

---

# PGDIT Automation System

---

Software Design and Analysis

---

Team Outlaws

---

Institute of Information Technology

University of Dhaka

# PGDIT Automation System

## Software Design Report

Submitted to

**Nadia Nahar**

Lecturer

Institute of Information Technology

University of Dhaka

Submitted by

Team Name:

**Outlaws**

Team Members:

**Md. Reshad Mollick – 0709**

**Md. Nafis Faysal - 0707**

**Md. Aquib Azmain – 0718**

**Abu Rafe Md. Jamil – 0722**

**Sabbir Hussain Meraj - 0726**

BSSE Session: 2014-2015



**Institute of Information Technology  
University of Dhaka**

16-11-2017

## Abstract

This report is about the software design and analysis report of project “PDGIT automation system”. It contains architectural design, component design and interface design of PGDIT automation system.

## TABLE OF CONTENTS

Chapter 1: Architectural Design.....	10
1.1    Representing the system in context .....	10
1.2    Define archetypes .....	11
1.3    Refine the Architecture into Components.....	13
1.4    Describe Instantiations of the System .....	14
1.5    Structured Design.....	15
Chapter 2: Component Design.....	29
2.1    Design classes that correspond to the problem domain .....	29
2.2    Design classes that correspond to the infrastructure domain .....	29
2.3    Class elaboration .....	30
2.3    a) Class collaboration .....	36
2.3    b) Method call sequence for individual modules.....	41
2.3    c) Appropriate Interfaces.....	46
2.3    d) Elaborate Attributes.....	46
2.3    e) Describe Processing Flow.....	48
2.3    f) State Diagram.....	75
2.4    Persistent Data.....	80
2.5    Elaborated Deployment .....	80
Chapter 3: Interface Design .....	81
3.1    Interface Analysis .....	81
User Analysis .....	81
Task Analysis .....	84
3.2    Interface Design Steps .....	88
Define interface objects and actions .....	88

Define events that will cause the state of the user interface to change ..... 112

Depict each interface state as it look to end user..... 115

## Content of Figures

Figure 1: PGDIT Automation System in context .....	10
Figure 2: Archetypes - attributes.....	11
Figure 3: Archetypes - Methods.....	12
Figure 4: Components of PGDIT System .....	13
Figure 5: Instantiations of PDGIT system.....	14
Figure 6: Level-0 DFD .....	15
Figure 7: Level-1 DFD .....	16
Figure 8: Level-2(Authentication) DFD .....	16
Figure 9: DFD of Registration process.....	17
Figure 10: Mapping of registration process.....	17
Figure 11: DFD of password recovery .....	18
Figure 12: Mapping of password recovery .....	18
Figure 13: DFD of Sign in.....	19
Figure 14: Transaction mapping of sign in.....	19
Figure 15: DFD of Admission administration .....	20
Figure 16: Mapping of admission administration .....	20
Figure 17: DFD of application payment .....	21
Figure 18: Mapping of application payment .....	21
Figure 19: DFD of scrutinizing process .....	22
Figure 20: Mapping of scrutinizing process .....	22
Figure 21: DFD of seat plan generation process .....	23
Figure 22: Mapping of seat plan generator .....	23
Figure 23: Written mark process.....	24
Figure 24: Mapping of written exam mark process .....	24
Figure 25: Mapping of viva schedule generator .....	25
Figure 26: DFD of admission payment process.....	26
Figure 27: Mapping of admission payment .....	26
Figure 28: DFD of course result generation process.....	27
Figure 29: Mapping of result generation process .....	27
Figure 30: DFD of result publication process .....	28

Figure 31: Mapping of course result publisher .....	28
Figure 32: Class elaboration of User .....	30
Figure 33: Class elaboration of Student.....	30
Figure 34: Class elaboration of Teacher.....	31
Figure 35: Class elaboration of Receptionist .....	32
Figure 36: Class elaboration of Scrutinizer.....	32
Figure 37: Program Chairperson.....	33
Figure 38: Class elaboration of Exam Controller .....	34
Figure 39: Class elaboration of Applicant.....	35
Figure 40: Class elaboration of Database operator.....	35
Figure 41: Class collaboration of authentication.....	36
Figure 42: Class collaboration of update Profile .....	36
Figure 43: Class collaboration of new admission session .....	36
Figure 44: Class collaboration of Apply .....	37
Figure 45: Class collaboration of view payment board.....	37
Figure 46: Class collaboration of update payment status .....	37
Figure 47: Class collaboration of update status of eligibility .....	38
Figure 48: Class collaboration of add viva schedule .....	38
Figure 49: Class collaboration of add new room.....	39
Figure 50: Class collaboration of result submission.....	39
Figure 51: Class collaboration of view result .....	39
Figure 52: Class collaboration of update result.....	40
Figure 53: Class collaboration of view supplementary list .....	40
Figure 54: Class collaboration f view course result .....	40
Figure 55: Method call sequence of Authentication .....	41
Figure 56: Method call sequence of Teacher registration.....	41
Figure 57: Method call sequence of Student account creation.....	42
Figure 58: Method call sequence of Candidate management .....	42
Figure 59: Method call sequence of Course management .....	43
Figure 60: Method call sequence of Result management.....	43
Figure 61: Method call sequence of User management.....	44
Figure 62: Method call sequence of Admission session .....	44

Figure 63: Method call sequence of Exam schedule manage .....	44
Figure 64: Method call sequence of Result manage .....	45
Figure 65: Method call sequence of Application processing .....	45
Figure 66: Processing flow of chooseOption.....	48
Figure 67: Processing flow of viewListOfCourse() .....	49
Figure 68: Processing flow of viewResultSheet(courseID,year) .....	49
Figure 69: Processing flow of viewSupplementaryList(courseID,year).....	50
Figure 70: Processing flow of viewCourseInformation(courseID,year) .....	50
Figure 71: Processing flow of updateStudentMark(courseID, year, mark, studentID) .....	51
Figure 72: Processing flow of updateCourseInformation(courseID, year, outline, description).....	51
Figure 73: Processing flow of updatePDF(courseID,year,username) .....	52
Figure 74: Processing flow of notifyForSupplementaryExam(courseID,year) .....	52
Figure 75: Processing flow of Connect().....	53
Figure 76: Processing flow of executeSelectCommand(sql).....	53
Figure 77: Processing flow of executeUpdateCommand(sql) .....	54
Figure 78: Processing flow of executeInsertCommand(sql) .....	54
Figure 79: Processing flow of viewPaymentBoard().....	55
Figure 80: Processing flow of getPaymentType () .....	55
Figure 81: Processing flow of updatePaymentStatus() .....	56
Figure 82: Processing flow of viewListOfCourses() .....	56
Figure 83: Processing flow of getCourseID () .....	57
Figure 84: Processing flow of viewResult() .....	57
Figure 85: Processing flow of changeInstructor() .....	58
Figure 86: Processing flow of lockResult() .....	58
Figure 87: Processing flow of viewListOfUsers() .....	59
Figure 88: Processing flow of addStudents() .....	59
Figure 89: Processing flow of viewUserInfo () .....	60
Figure 90: Processing flow of getUserInfo() .....	60
Figure 91: Processing flow of updateInfo() .....	61
Figure 92: Processing flow of downloadAdmitCard (applicationID,dateOfBirth) .....	61

Figure 93: Processing flow of apply(name, fathersName, mothersName, email, contactNo, photo, dateOfBirth, nationality, permanentAddress, sscO, hscA, bachelor, hscMathMark, bachelorMathMark).....	62
Figure 94: Processing flow of veiwHeader() .....	62
Figure 95: Processing flow of veiwDashboard() .....	63
Figure 96: Processing flow of viewNotification (notificationID) .....	63
Figure 97: Processing flow of updatePersonalDate(username, email, mobile, address, password, picture).....	64
Figure 98: Processing flow of verifyPassword(username, password) .....	64
Figure 99: Processing flow of viewListOfRoom() .....	65
Figure 100: Processing flow of updateRoomCapacity() .....	65
Figure 101: Processing flow of insertRoomDatabase() .....	66
Figure 102: Processing flow of viewVivaSchedule() .....	66
Figure 103: Processing flow of insertScheduleIntoDatabase() .....	67
Figure 104: Processing flow of insertAdmisionInfointoDatabase() .....	67
Figure 105: Processing flow of viewListOfEligibleCandidatesOnScrutinize() .....	68
Figure 106: Processing flow of viewListOfEligibleCandidatesAfterScrutinize() .....	68
Figure 107: Processing flow of viewListOfEligibleCandidatesAfterWrittenExam() .....	69
Figure 108: Processing flow of updateAdmissionInfo .....	69
Figure 109: Processing flow of viewCourseInformation(courseID,year) .....	70
Figure 110: Processing flow of viewResultsheet() .....	70
Figure 111: Processing flow of viewRunningCourses() .....	71
Figure 112: Processing flow of getapplicationId() .....	71
Figure 113: Processing flow of getIndex().....	72
Figure 114: Processing flow of UpdateStatus(applicationId,index) .....	72
Figure 115: Processing flow of searchByApplicationID(applicationId) .....	73
Figure 116: Processing flow of searchByIndex(index) .....	73
Figure 117: Processing flow of searchByIndex .....	74
Figure 118: State diagram of User class.....	75
Figure 119: State diagram of Student class .....	75
Figure 120: State diagram of Teacher class.....	76
Figure 121: State diagram of Receptionist class .....	76

Figure 122: State diagram of Exam Controller class .....	77
Figure 123: State diagram of Scrutinizer class.....	77
Figure 124: State diagram of Program Chairman class .....	78
Figure 125: State diagram of Applicant class.....	79
Figure 126: State diagram of Database Operator .....	79
Figure 127: Elaborated deployment of PGDIT Automation System .....	80
Figure 128: External (Home) .....	88
Figure 129: External (Admission – Session ended) .....	89
Figure 130: External (Admission - Session going admission form) .....	89
Figure 131: External (Admission - Session going - Admission form - Submit) .....	90
Figure 132: External (Admission – Session going – Admit card - Available) .....	90
Figure 133: External (Admission – Session going – Admit card – Not Available) .....	91
Figure 134: External (Create Account) .....	91
Figure 135: External (Sing in) .....	92
Figure 136: External (Sing in-Forgot password) .....	92
Figure 137: External (Sing in - Blocked) .....	93
Figure 138: Internal (Teacher: Home) .....	93
Figure 139: Internal (Teacher: Manage Course- Course Details).....	94
Figure 140: Internal (Teacher: Manage Course- Update Marks) .....	94
Figure 141: Internal (Teacher: Manage Course- List of Students (Suppli)) .....	95
Figure 142: Internal (Teacher: Viva Schedule) .....	95
Figure 143: Internal (Member: Profile) .....	96
Figure 144: Internal (Student: Home) .....	96
Figure 145: Internal (Student: View Course) .....	97
Figure 146: Internal (Student: View Course .....	97
Figure 147: Internal (Student: Profile).....	98
Figure 148: Internal (Receptionist: Home) .....	98
Figure 149: Internal (Receptionist: Search Applicant-By name) .....	99
Figure 150: Internal (Receptionist: Search Applicant-By mobile no).....	99
Figure 151: Internal (Receptionist: Payment-Application payment).....	100
Figure 152: Internal (Receptionist: Payment-Admission fees) .....	101
Figure 153: Internal (Receptionist: Profile) .....	102

Figure 154: Internal (Scrutinizer: Home) .....	102
Figure 155: Internal (Scrutinizer: Search Applicant) .....	103
Figure 156: Internal (Scrutinizer: Profile) .....	103
Figure 157: Internal (Program chairperson: Home) .....	104
Figure 158: Internal (Program chairperson: Manage Course- Details) .....	104
Figure 159: Internal (Program chairperson: Manage Course- Add) .....	105
Figure 160: Internal (Program chairperson: Manage User- View) .....	105
Figure 161: Internal (Program chairperson: Manage User- View - Edit) .....	106
Figure 162: Internal (Program chairperson: Manage User- New User Request) .....	107
Figure 163: Internal (Program chairperson: Manage User- Create Student Account) ...	107
Figure 164: Internal (Program Chairman: Profile) .....	108
Figure 165: Internal (Exam controller: Home).....	108
Figure 166: Internal (Exam controller: Admission-Session) .....	109
Figure 167: Internal (Exam controller: Admission-Eligible Applicants) .....	109
Figure 168: Internal (Exam controller: Manage Schedule-Viva) .....	110
Figure 169: Internal (Exam controller: Manage Schedule-Update room) .....	110
Figure 170: Internal (Exam controller: Profile).....	111
Figure 171: Home .....	115
Figure 172: Notices .....	115
Figure 173: Options .....	116
Figure 174: Admission form .....	116
Figure 175: Dashboard of Teacher .....	117
Figure 176: Profile .....	117
Figure 177: Course details.....	118
Figure 178: Update marks.....	118
Figure 179: Supplementary exam list .....	119
Figure 180: Viva schedule.....	119
Figure 181: Dashboard of exam controller .....	120
Figure 182: Admission session .....	120
Figure 183: Viva schedule manage .....	121
Figure 184: Update room.....	121

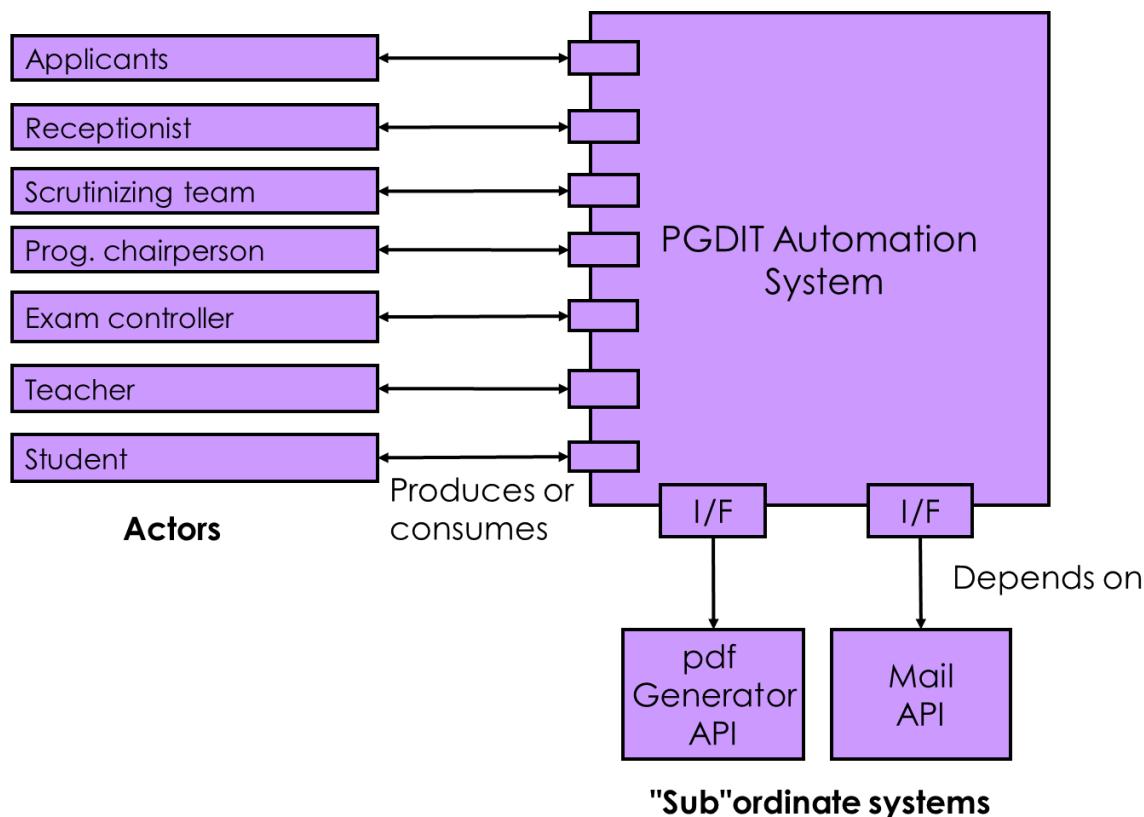
# CHAPTER 1: ARCHITECTURAL DESIGN

Software architectural design represents the structure of the data and program components that are required to build a computer-based system. There are 4 steps in architectural design:

- 1) Represent the system in context
- 2) Define archetypes
- 3) Refine the architecture into components
- 4) Describe instantiations of the system

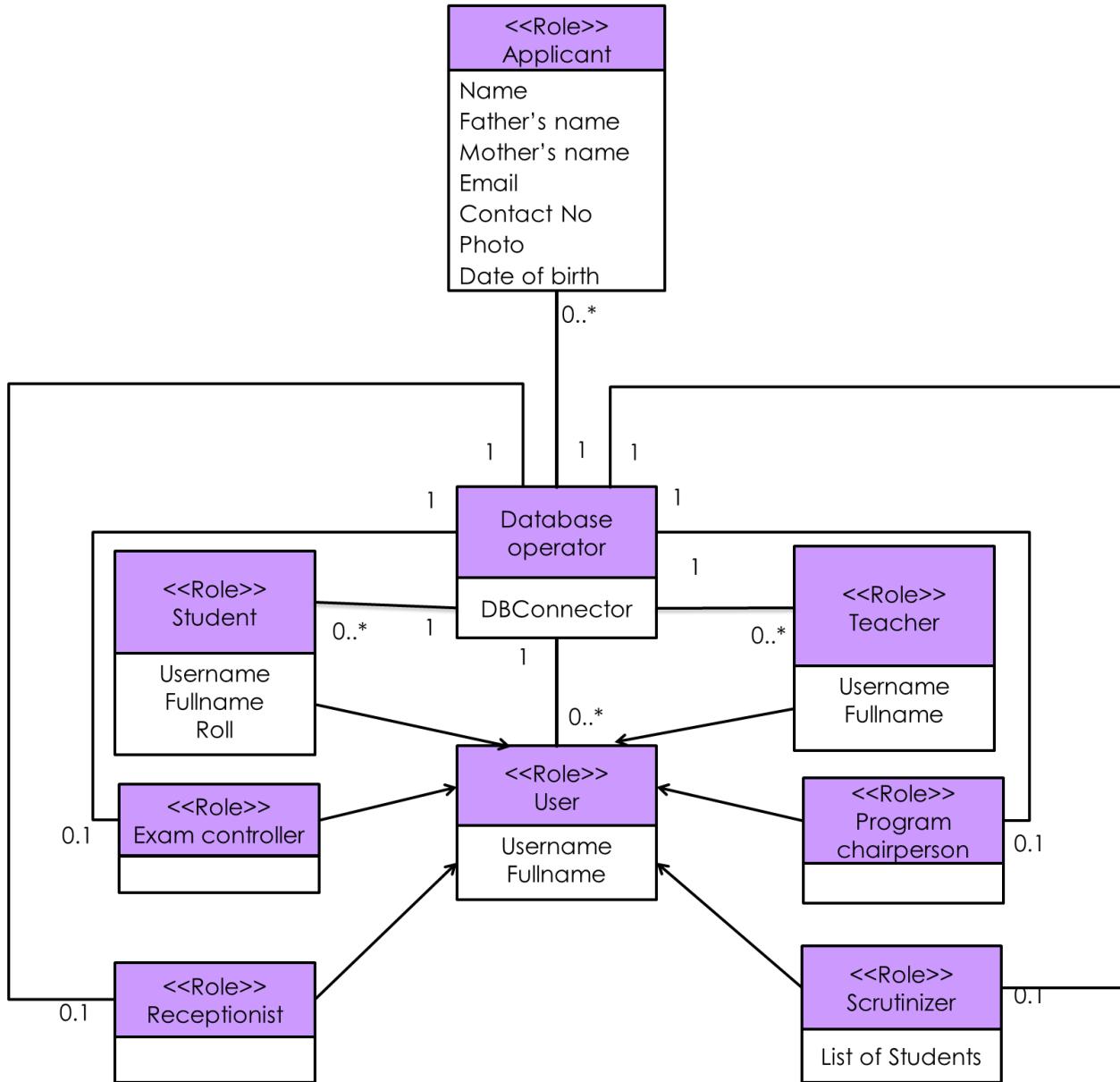
## 1.1 Representing the system in context

Following is the ACD (Architectural Context Diagram) of "PGDIT automation system".

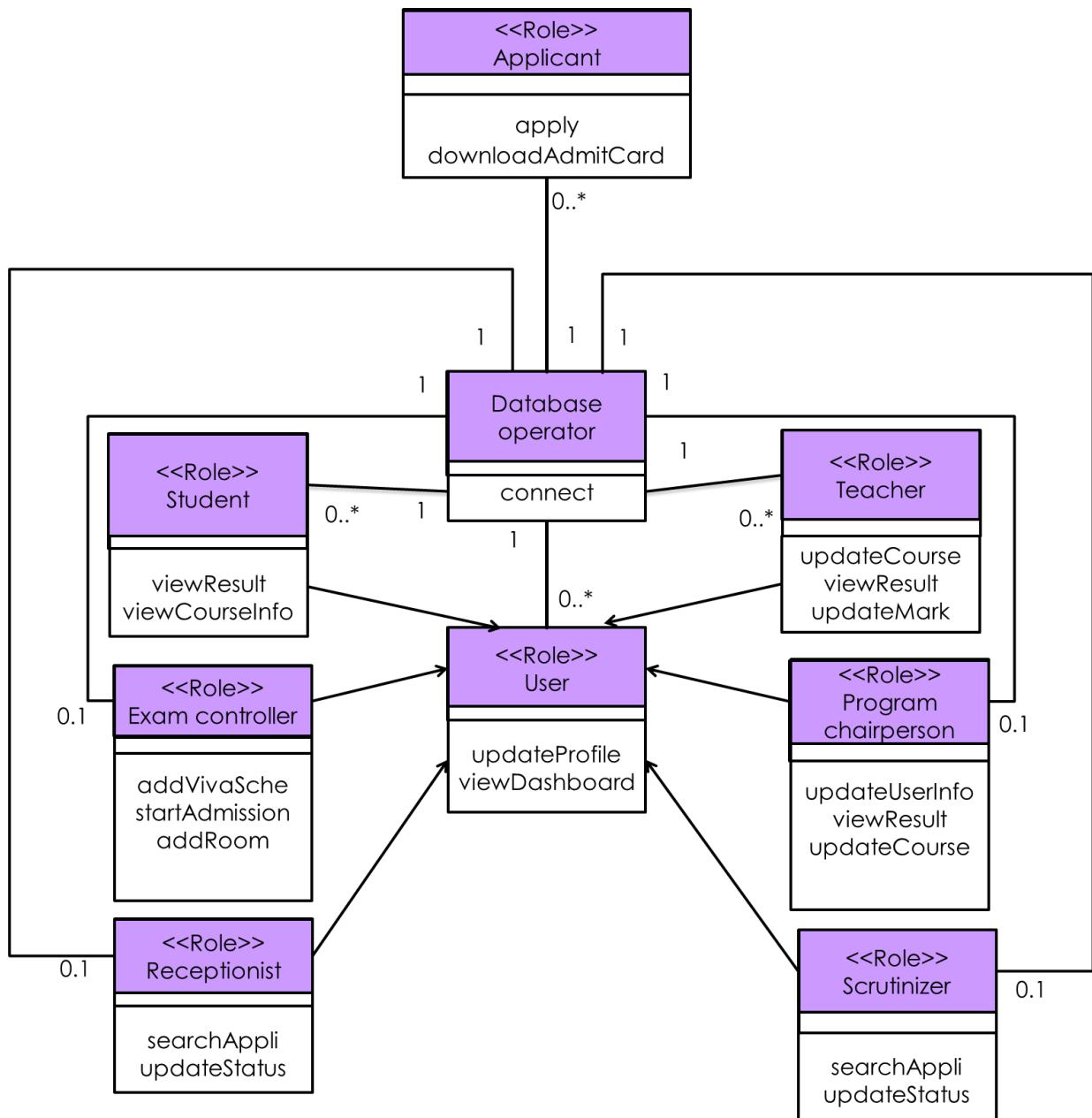


**Figure 1: PGDIT Automation System in context**

## 1.2 Define archetypes

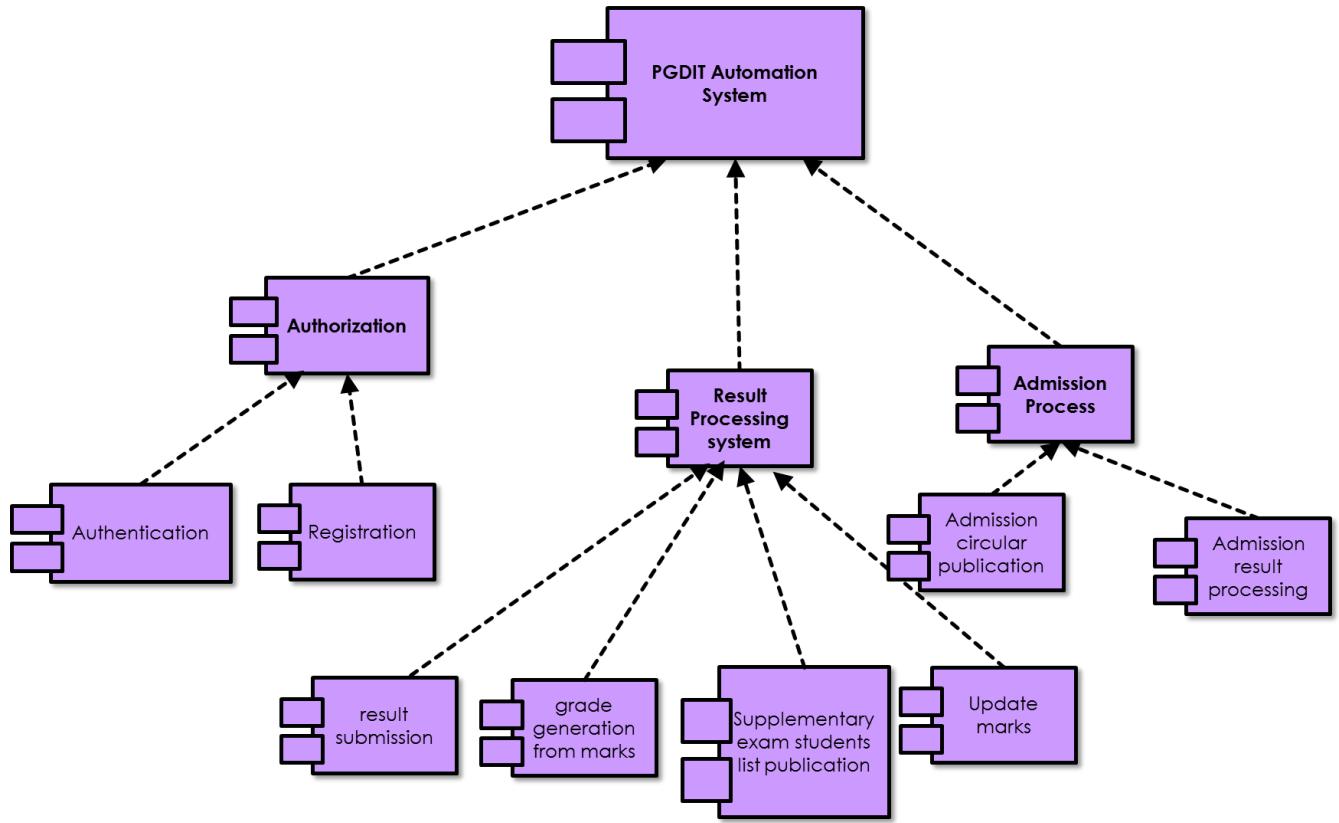


**Figure 2: Archetypes - attributes**



**Figure 3: Archetypes - Methods**

### 1.3 Refine the Architecture into Components



**Figure 4: Components of PGDIT System**

## 1.4 Describe Instantiations of the System

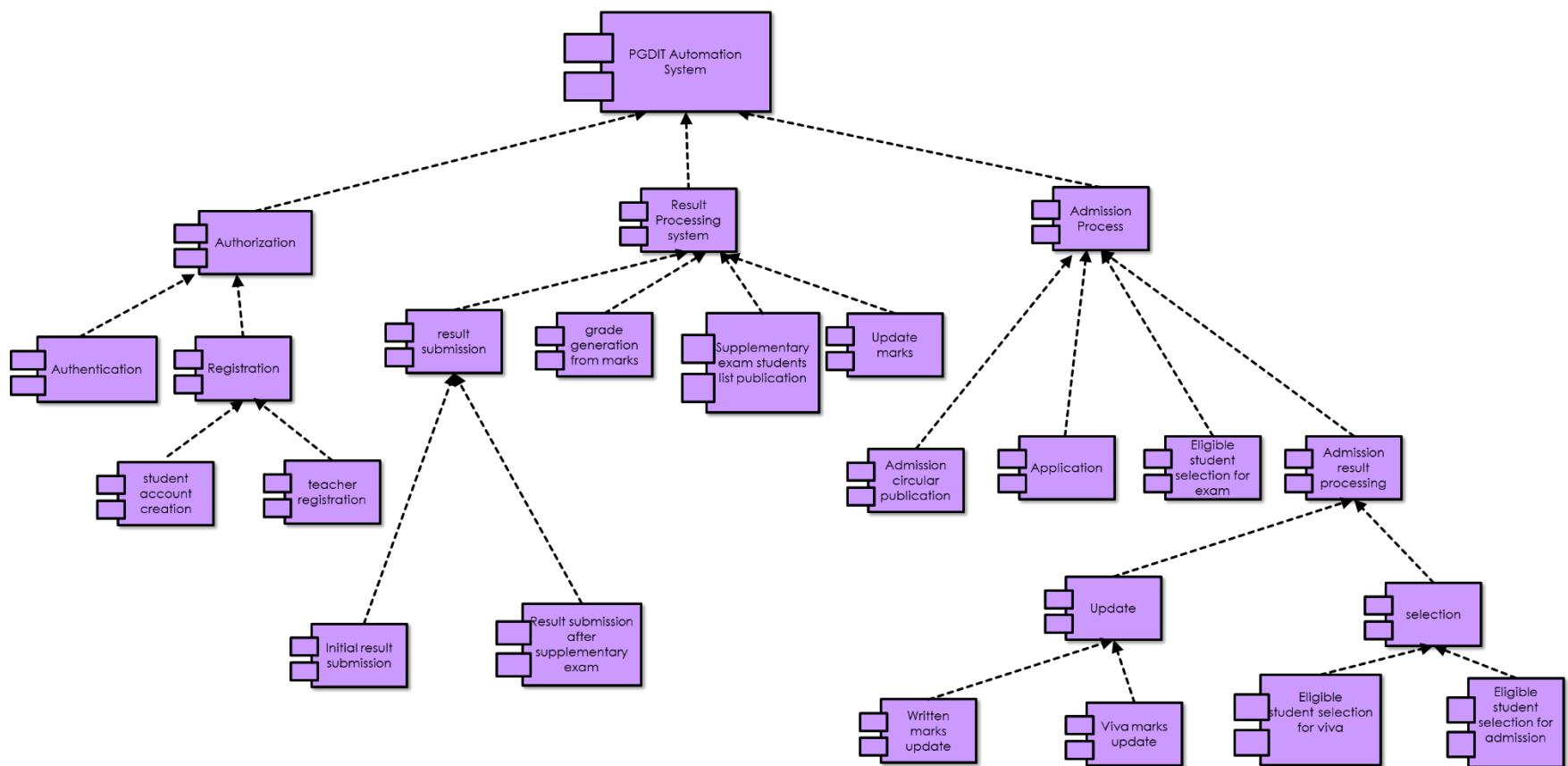
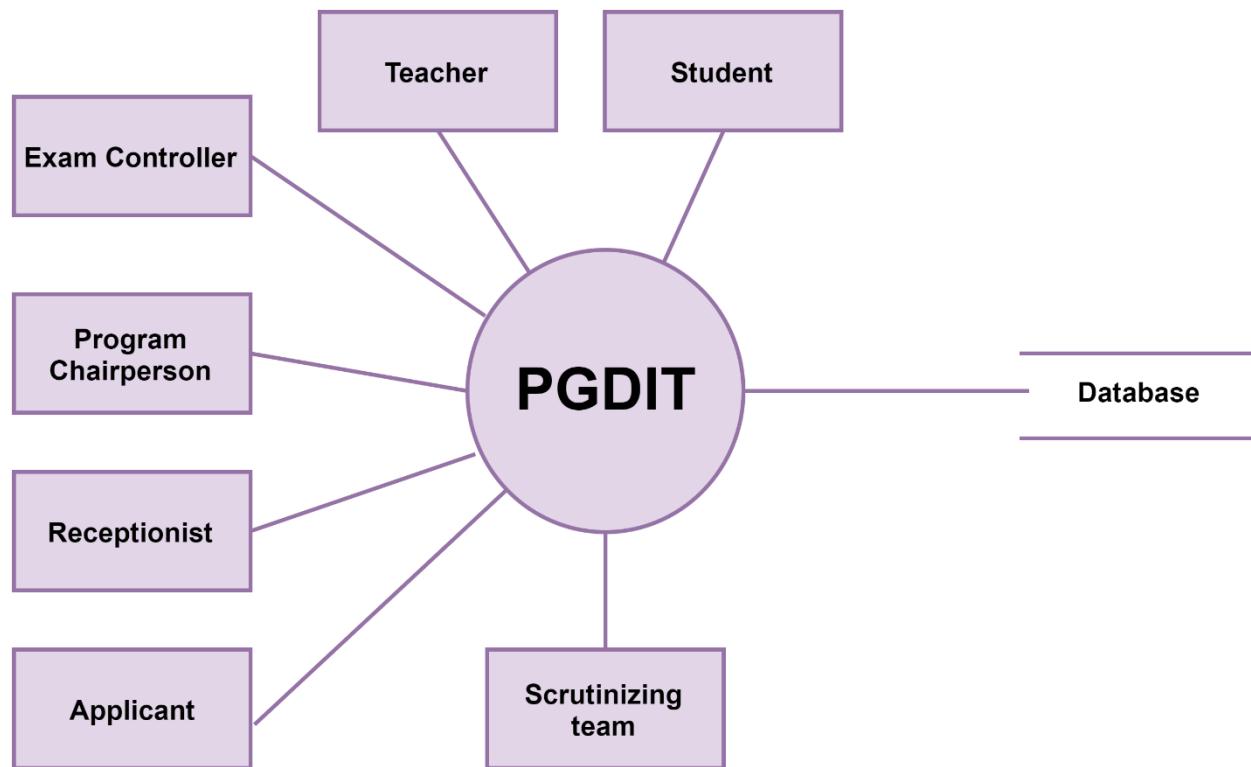


Figure 5: Instantiations of PDGIT system

## 1.5 Structured Design

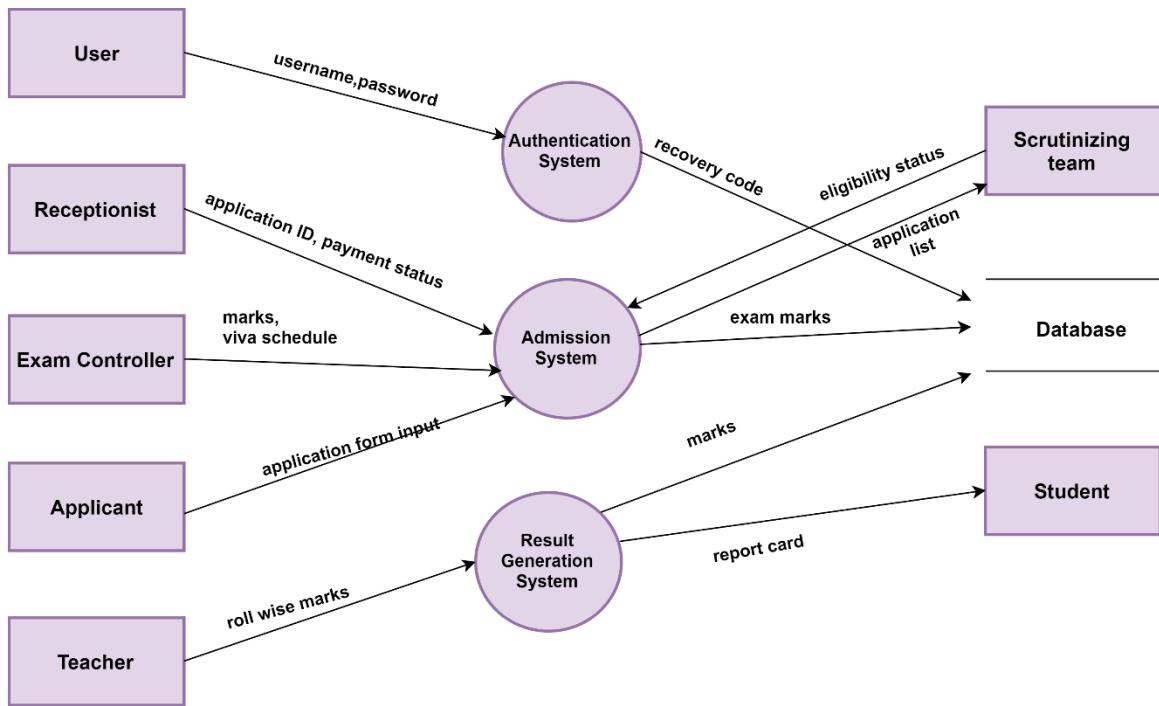
A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination.

