion | isDeleted | createdAt | updatedAt | |

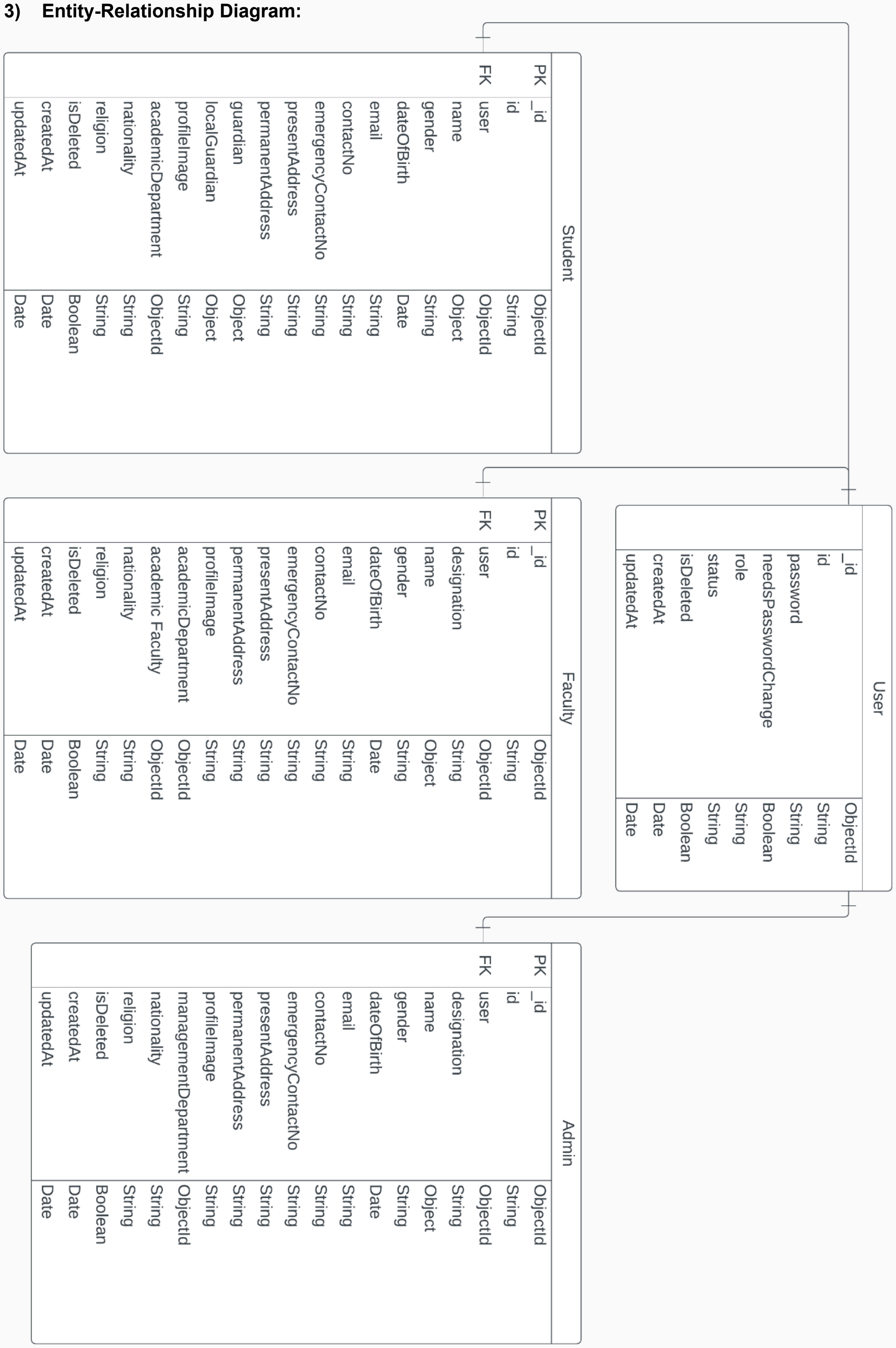
### c) Faculty:

| | | | | |
|---|---|---|---|---|
| _id | id (generated) | designation | name | gender |
| dateOfBirth | email | contactNo | emergency ContactNo | present Address |
| permanent Address | profileImage | academic Department | academic Faculty | nationality |
| religion | isDeleted | createdAt | updatedAt | |

### d) Admin:

| | | | | |
|---|---|---|---|---|
| _id | id (generated) | designation | name | gender |
| dateOfBirth | email | contactNo | emergency ContactNo | present Address |
| permanent Address | profileImage | management Department | nationality | religion |
| isDeleted | createdAt | updatedAt | | |

### 3) Entity-Relationship Diagram:



Student

| | | |
|---|---|---|
| PK | _id | ObjectId |
| | id | String |
| FK | user | ObjectId |
| | name | Object |
| | gender | String |
| | dateOfBirth | Date |
| | email | String |
| | contactNo | String |
| | emergencyContactNo | String |
| | presentAddress | String |
| | permanentAddress | String |
| | guardian | Object |
| | localGuardian | Object |
| | profileImage | String |
| | academicDepartment | ObjectId |
| | nationality | String |
| | religion | String |
| | isDeleted | Boolean |
| | createdAt | Date |
| | updatedAt | Date |

Faculty

| | | |
|---|---|---|
| PK | _id | ObjectId |
| | id | String |
| FK | user | ObjectId |
| | designation | String |
| | name | Object |
| | gender | String |
| | dateOfBirth | Date |
| | email | String |
| | contactNo | String |
| | emergencyContactNo | String |
| | presentAddress | String |
| | permanentAddress | String |
| | profileImage | String |
| | academicDepartment | ObjectId |
| | academic Faculty | ObjectId |
| | nationality | String |
| | religion | String |
| | isDeleted | Boolean |
| | createdAt | Date |
| | updatedAt | Date |

User

| | |
|---|---|
| _id | ObjectId |
| id | String |
| password | String |
| needsPasswordChange | Boolean |
| role | String |
| status | String |
| isDeleted | Boolean |
| createdAt | Date |
| updatedAt | Date |

Admin

| | | |
|---|---|---|
| PK | _id | ObjectId |
| | id | String |
| FK | user | ObjectId |
| | designation | String |
| | name | Object |
| | gender | String |
| | dateOfBirth | Date |
| | email | String |
| | contactNo | String |
| | emergencyContactNo | String |
| | presentAddress | String |
| | permanentAddress | String |
| | profileImage | String |
| | managementDepartment | ObjectId |
| | nationality | String |
| | religion | String |
| | isDeleted | Boolean |
| | createdAt | Date |
| | updatedAt | Date |

4) **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