Software Requirements Specification (SRS) for a School Management System. An SRS is a document that outlines the requirements, features, and functionalities of a software system. Here's a template for the SRS of a School Management System:

1. **Introduction**
   * Purpose of the document
   * Scope of the system
   * Definitions, acronyms, and abbreviations
2. **Overall Description**
   * Product perspective
   * System interfaces
   * User characteristics
   * Constraints
   * Assumptions and dependencies
3. **Specific Requirements**
   * **Functional Requirements**
     + Student Management
       - Student registration
       - Student records management
       - Attendance tracking
       - Grade management
     + Teacher Management
       - Teacher registration
       - Class assignments
       - Attendance tracking
       - Grade management
     + Course Management
       - Course creation and management
       - Curriculum management
       - Scheduling
     + Fee Management
       - Fee structure management
       - Fee payment tracking
     + Library Management
       - Book management
       - Book issue/return
     + Examination Management
       - Exam scheduling
       - Result management
     + Reports and Analytics
       - Student reports
       - Teacher reports
       - Attendance reports
       - Grade reports
     + User Management
       - User roles and permissions
       - User authentication and authorization
   * **Non-functional Requirements**
     + Performance requirements
     + Security requirements
     + Usability requirements
     + Reliability requirements
     + Maintainability requirements
     + Portability requirements
4. **Appendices**
   * Glossary
   * Use case diagrams
   * Data flow diagrams
   * Entity-Relationship diagrams
   * Interface mockups

Laravel

-- Users table: stores all users (students, teachers, staff)

CREATE TABLE users (

id INT AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(255) NOT NULL,

email VARCHAR(255) UNIQUE NOT NULL,

password VARCHAR(255) NOT NULL,

role ENUM('student', 'teacher', 'staff', ‘admin’) NOT NULL,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP

);

-- Courses table: stores information about courses

CREATE TABLE subjects (

id INT AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(255) NOT NULL,

description TEXT,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP

);

--shift table

Id

Shift

Created at

Updated at

--section table

Id

Section

Created at

Updated at

-- Classes table: stores information about classes, including shift and section

CREATE TABLE classes (

id INT AUTO\_INCREMENT PRIMARY KEY,

subject\_id INT NOT NULL,

teacher\_id INT NOT NULL,

session INT,

shift\_id int NOT NULL,

section\_id int NOT NULL,

version set(‘bangla’,’english’)

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

status Boolean

FOREIGN KEY (course\_id) REFERENCES courses(id),

FOREIGN KEY (teacher\_id) REFERENCES users(id)

);

-- Enrollments table: stores information about student enrollments in classes

CREATE TABLE enrollments (

id INT AUTO\_INCREMENT PRIMARY KEY,

student\_id INT NOT NULL,

class\_id INT NOT NULL,

enrollment\_date DATE,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

FOREIGN KEY (student\_id) REFERENCES users(id),

FOREIGN KEY (class\_id) REFERENCES classes(id)

);

-- Grades table: stores grades for students in classes

CREATE TABLE grades (

id INT AUTO\_INCREMENT PRIMARY KEY,

enrollment\_id INT NOT NULL,

grade DECIMAL(5,2),

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

FOREIGN KEY (enrollment\_id) REFERENCES enrollments(id)

);

-- Attendance table: stores attendance records for students in classes

CREATE TABLE attendance (

id INT AUTO\_INCREMENT PRIMARY KEY,

class\_id int not null

enrollment\_id INT NOT NULL,

date DATE NOT NULL,

status ENUM('present', 'absent', 'late', 'excused') NOT NULL,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

FOREIGN KEY (enrollment\_id) REFERENCES enrollments(id)

);

-- Exams table: stores information about exams

CREATE TABLE exams (

id INT AUTO\_INCREMENT PRIMARY KEY,

class\_id INT NOT NULL,

name VARCHAR(255) NOT NULL,

date DATE NOT NULL,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

FOREIGN KEY (class\_id) REFERENCES classes(id)

);

-- Exam Results table: stores results of exams for students

CREATE TABLE exam\_results (

id INT AUTO\_INCREMENT PRIMARY KEY,

exam\_id INT NOT NULL,

student\_id INT NOT NULL,

score DECIMAL(5,2),

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

FOREIGN KEY (exam\_id) REFERENCES exams(id),

FOREIGN KEY (student\_id) REFERENCES users(id)

);

-- Tuition Fees table: stores tuition fee records for students

CREATE TABLE tuition\_fees (

id INT AUTO\_INCREMENT PRIMARY KEY,

student\_id INT NOT NULL,

month YEAR\_MONTH NOT NULL,

amount DECIMAL(10,2) NOT NULL,

status ENUM('paid', 'unpaid') NOT NULL,

due\_date DATE NOT NULL,

paid\_date DATE,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

FOREIGN KEY (student\_id) REFERENCES users(id)

);

Guardian table( id, enrollment\_id, guardianname1, relation1, phone1, email1, guardianname2, relation2, phone2, email2)

ID Cards

(id, name, class, shift, section)

Subject(SR No. Name Code Type Status)