**Level-0 DFD**



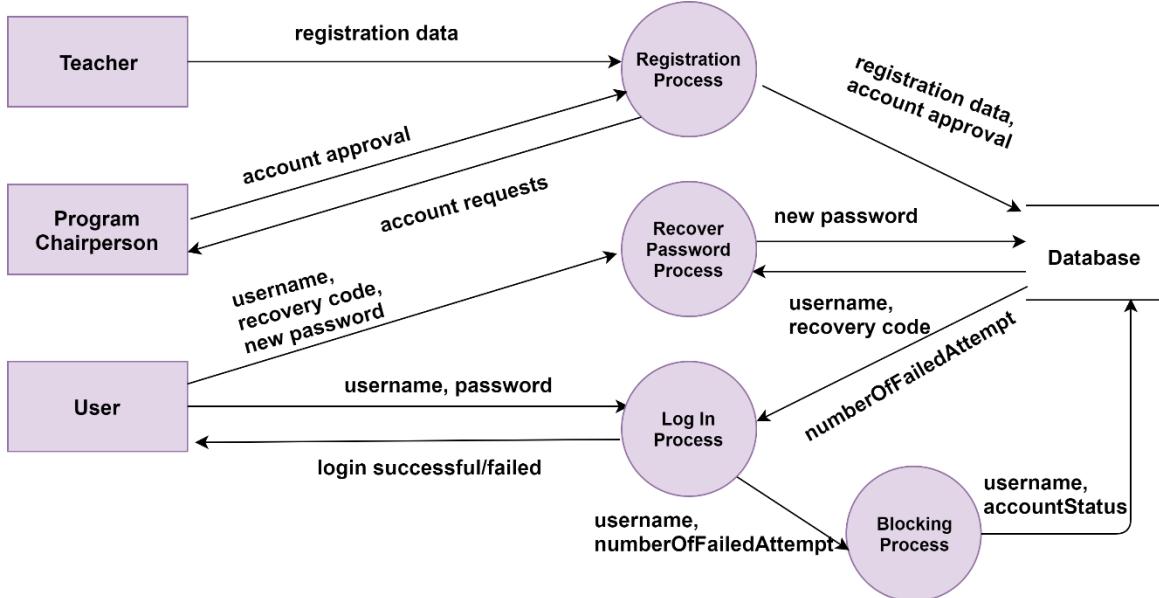
**Figure 6: Level-0 DFD**

**Level-1 DFD**



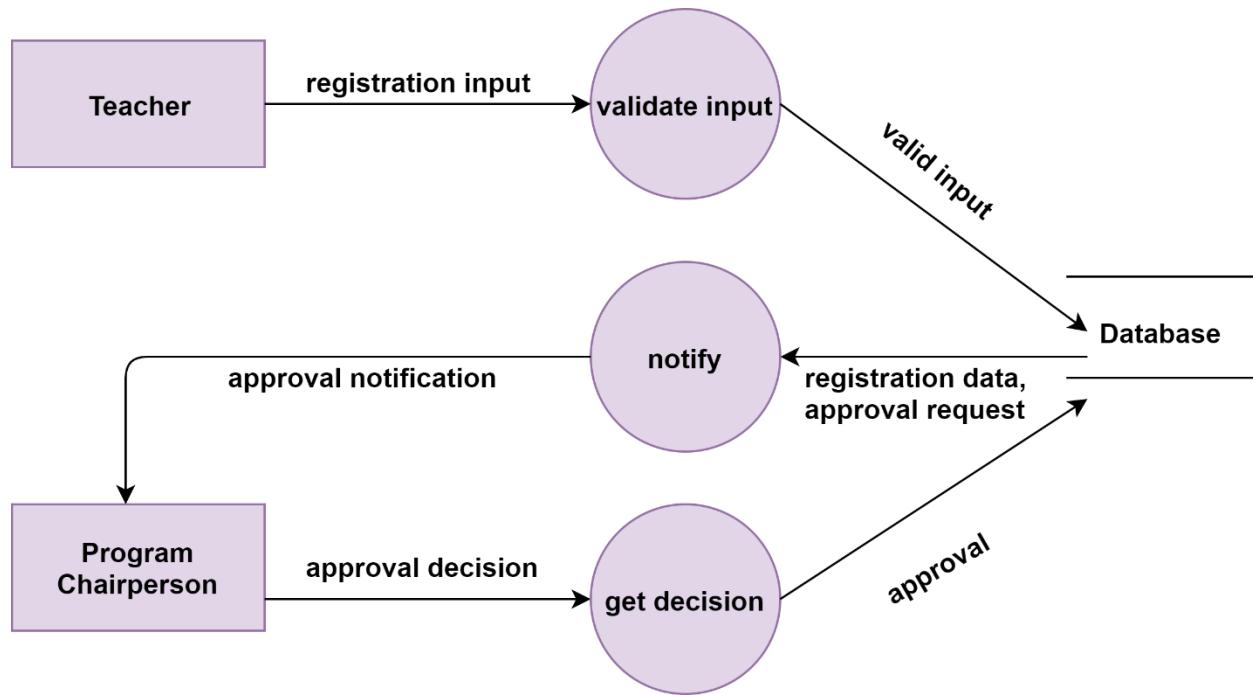
**Figure 7: Level-1 DFD**

**Level-2 DFD (Authentication)**

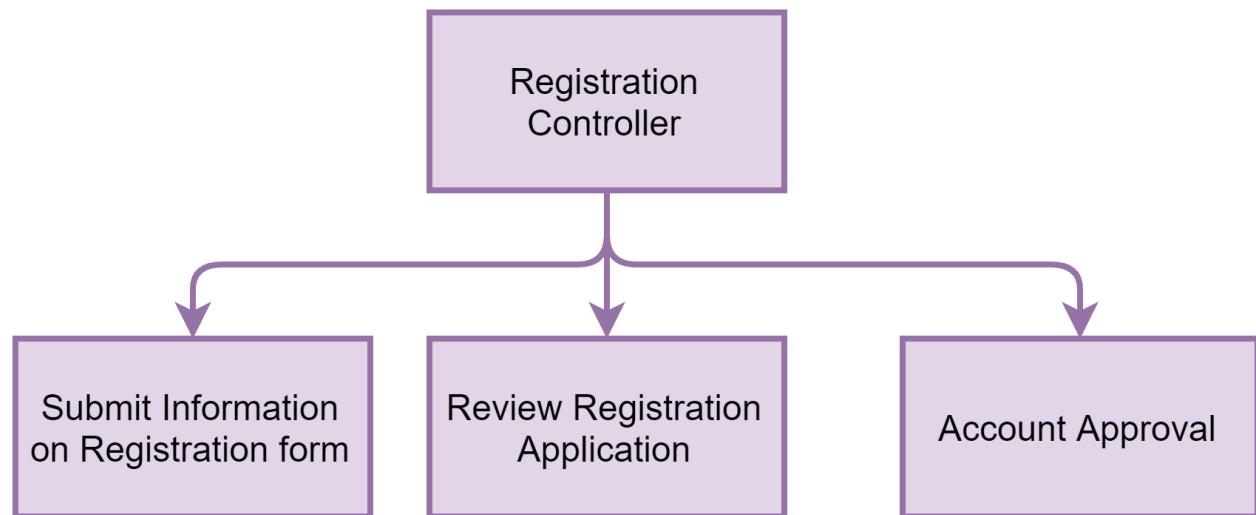


**Figure 8: Level-2(Authentication) DFD**

### Level-2.1 Registration Process



**Figure 9: DFD of Registration process**



**Figure 10: Mapping of registration process**

## Level 2.2 Password Recovery

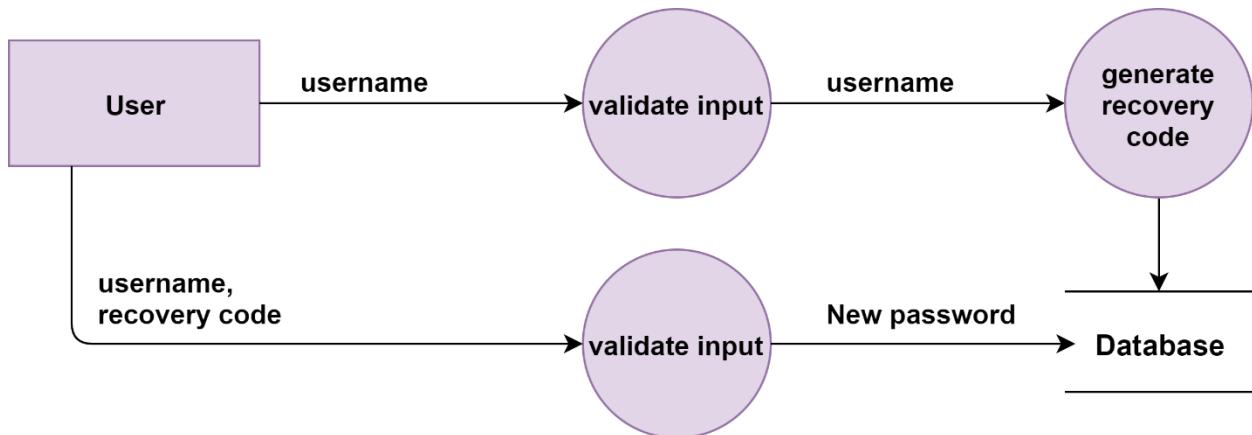


Figure 11: DFD of password recovery

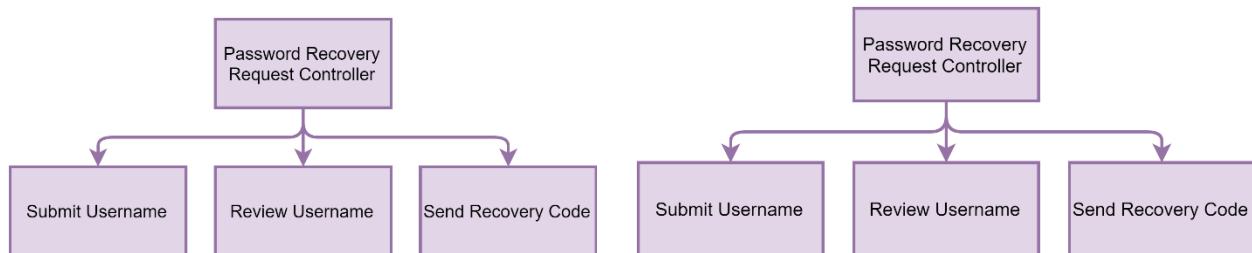


Figure 12: Mapping of password recovery

### Level-2.3 Sign in Process

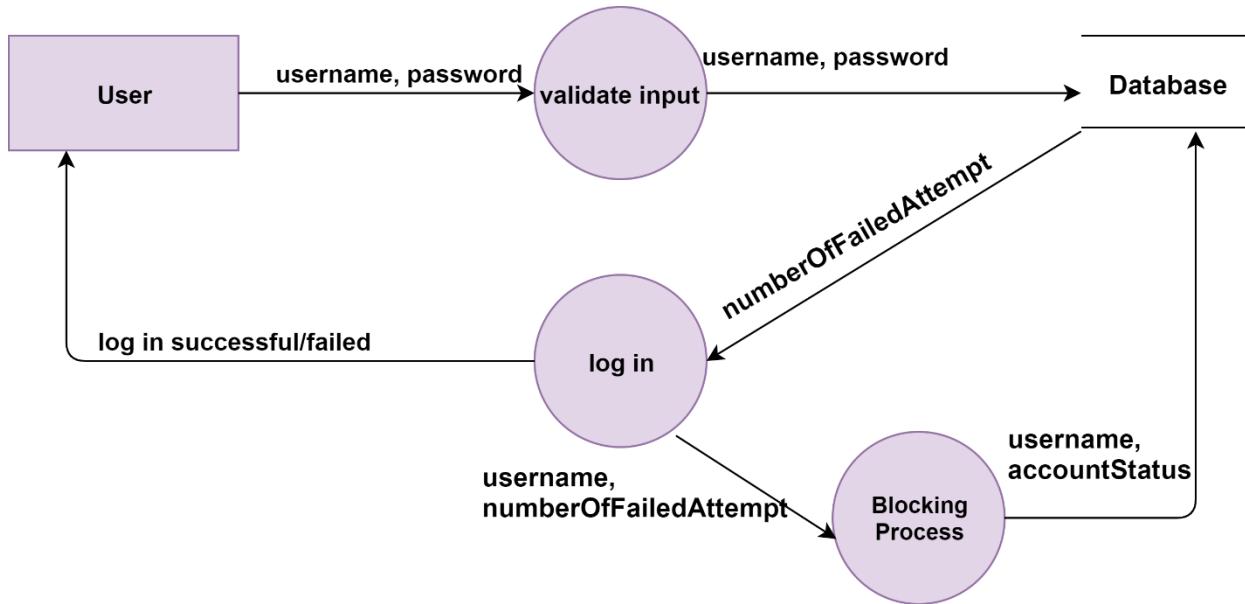


Figure 13: DFD of Sign in

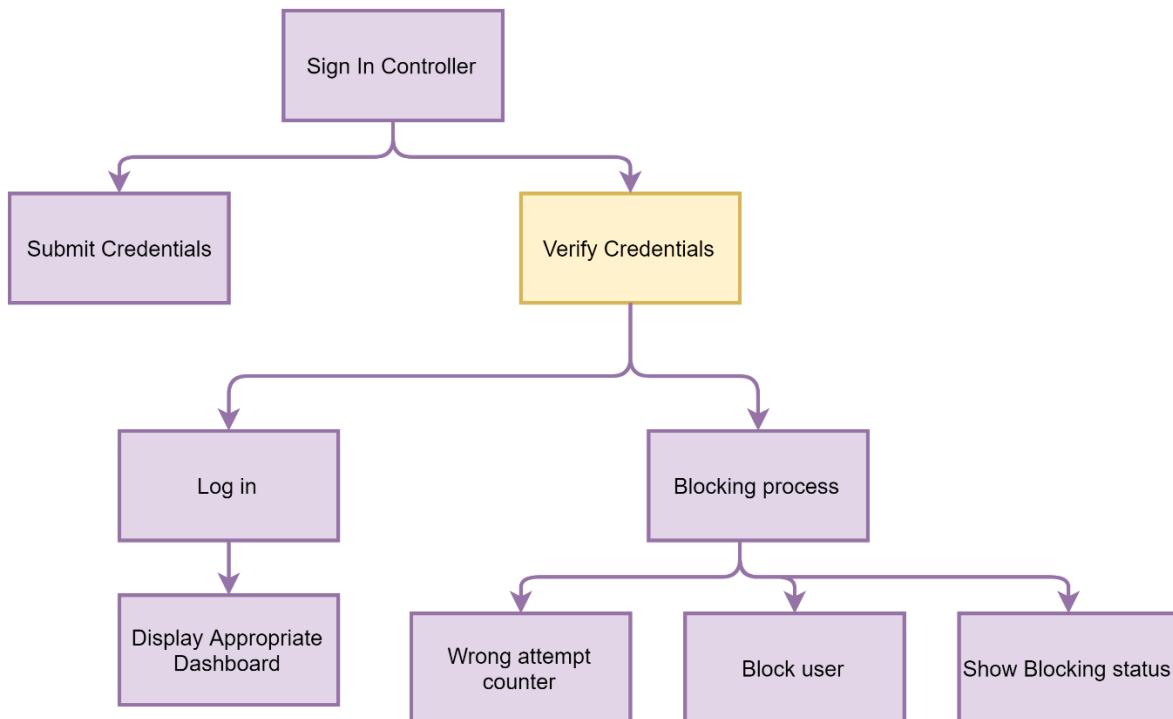


Figure 14: Transaction mapping of sign in

#### Level-2.4 Admission administration

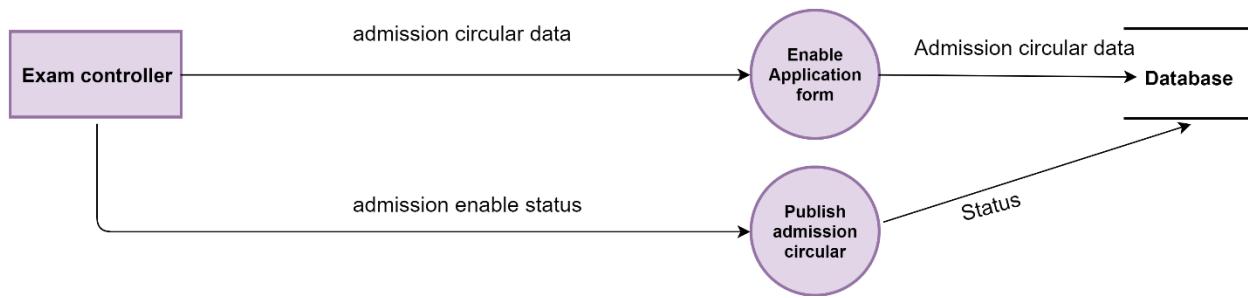


Figure 15: DFD of Admission administration

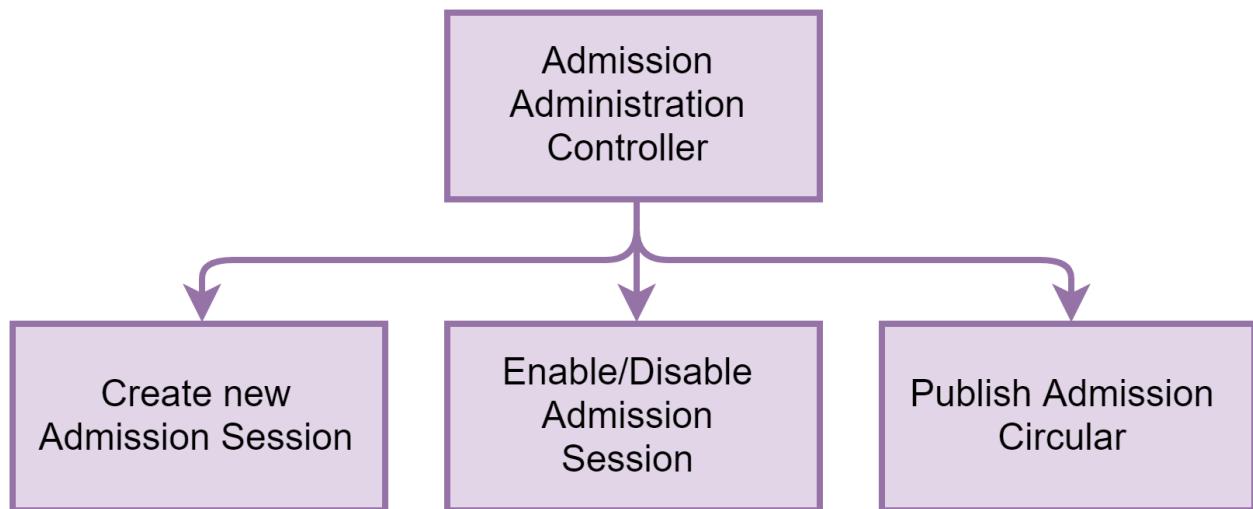


Figure 16: Mapping of admission administration

### Level-2.5 Application payment

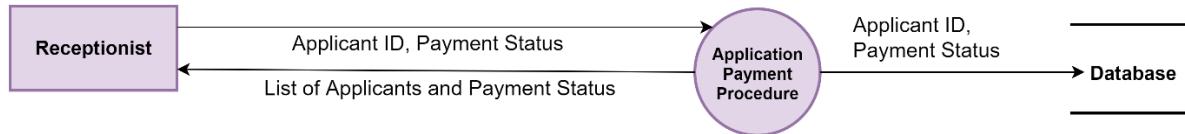


Figure 17: DFD of application payment

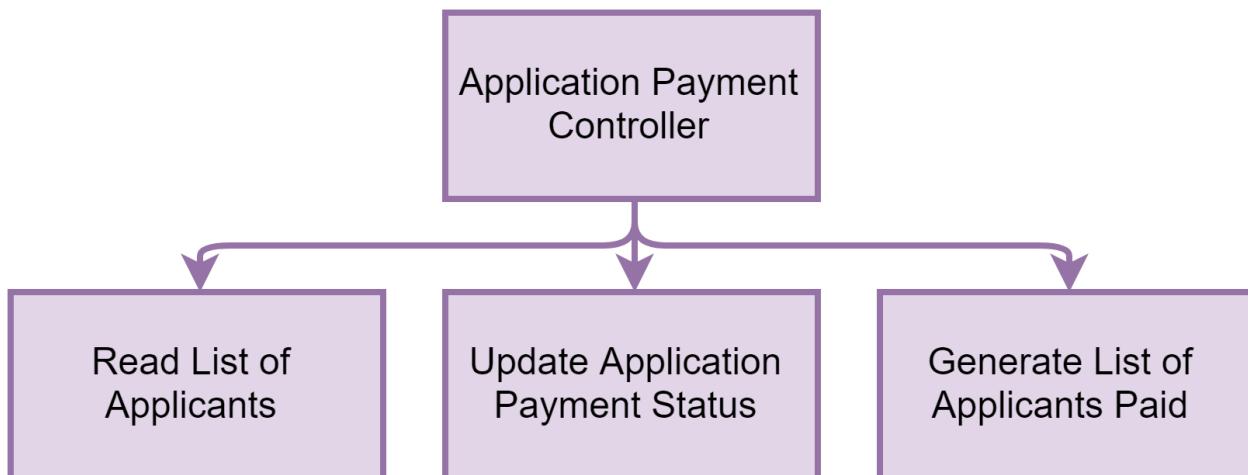


Figure 18: Mapping of application payment

## Level-2.6 Scrutinizing process

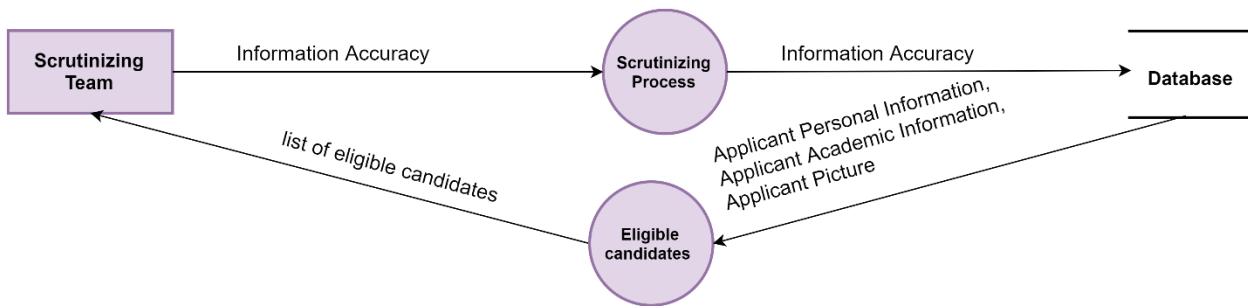


Figure 19: DFD of scrutinizing process

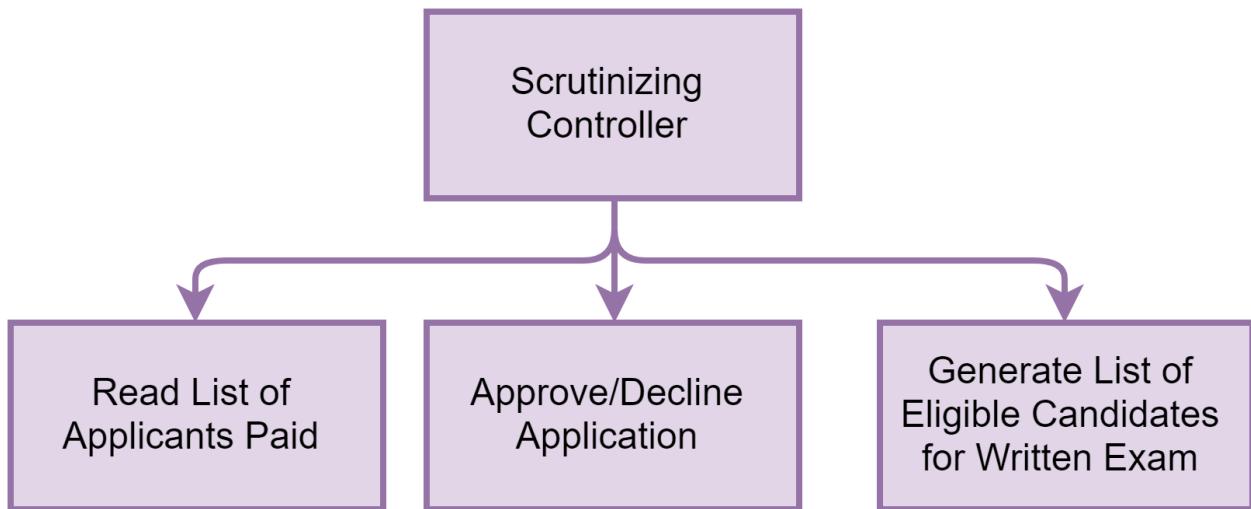


Figure 20: Mapping of scrutinizing process

### Level-2.6 Seat plan generation process

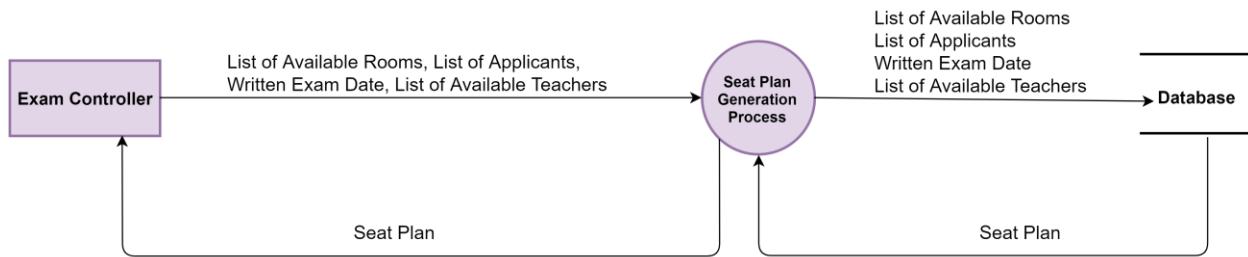


Figure 21: DFD of seat plan generation process

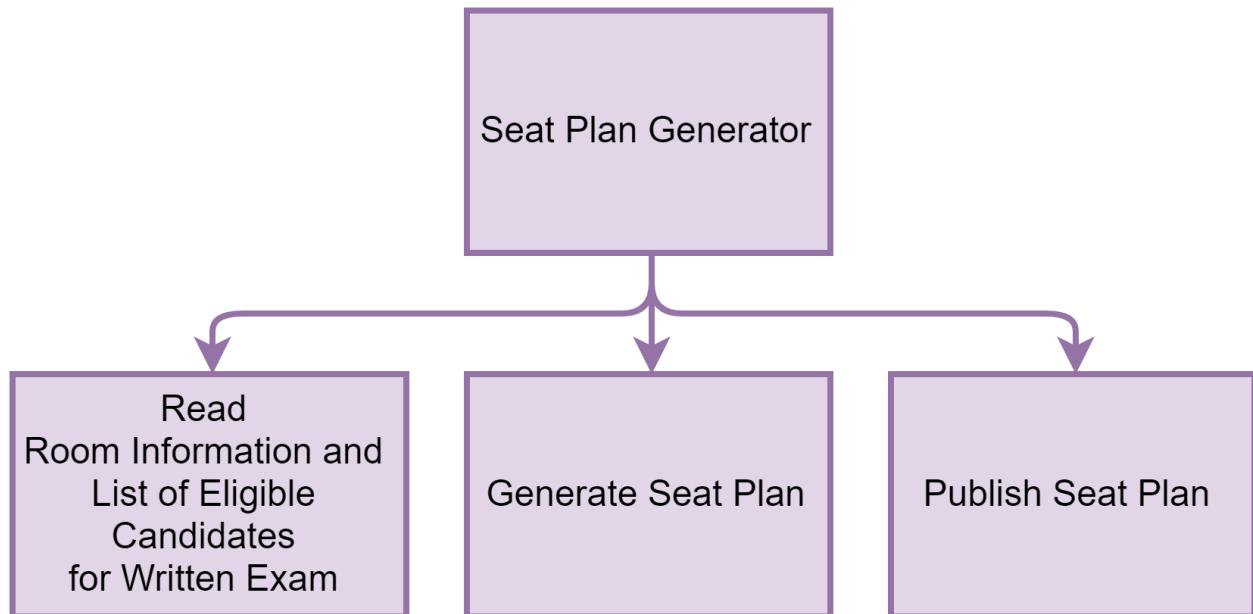


Figure 22: Mapping of seat plan generator

### Level-2.7 Written mark process

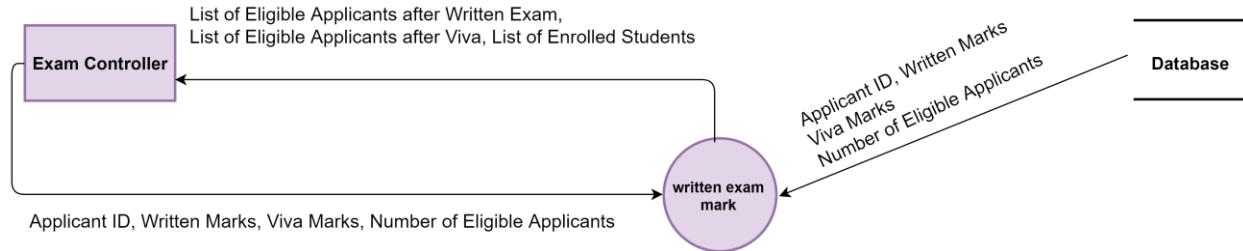


Figure 23: Written mark process

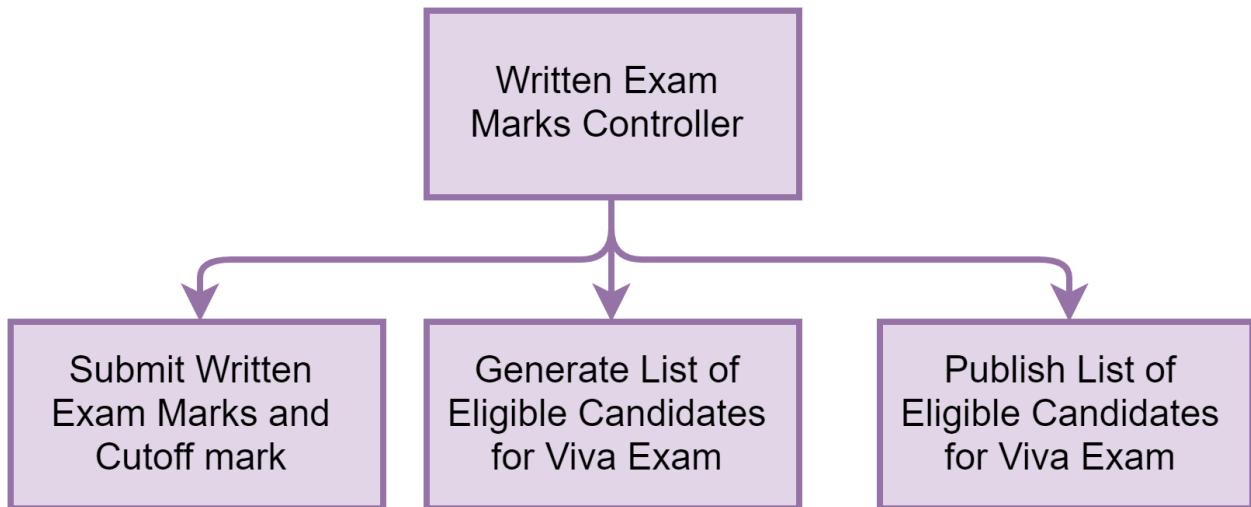


