STUDENT

```
body: {
   studentID: String, Required
   name: String
   photo: File - jpg/jpeg/png (input name must be “student_photo”)
   gender: String
   religion: String
   fatherName: String
   motherName: String
   guardianName: String
   guardianPhone: String
   bloodGroup: String
   birth_date: Date
   admission_date: String
   batch: String
   session: String
   semester: String
   group: String
   nid: String
   permanentAddress: String
   presentAddress: String
   email: String
   phone: String
   website: String
   trainingExprience: String
   teachingArea: String
   password: String, default: "password",
   cv: File – pdf (input name must be “student_cv”)
   activeStatus: Boolean, default: true, (Only can change SUPER-ADMIN)
}
```

TEACHER

```
body: {
   teacherID: String, Required
   name: String
   photo: File - jpg/jpeg/png (input name must be “student_photo”)
   gender: String
   religion: String
   bloodGroup: String
   birth_date: Date
   designation: String
   nid: String
   faculty: String
   address: String
   email: String
   phone: String
   website: String
   academicQualification: String
   trainingExprience: String
   teachingArea: String
   password: String, default: "password"
   activeStatus: Boolean, default: true, (Only can change SUPER-ADMIN)
}
```

EVENTS

```
body: {
   title: String, Required
   description: String
   photo: File - jpg/jpeg/png (input name must be “student_photo”)
   place: String
   startTime: String
   endTime: String
   status: ["active", "inactive"], default: "inactive"
   eventCreator: id of who created the event.
 }
```

TEACHER

```
body: {
   adminID: String, Required
   name: String
   photo: File - jpg/jpeg/png (input name must be “student_photo”)
   email: String
   phone: String
   password: String, default: "password"
}
```

STUDENT API

Route	Data	Method
/student/create	Create student	POST
/student/all	Get all student	GET
/student/find/_id	Get specific student	GET
/student/delete/_id	Delete student	DELETE
/student/update/_id	Update student data	PUT

Note: Send the token with req.headers.authorization

authorization : eyJhbGci5.eyJzdWIiOiIxMjM0DIyfQ.SflKxweJf36POk6yJV

TEACHER API

Route	Data	Method
/teacher/create	Create teacher	POST
/teacher/all	Get all teacher	GET
/teacher/find/_id	Get specific teacher	GET
/teacher/delete/_id	Delete teacher	DELETE
/teacher/update/_id	Update teacher data	PUT

Note: Send the token with req.headers.authorization

EVENT API

Route	Data	Method
/event/create	Create event	POST
/event/all	Get all event	GET
/ event/find/_id	Get specific event	GET
/ event/delete/_id	Delete event	DELETE
/event/update/_id	Update event data	PUT

Note: Send the token with req.headers.authorization

SUPER-ADMIN API

Route	Data	Method
/admin/create-admin	Create admin	POST
/admin/update-admin	Update admin data	PUT

Note: Send the token with req.headers.authorization

AUTHENTICATION API

Route	Field	Method
/auth/login	studentID/teacherID/adminID, password	POST

Note: Send the token with req.headers.authorization