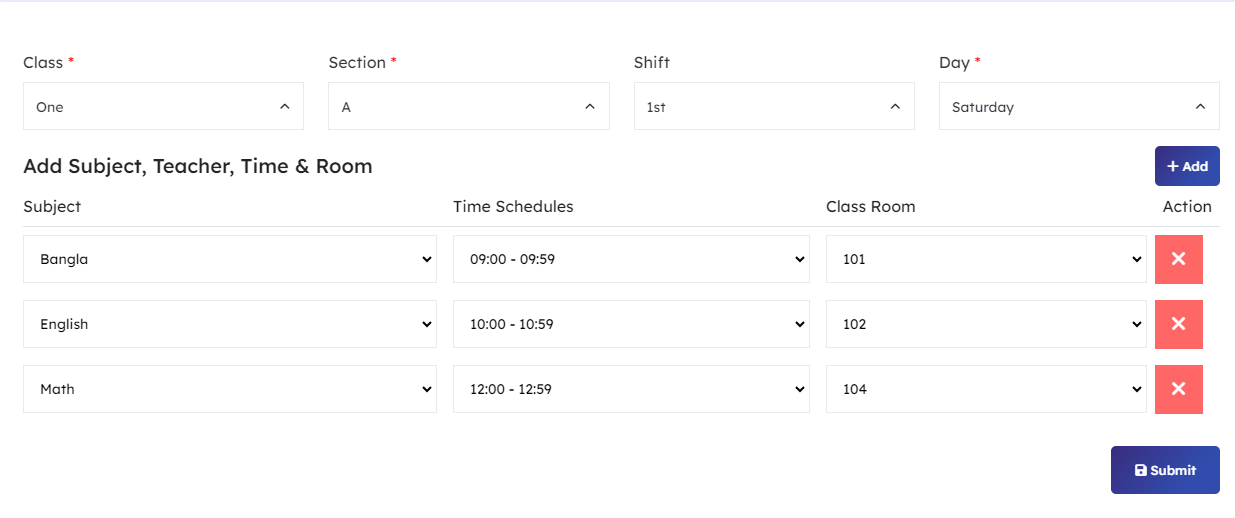
Classwise subject and teacher

Time Schedule(SR No. Type Start Time End Time Status)

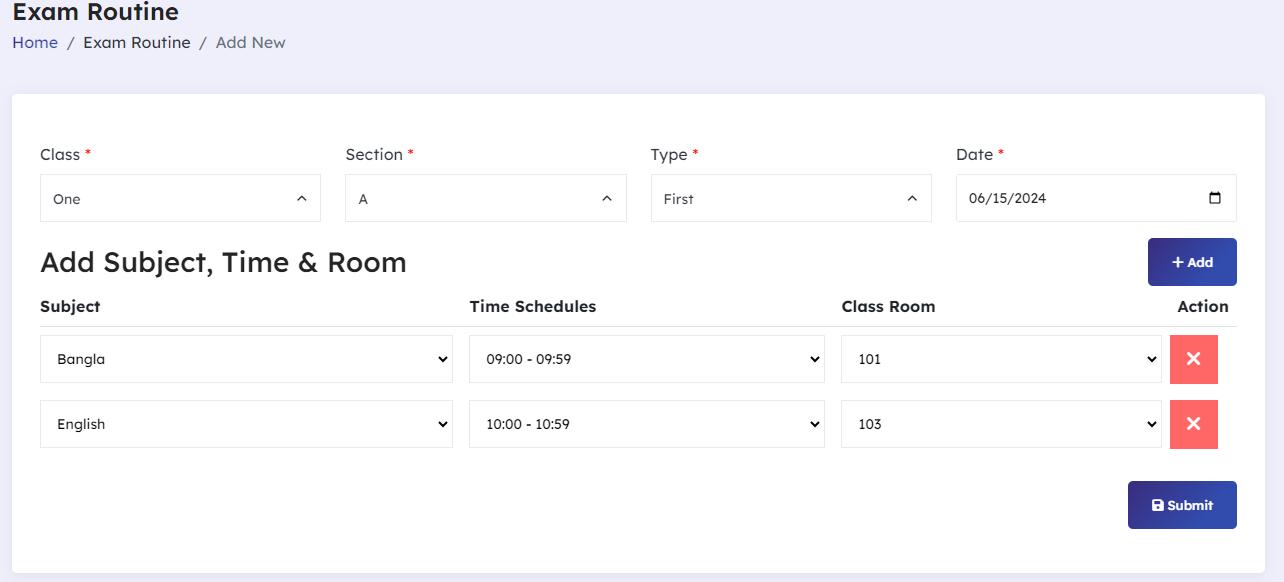