Figure 24: Mapping of written exam mark process

### Level-2.7 Viva schedule generation

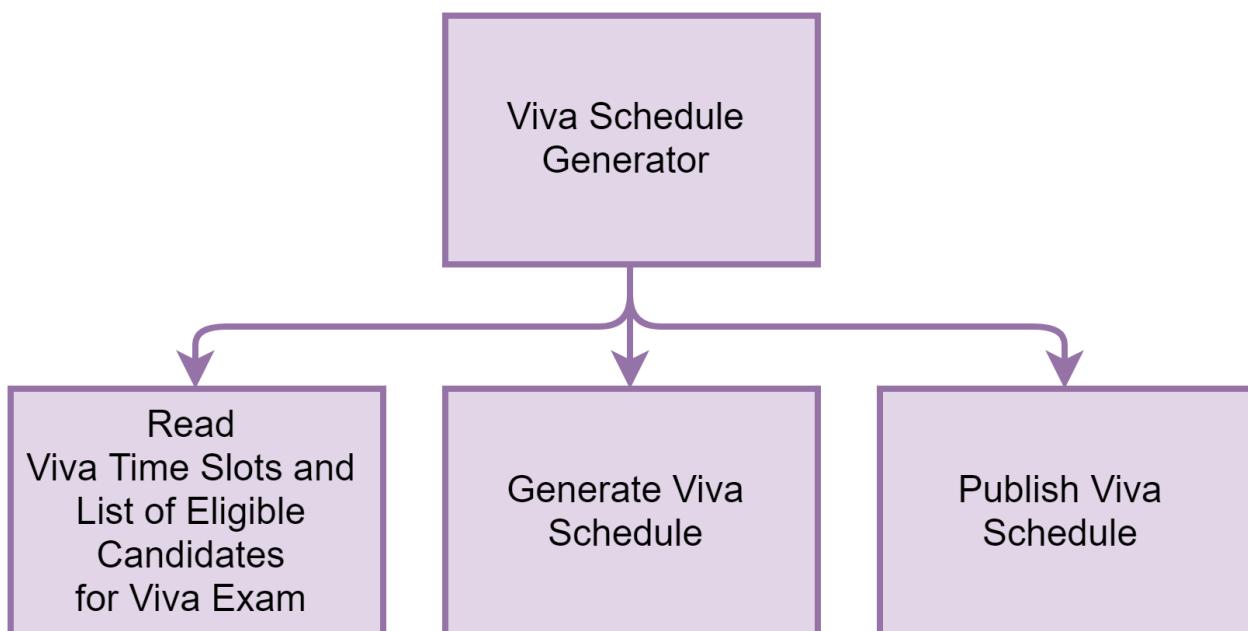
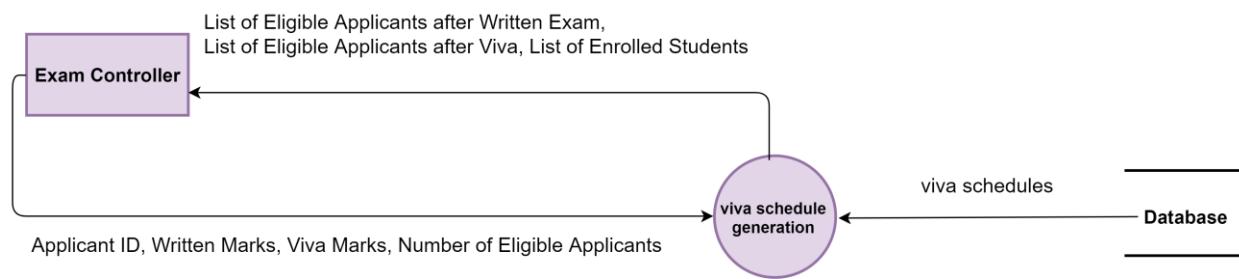


Figure 25: Mapping of viva schedule generator

### Level-2.8 Admission payment process

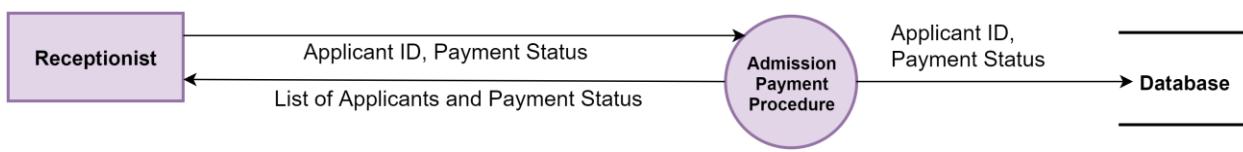


Figure 26: DFD of admission payment process

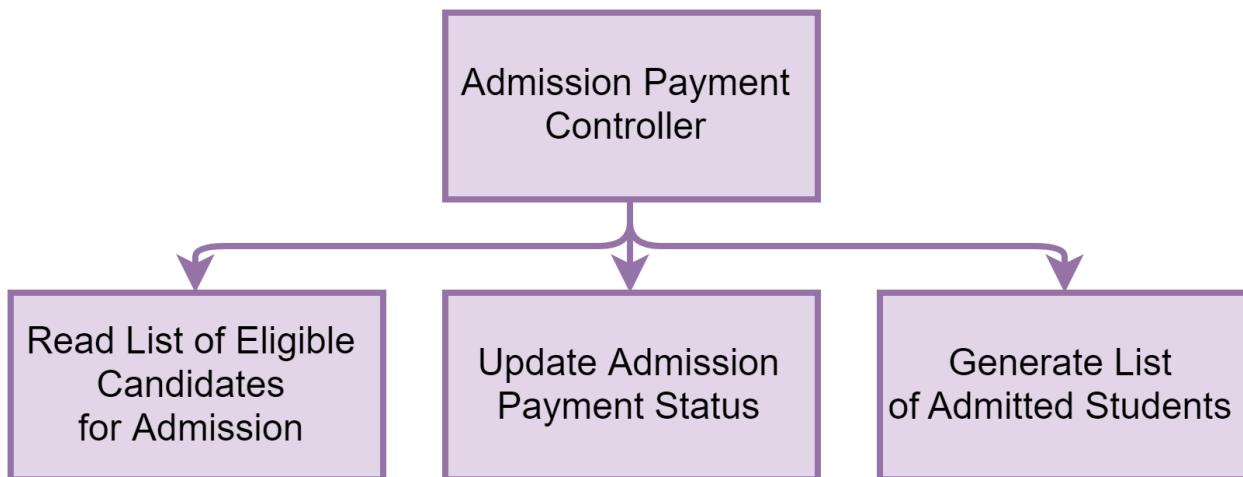


Figure 27: Mapping of admission payment

### Level-2.9 Course result generation process

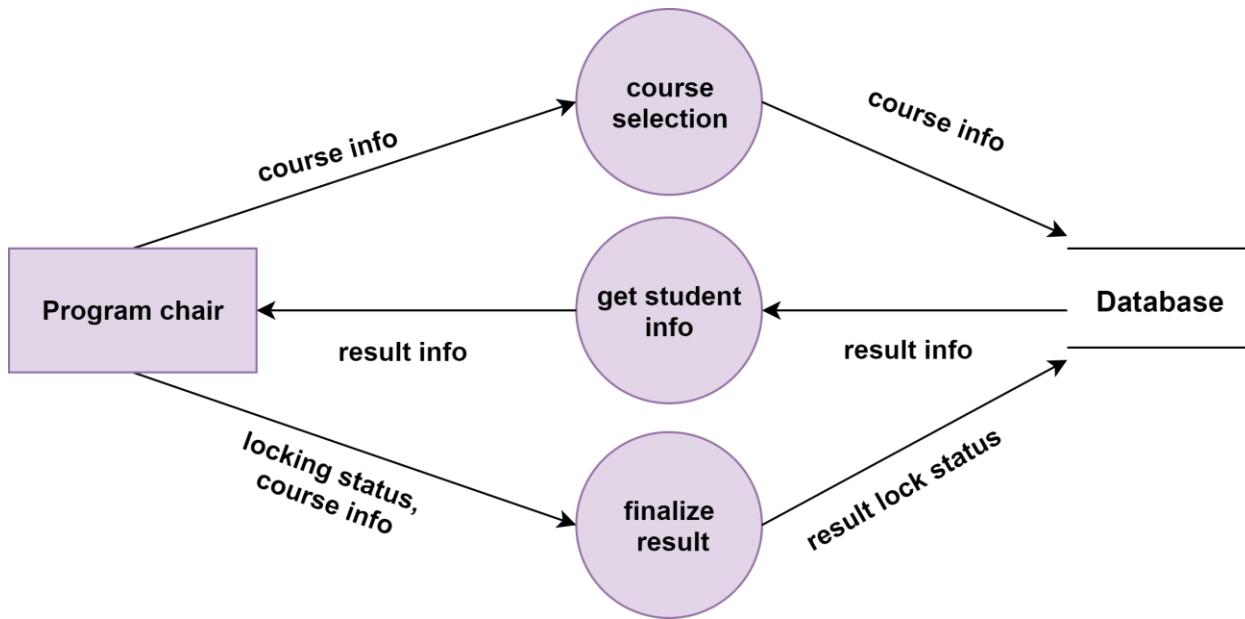


Figure 28: DFD of course result generation process

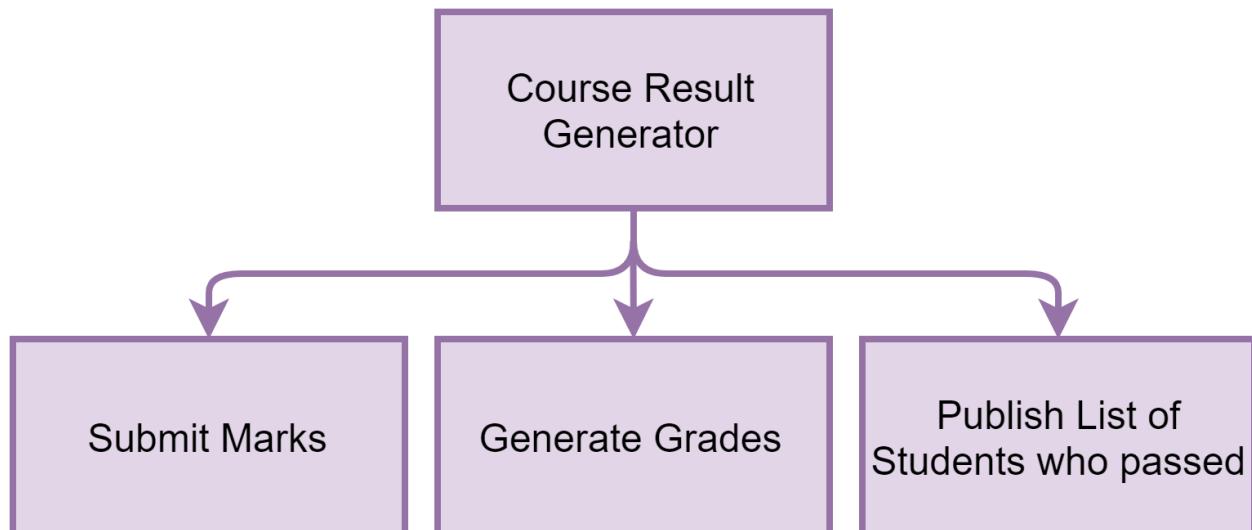


Figure 29: Mapping of result generation process

### Level-2.10 Result publication process

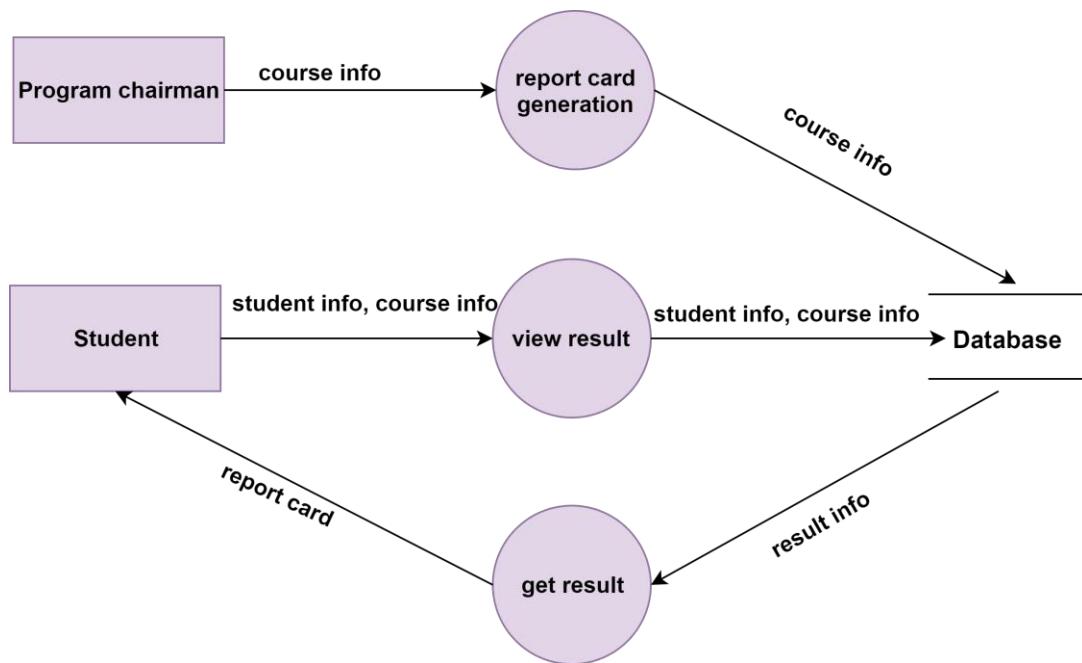


Figure 30: DFD of result publication process

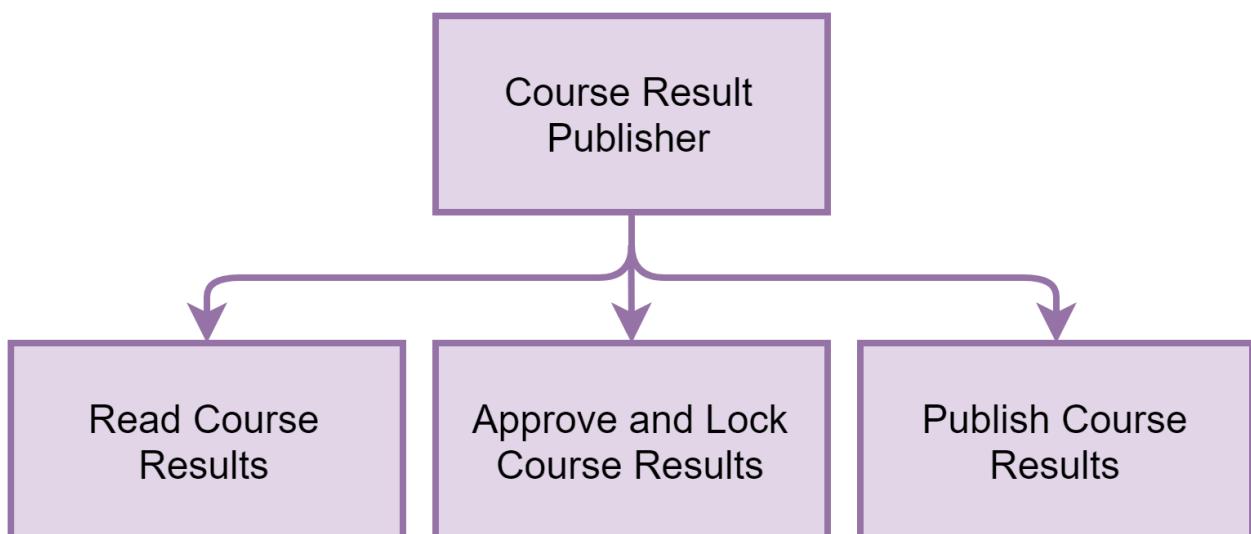


Figure 31: Mapping of course result publisher

## CHAPTER 2: COMPONENT DESIGN

A component-level design can be represented using some intermediate representation (e.g. graphical, tabular, or text-based) that can be translated into source code.

### 2.1 Design classes that correspond to the problem domain

User

Receptionist

Student

Scrutinizing Team

Teacher

Program Chairperson

Exam Controller

Applicant

### 2.2 Design classes that correspond to the infrastructure domain

Database Operator

## 2.3 Class elaboration

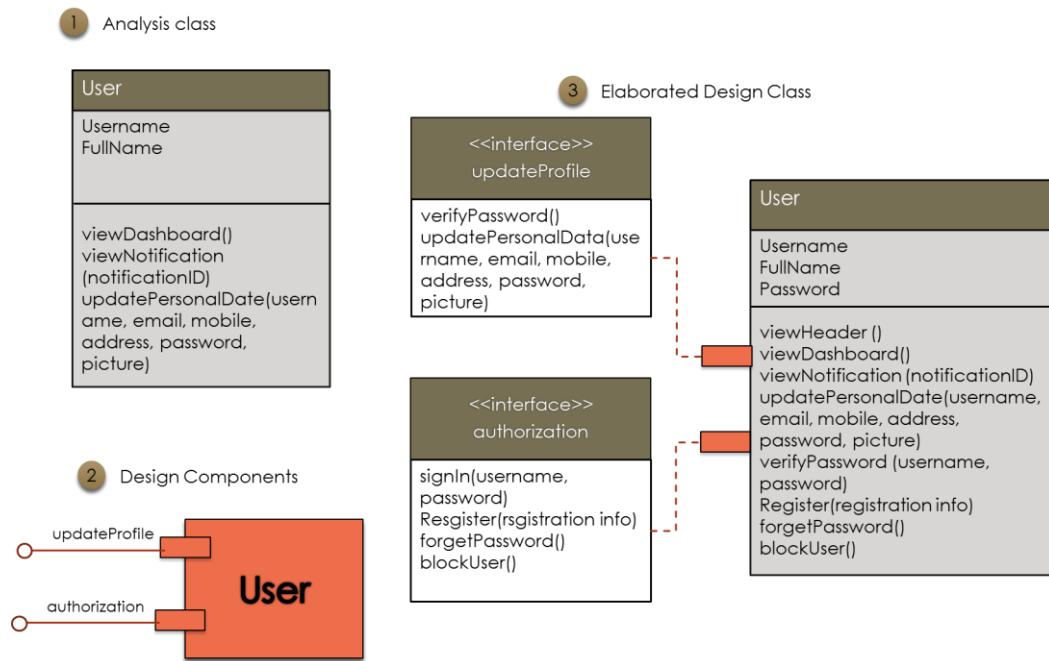


Figure 32: Class elaboration of User

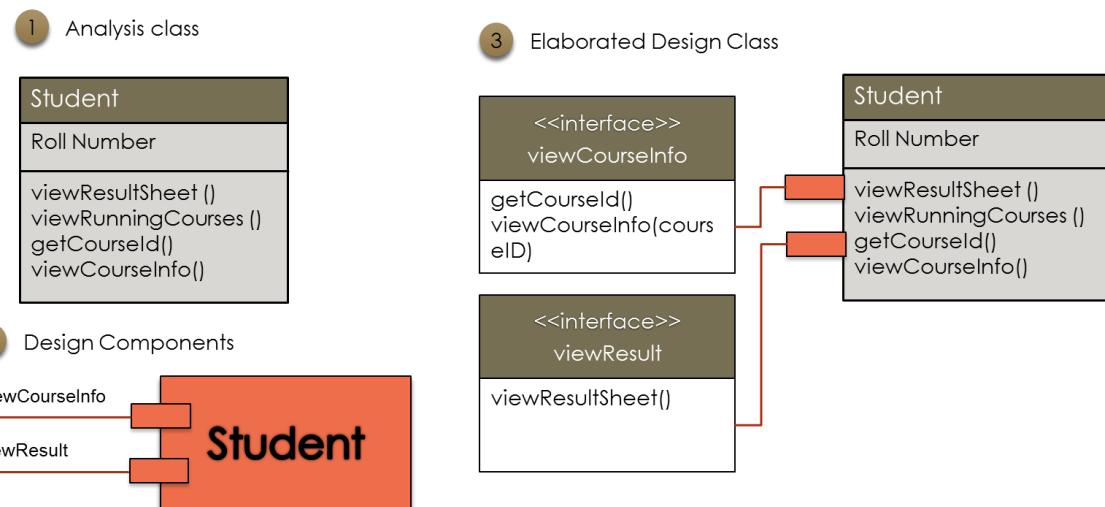
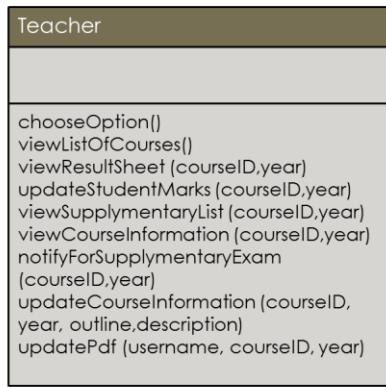
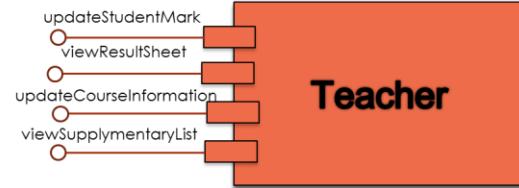


Figure 33: Class elaboration of Student

1 Analysis class



2 Design Components



3 Elaborated Design Class

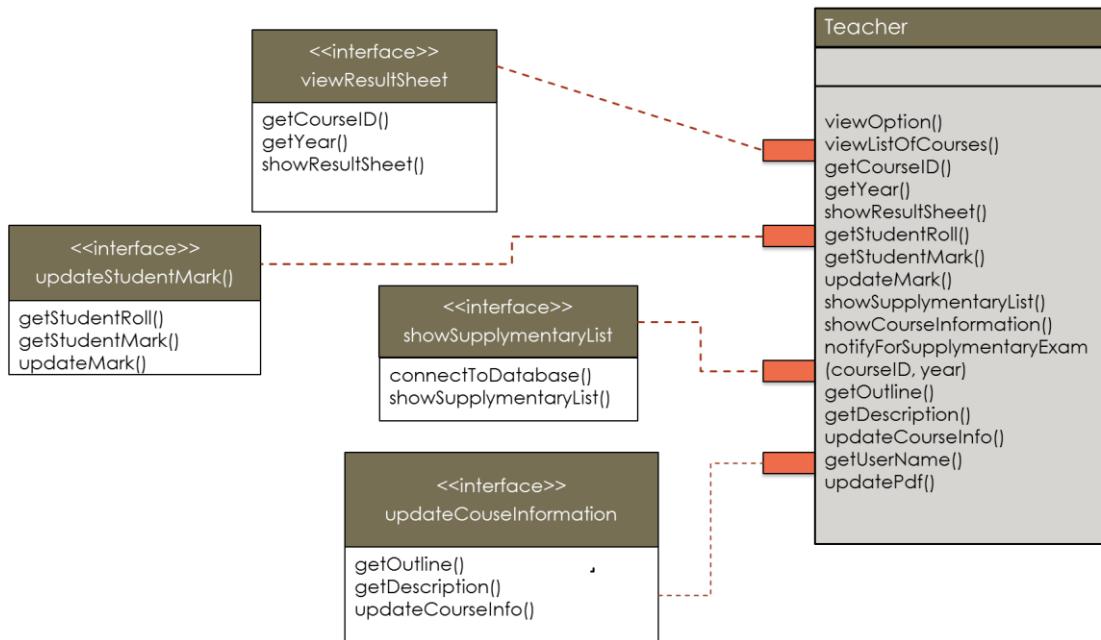
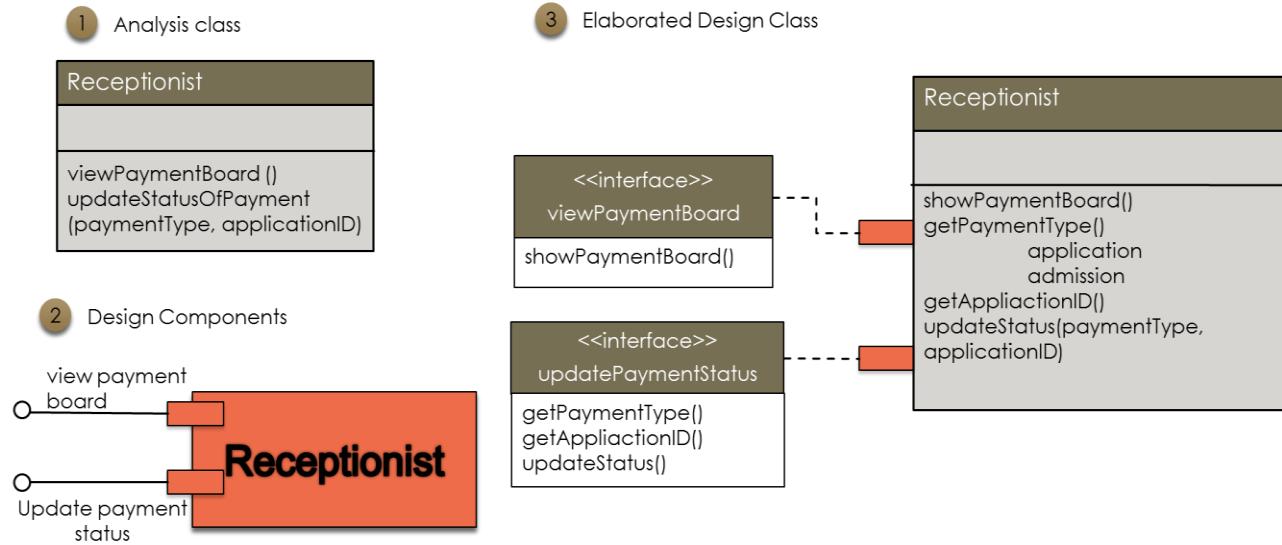
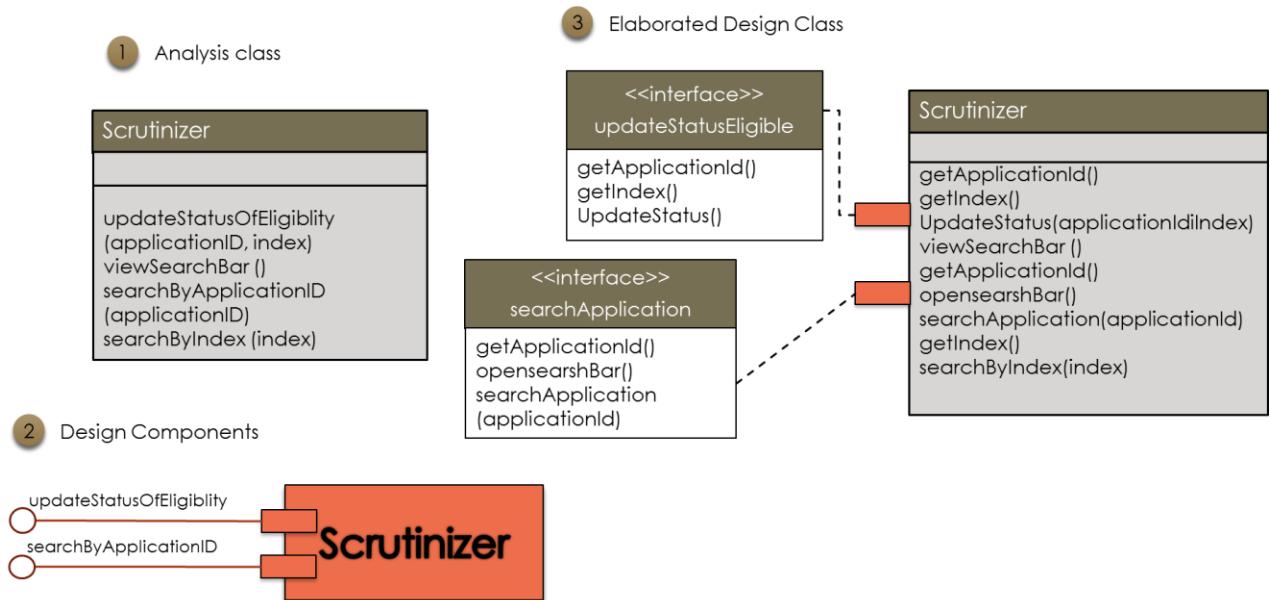


