**University Management Project**

We are building a university management system where, we have **three** typesroles such as, Admin, Student, Faculty.

***Functional requirements:***

* Student:
  + Student can login and logout.
  + Student can maintain their own profile (manage, update).
  + Student can update certain fields in his/her account.
  + Student can enroll in a semester.
  + Student can enroll in offered courses for a specific semester.
  + Student can pay their tuition fees through offline/online (partial or full payment).
  + Student can see their transaction history.
  + Student can see their class routines.
  + Student can see their notice board and events
  + Student can see their result (Full or semester-wise).
  + Student can evaluate their teachers.
* Admin:
  + Admin can login and logout.
  + Admin can manage update their profile.
  + Admin can only update some certain fields.
  + Admin can manage (**block/unblock, change password, force logout,** ) user (student, teacher etc.) account.
  + Admin can manage multiple (**semester, offered courses, section, faculty, student, building, payment, activity**) procedure.
* Factulty:
  + Faculty can login and logout.
  + Faculty can manage update their profile.
  + Faculty can only update some certain fields.
  + Faculty can manage user (student, teacher etc.) account.
  + Faculty can manage student grade.
  + Faculty can have access to personal information.
  + Faculty can manage their lectures / lecture resources.

Authentication Application

Permission:

1. title

UserPermission:

1. permissionID
2. UserID (studentID/facultyID/adminID)

User:

1. \_id
2. Role
3. Password
4. Created date
5. Updated date
6. StudentID/adminID/facultyID (\_id reference for the user)

Student:

1. \_id
2. Name
   1. First name
   2. Middle name
   3. Last name
3. Gender
4. Birthdate
5. Guardian
6. Contact No.
7. Email
8. Emergency contact
9. Address
   1. Present address
   2. Permanent address
10. Department
11. Subject

Admin:

1. \_id
2. Name
   1. First name
   2. Middle name
   3. Last name
3. Gender
4. Birthdate
5. Contact No.
6. Emergency contact
7. Department
8. Email
9. Designation

Faculty:

1. \_id
2. Name
   1. First name
   2. Middle name
   3. Last name
3. Contact no
4. Emergency contact no
5. Faculty
6. Designation
7. Gender
8. Birthdate
9. Email

API Endpoints

* User:
  + users/create-students (POST)
  + users/create-faculty (POST)
  + users/:id (GET)
  + users/:id (PATCH)
  + users/create-admin (POST)
  + users/my-profile (GET)
  + users/:id/assign-permissions (POST)
  + users/:id/remove-permissions (POST)
  + users?page=1?limit=10 (GET)
  + users/:id/force-logout
  + users/:id/available-permissions?page=1?limit=10 (GET)
  + users/:id/assigned-permissions?page=1?limit=10 (GET)
  + users/:id (DELETE)
* Students:
  + students/:id/update (PATCH)
  + students/:id/profile (GET)
  + students?page=1?limit=10 (GET)
* Faculty:
  + faculties/:id/update (PATCH)
  + faculties/:id/profile (GET)
  + faculties?page=1?limit=10 (GET)
* Admin:
  + admins/:id/update (PATCH)
  + admins/:id/profile (GET)
  + admins?page=1?limit=10 (GET)
* Permission:
  + permissions (POST)
  + permissions?page=1?limit=10 (GET)
  + permissions/:id (GET)
  + permissions/:id (PATCH)
  + permissions/:id (DELETE)
* Auth:
  + auth/login (POST)
  + auth/refresh-token (POST)
  + auth/change-password (POST)
  + auth/forgot-password (POST)
  + auth/reset-password (POST)

**Plans:**

1. project setup, .gitignore
2. eslint
3. prettier
4. husky and lint stage