Class Room(SR No. Room No Capacity Status)

Routines:

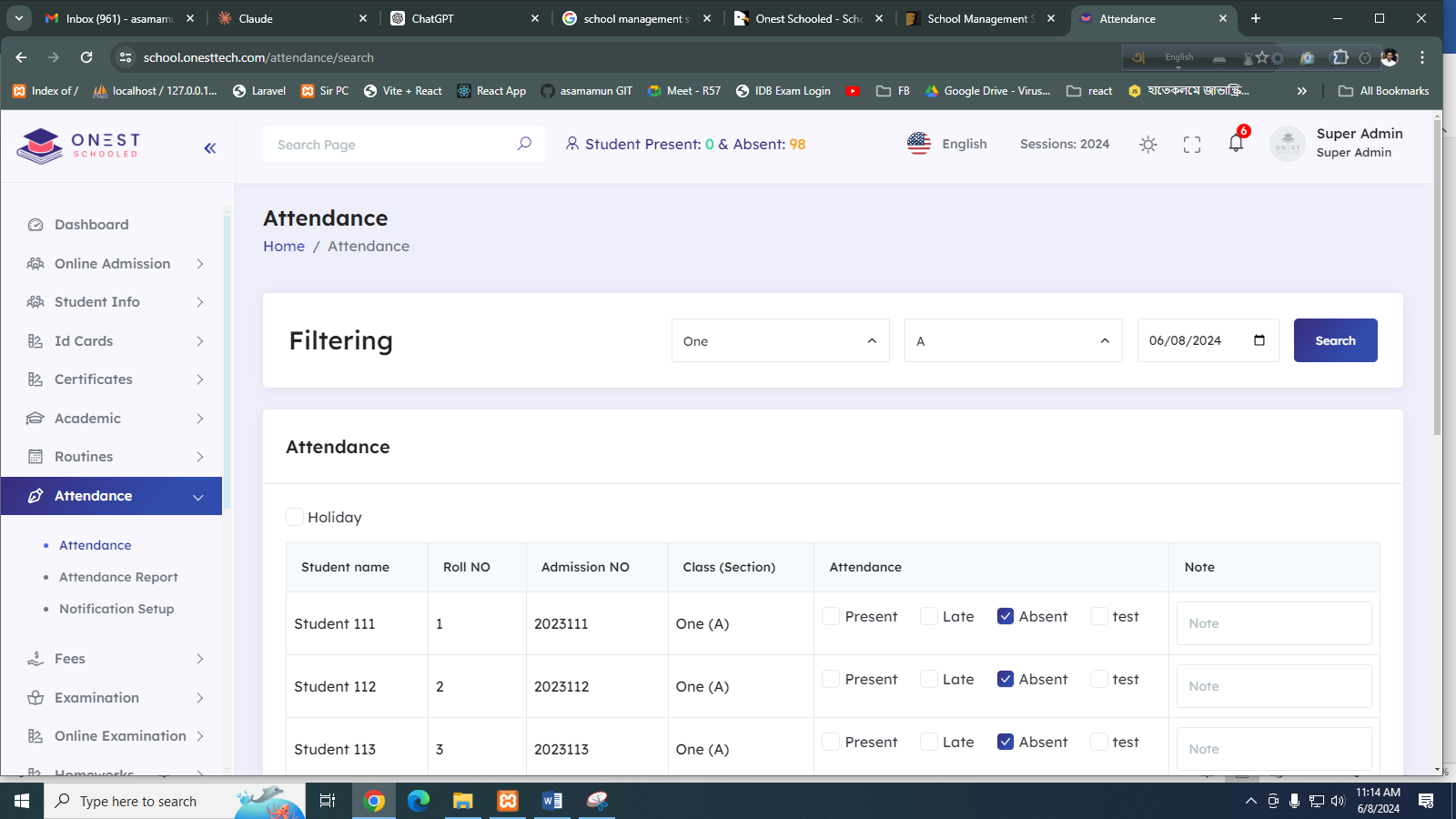
Class Routine(class, day, subject, time schedule, classroom)



