***“UniMate University”***

***Requirement Analysis - MVP Project***

**MERN Stack MVP Web Application**

1. **Functional Requirements:**
   1. **Authentication:** 
      1. **Student:**
         1. Can log in and log out securely.
         2. Can update their password.
      2. **Faculty:**
         1. Can log in and log out securely.
         2. Can update their password.
      3. **Admin:**
         1. Can log in and log out securely.
         2. Can update their password.
   2. **Profile Management:**
      1. **Student:**
         1. Can manage and update their profile information.
         2. Can update certain fields only.
      2. **Faculty:**
         1. Can manage and update their profile information.
         2. Can update certain fields only.
      3. **Admin:**
         1. Can manage and update their profile information.
         2. Can update certain fields only.
   3. **Academic Procedures:**
      1. **Student:** 
         1. Can enrol in the 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.
      2. **Faculty:**
         1. Can manage students’ grades.
         2. Can access students’ personal and academic information.
      3. **Admin:**
         1. **Can manage multiple processes:**
            1. Semesters;
            2. Courses;
            3. Offered Courses;
            4. Sections;
            5. Rooms;
            6. Buildings;
   4. **User Management:**
      1. **Admin:**
         1. Can manage multiple types of users.
         2. Can block/unblock users.
         3. Can change users’ passwords.
2. **Data Model:**

| **Academic Faculty** | | **Academic Department** | |
| --- | --- | --- | --- |
| * \_id * name * createdAt * updatedAt | | * \_id * name * academicFaculty * createdAt * updatedAt | |
| **Management Department** | | **Course** | |
| * \_id * name * createdAt * updatedAt | | * \_id * title * prefix * code * credits * prerequisiteCourse:   + course   + isDeleted * isDeleted * createdAt * updatedAt | |
| **Academic Semester** | | **Semester Registration** | |
| * \_id * name * year * code * startMonth * endMonth * createdAt * updatedAt | | **Client Model** | **Database Model** |
| * academicSemester * startDay * endDay * startTime * endTime * minCredit * maxCredit | * \_id * academicSemester * status * startDate * endDate * minCredit * maxCredit * createdAt * updatedAt |
| **Offered Course** | | **User** | |
| **Client Model** | **Database Model** | * \_id * id (generated) * password * needsPasswordChange * role * status * createdAt * updatedAt * isDeleted | |
| * \_id * semesterRegistration * academicSemester * academicFaculty * academicDepartment * course * faculty * maxCapacity * remainingCapacity * section * days * startTime * endTime * createdAt * updatedAt | * semesterRegistration * academicDepartment * course * faculty * maxCapacity * remainingCapacity * section * days * timeSlot |

| **Student** | **Faculty** |
| --- | --- |
| * \_id * id (generated) * user * name   + firstName   + middleName   + lastName * gender * dateOfBirth * email * contactNo * emergencyContactNo * bloodGroup * presentAddress * permanentAddress * guardian   + father     - name       * firstName       * middleName       * lastName     - occupation     - contactNo     - email   + mother     - name       * firstName       * middleName       * lastName     - occupation     - contactNo     - email * localGuardian   + name     - firstName     - middleName     - lastName   + occupation   + contact No   + email   + address   + relationship * profileImage * admissionSemester * nationality * Religion * isDeleted | * \_id * id (generated) * user * designation * name   + firstName   + middleName   + lastName * gender * dateOfBirth * email * contactNo * emergencyContactNo * bloodGroup * presentAddress * permanentAddress * profileImage * academicDepartment * joiningDate * nationality * religion * isDeleted |
| **Admin** |
| * \_id * id (generated) * user * designation * name   + firstName   + middleName   + lastName * gender * dateOfBirth * email * contactNo * emergencyContactNo * bloodGroup * presentAddress * permanentAddress * profileImage * managementDepartment * joiningDate * nationality * religion * isDeleted |

1. **API Endpoints and CRUD Operation Methods:**Most **GET** endpoints (especially those returning lists like students, faculties, departments, courses, etc.) support:  
     
