|  |  |  |
| --- | --- | --- |
| **CLASS** | **ATTRIBUTES** | **BEHAVIOURS** |
| User Class | name  email  password | signup()  login()  viewandmanageProfile() |
|  |  |  |
| Exam Officer (Inherits from User) | exo\_name (String)  exo\_email (String)  exo\_staffNo (String)  exo\_password (String) | signUp(exo\_Name, exo\_StaffNo, exo\_email, exo\_password)  login(exo\_staffNo, exo\_password)  viewandmanageProfile(exo\_Name, exo\_StaffNo, exo\_email, exo\_password)  createandmanageTeacherDetails(t\_name, t\_staffNo, t\_ID, t\_CellNo)  createandmanageStudentDetails(s\_name, s\_admno, s\_parentCellNo)  GenerateReports() |
|  |  |  |
| Teacher Class (Inherits from User) | t\_name (String)  t\_staffNo (String)  t\_email (String)  t\_password (String) | signUp(t\_staffNo, t\_email, t\_password, t\_passportPhoto)  logIn(t\_staffNo, t\_password)  viewandmanageProfile()  selectSubject()  createEvaluationTool(evalName, evalMaxMarks)  addStudent(s\_admno, s\_name)  enterStudentMarks(s\_EvalMark)  sortResults()  downloadResults()  submitSubjectResults() |
|  |  |  |
| Student Class (Inherits from User) | s\_name (String)  s\_admno (String)  s\_email (String)  s\_parentCellNo (Integer)  s\_password (String) | signUp(s\_admno, s\_email, s\_password, s\_passportPhoto)  logIn(s\_admno, s\_password)  viewandmanageProfile()  view\_s\_Results()  download\_s\_Results() |
|  |  |  |
| Subject Class | sub\_Name (String)  sub\_Code (int) |  |
|  |  |  |
| Level | formNumber  termNameandNumber  termYear |  |
|  |  |  |
| EvaluationTool Class | Type (Enum: Assignment, CAT, GroupWork, Exam)  maxMarks (int) | convertCourseworktoFiftyPercent()  convertExamtoFiftyPercent() |
|  |  |  |
| Result Class | s\_Name  s\_AdmNo | calcStudentSubjectTotalMarks()  calcStudentSubjectAverageMarks()  calcStudentSubjectGrade()  calcStudentSubjectRemark()  calcStudentOverallAverageMark()  calcStudentOverallGrade()  calcStudentOverallRemark()  calcClassAverage()  calcClassGrade()  calcClassRemark() |

**Roadmap:**

* **UI: User Interfaces**
* **Database: normalization**
* **Logic**

(Database + Logic: Backend)

**UI Tools:**

|  |  |
| --- | --- |
| **User** | **UI Tools** |
|  | **Main dashboard** |
| **Exam Officer** | signUp form (exo\_Name, exo\_StaffNo, exo\_email, exo\_password)  login page (exo\_staffNo, exo\_password)  Exam Officer dashboard:   * My Profile section * Students section (s\_name, s\_admno, s\_parentCellNo) * Teachers section (t\_name, t\_staffNo, t\_ID, t\_passportPhoto, t\_CellNo) * Subjects section (subName, subCode) * Terms Section (termNumber, termDuration, termYear) * Reports Section |
| **Teacher** | signUp form (t\_staffNo, t\_email, t\_password)  logIn page (t\_staffNo, t\_password)  Teacher’s dashboard:   * My Profile Section * Marks Entry Section * Create Evaluation Tools Feature * Import Students Section * Marks Input Form |
| **Student** | signUp form (s\_admno, s\_email, s\_password)  logIn page (s\_admno, s\_password)  Student dashboard:   * My Profile Section * My Results Section |