Figure 34: Class elaboration of Teacher

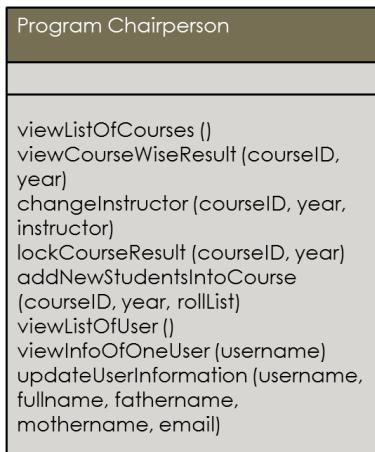


**Figure 35: Class elaboration of Receptionist**

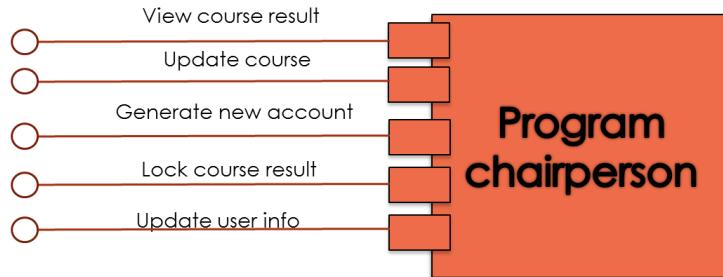


**Figure 36: Class elaboration of Scrutinizer**

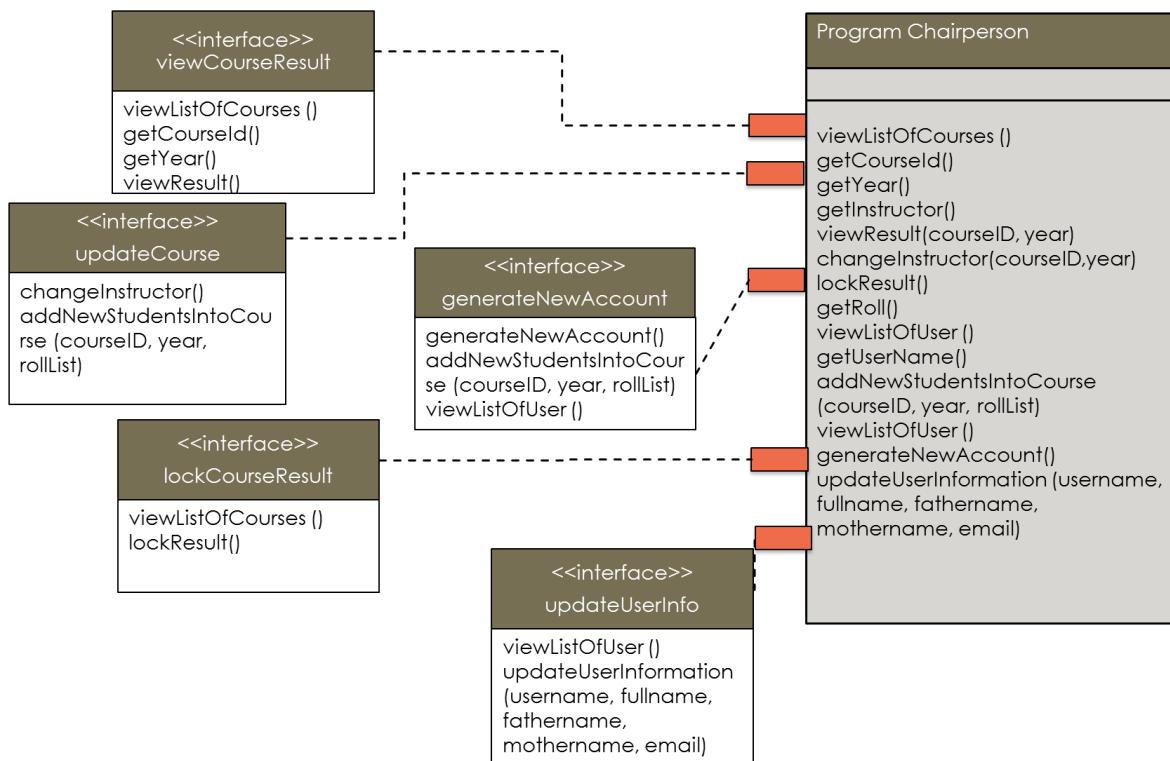
1 Analysis class



2 Design Components



3 Elaborated Design Class

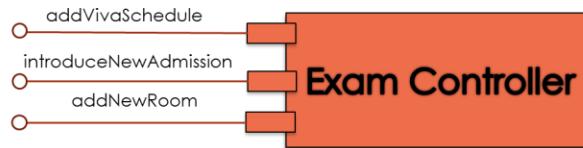


**Figure 37: Program Chairperson**

1 Analysis class

Exam Controller	
viewListOfRoom() updateRoomCapacity(roomno, capacity) addNewRoom(roomno, capacity) viewVivaSchedule() addVivaSchedule(usernameList, vivaDate, startTime, endTime, roomNo) viewListOfApplicants() introduceNewAdmission(AdmissionYear, FormFillUpStartAt, FormFillUpEndAt, ApplicationPaymentStartAt, ApplicationPaymentEndAt, ScrutinizingStartAt, AdmissionDate, AdmissionPaymentStartAt, AdmissionPaymentEndAt, AdmissionPaymentStartAt, AdmissionPaymentEndAt) viewAdmissionPanel() viewListOfEligibleCandidatesOnScrutinize()	changeScrutinizingStatus() viewListOfEligibleCandidatesAfterScrutinize() updateAvailabilityOfAdmitCard() viewListOfEligibleCandidatesAfterWrittenExam() updateWrittenMarks(applicantList, markList) lockWrittenMarks(capacity) viewListOfEligibleCandidatesAfterVIVAExam() updateVivaMarks(applicantList, markList) lockVivaMarks(capacity) viewListOfEligibleCandidatesAfterFullResult() updateAdmissionInfo(FormFillUpStartAt, FormFillUpEndAt, ApplicationPaymentStartAt, ApplicationPaymentEndAt, ScrutinizingStartAt, AdmissionDate, AdmissionPaymentStartAt, AdmissionPaymentEndAt)

2 Design Components



3 Elaborated Design Class

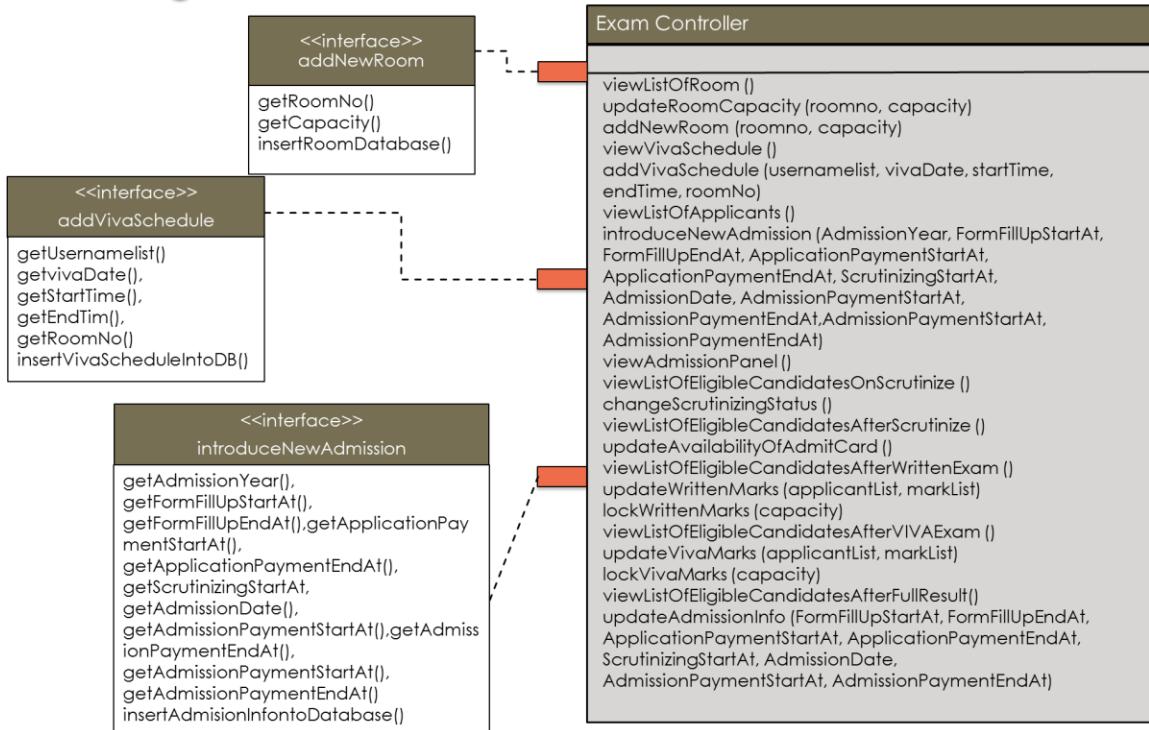


Figure 38: Class elaboration of Exam Controller

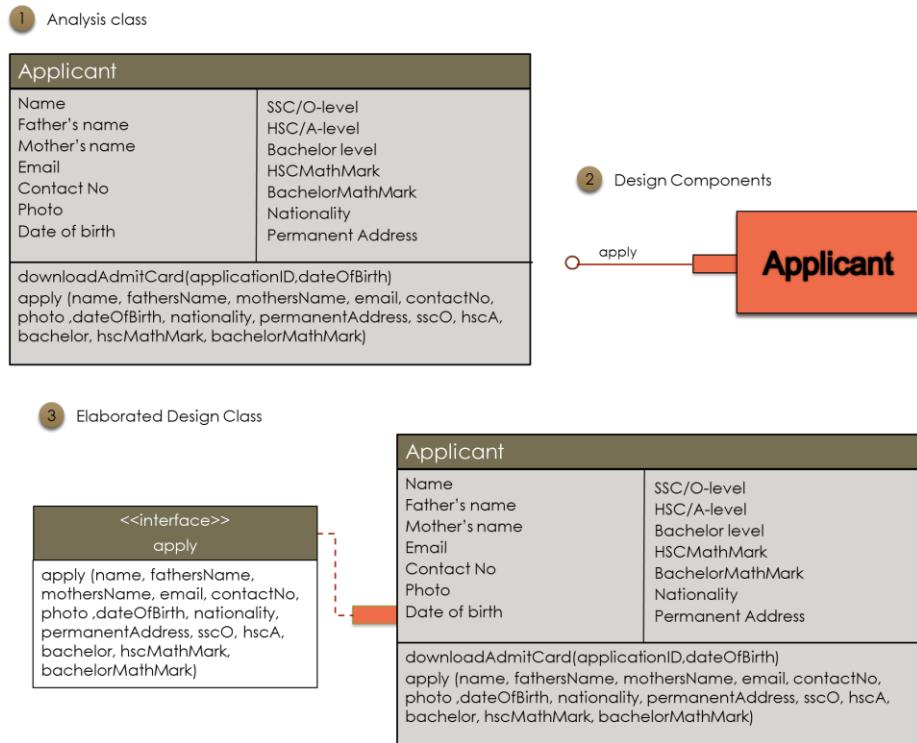


Figure 39: Class elaboration of Applicant

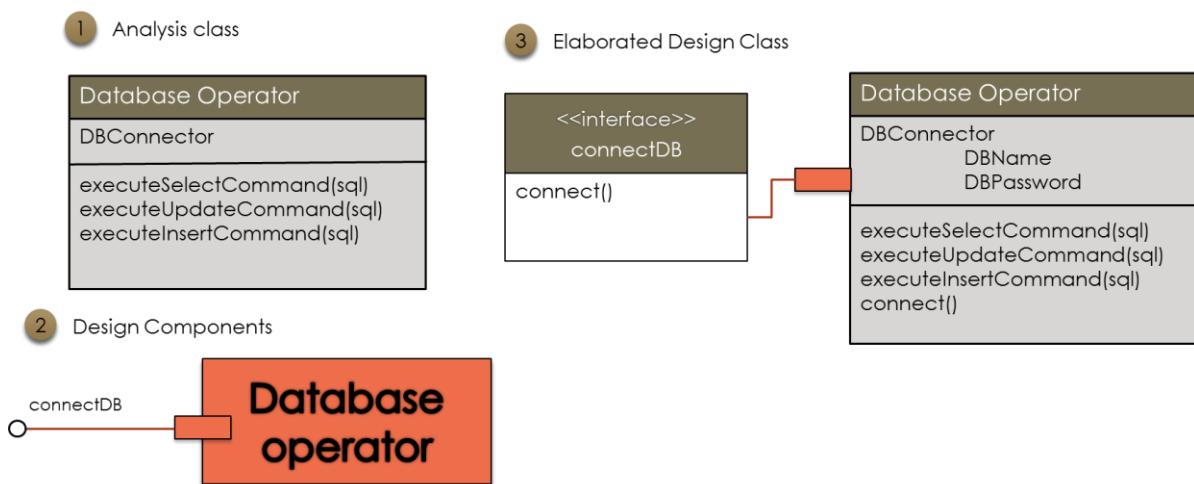


Figure 40: Class elaboration of Database operator

## 2.3 a) Class collaboration

### Sign in

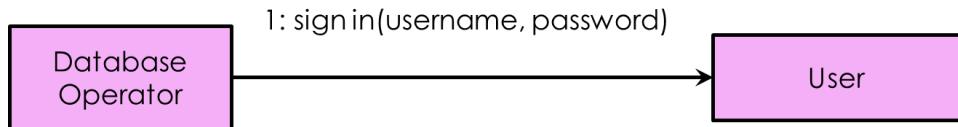


Figure 41: Class collaboration of authentication

### Update Profile

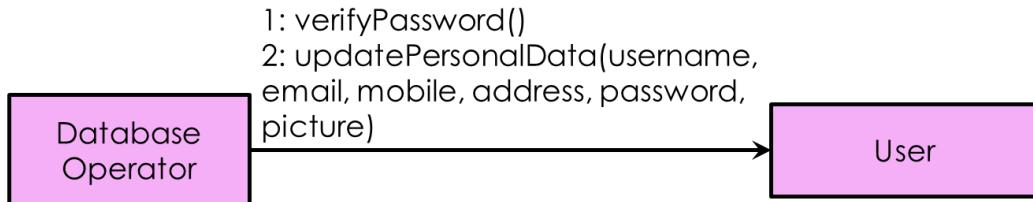


Figure 42: Class collaboration of update Profile

### New Admission Session

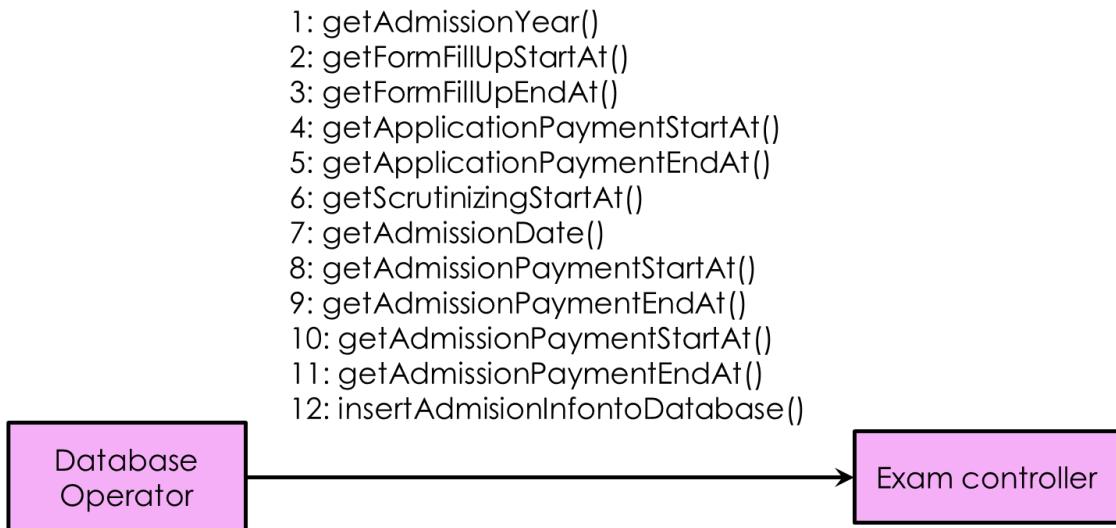


Figure 43: Class collaboration of new admission session

## Apply

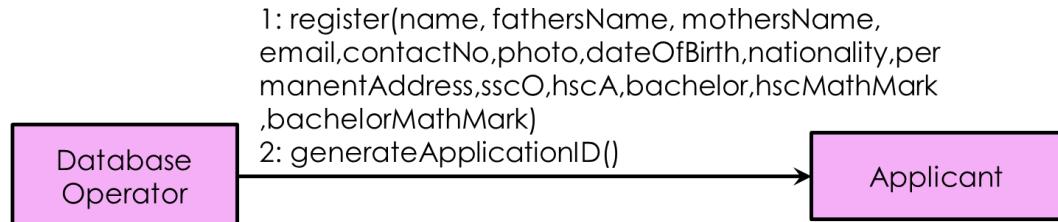


Figure 44: Class collaboration of Apply

## View Payment Board



Figure 45: Class collaboration of view payment board

## Update Payment Status

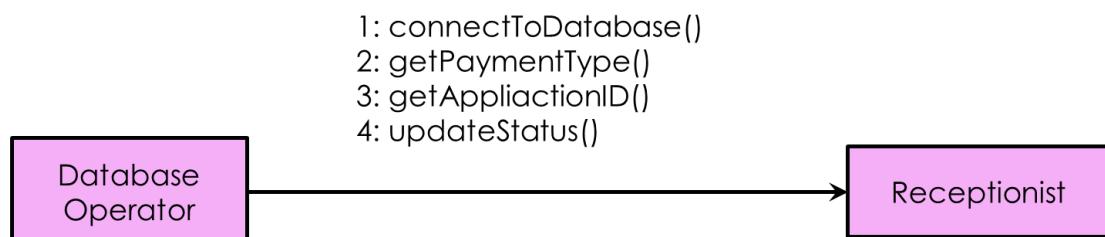
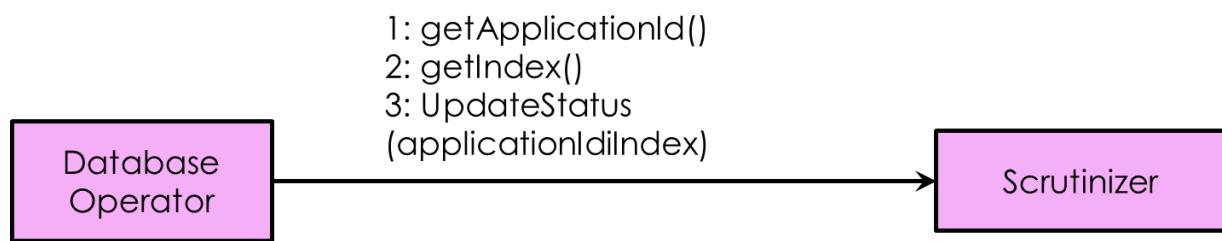


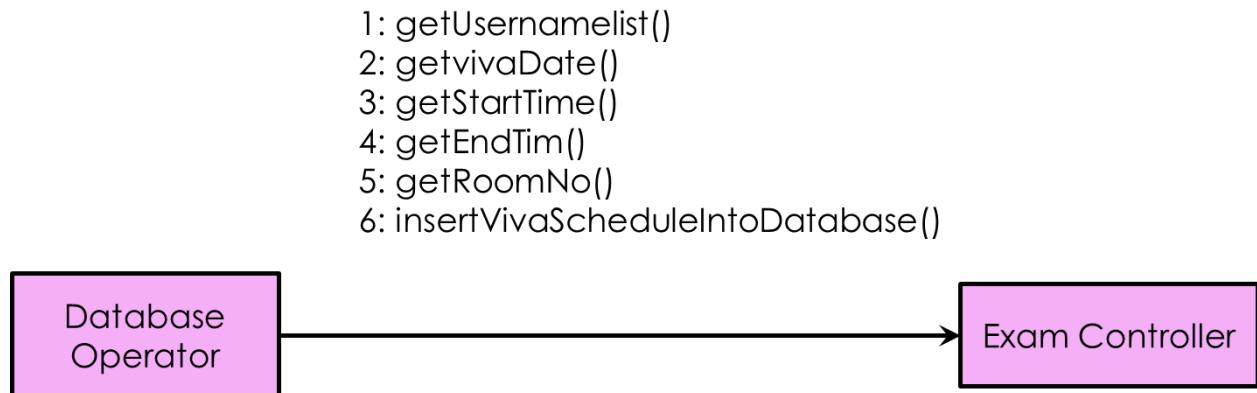
Figure 46: Class collaboration of update payment status

## Update Status of Eligibility



**Figure 47: Class collaboration of update status of eligibility**

## Add Viva Schedule



**Figure 48: Class collaboration of add viva schedule**

## Add new room

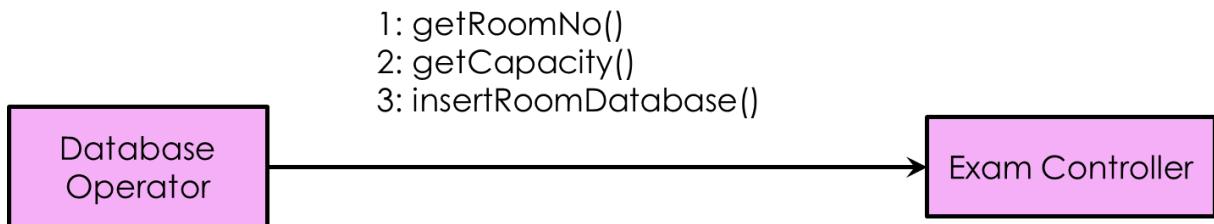


Figure 49: Class collaboration of add new room

## Result submission

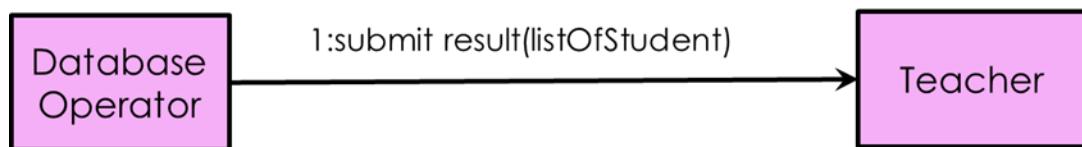


Figure 50: Class collaboration of result submission

## View Result

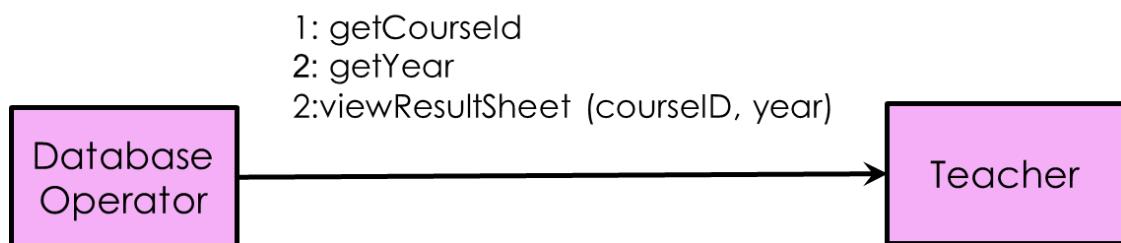


Figure 51: Class collaboration of view result

## Update Result

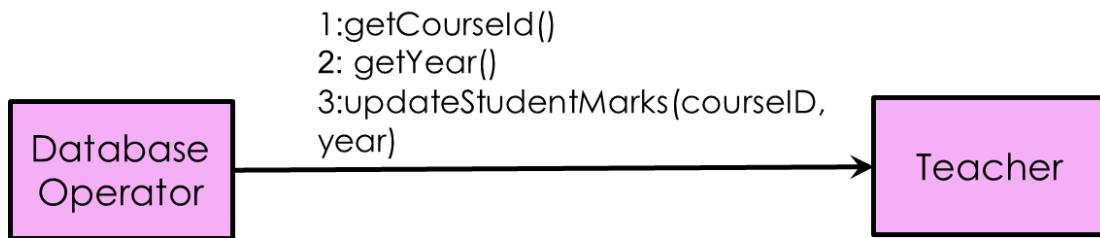


Figure 52: Class collaboration of update result

## View SupplementaryList

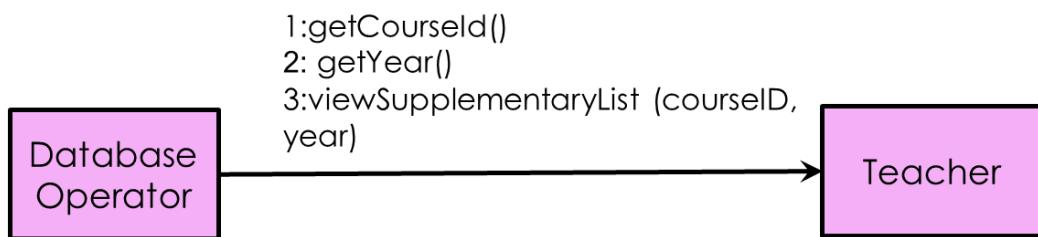


Figure 53: Class collaboration of view supplementary list

## View Course Result

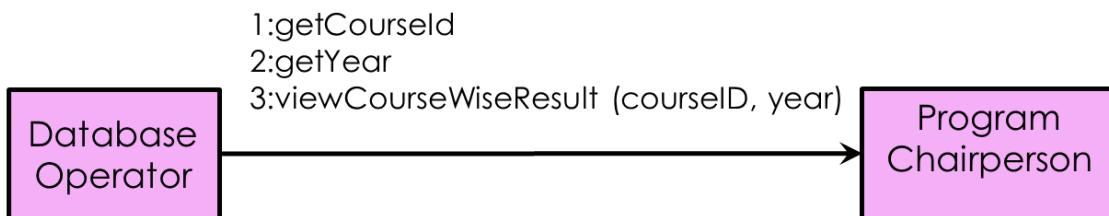


Figure 54: Class collaboration f view course result

## 2.3 b) Method call sequence for individual modules

### Authentication



Figure 55: Method call sequence of Authentication

### Teacher Registration

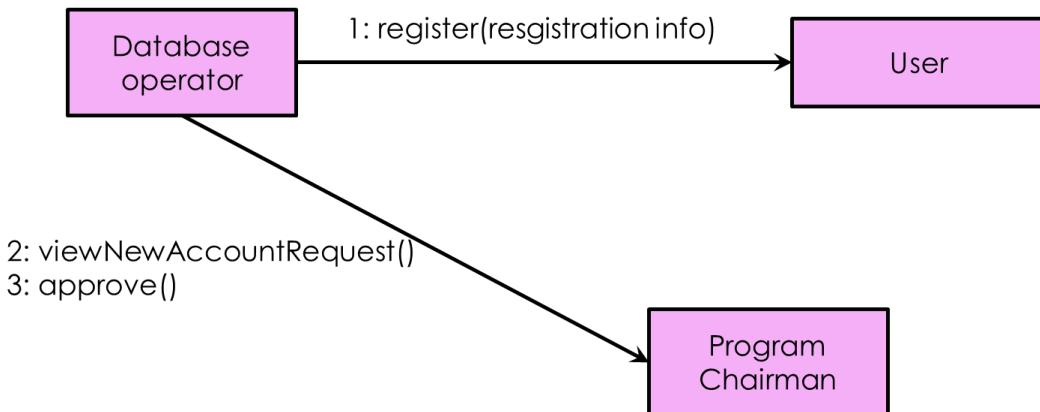


Figure 56: Method call sequence of Teacher registration

## Student Account Creation



Figure 57: Method call sequence of Student account creation

## Candidate Management

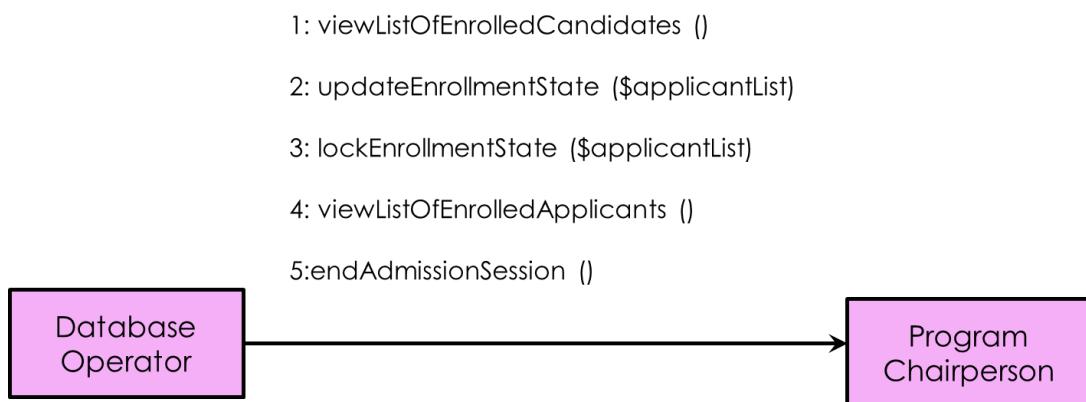


Figure 58: Method call sequence of Candidate management

## Course Management

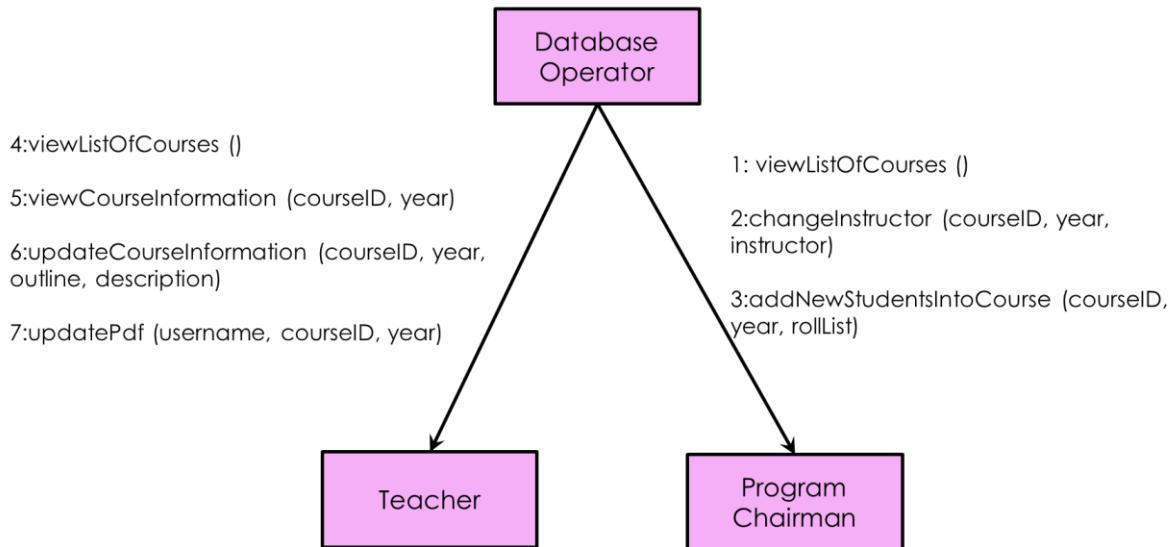


Figure 59: Method call sequence of Course management

## Result Management

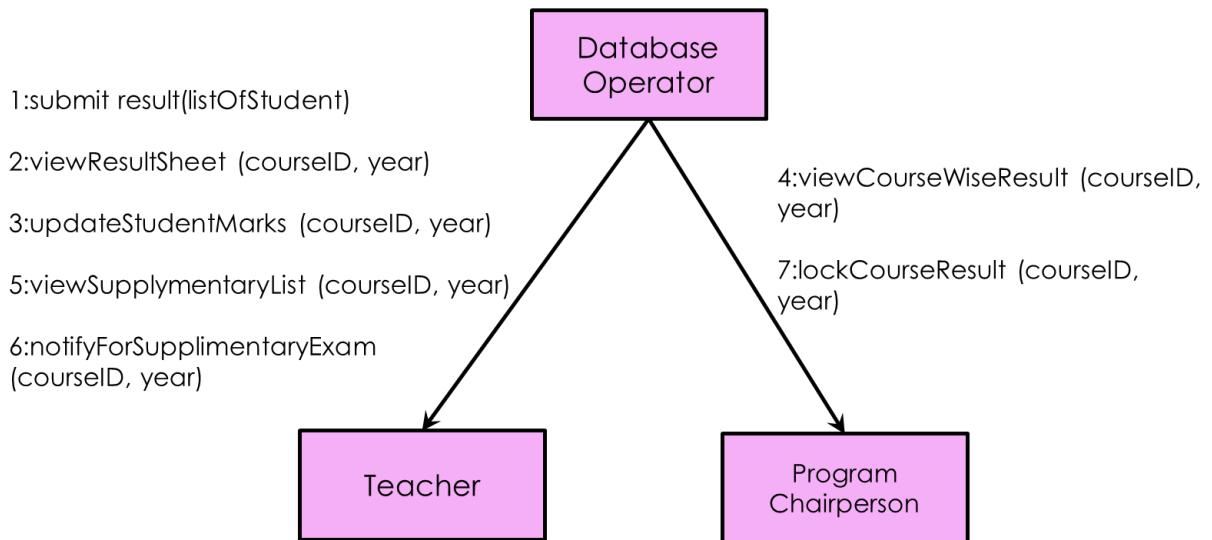
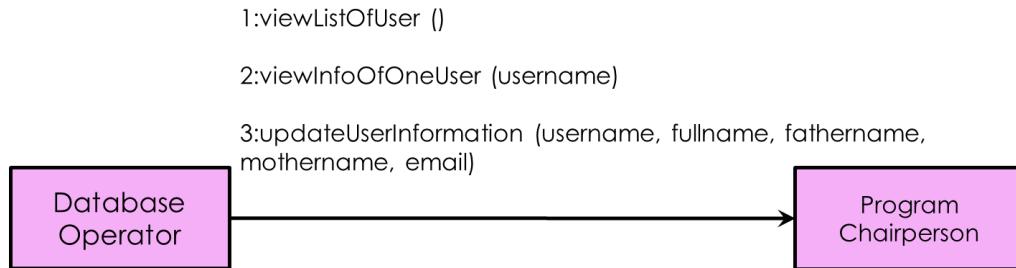


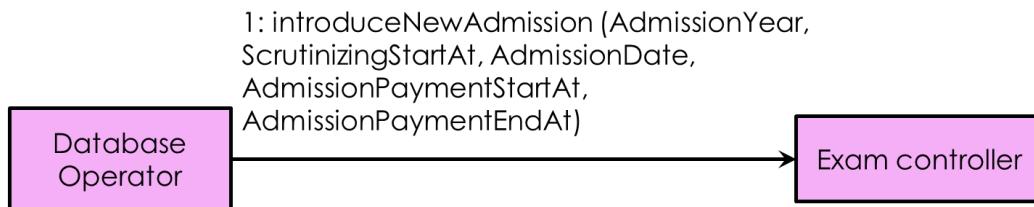
Figure 60: Method call sequence of Result management

## User Management



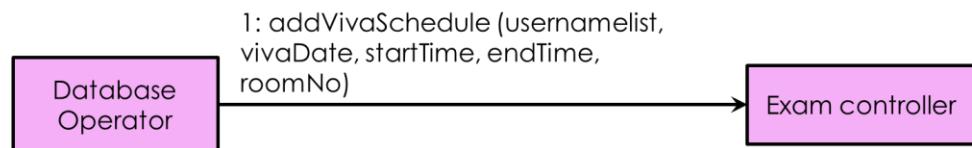
**Figure 61: Method call sequence of User management**

## Admission Session



**Figure 62: Method call sequence of Admission session**

## Exam Schedule Manage



**Figure 63: Method call sequence of Exam schedule manage**

## Result Manage

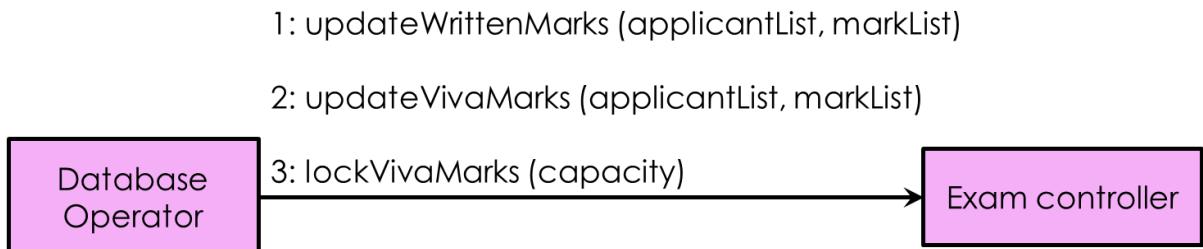


Figure 64: Method call sequence of Result manage

## Application Processing

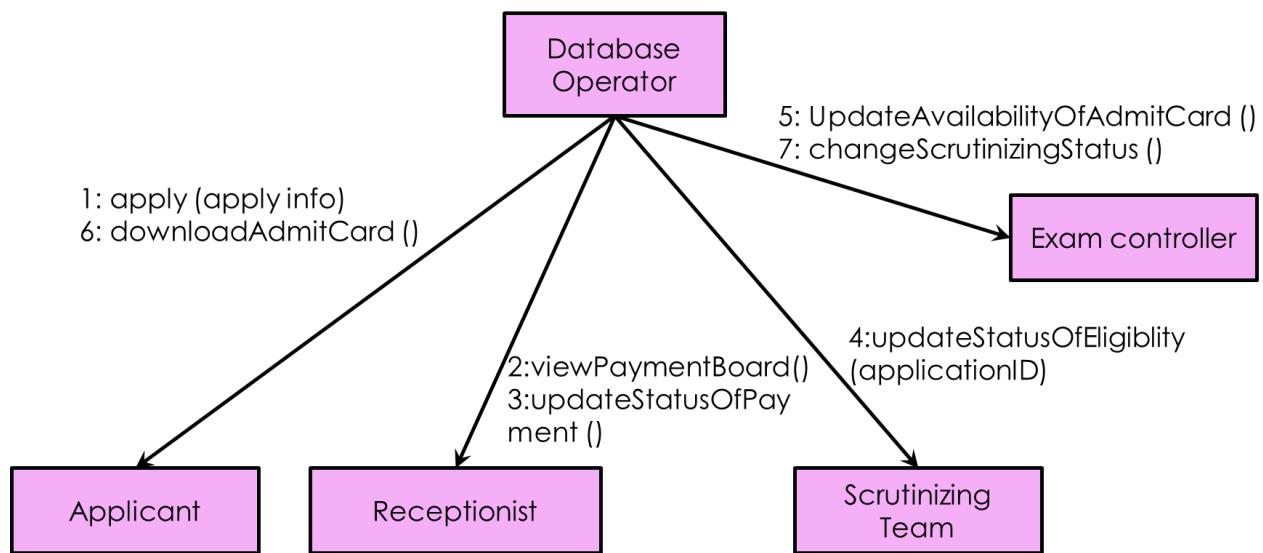


Figure 65: Method call sequence of Application processing

## 2.3 c) Appropriate Interfaces

There is no necessity to divide the classes in subclasses as they exhibit sufficient cohesion. So there is no need to define appropriate interfaces.