Exam Routine(class, exam\_id, date, subject, time, room)



Attendance report:( class, shift, section, date, student\_id, status(present, absent)

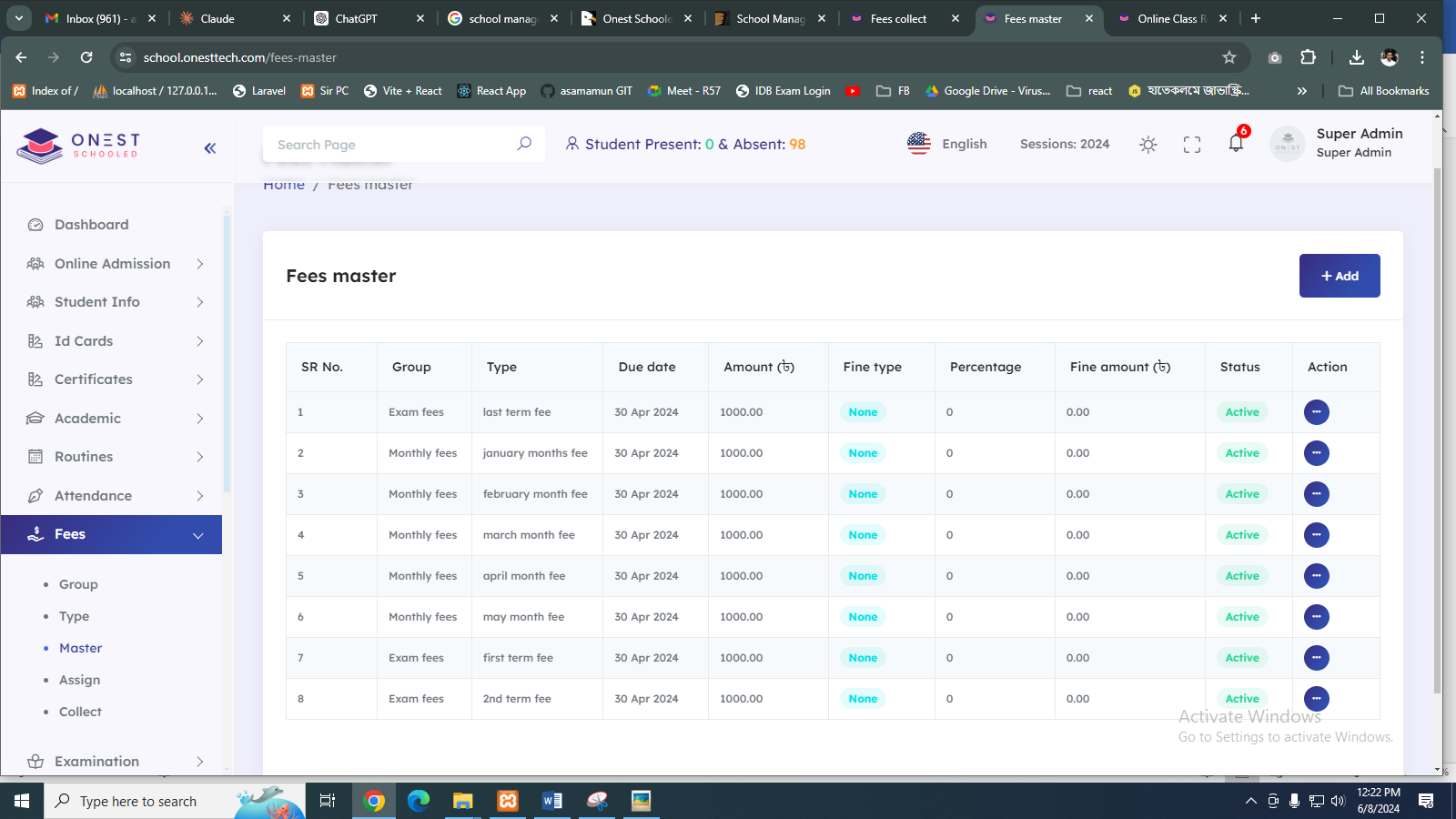


Exam Fees

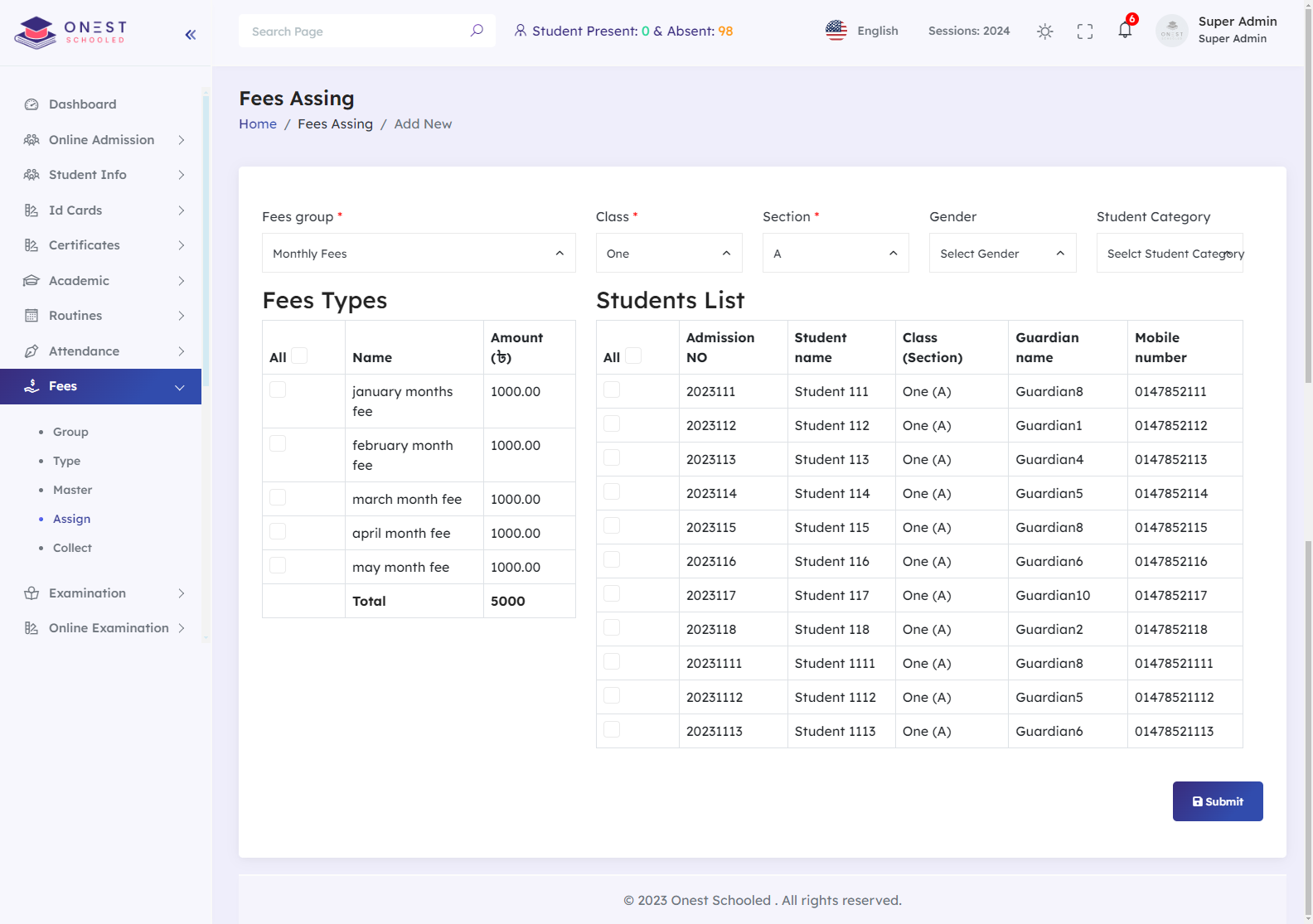
Fees group(id, name, description, status)

Fees type(id, name,. code, details)

Fees master(id, group\_id, type\_id, duedate, amount, finetype, status)