**Normalization**

|  |  |  |  |
| --- | --- | --- | --- |
| **User Activities & System Processes** | **1NF** | **2NF** | **3NF** |
| Exam Officer   * Sign Up * Login * Manage their profile * Manage Student Details   (system will move students to the next form and/or term based on exam results)   * Manage Teacher Details * Generate Subject Results (by form and term) * Generate Student Results * Logout   Teacher   * Sign Up * Log in * Manage their profile * View & Select subjects, and: * Enroll students (filter by form and term) * Enter student Marks (Assignment 1, CAT 1, Assignment 2, CAT 2, Exam) * Sort the results (by total, in ascending or descending order) * Download Subject Results * Submit Results (students see, exam officer sees, no change can be made) * Log out   Student:   * Sign up * Log in * Manage their profile * View/Download Results (filter by form and term) * Log out | * exoName * exoStaffNo * exoEmail * exoPassword * exoStaffNo * exoPassword * exoName * exoStaffNo * exoEmail * exoPassword * studentName * studentAdmno * studentParentCell * studentParentEmail * studentForm * studentTerm * teacherName * teacherStaffNo * subjectName * subjectCode * subjectAverage * subjectRemark * studentSubjectTotal * studentSubjectAverage * studentSubjectGrade * studentSubjectRemark * teacherStaffNo * teacherEmail * teacherPassword * teacherStaffNo * teacherPassword * teacherEmail * teacherPassword * subjectName * subjectCode * studentName * studentAdmno * studentForm * studentTerm * studentAsgn1Mark * studentAsgn2Mark * studentCAT1Mark * studentCAT2Mark * studentExamMark * studentSubjectTotal * studentSubjectAverage * studentSubjectGrade * studentSubjectRemark * studentAdmno * studentEmail * studentPassword * studentAdmno * studentPassword * studentEmail * studentPassword * studentName * studentAdmno * subjectName * subjectCode * studentSubjectTotal * studentSubjectAverage * studentSubjectGrade * studentSubjectRemark * studentOverallTermAverage * studentOverallTermGrade * studentOverallTermRemark | * exoName * exoStaffNo * exoEmail * exoPassword * studentName * studentAdmno * studentEmail * studentPassword * studentParentCell * studentParentEmail * studentForm * studentTerm * teacherName * teacherStaffNo * teacherEmail * teacherPassword * subjectName * subjectCode * subjectAverage * subjectRemark * studentSubjectTotal * studentSubjectAverage * studentSubjectGrade * studentSubjectRemark * studentAsgn1Mark * studentAsgn2Mark * studentCAT1Mark * studentCAT2Mark * studentExamMark * studentSubjectTotal * studentSubjectAverage * studentSubjectGrade * studentSubjectRemark * studentOverallTermAverage * studentOverallTermGrade * studentOverallTermRemark | **ExamOfficerInfo**   * exoName * exoStaffNo * exoEmail * exoPassword   **TeacherInfo**   * teacherName * teacherStaffNo * teacherEmail * teacherPassword   **Level**   * form * term * stream   **SubjectInfo**   * subjectName * subjectCode * subjectAverage * subjectRemark   **StudentInfo**   * studentName * studentAdmno * studentEmail * studentPassword * studentParentCell * studentParentEmail   **Results**   * studentSubjectTotal * studentSubjectAverage * studentSubjectGrade * studentSubjectRemark * studentAsgn1Mark * studentAsgn2Mark * studentCAT1Mark * studentCAT2Mark * studentExamMark * studentOverallTermAverage * studentOverallTermGrade * studentOverallTermRemark * subjectAverage * subjectRemark |

**Class Diagram**

**Subject**

sub\_Name  
sub\_Code

**//inherits methods of Users**addnewTeacher()  
editTeacherDetails()  
deleteTeacher()  
addnewStudent()  
editStudentDetails()  
deleteStudent()  
generateExamReports()

**ExamOfficer: User**

**//inherits attributes of Users**staffNo

**//inherits methods of Users**addnewTeacher()  
editTeacherDetails()  
deleteTeacher()  
addnewStudent()  
editStudentDetails()  
deleteStudent()  
generateExamReports()

**User**

name  
email  
password

Signup()  
login()  
ViewandManageProfile()

**Teacher: User**

**//inherits attributes of Users**staffNo

**//inherits methods of Users**selectSubject()  
createEvaluationTool()  
addStudenttoSubject()  
enterStudentMarks()  
sortResults()  
downloadSubjectResults()  
submitSubjectResults()

**Student: User**

**//inherits attributes of Users**s\_admno  
s\_parentCellNo

**//inherits methods of Users**generatePersonalResults()  
downloadPersonalResults()

1. Lecture 1, slide 10-12
2. Class diagram for the system scenario

Real world scenario:

* Characteristics:

Box – characteristics of a box(color, weight, no.of sides, )//attributes of a box

Things you do to/with a box (//functions or methods to be called) e.g. calculateVolume(), calculateWeight()

doubleVolumeofSquareBox(){

return length\*width\*height;

}

doubleVolumeofRoundBox(){

return pie\*radius\*radius;

}

Primitive data types = int, double, float, String, bool,

int sum(){

s=a+b;

}

Mathematical operators

+, -, =, /, %

Comparison Operators:

>, <, >=, <=, ==