## 2.3 d) Elaborate Attributes

1. Username: string = null { contains name value 20 character max}
2. EmailAddress: string = null { contains email – xyz@gmail.com}
3. Password: string = null {4<=number of character<=20}
4. MobileNumber: string = null { contains number value 11 character max}
5. Name: string = null { contains name value 50 character max}
6. RollNumber: int = null {10 character max}
7. DBName: string = null {contains database name value 10 character max}
8. DBPassword: string = null {5<=number of characters<=10}
9. CourseID: int = null {5 digit}
10. Year: int = null {4 digit}
11. Outline: string = null {number of characters<=500}
12. Description:String=null {number of characters<=1000}
13. Username:String=null {number of characters<=15}
14. FullName:String=null {number of characters<=40}
15. FathersName:String=null {number of characters<=40}
16. MothersName : String=null {number of characters<=40}
17. Instructor : String=null {number of characters<=40}
18. Email : String=null {number of characters<=40}
19. RollList : int =null {List of students roll number}
20. Photo : String=null{filepath}
21. Nationality : String=null{number of characters<=15}
22. ContactNumber : int =null {15 digit}
23. PermanentAddress : String =null {number of characters<=100}
24. HscMathMark : int=null {3 digits}
25. BachelorMathMark : int=null {3 digit}
26. schoolName: String =null {number of characters<=100}

27. schoolBoard: String =null {number of characters<=100}  
28. schoolGroup: String =null {number of characters<=100}  
29. schoolYear: int = null {4 digit}  
30. schoolGPA: String =null {number of characters<=100}  
31. collegeName: String =null {number of characters<=100}  
32. collegeBoard: String =null {number of characters<=100}  
33. collegeGroup: String =null {number of characters<=100}  
34. collegeYear: int = null {4 digit}  
35. collegeGPA: String =null {number of characters<=100}  
36. universityName: String =null {number of characters<=100}  
37. departmentName: String =null {number of characters<=100}  
38. universityYear: int = null {4 digit}  
39. universityGPA: String =null {number of characters<=100}

## 2.3 e) Describe Processing Flow

### Processing flow of the methods: Teacher class

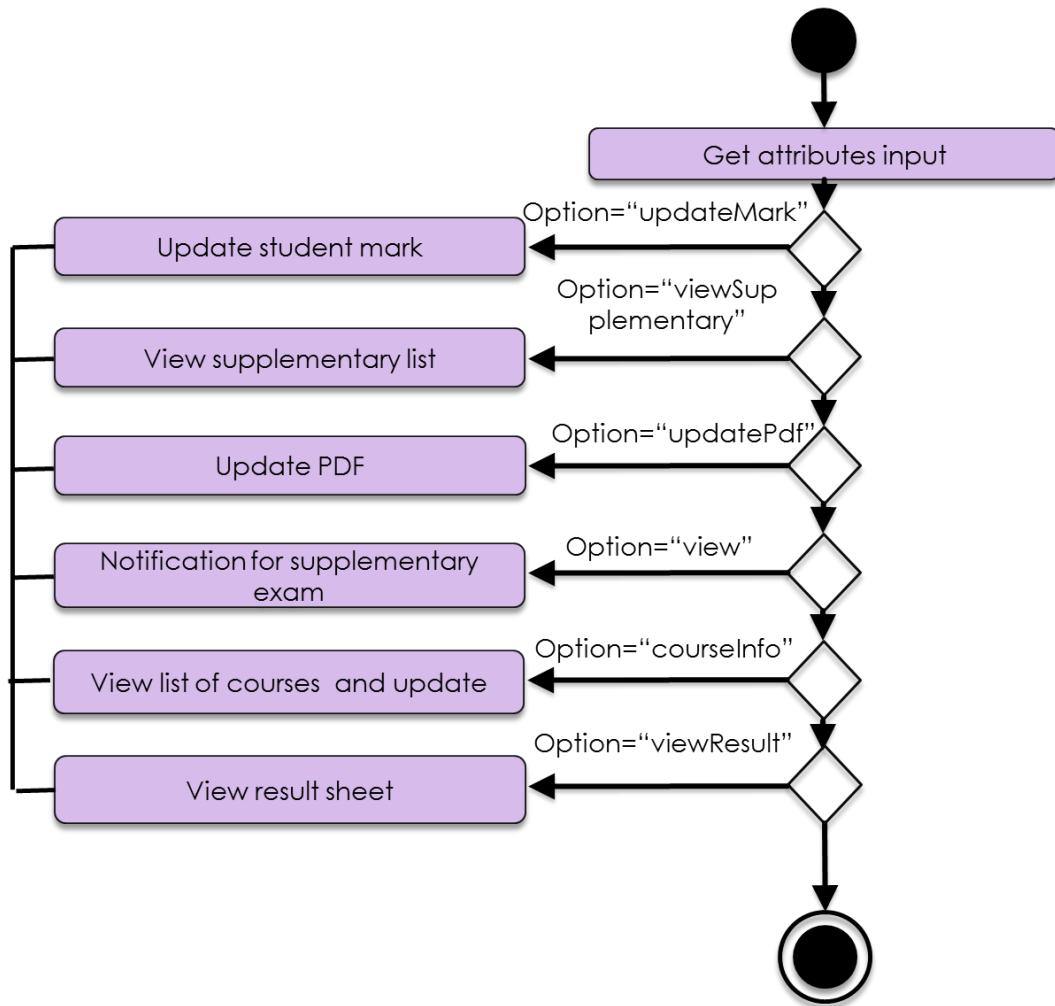
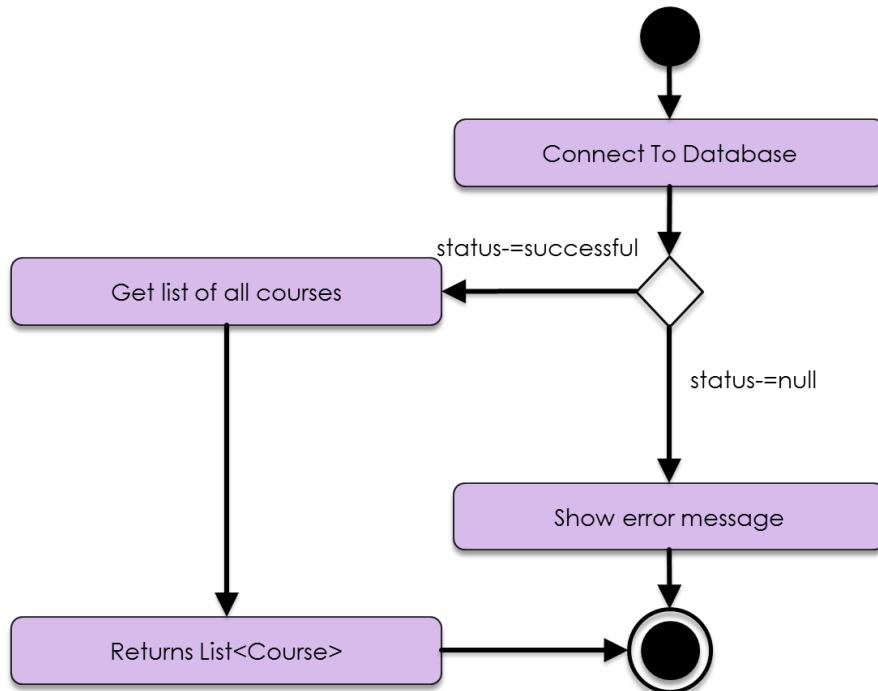
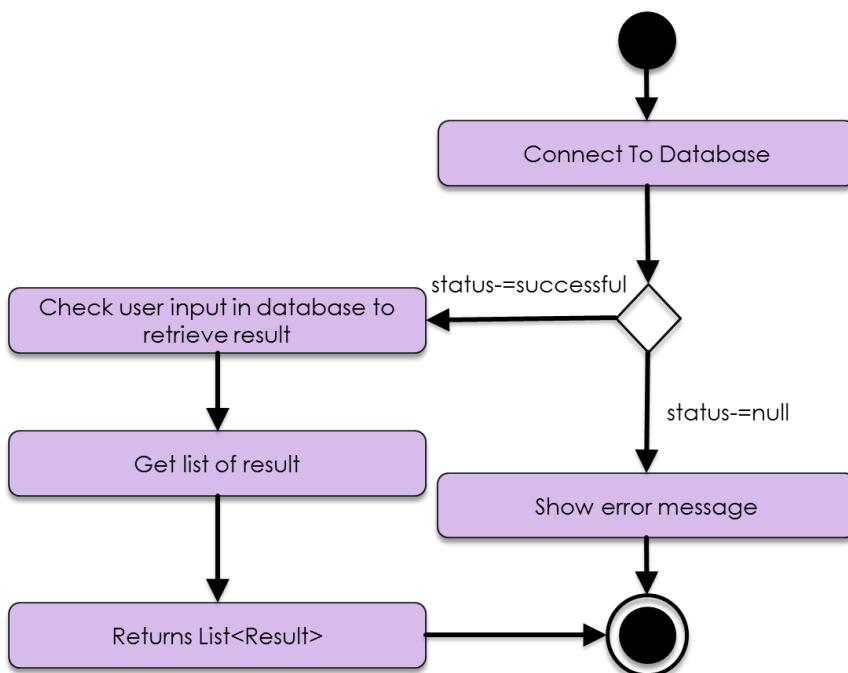


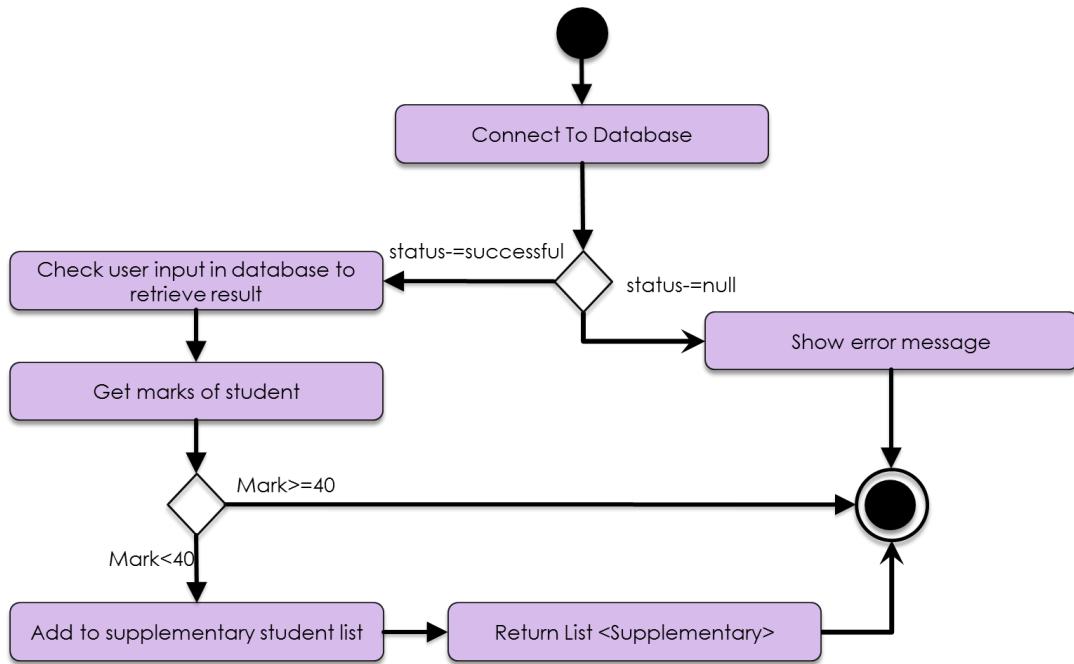
Figure 66: Processing flow of `chooseOption`



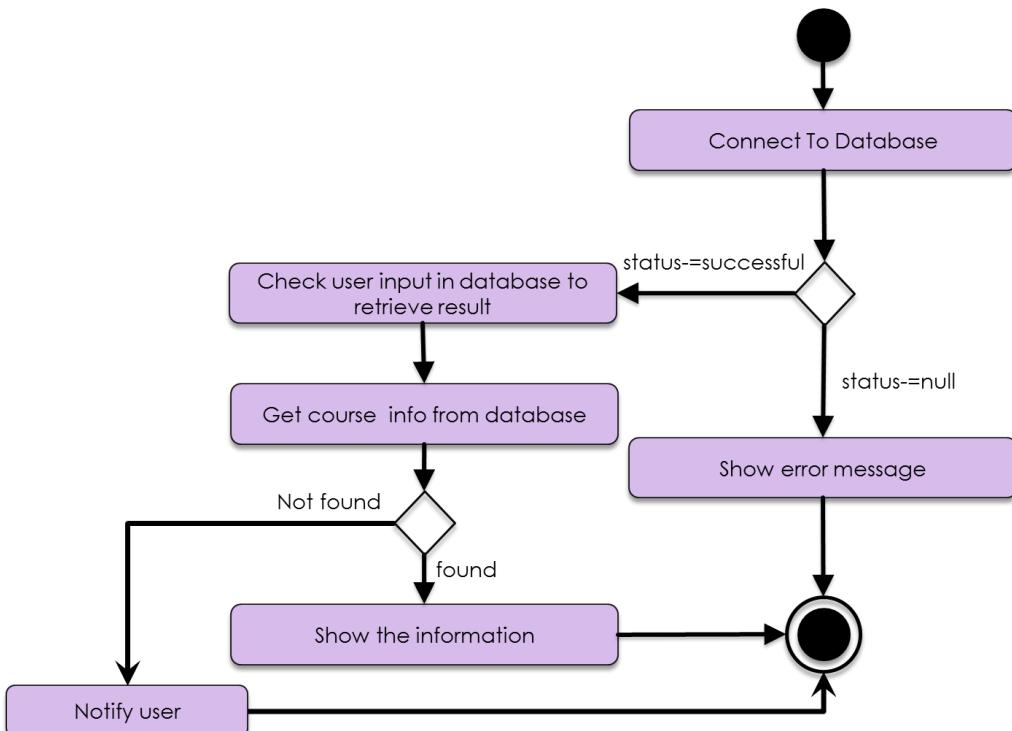
**Figure 67: Processing flow of `viewListOfCourse()`**



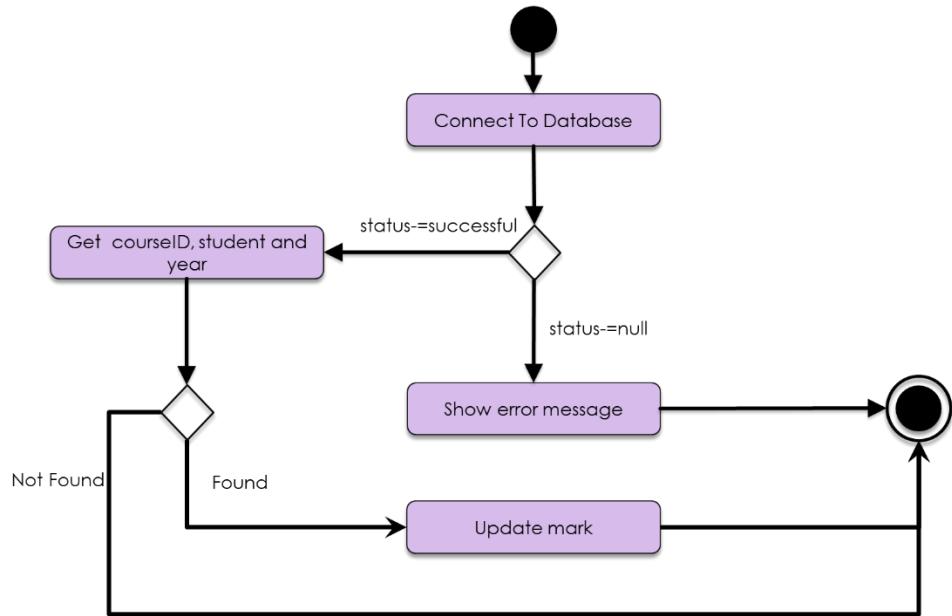
**Figure 68: Processing flow of `viewResultSheet(courseID,year)`**



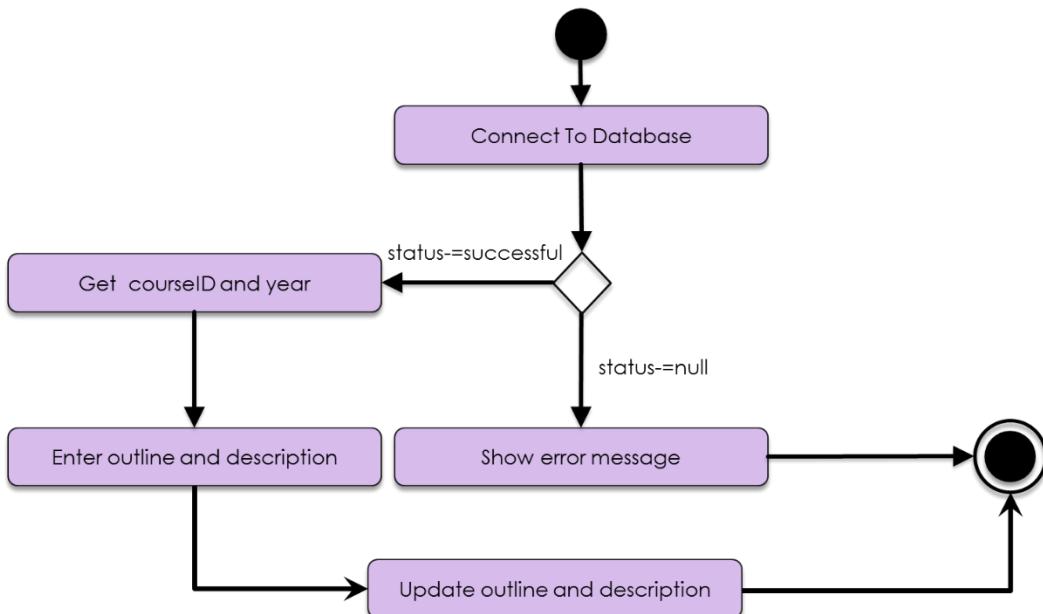
**Figure 69: Processing flow of `viewSupplementaryList(courseID,year)`**



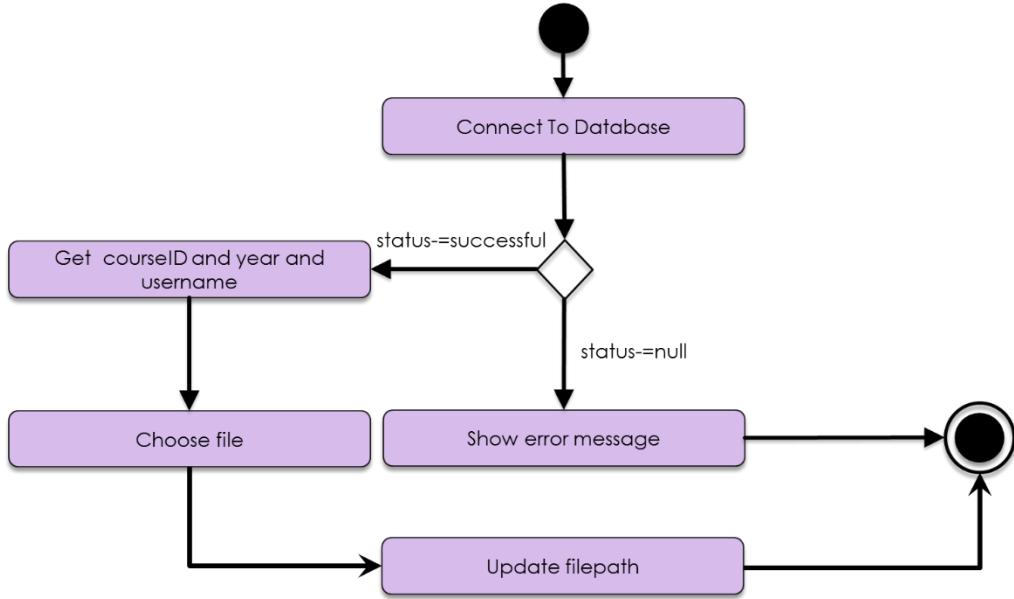
**Figure 70: Processing flow of `viewCourseInformation(courseID,year)`**



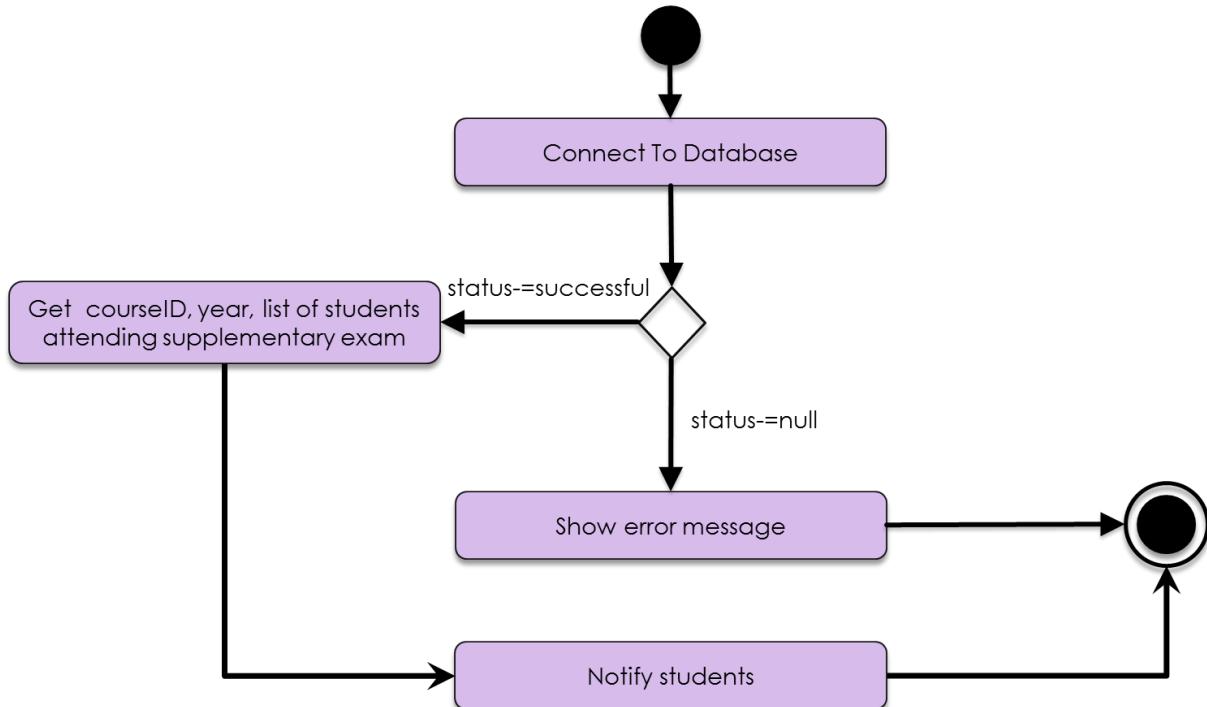
**Figure 71: Processing flow of `updateStudentMark(courseID, year, mark, studentID)`**



**Figure 72: Processing flow of `updateCourseInformation(courseID, year, outline, description)`**



**Figure 73: Processing flow of `updatePDF(courseID, year, username)`**



**Figure 74: Processing flow of `notifyForSupplementaryExam(courseID, year)`**

## Processing flow of the methods: Database Operator class

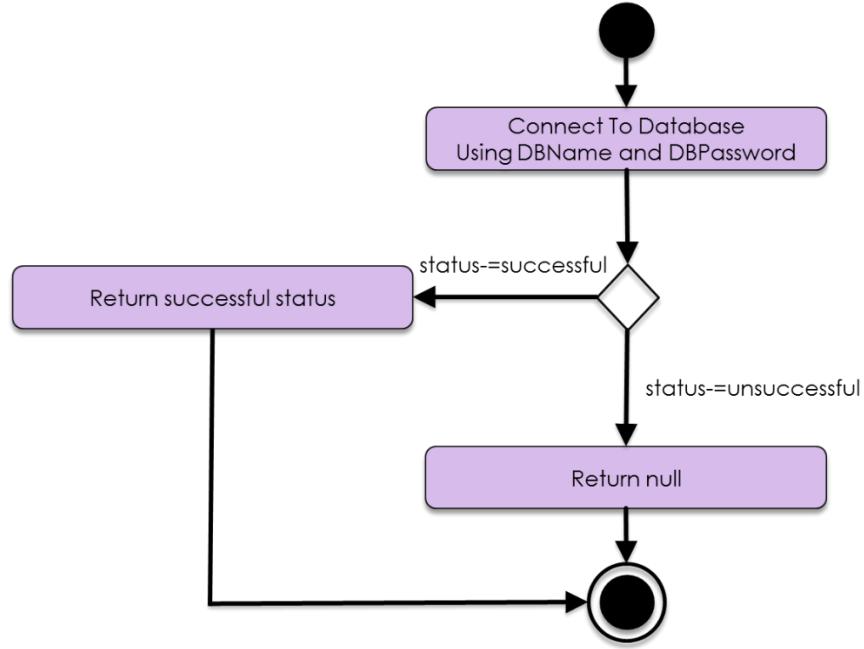


Figure 75: Processing flow of `Connect()`

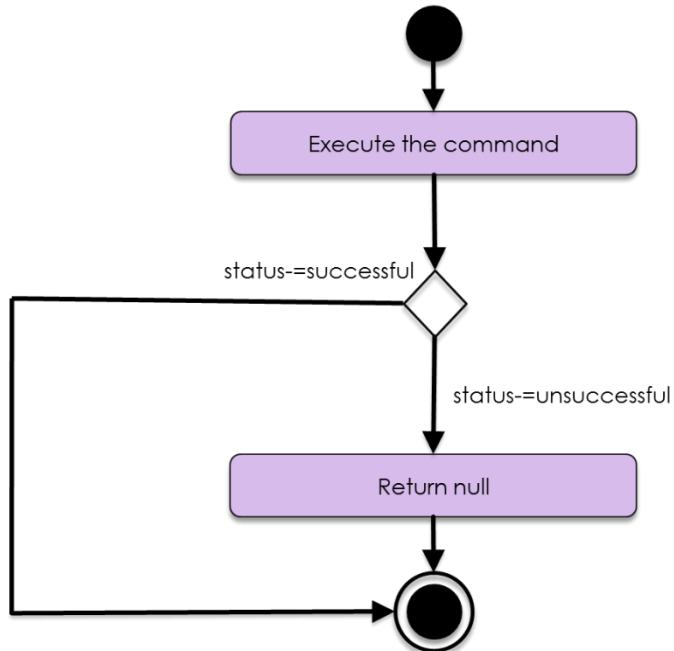
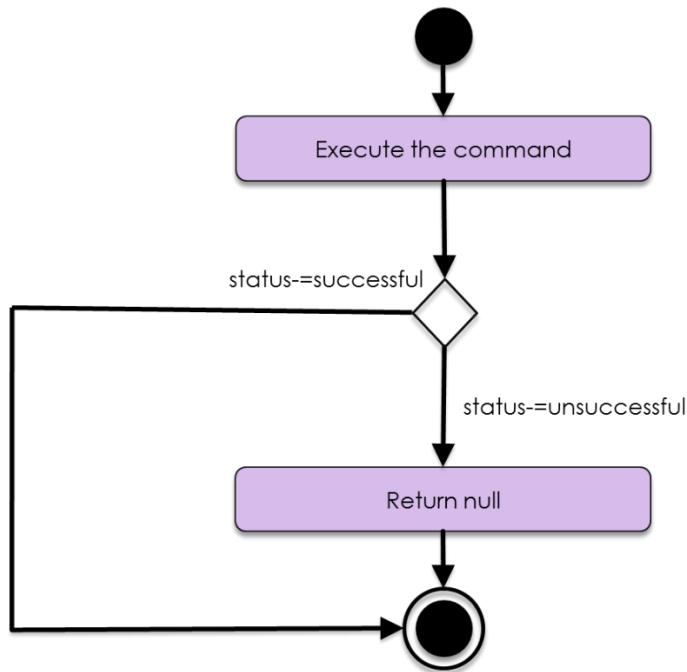
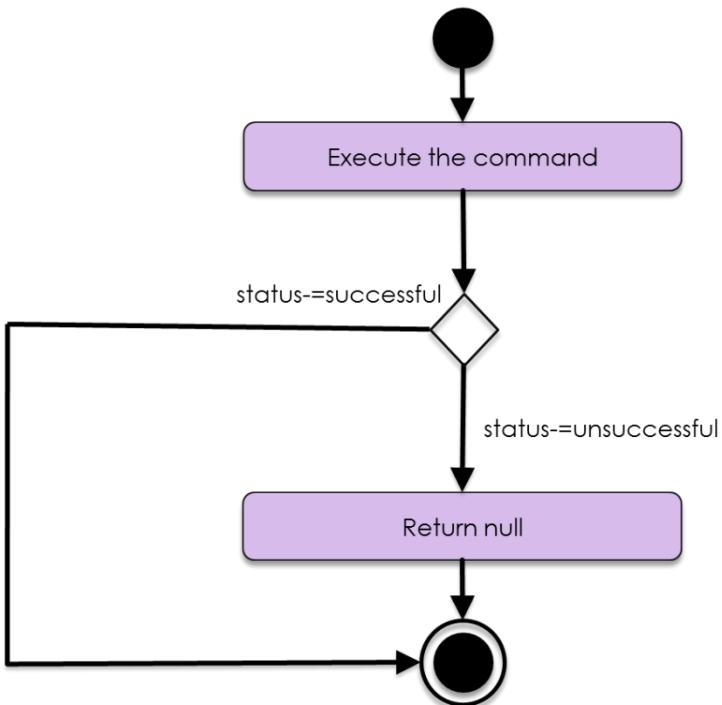