    • **searchTerm=<keyword>:** for partial name/code/id searches  
    • **limit=<number>:** to limit results per page  
    • **page=<keyword>**: to paginate results  
    • **sort=<field>**: for sorting  
    • **Field-based filters:** (e.g., gender=male&bloodGroup=B+)
   1. **Academic Faculty:**
      1. **Create an academic faculty:** academic-faculties/create-academic-faculty **(POST)**
      2. **Get all academic faculties:** academic-faculties **(GET)**
      3. **Get a specific academic faculty:** academic-faculties/:id **(GET)**
      4. **Update a specific academic faculty:** academic-faculties/:id/: **(PATCH)**
   2. **Academic Department:**
      1. **Create an academic department:** academic-departments/create-academic-department **(POST)**
      2. **Get all academic departments:** academic-departments **(GET)**
      3. **Get a specific academic department:** academic-departments/:id **(GET)**
      4. **Update a specific academic department:** academic-departments/:id **(PATCH)**
   3. **Management Department:**
      1. **Create a management department:** management-departments/create-management-department **(POST)**
      2. **Get all management departments:** management-departments **(GET)**
      3. **Get a specific management department:** management-departments/:id **(GET)**
      4. **Update a specific management department:**management-departments/:id **(PATCH)**
   4. **Course:**
      1. **Create a course:** courses/create-course**(POST)**
      2. **Get all courses:** courses**(GET)**
      3. **Get a specific course:** courses/:id **(GET)**
      4. **Update a specific course:** courses/:id **(PATCH)**
      5. **Delete a course:** courses/:id **(DELETE)**
      6. **Get all assigned faculties of a course:** courses/faculties/:id **(GET)**
      7. **Assign faculties to a course:** courses/:id/assign-faculties **(PUT)**
      8. **Remove faculties from a course:** courses/:id/remove-faculties **(DELETE)**
   5. **Academic Semester:**
      1. **Create an academic semester:**academic-semesters/create-academic-semester **(POST)**
      2. **Get all academic semesters:** academic-semesters **(GET)**
      3. **Get a specific academic semester:** academic-semesters/:id **(GET)**
      4. **Update a specific academic semester:** academic-semesters/:id **(PATCH)**
   6. **Semester Registration:**
      1. **Create a semester registration:**semester-registrations/create-semester-registration **(POST)**
      2. **Get all semester registrations:** semester-registrations **(GET)**
      3. **Get a specific semester registration:** semester-registrations/:id **(GET)**
      4. **Update a specific semester registration:** semester-registrations/:id **(PATCH)**
      5. **Update all semester registration statuses:**semester-registrations/update/status **(PATCH)**
   7. **Offered Course:**
      1. **Create an offered course:**offered-courses/create-offered-course **(POST)**
      2. **Get all offered courses:** offered-courses **(GET)**
      3. **Get a specific offered course:** offered-courses/:id **(GET)**
      4. **Update a specific offered course:** offered-courses/:id **(PATCH)**
   8. **User:**
      1. **Create different types of users:**
         1. **Student:** users/create-student **(POST)**
         2. **Faculty:** users/create-faculty **(POST)**
         3. **Admin:** users/create-admin **(POST)**
   9. **Student:**
      1. **Get all students:** students **(GET)**
      2. **Get a specific student:** students/:id **(GET)**
      3. **Update a specific student:** students/:id **(PATCH)**
      4. **Delete a specific student:** students/:id **(DELETE)**
      5. **Get students’ profile:** students/my-profile **(GET)**
   10. **Faculty:**
       1. **Get all faculties:** faculties **(GET)**
       2. **Get a specific faculty:** faculties/:id **(GET)**
       3. **Update a specific faculty:** faculties/:id **(PATCH)**
       4. **Delete a specific faculty:** faculties/:id **(DELETE)**
       5. **Get faculties’** **profile:** faculties/my-profile **(GET)**
   11. **Admin:**
       1. **Get all admins:** admins **(GET)**
       2. **Get a specific admin:** admins/:id **(GET)**
       3. **Update a specific admin:** admins/:id **(PATCH)**
       4. **Delete a specific admin:** admins/:id **(DELETE)**
       5. **Get admins’ profile:** admins/my-profile **(GET)**
   12. **Auth:**
       1. **Login:** auth/login
       2. **Register:** auth/register
       3. **Refresh Token:** auth/refresh-token
       4. **Change Password:** auth/change-password
       5. **Forgot Password:** auth/forgot-password
       6. **Reset Password:** auth/reset-password