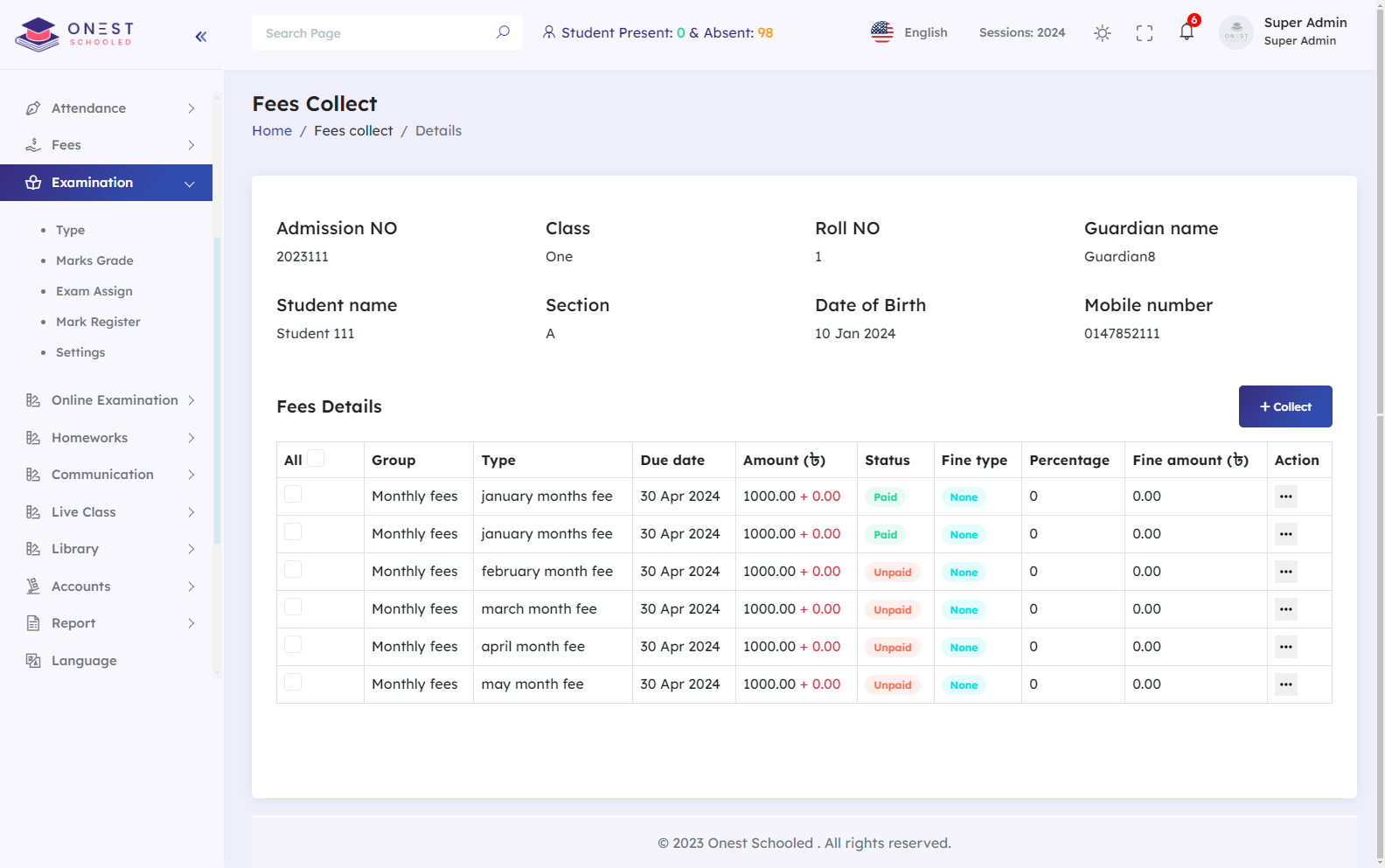


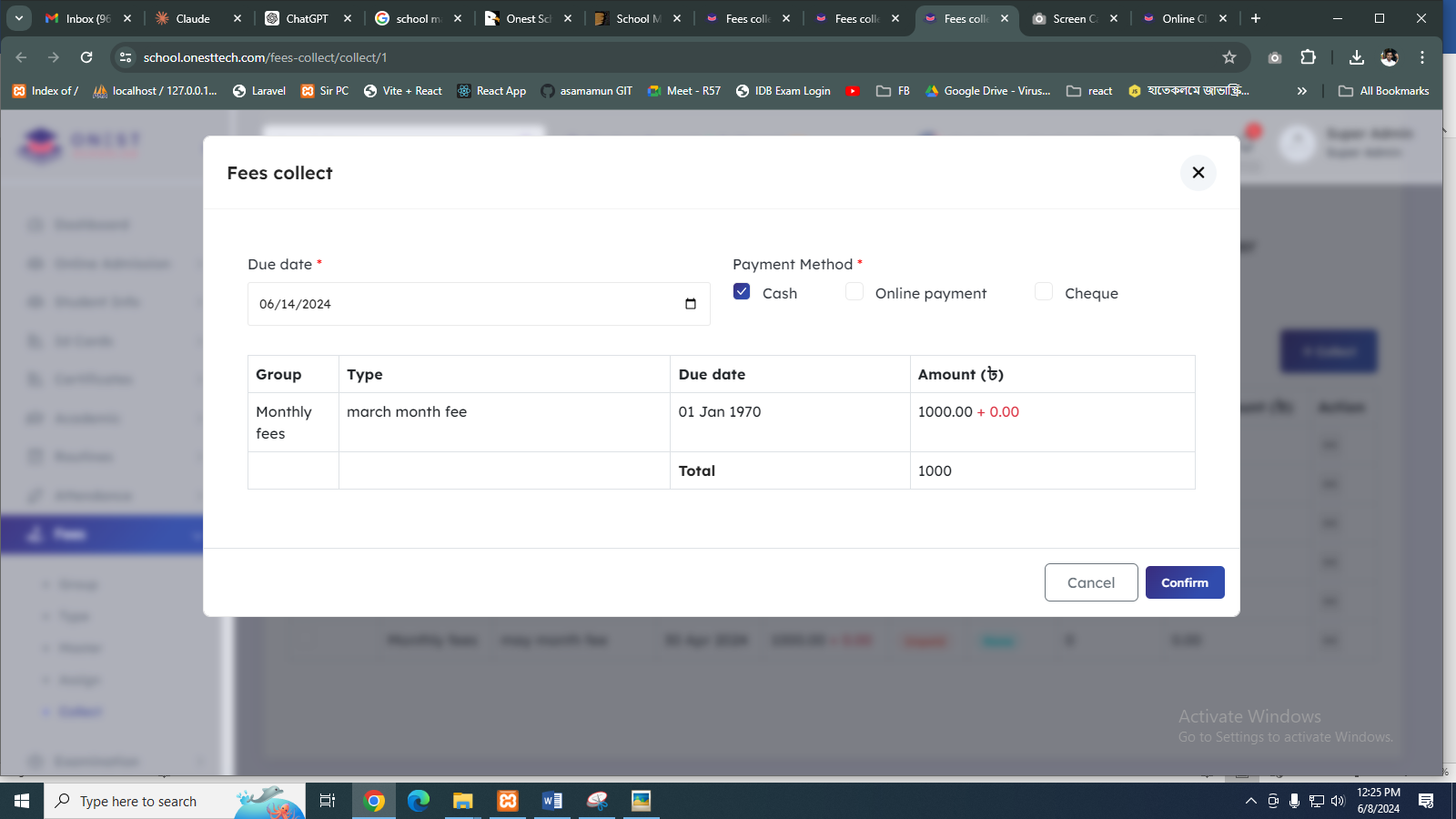
Fees assign:



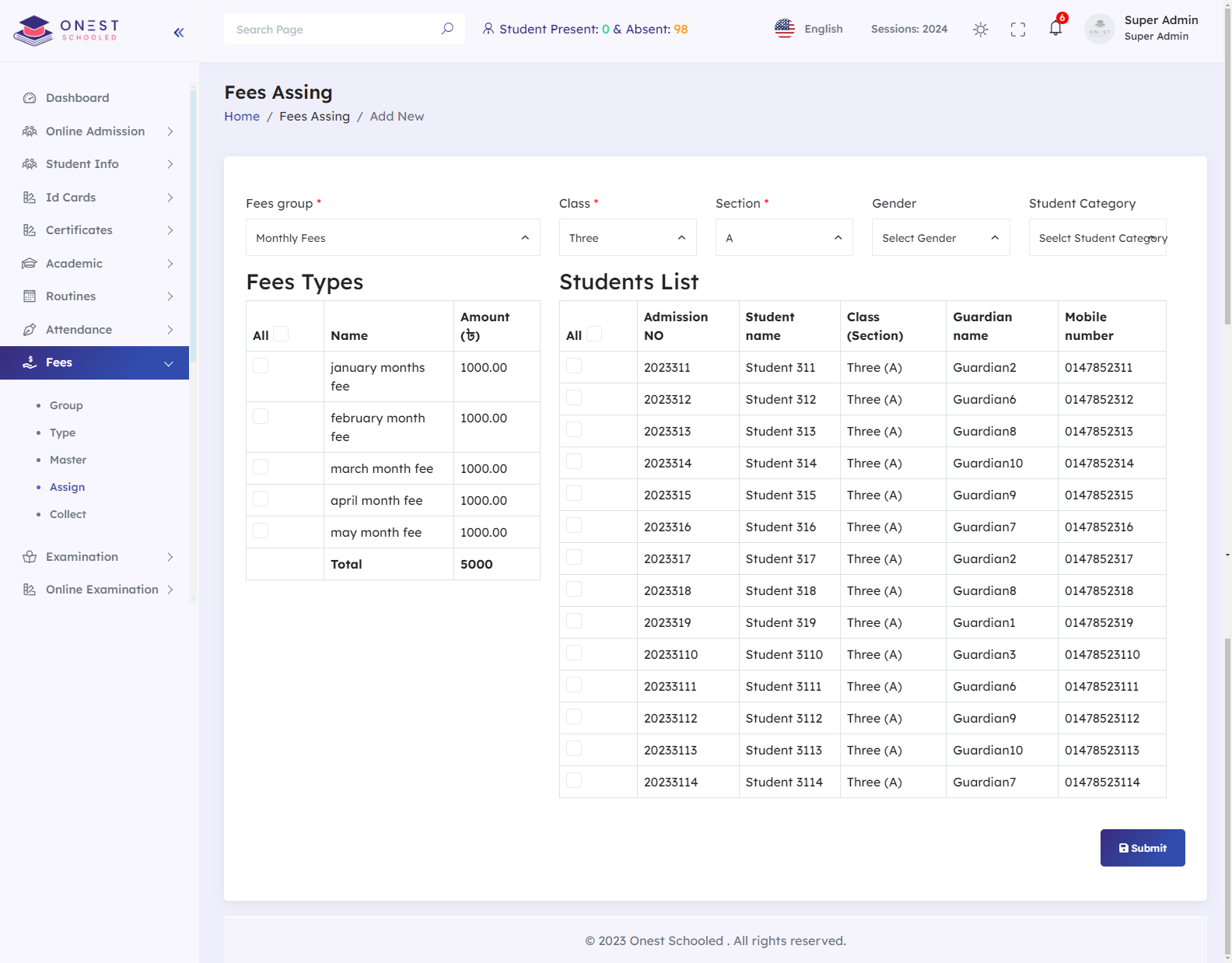
Monthly fees

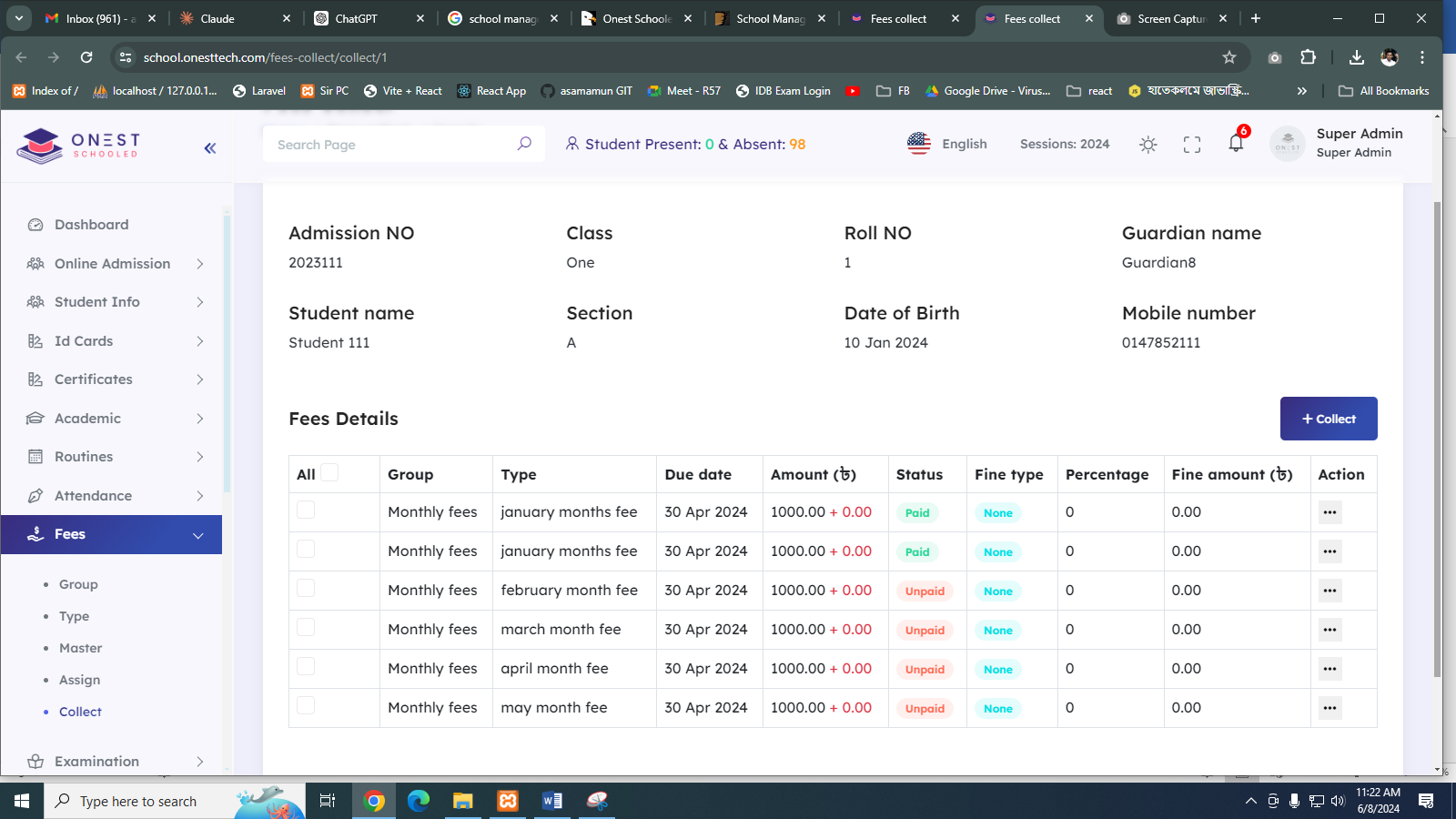
Fees collect