Figure 76: Processing flow of `executeSelectCommand(sql)`



**Figure 77: Processing flow of executeUpdateCommand(sql)**



**Figure 78: Processing flow of executeInsertCommand(sql)**

## Processing flow of the methods: Receptionist class

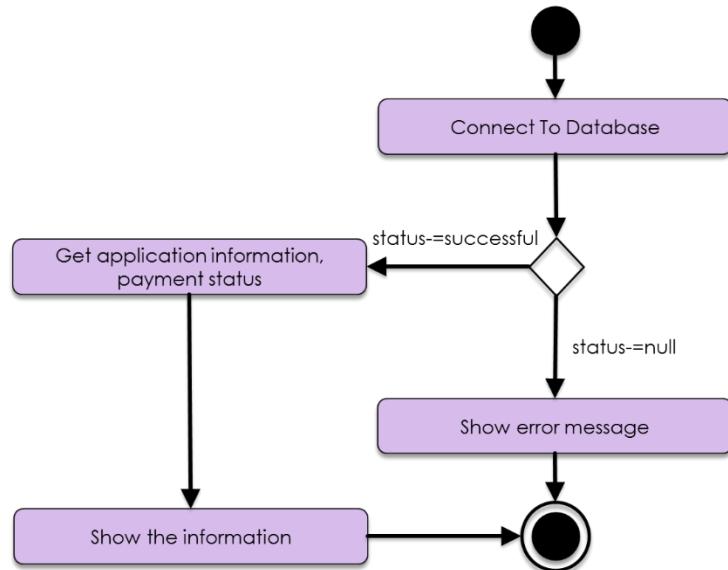


Figure 79: Processing flow of `viewPaymentBoard()`

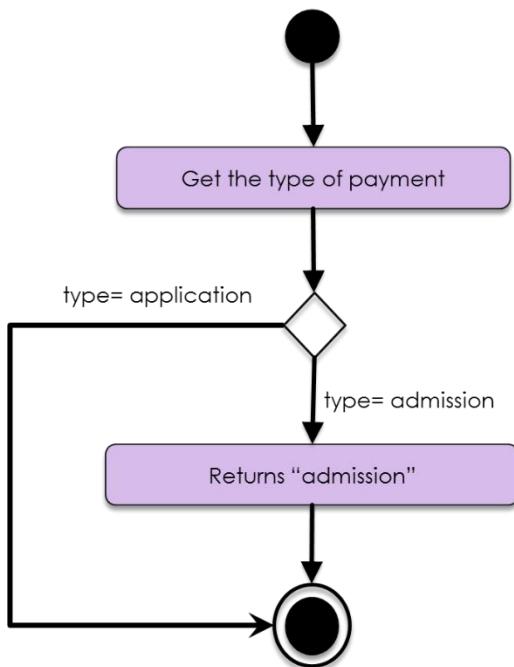
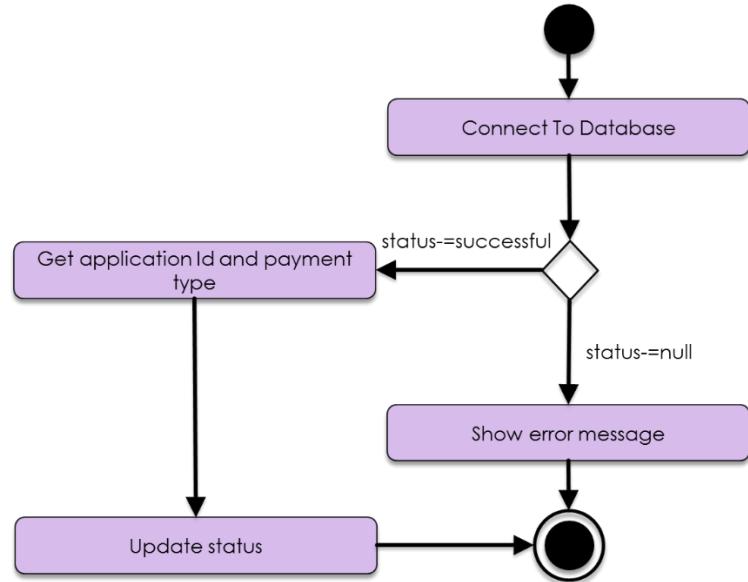
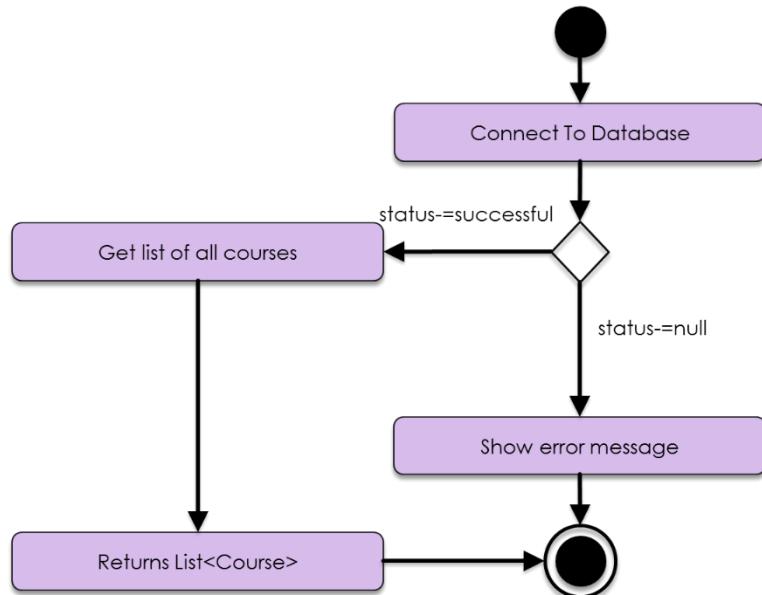


Figure 80: Processing flow of `getPaymentType ()`

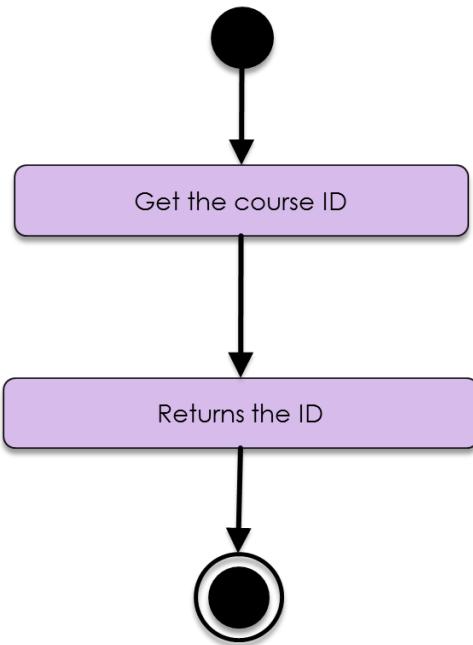


**Figure 81: Processing flow of `updatePaymentStatus()`**

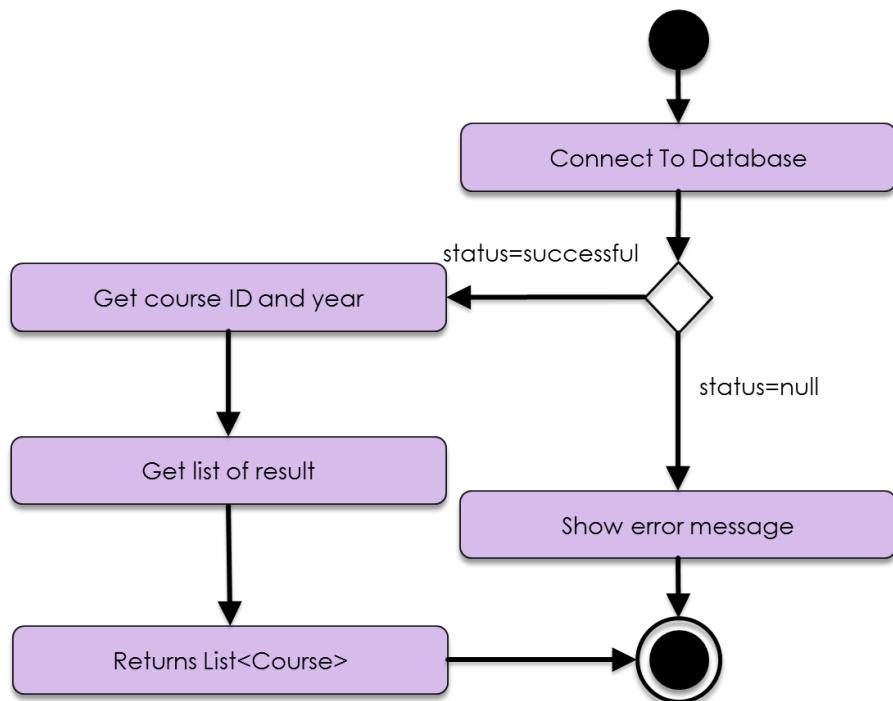
### Processing flow of the methods: Program Chairperson Class



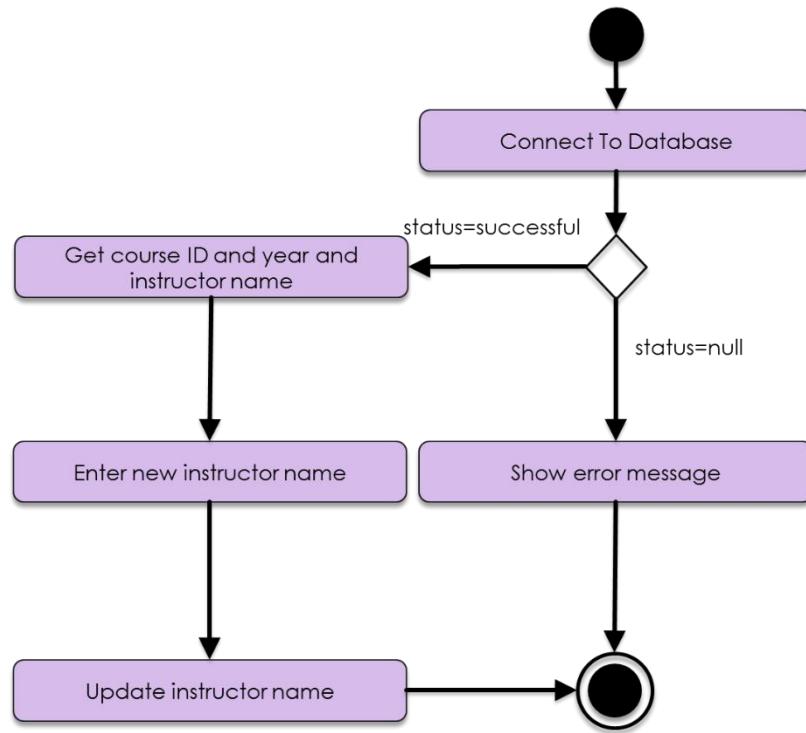
**Figure 82: Processing flow of `viewListOfCourses()`**



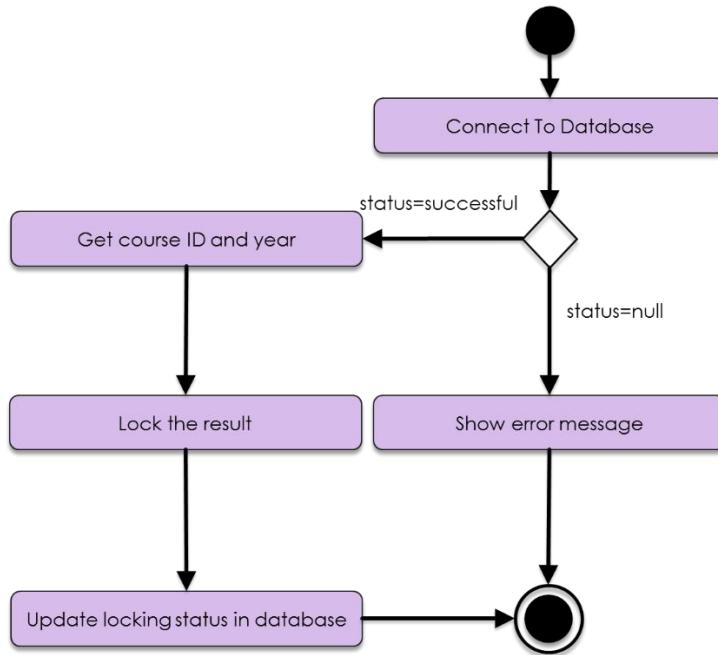
**Figure 83: Processing flow of `getCourseID()`**



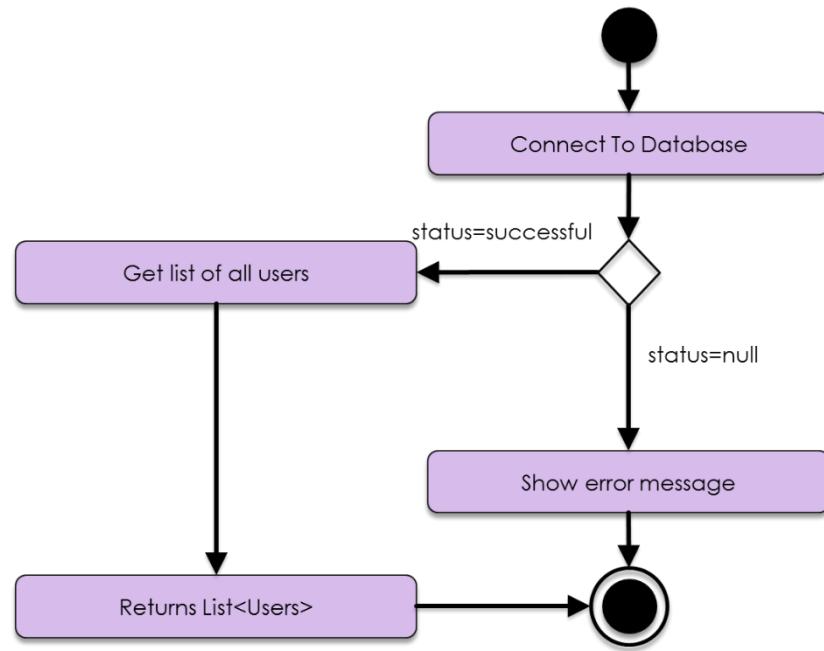
**Figure 84: Processing flow of `viewResult()`**



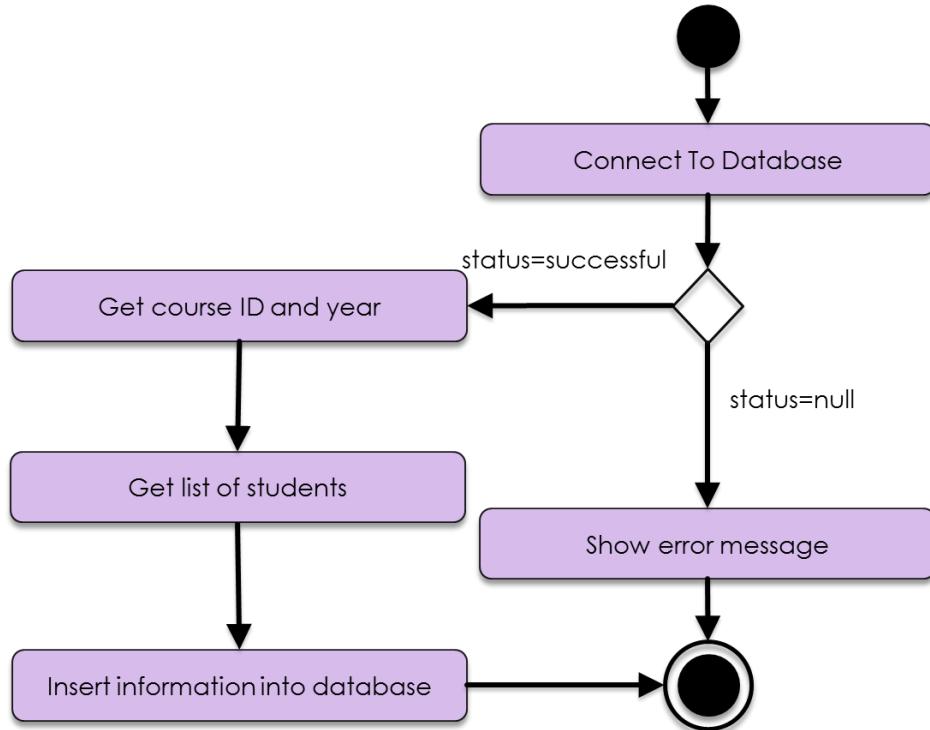
**Figure 85: Processing flow of `changeInstructor()`**



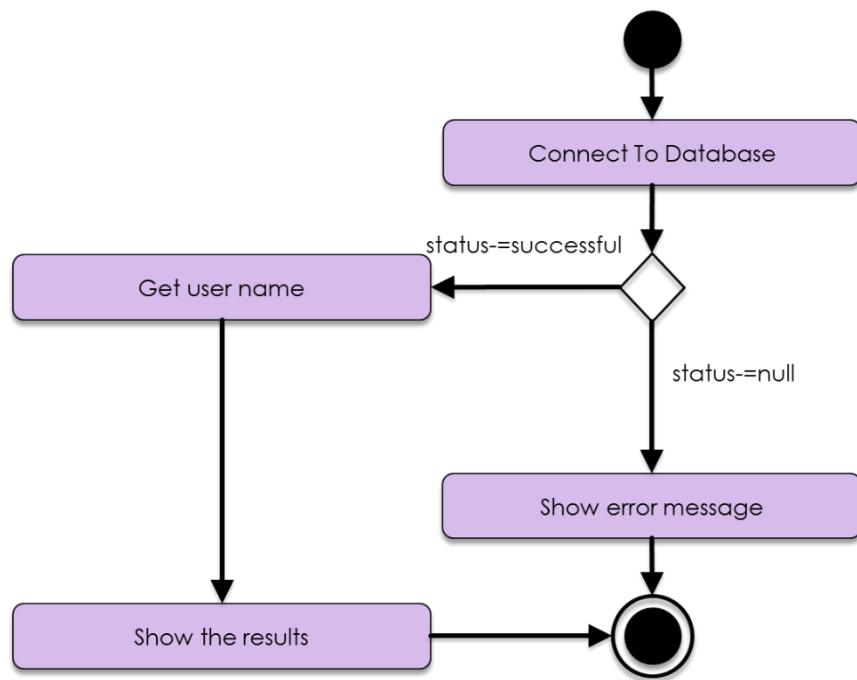
**Figure 86: Processing flow of `lockResult()`**



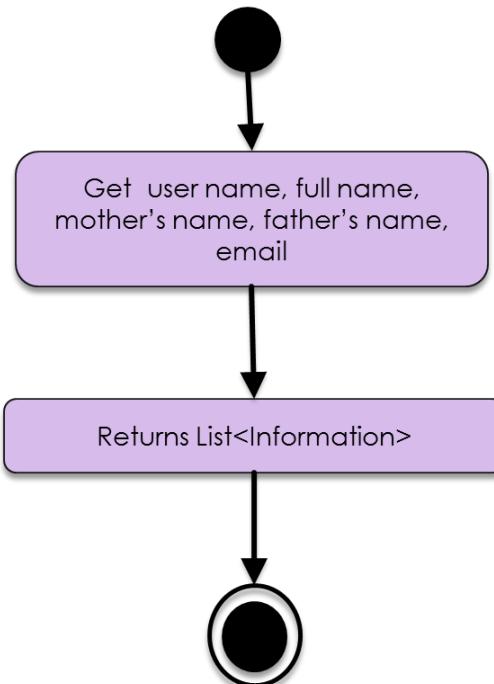
**Figure 87: Processing flow of `viewListOfUsers()`**



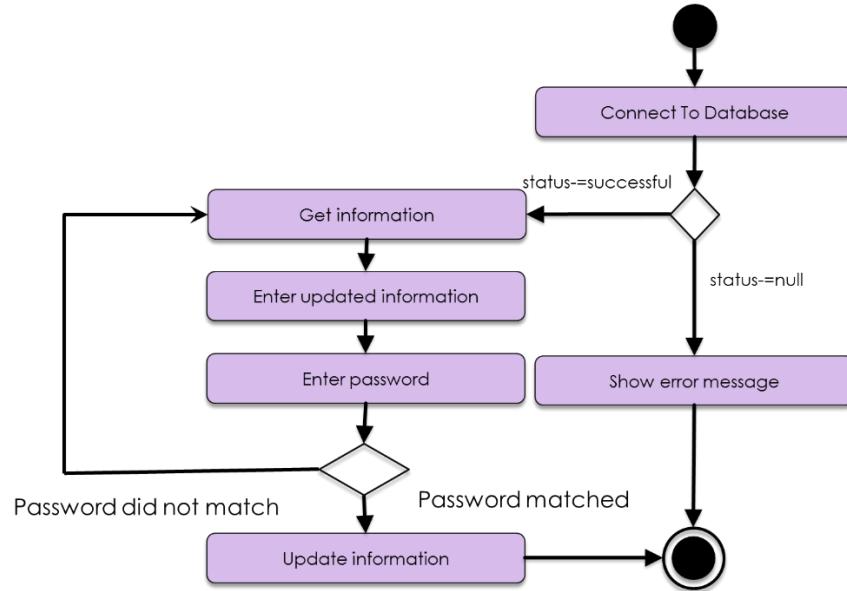
**Figure 88: Processing flow of `addStudents()`**



**Figure 89: Processing flow of `viewUserInfo()`**

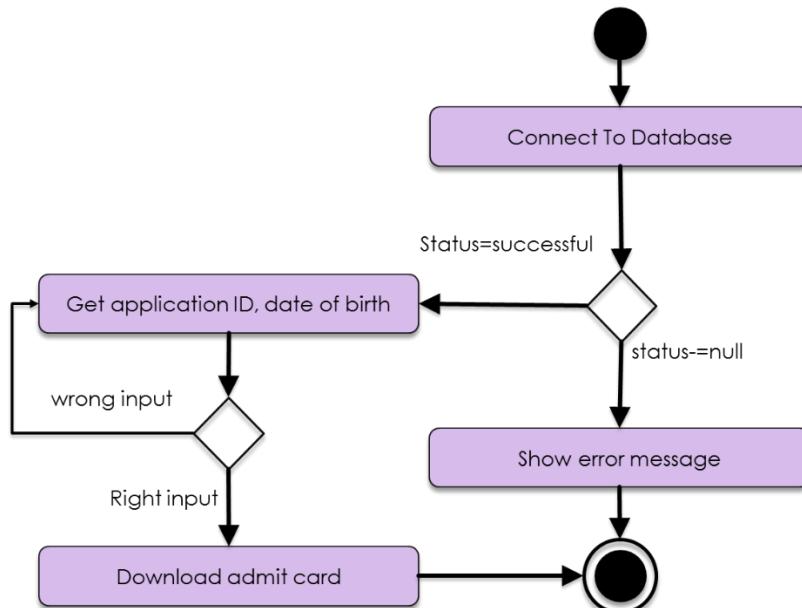


**Figure 90: Processing flow of `getUserInfo()`**



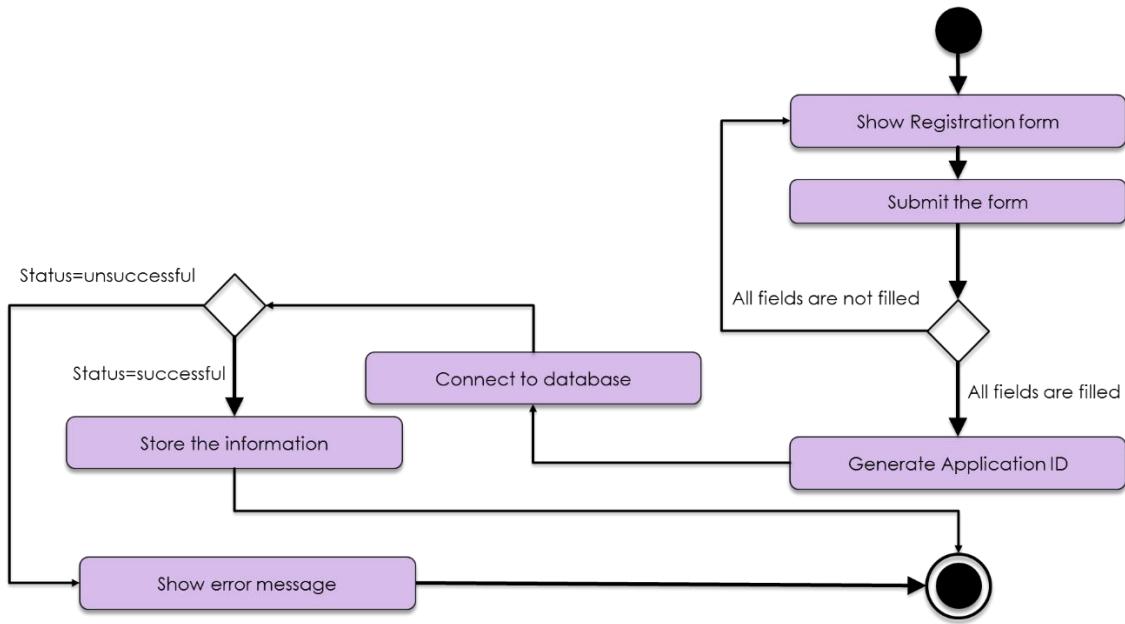
**Figure 91: Processing flow of `updateInfo()`**

### Processing flow of the methods: Applicant class



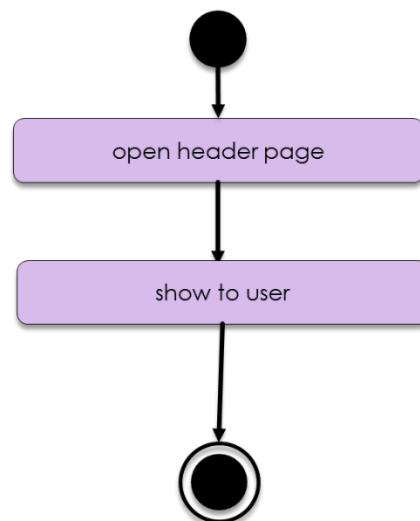
**Figure 92: Processing flow of `downlaodAdmitCard`**

**(`applicationID,dateOfBarth`)**

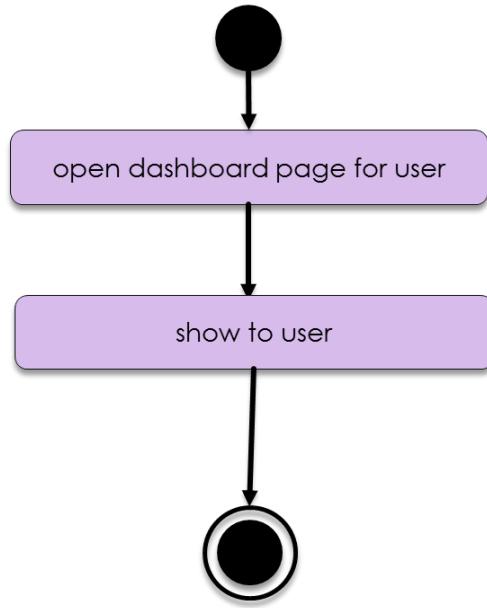


**Figure 93: Processing flow of apply(name, fathersName, mothersName, email, contactNo, photo, dateOfBirth, nationality, permanentAddress, sscO, hscA, bachelor, hscMathMark, bachelorMathMark)**

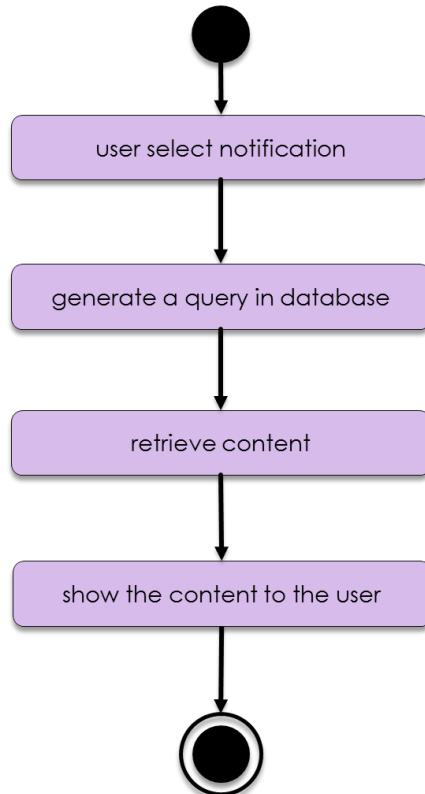
### Processing flow of the methods: User class



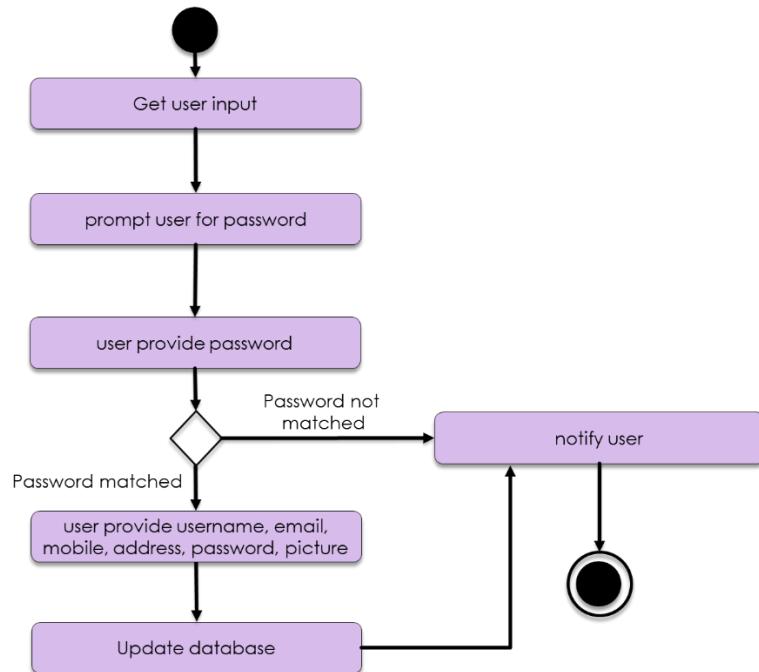
**Figure 94: Processing flow of viewHeader()**



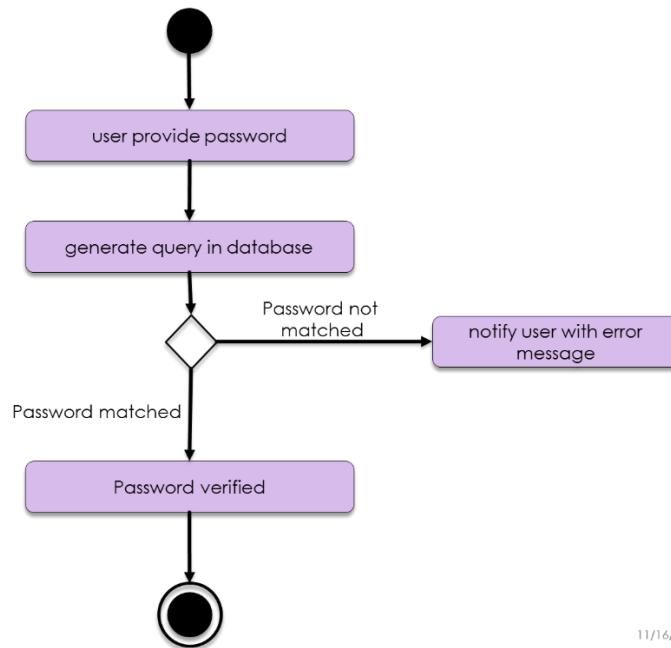
**Figure 95: Processing flow of `viewDashboard()`**



**Figure 96: Processing flow of `viewNotification(notificationID)`**



**Figure 97: Processing flow of updatePersonalData(username, email, mobile, address, password, picture)**



11/16/2017

**Figure 98: Processing flow of verifyPassword(username, password)**

## Processing flow of the methods: Exam Controller Class

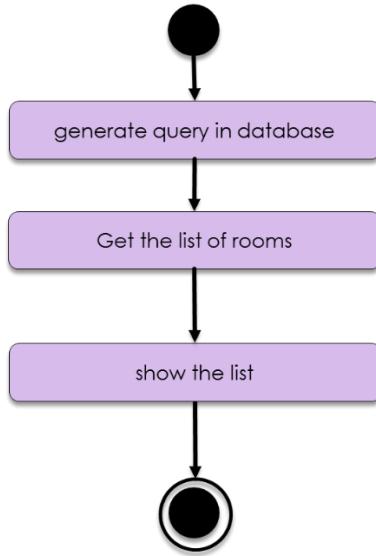


Figure 99: Processing flow of `viewListOfRoom()`

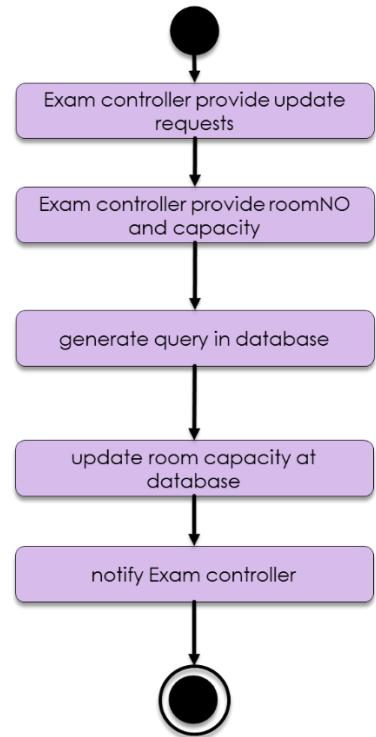
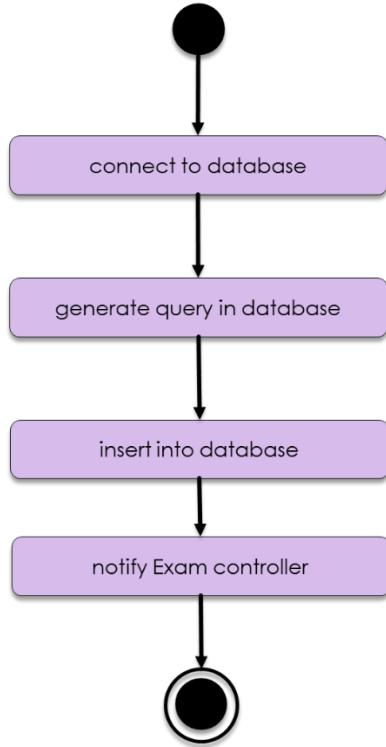
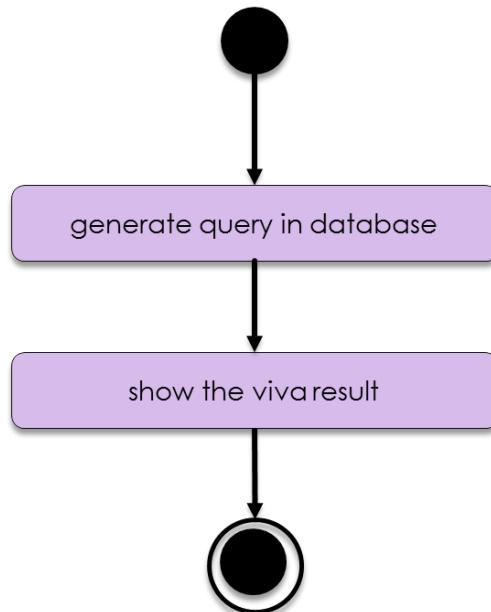


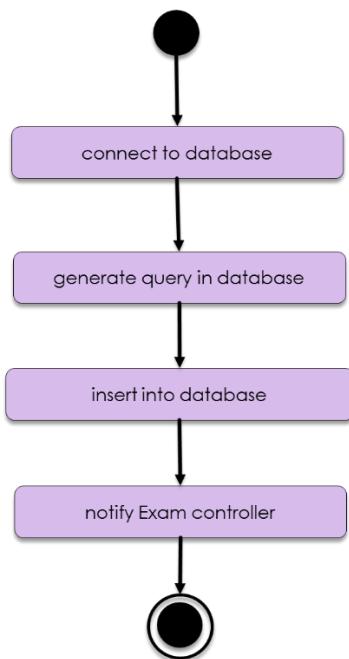
Figure 100: Processing flow of `updateRoomCapacity()`



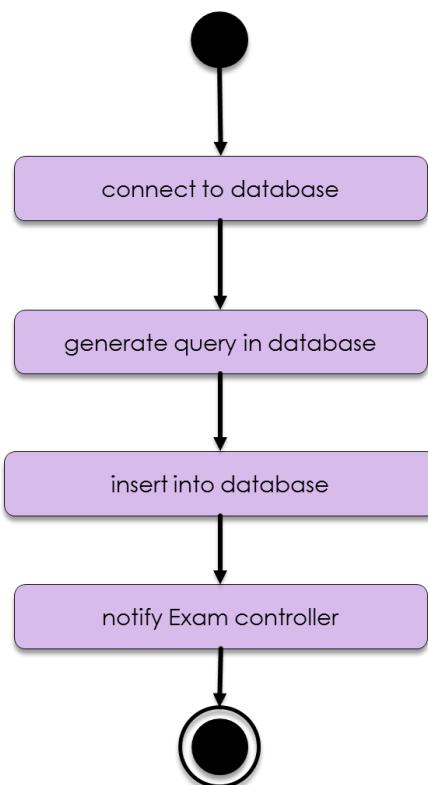
**Figure 101: Processing flow of insertRoomDatabase()**



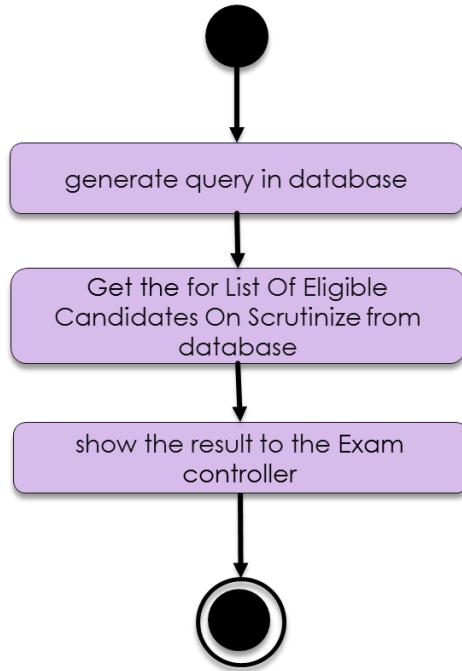
**Figure 102: Processing flow of viewVivaSchedule()**



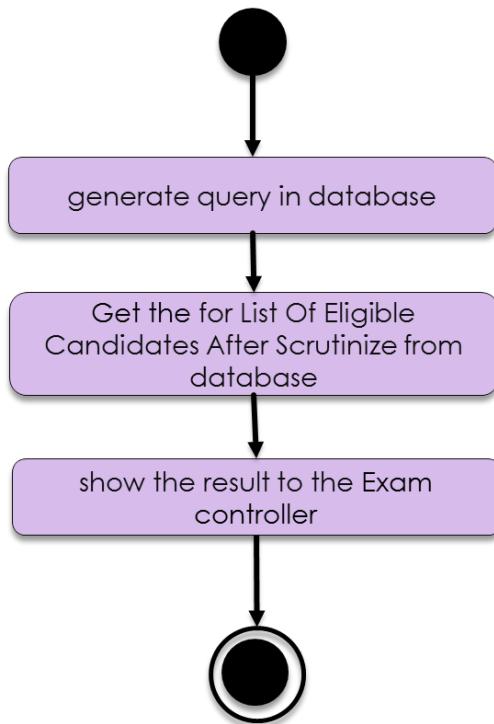
**Figure 103: Processing flow of `insertScheduleIntoDatabase()`**



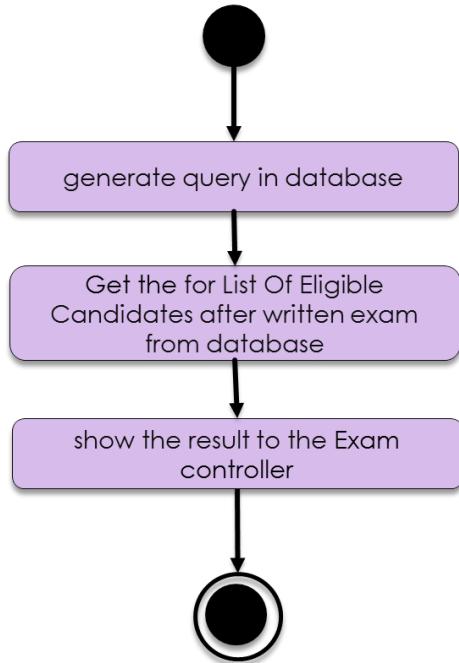
**Figure 104: Processing flow of `insertAdmisionInfoIntoDatabase()`**



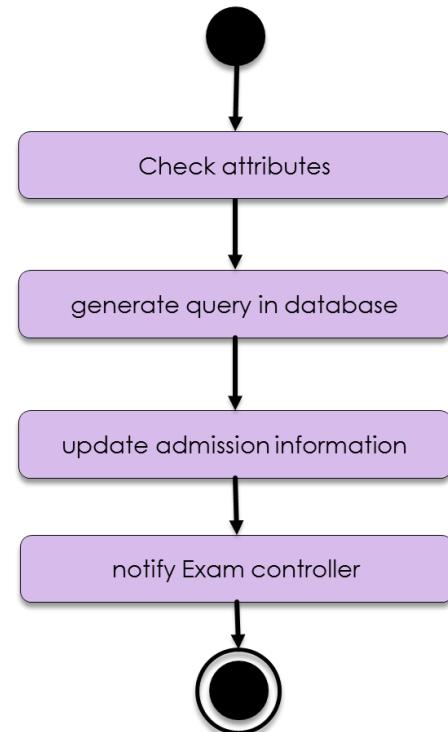
**Figure 105: Processing flow of `viewListOfEligibleCandidatesOnScrutinize()`**



**Figure 106: Processing flow of `viewListOfEligibleCandidatesAfterScrutinize()`**



**Figure 107: Processing flow of  
`viewListOfEligibleCandidatesAfterWrittenExam()`**



**Figure 108: Processing flow of `updateAdmissionInfo`**

## Processing flow of the methods: Student Class

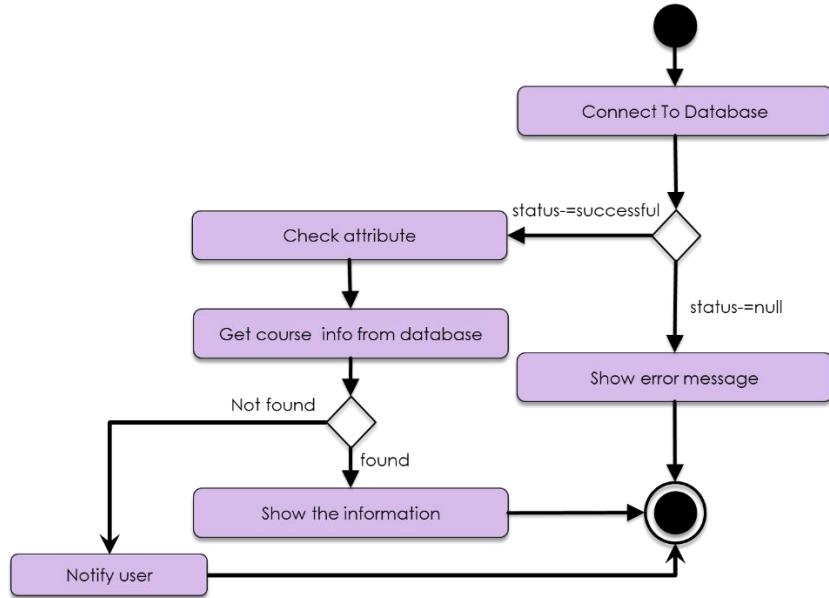


Figure 109: Processing flow of `viewCourseInformation(courseID,year)`

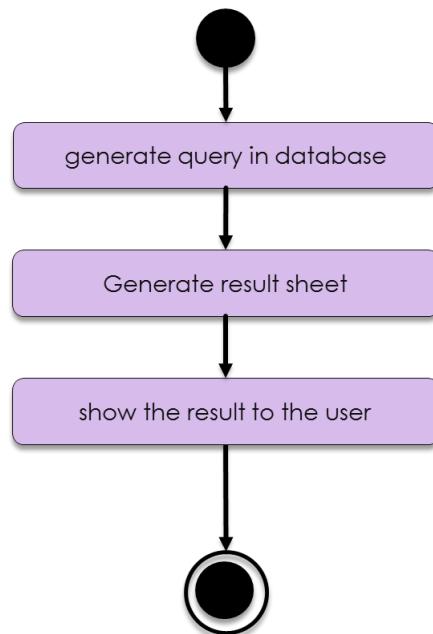
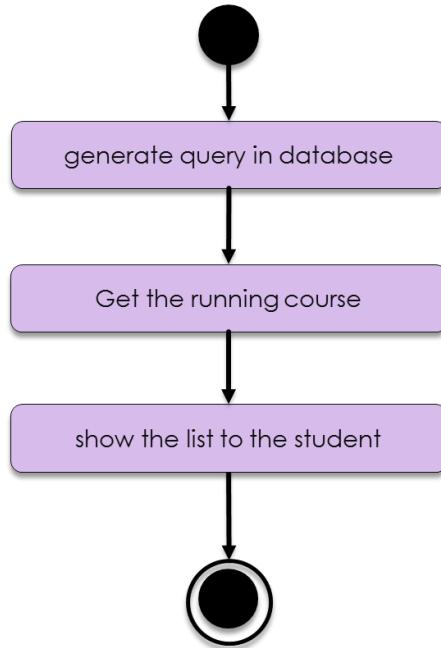
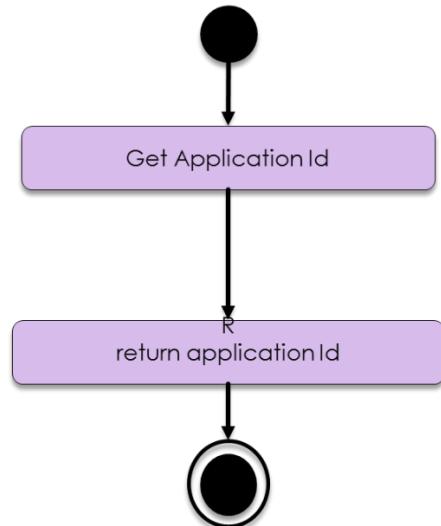


Figure 110: Processing flow of `viewResultsheet()`

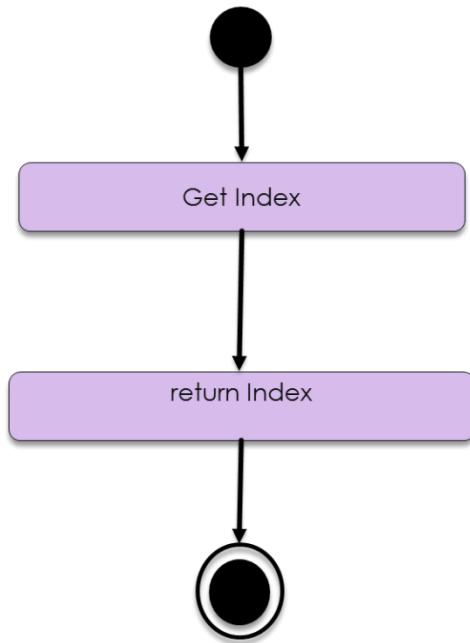


**Figure 111: Processing flow of viewRunningCourses()**

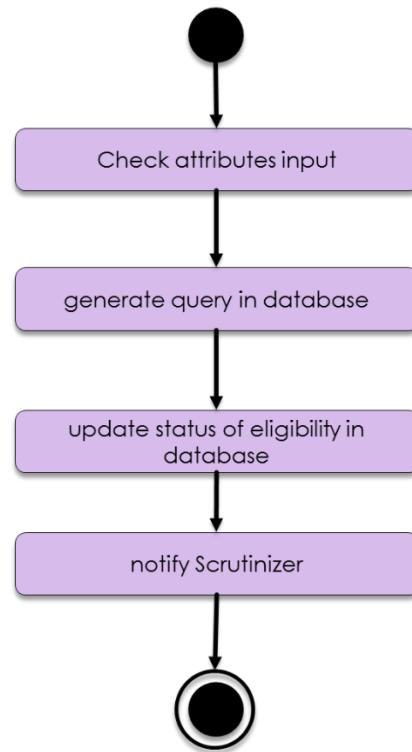
### Processing flow of the methods: Scrutinizer Class



**Figure 112: Processing flow of getapplicationId()**

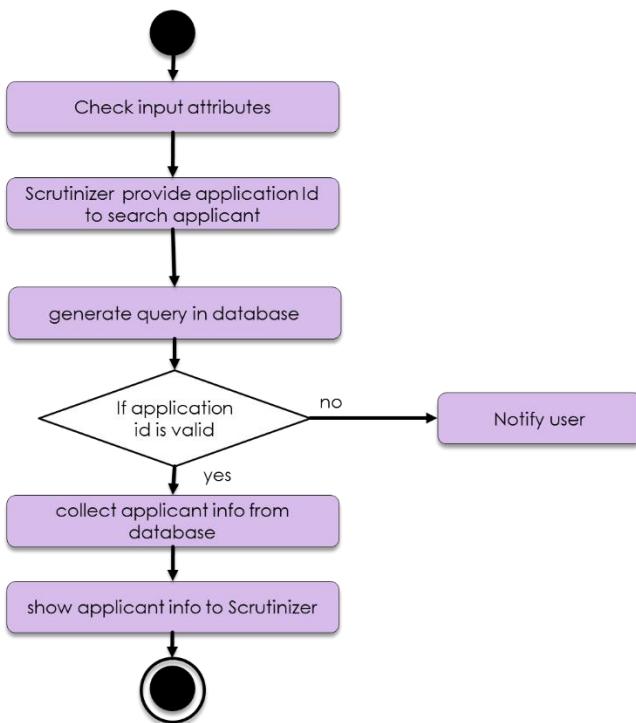


**Figure 113: Processing flow of getIndex()**

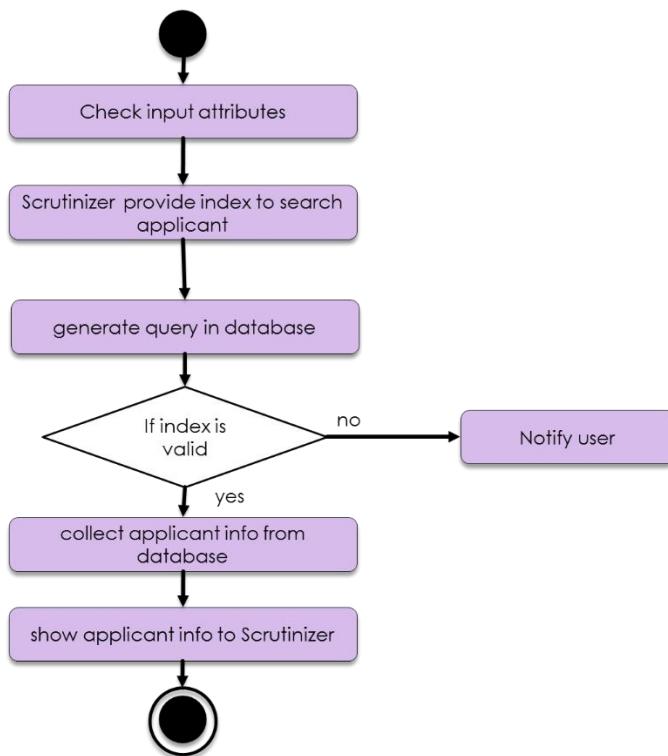


11/16/2017

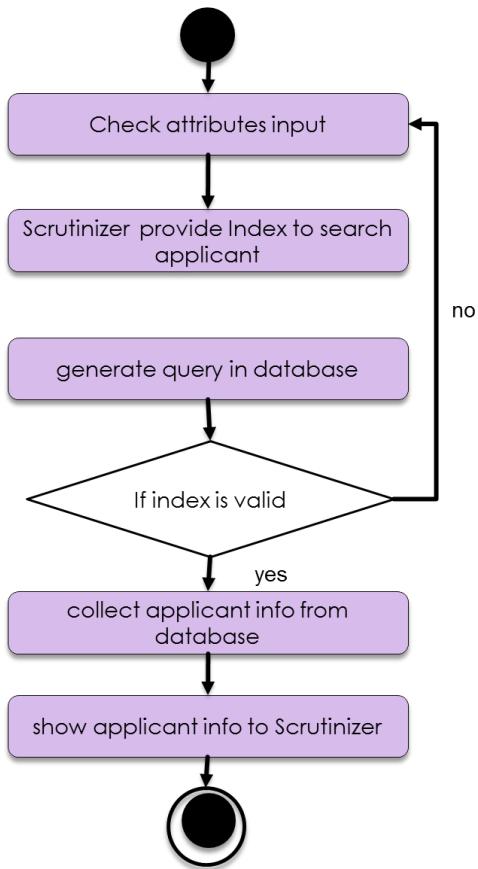
**Figure 114: Processing flow of UpdateStatus(applicationId,index)**



**Figure 115: Processing flow of searchByApplicationID(applicationId)**

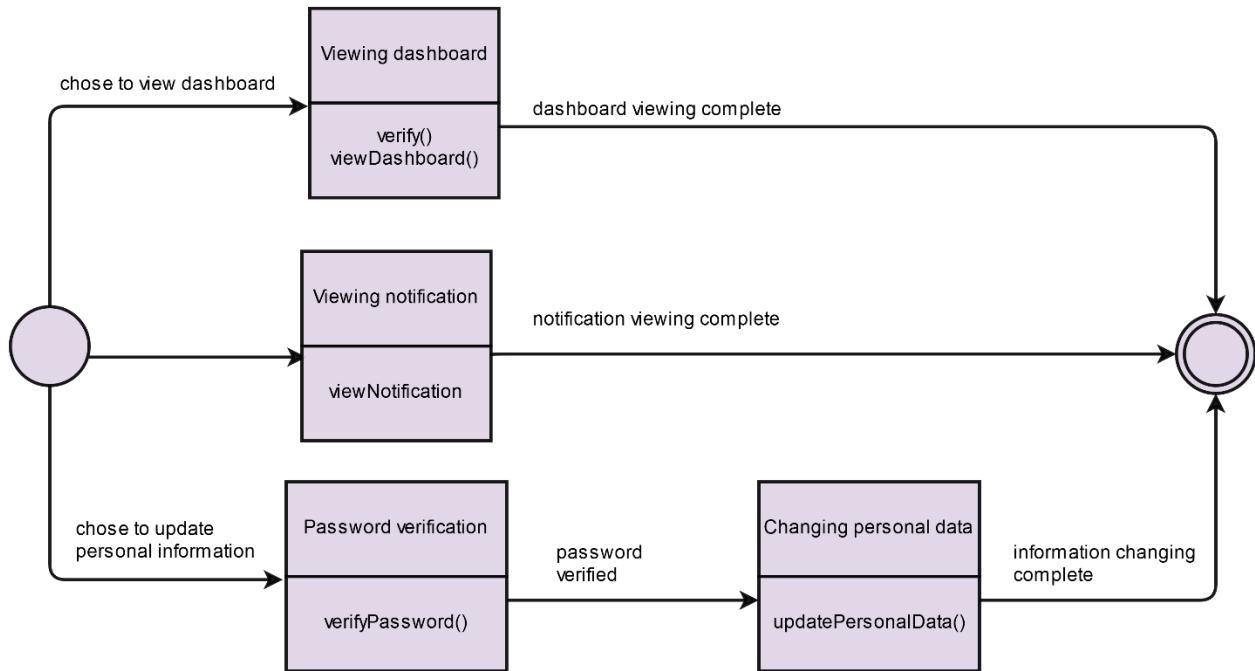


**Figure 116: Processing flow of searchByIndex(index)**

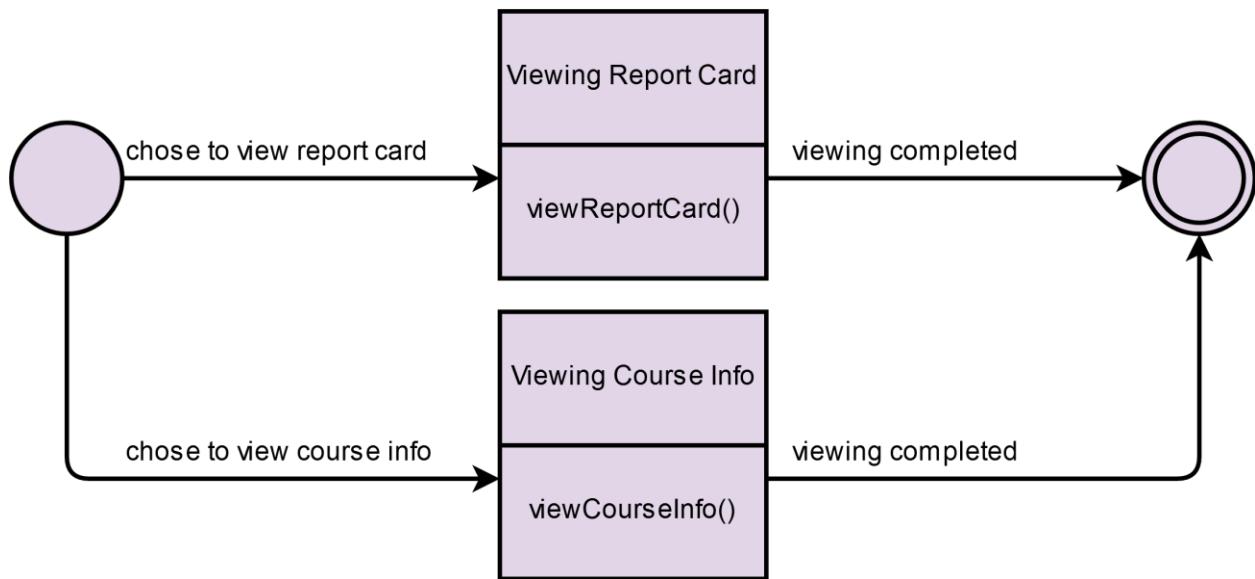


**Figure 117: Processing flow of `searchByIndex`**

## 2.3 f) State Diagram



**Figure 118: State diagram of User class**



**Figure 119: State diagram of Student class**

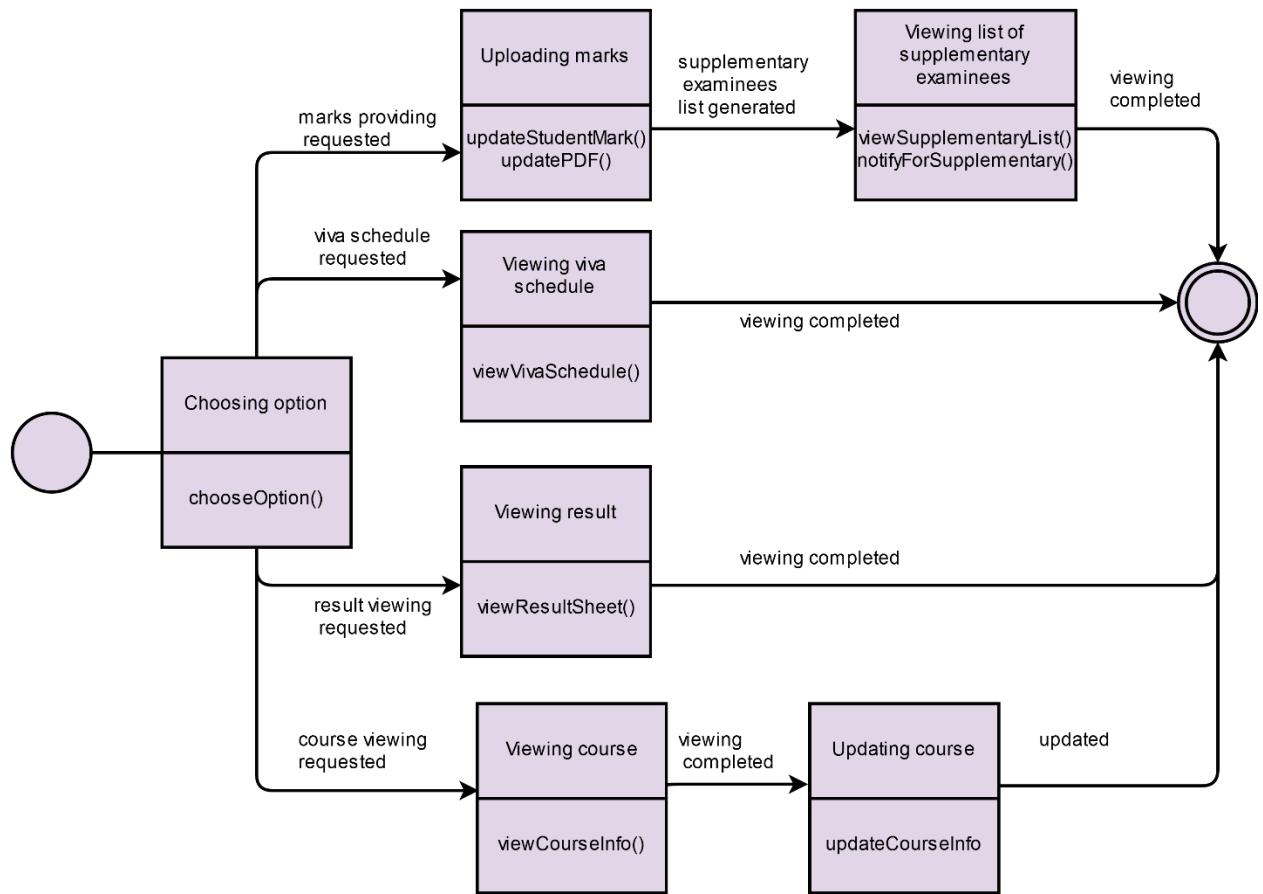


Figure 120: State diagram of Teacher class

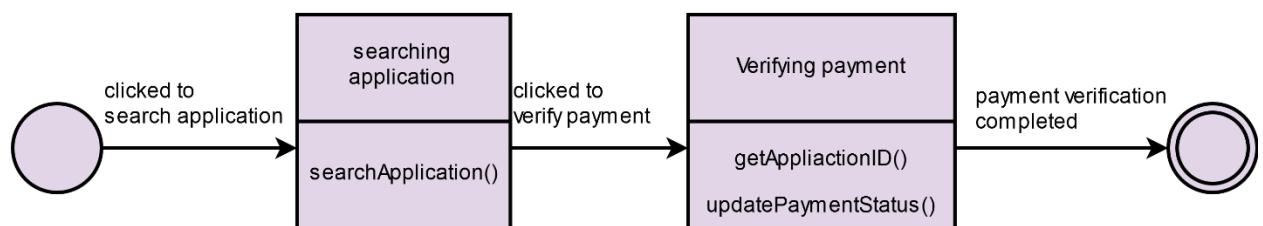
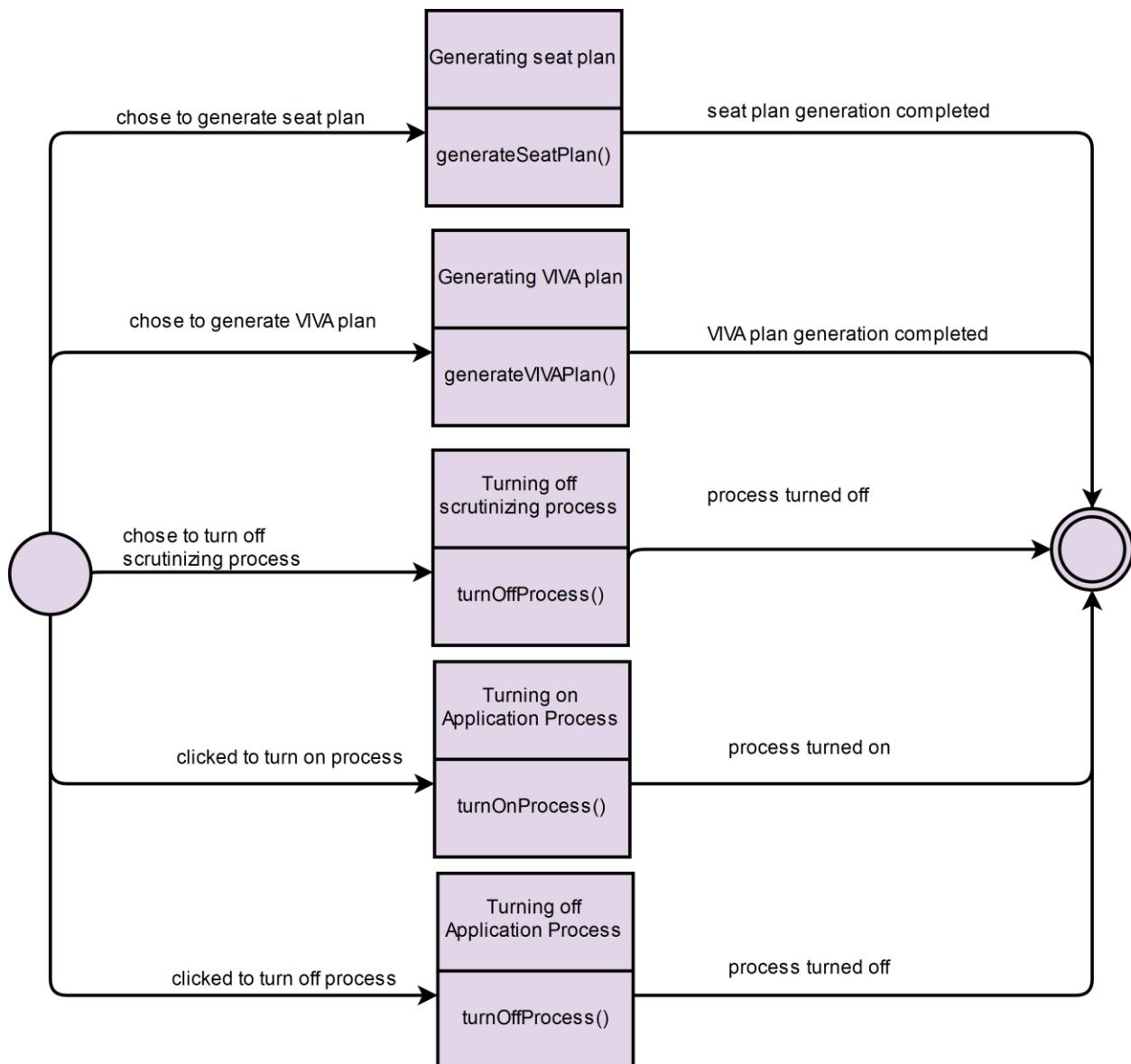
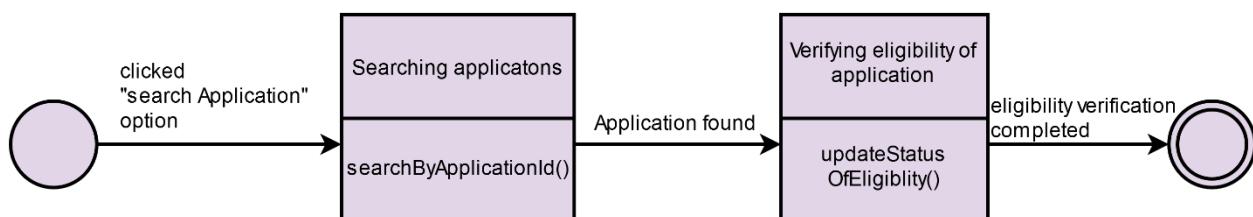