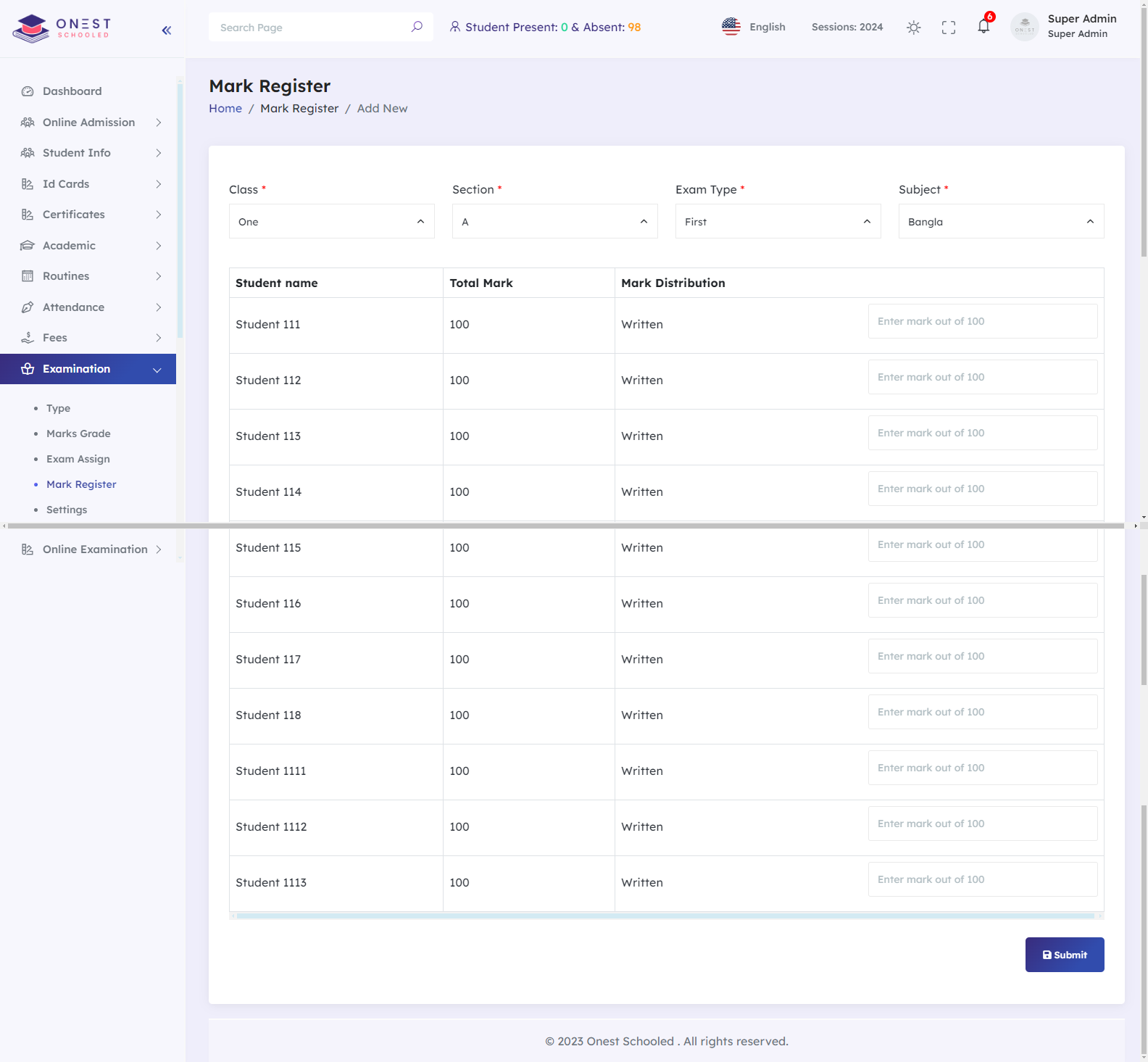




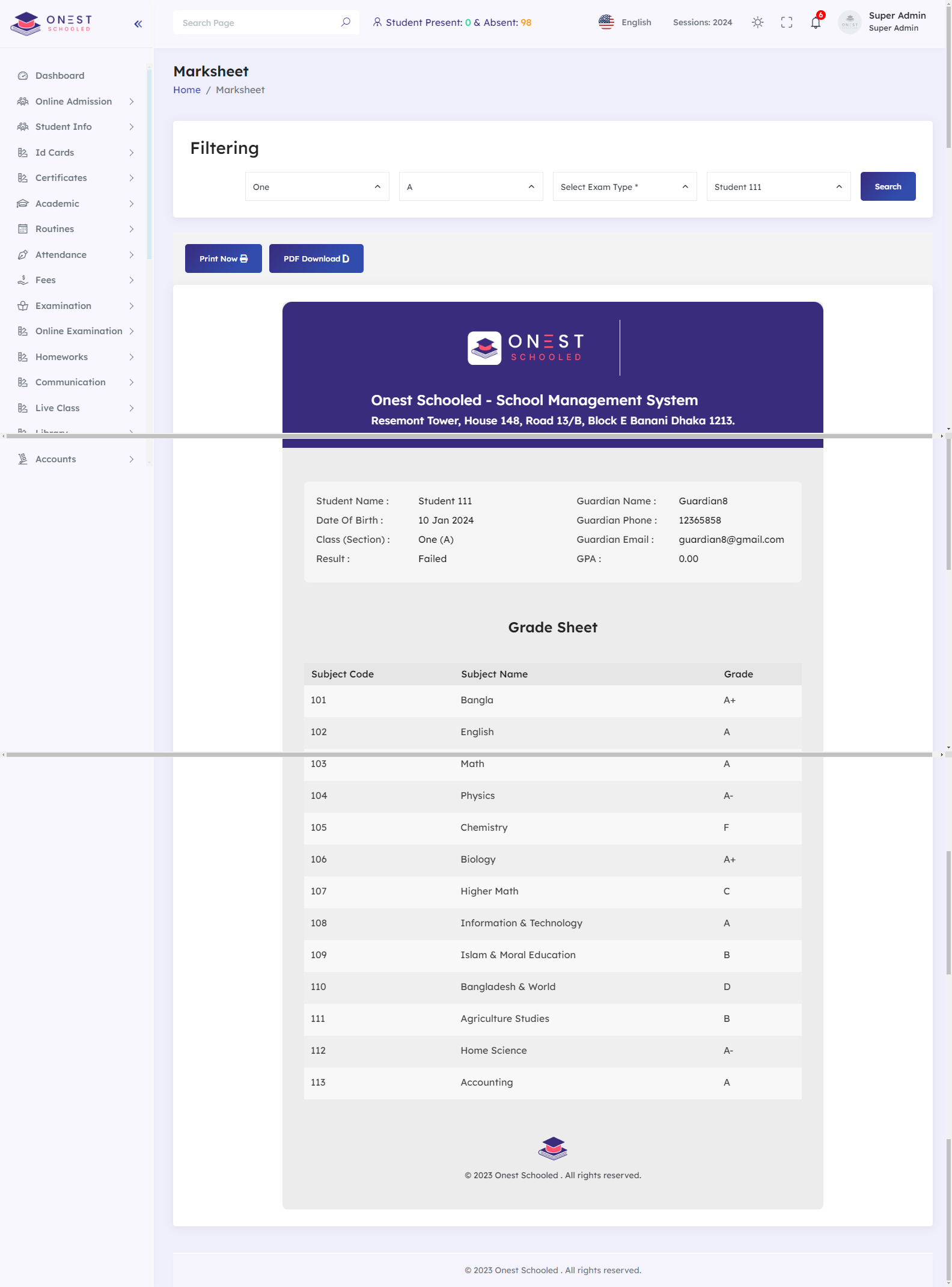
Exam-month-fees(SR No. Group Type Due date Amount (৳) Fine type Percentage Fine amount (৳))



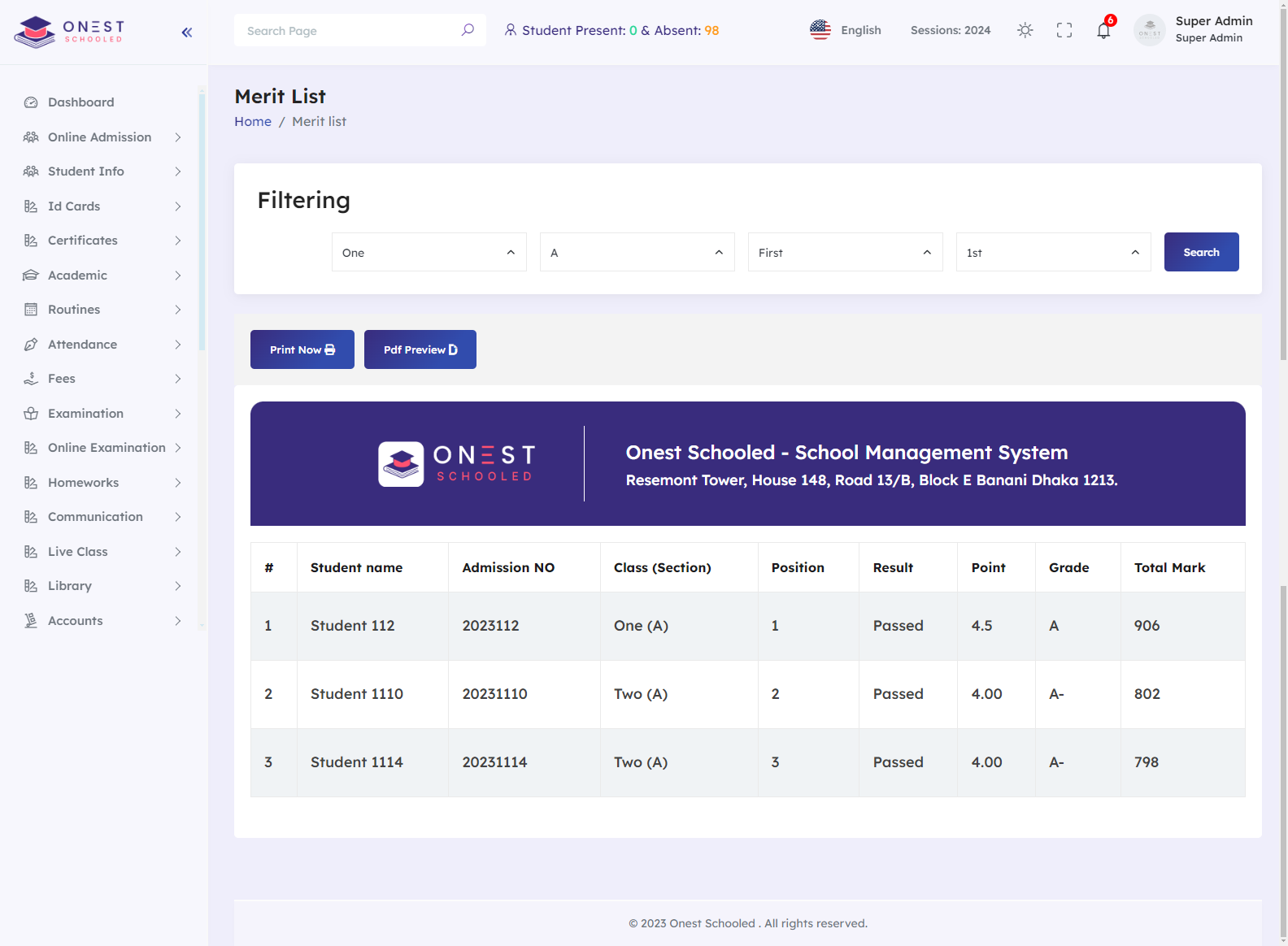




Marksheet



Merit List



Notice Board