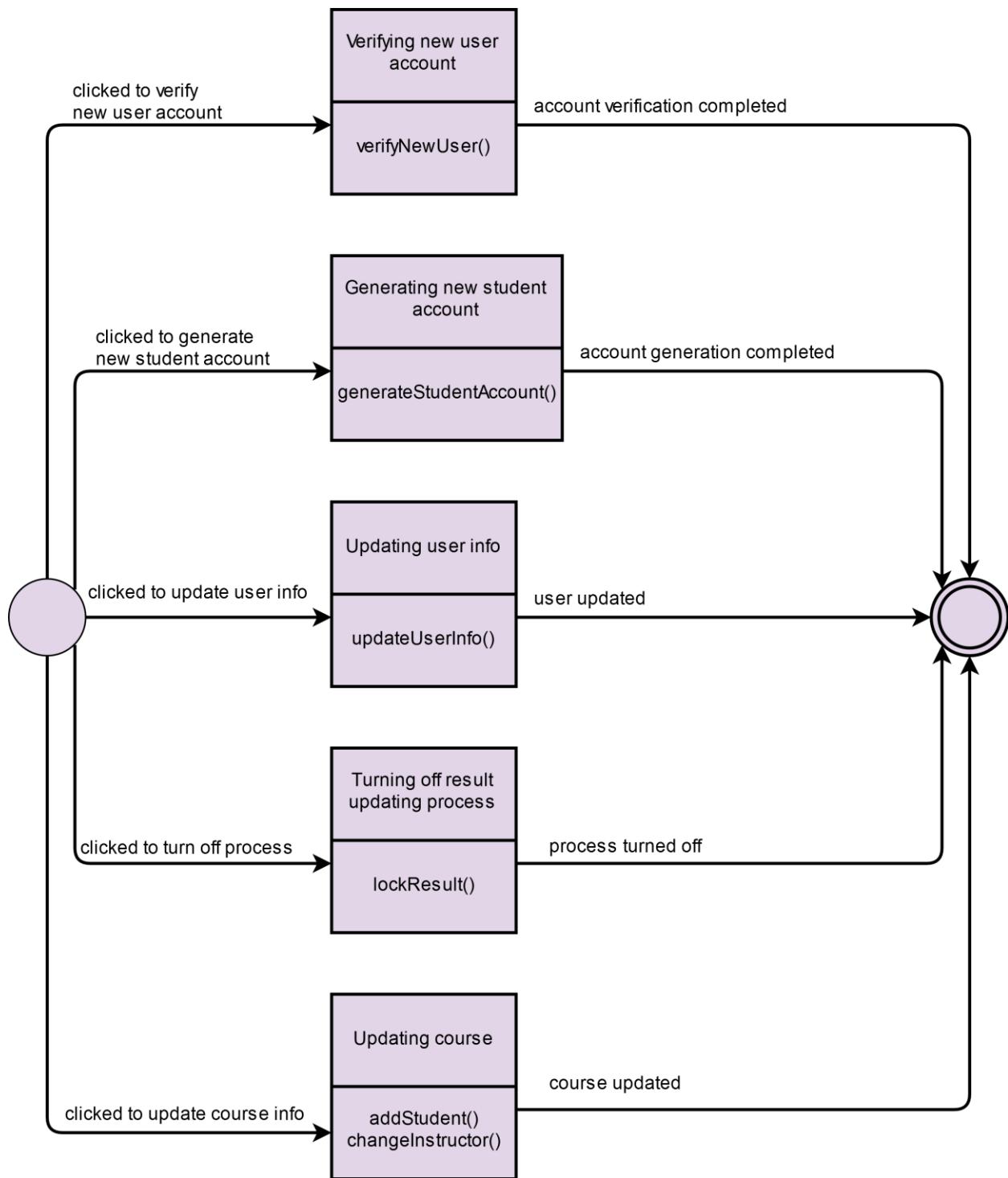
Figure 121: State diagram of Receptionist class



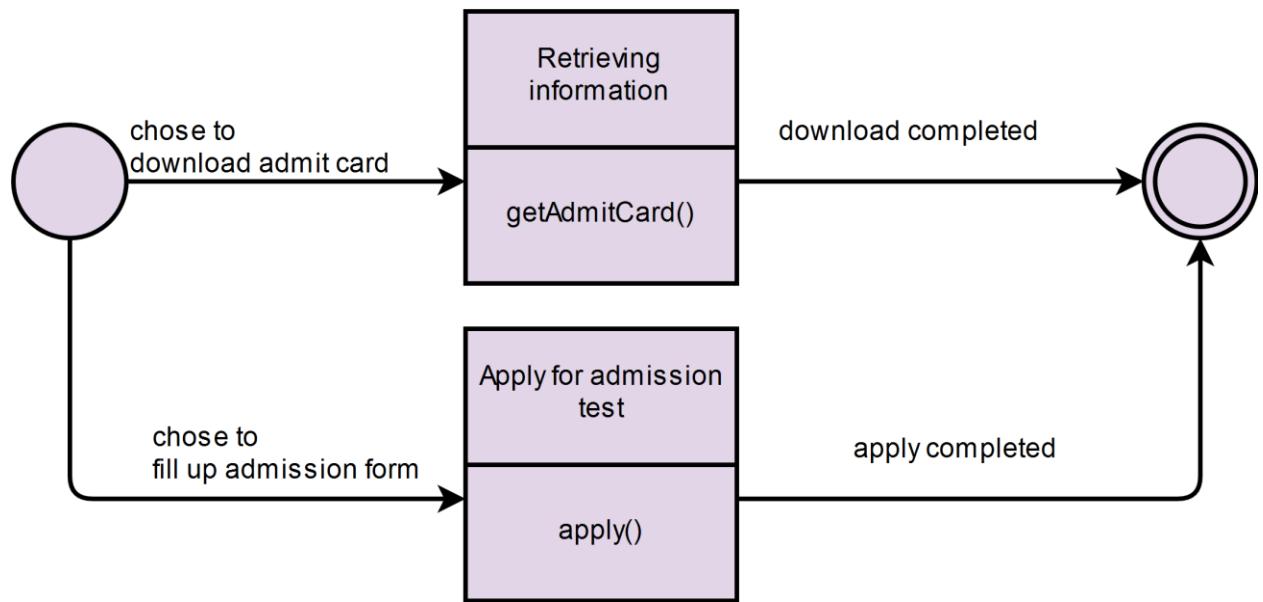
**Figure 122: State diagram of Exam Controller class**



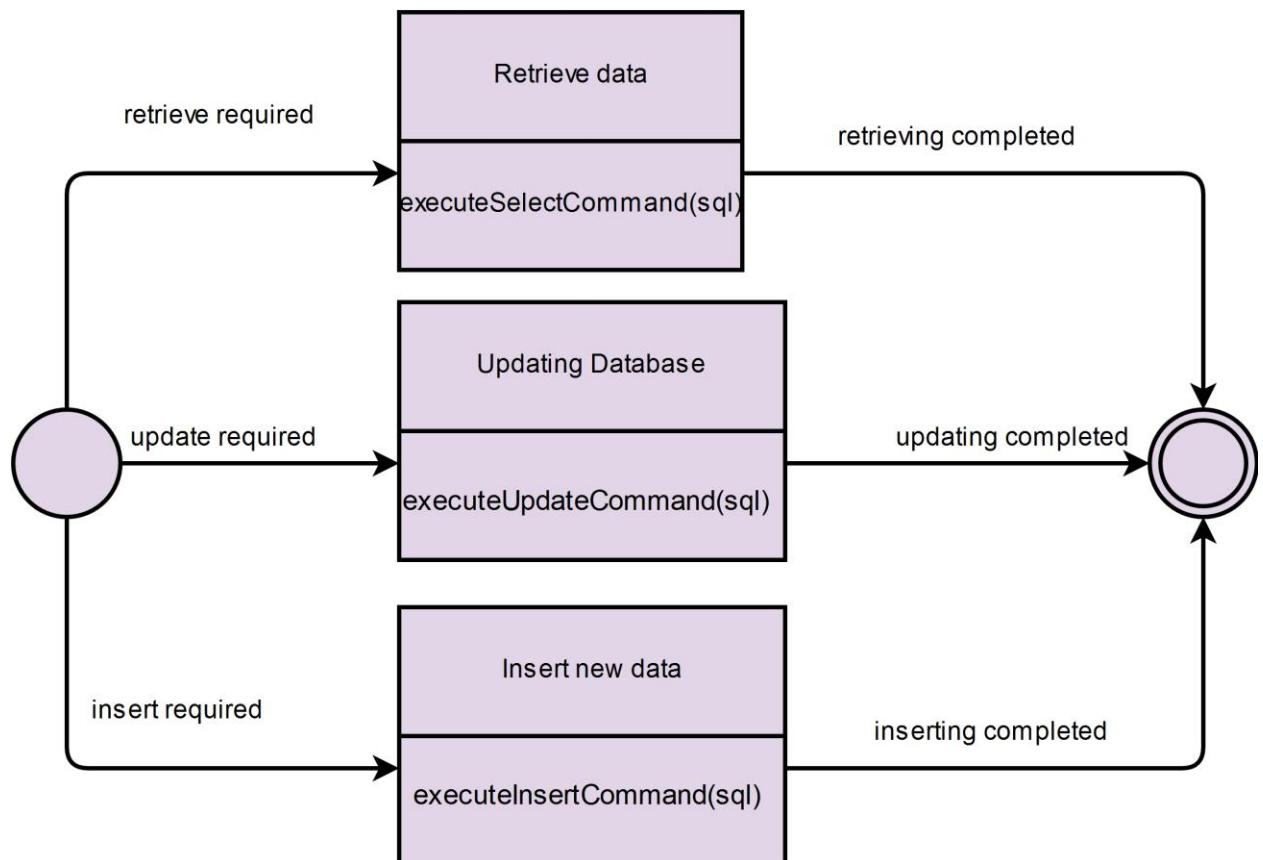
**Figure 123: State diagram of Scrutinizer class**



**Figure 124: State diagram of Program Chairman class**



**Figure 125: State diagram of Applicant class**



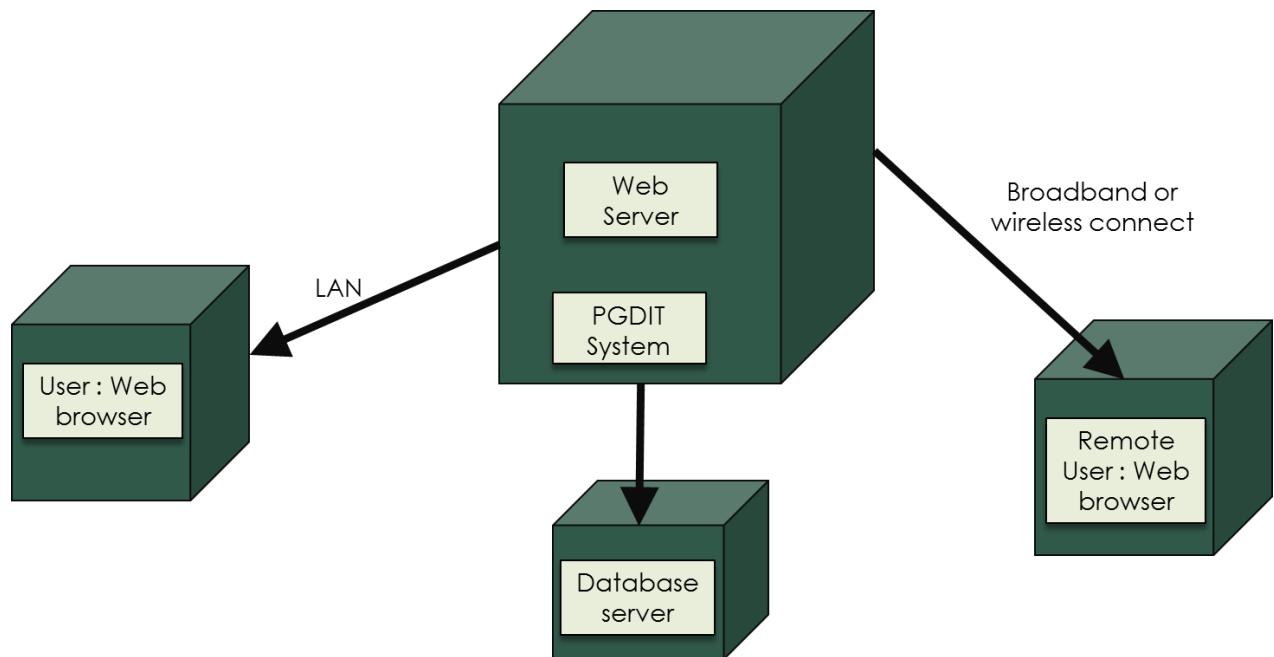
**Figure 126: State diagram of Database Operator**

## 2.4 Persistent Data

**MySQL database:** An open source relational database management system which is powerful and free that delivers a rich and reliable data store for lightweight Web Sites and desktop applications.

**Class required to manage DB:** Database Operator

## 2.5 Elaborated Deployment



**Figure 127: Elaborated deployment of PGDIT Automation System**

# CHAPTER 3: INTERFACE DESIGN

User Interface Design is the design of websites, computers, appliances, machines, mobile communication devices, and software applications with the focus on the user's experience and interaction.

## 3.1 Interface Analysis

We divide interface analysis into the following part-

1. User analysis
2. Task analysis

### USER ANALYSIS

There are two steps in this part-

1. Identify user
2. Know user

#### Identify User

From the requirement and specification document, we have found six types of user-

1. Teacher
2. Student
3. Exam Controller
4. Program Chairperson
5. Scrutinizing team
6. Receptionist
7. Applicant

#### Know User

We collect the following information about the users.

Teacher

- Age: 25-60
- Skills: Above Average
- Domain expert: Yes

- Application expert: No
- Office hour: Normal
- Frequency of use: Occasionally
- Consequence of a mistake: Low
- General computer experience: Yes

Student

- Age: 20-50
- Skills: Average
- Frequency of use: Occasionally
- Consequence of a mistake: Low
- General computer experience: Yes

Program Chairperson

- Age: 30-60
- Skills: Above Average
- Domain expert: Yes
- Application expert: No
- Office hour: Normal
- Frequency of use: Occasionally
- Consequence of a mistake: High
- General computer experience: Yes

Exam Controller

- Age: 30-60
- Skills: Above Average
- Domain expert: Yes
- Application expert: No
- Office hour: Normal
- Frequency of use: Occasionally
- Consequence of a mistake: High
- General computer experience: Yes

### Scrutinizing Team

- Age: 30-60
- Skills: Above Average
- Application expert: No
- Frequency of use: Occasionally
- Consequence of a mistake: High
- General computer experience: Yes

### Receptionist

- Age: 30-60
- Skills: Above Average
- Office hour: Normal
- Frequency of use: Occasionally
- Consequence of a mistake: Low
- General computer experience: Yes

### Applicant

- Age: 20-60
- Frequency of use: Frequently
- Consequence of a mistake: Low
- General computer experience: No

## TASK ANALYSIS

We identify and analysis the tasks of all the users separately.

### Teacher

The Teacher perform the following tasks.

#### 1. Registration

**Goal:** Register to the system as a teacher.

**Pre-condition:**

- Need to be a teacher of IIT

**Sub-tasks:**

- Apply for registration by fulfilling the form
- Approve by program chairperson

#### 2. Manage Courses

**Goal:** Update courses details, marks and notify students.

**Pre-condition:**

- Logged in as a teacher.

**Sub-tasks:**

- Update courses details.
- Update marks.
- Notify students.

#### 3. View Viva Schedule

**Goal:** View and download the viva notices.

**Pre-condition:**

- Logged in as a teacher.

**Sub-tasks:**

- View the notices.
- Download from the web.

## Student

The student performs the following tasks.

### 1. View Mark, CGPA and Result

**Goal:** View the mark sheet, CGPA and result of students.

**Pre-condition:**

- Logged in as a student.

**Sub-tasks:**

- View mark sheet, CGPA and result in the dashboard.

## Program Chairperson

The program chairperson performs the following tasks.

### 1. Manage Users

**Goal:** Create, update user accounts and approve/deny new accounts request.

**Pre-condition:**

- Logged in as program chairperson.
- Request from teachers.

**Sub-tasks:**

- Create user account.
- Update user account.
- Approve/deny teacher's account request.

### 2. Manage Courses

**Goal:** Add/Change courses and add students to courses.

**Pre-condition:**

- Logged in as program chairperson.

**Sub-tasks:**

- Add/Change courses.
- Add students to courses.

## Exam Controller

The exam controller performs the following tasks.

### 1. Initiate the Admission Process

**Goal:** Initiate the admission process by uploading the admission circular.

**Pre-condition:**

- Logged in as exam controller.

**Sub-tasks:**

- Upload the circular on the web.

### 2. Manage Schedule

**Goal:** Add viva schedule and update exam room.

**Pre-condition:**

- Logged in as exam controller.
- Scrutinizing team filters the applications.

**Sub-tasks:**

- Add viva schedule.
- Update exam rooms.

## Scrutinizing Team

The scrutinizing team perform the following tasks.

### 1. Download Applications to Filter

**Goal:** Download all the applications to filter them manually.

**Pre-condition:**

- Logged in as scrutinizing team
- The applicant's payment verified by the receptionist.

**Sub-tasks:**

- Download the applications.

## Receptionist

The receptionists perform the following tasks.

### 1. Update Payment Status

**Goal:** Update the payment status of applicants after paying the fees.

**Pre-condition:**

- Logged in as receptionist.
- The applicant paid the fees manually.

**Sub-tasks:**

- Update payment status.

### 2. Search Applicants

**Goal:** Search applicants by name or phone number.

**Pre-condition:**

- Logged in as receptionist.

**Sub-tasks:**

- Search by name or phone number.

## Applicant

The applicants can perform the following tasks.

### 1. Apply for Admission

**Goal:** apply for admitting in PGDIT.

**Pre-condition:**

- The exam controller uploads the circular.

**Sub-tasks:**

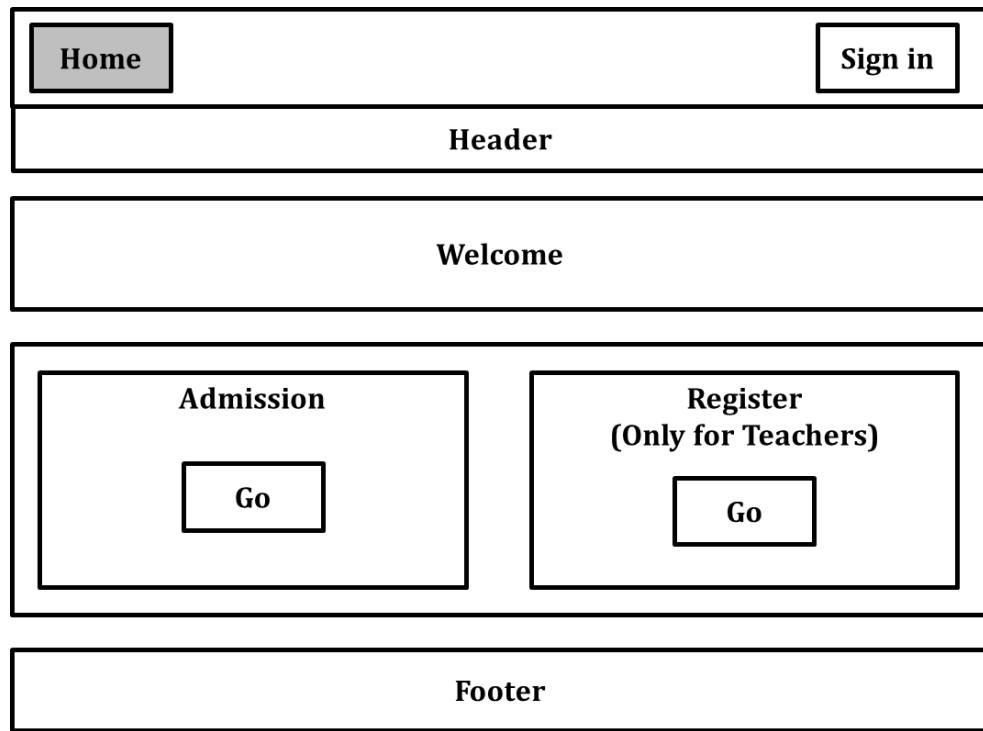
- Visit the website.
- Apply by fulfilling the form and submit.

## 3.2 Interface Design Steps

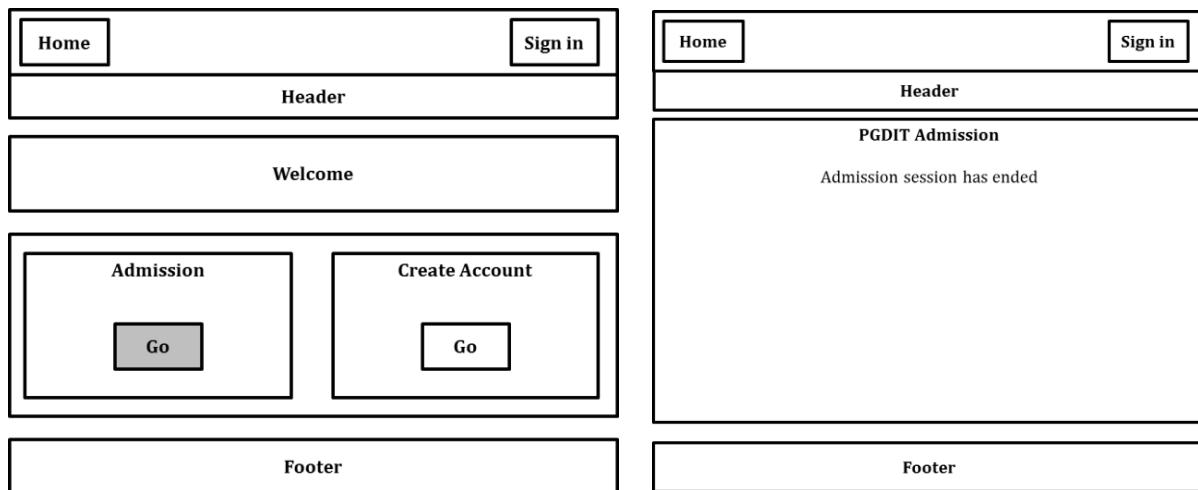
We follow the following steps to design the Library Circulation System (LCS) user interface.

- i. Define interface objects and actions
- ii. Define events that will cause the state of the user interface to change
- iii. Depict each interface state as it look to end user

### DEFINE INTERFACE OBJECTS AND ACTIONS



**Figure 128: External (Home)**



**Figure 129: External (Admission – Session ended)**

This wireframe shows a more detailed version of the website. It includes a 'Header' with 'Home' and 'Sign in' buttons. Below the header is a 'Welcome' section. The main content area contains a 'Registration Form' with fields for Name, Father's name, Mother's name, Email, Contact No, Photo, Date of birth, Nationality, and Permanent Address. To the right of these fields are corresponding input fields. Below this is an 'Academic career' section with fields for SSC/O-level, HSC/A-level, and Bachelor level. To the right of these are input fields. Further down is a 'Marks in mathematics' section with fields for HSC/A-level and Bachelor level. To the right of these are input fields. A 'Submit' button is located at the bottom right of the form area. The footer section is at the bottom of the page.

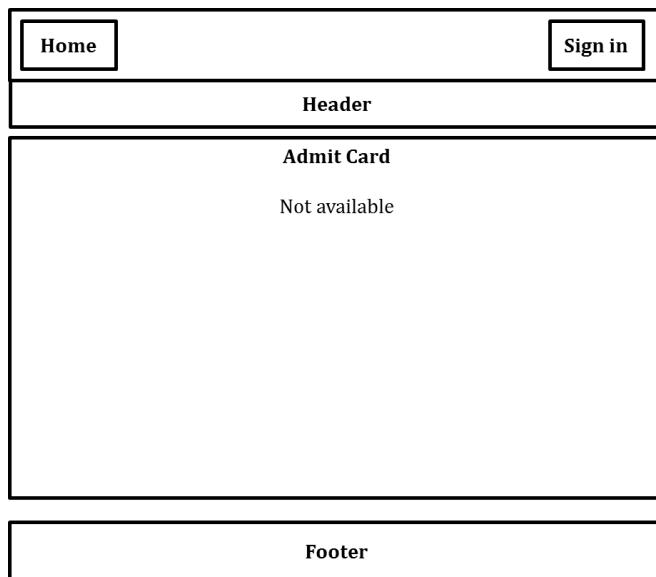
**Figure 130: External (Admission - Session going admission form)**

<b>Home</b>	<b>Sign in</b>																																		
<b>Header</b>																																			
<b>Registration Form</b> <table> <tr><td>Name</td><td><input type="text"/></td></tr> <tr><td>Father's name</td><td><input type="text"/></td></tr> <tr><td>Mother's name</td><td><input type="text"/></td></tr> <tr><td>Email</td><td><input type="text"/></td></tr> <tr><td>Contact No</td><td><input type="text"/></td></tr> <tr><td>Photo</td><td><input type="file"/></td></tr> <tr><td>Date of birth</td><td><input type="date"/></td></tr> <tr><td>Nationality</td><td><input type="text"/></td></tr> <tr><td>Permanent Address</td><td><input type="text"/></td></tr> <tr><td><b>Academic career</b></td><td><input type="text"/></td></tr> <tr><td>SSC/O-level</td><td><input type="text"/></td></tr> <tr><td>HSC/A-level</td><td><input type="text"/></td></tr> <tr><td>Bachelor level</td><td><input type="text"/></td></tr> <tr><td><b>Marks in mathematics</b></td><td><input type="text"/></td></tr> <tr><td>HSC/A-level</td><td><input type="text"/></td></tr> <tr><td>Bachelor level</td><td><input type="text"/></td></tr> </table>		Name	<input type="text"/>	Father's name	<input type="text"/>	Mother's name	<input type="text"/>	Email	<input type="text"/>	Contact No	<input type="text"/>	Photo	<input type="file"/>	Date of birth	<input type="date"/>	Nationality	<input type="text"/>	Permanent Address	<input type="text"/>	<b>Academic career</b>	<input type="text"/>	SSC/O-level	<input type="text"/>	HSC/A-level	<input type="text"/>	Bachelor level	<input type="text"/>	<b>Marks in mathematics</b>	<input type="text"/>	HSC/A-level	<input type="text"/>	Bachelor level	<input type="text"/>		
Name	<input type="text"/>																																		
Father's name	<input type="text"/>																																		
Mother's name	<input type="text"/>																																		
Email	<input type="text"/>																																		
Contact No	<input type="text"/>																																		
Photo	<input type="file"/>																																		
Date of birth	<input type="date"/>																																		
Nationality	<input type="text"/>																																		
Permanent Address	<input type="text"/>																																		
<b>Academic career</b>	<input type="text"/>																																		
SSC/O-level	<input type="text"/>																																		
HSC/A-level	<input type="text"/>																																		
Bachelor level	<input type="text"/>																																		
<b>Marks in mathematics</b>	<input type="text"/>																																		
HSC/A-level	<input type="text"/>																																		
Bachelor level	<input type="text"/>																																		
<b>Submit</b>																																			
<b>Footer</b>																																			
<b>Home</b>	<b>Sign in</b>																																		
<b>Header</b>																																			
<b>You have successfully applied for PGDIT Admission</b> <table> <tr><td><b>Application ID</b></td><td><input type="text"/></td></tr> <tr><td>Name</td><td><input type="text"/></td></tr> <tr><td>Father's name</td><td><input type="text"/></td></tr> <tr><td>Mother's name</td><td><input type="text"/></td></tr> <tr><td>Email</td><td><input type="text"/></td></tr> <tr><td>Contact No</td><td><input type="text"/></td></tr> <tr><td>Photo</td><td><input type="file"/></td></tr> <tr><td>Date of birth</td><td><input type="date"/></td></tr> <tr><td>Nationality</td><td><input type="text"/></td></tr> <tr><td>Permanent Address</td><td><input type="text"/></td></tr> <tr><td><b>Academic career</b></td><td><input type="text"/></td></tr> <tr><td>SSC/O-level</td><td><input type="text"/></td></tr> <tr><td>HSC/A-level</td><td><input type="text"/></td></tr> <tr><td>Bachelor level</td><td><input type="text"/></td></tr> <tr><td><b>Marks in mathematics</b></td><td><input type="text"/></td></tr> <tr><td>HSC/A-level</td><td><input type="text"/></td></tr> <tr><td>Bachelor level</td><td><input type="text"/></td></tr> </table>		<b>Application ID</b>	<input type="text"/>	Name	<input type="text"/>	Father's name	<input type="text"/>	Mother's name	<input type="text"/>	Email	<input type="text"/>	Contact No	<input type="text"/>	Photo	<input type="file"/>	Date of birth	<input type="date"/>	Nationality	<input type="text"/>	Permanent Address	<input type="text"/>	<b>Academic career</b>	<input type="text"/>	SSC/O-level	<input type="text"/>	HSC/A-level	<input type="text"/>	Bachelor level	<input type="text"/>	<b>Marks in mathematics</b>	<input type="text"/>	HSC/A-level	<input type="text"/>	Bachelor level	<input type="text"/>
<b>Application ID</b>	<input type="text"/>																																		
Name	<input type="text"/>																																		
Father's name	<input type="text"/>																																		
Mother's name	<input type="text"/>																																		
Email	<input type="text"/>																																		
Contact No	<input type="text"/>																																		
Photo	<input type="file"/>																																		
Date of birth	<input type="date"/>																																		
Nationality	<input type="text"/>																																		
Permanent Address	<input type="text"/>																																		
<b>Academic career</b>	<input type="text"/>																																		
SSC/O-level	<input type="text"/>																																		
HSC/A-level	<input type="text"/>																																		
Bachelor level	<input type="text"/>																																		
<b>Marks in mathematics</b>	<input type="text"/>																																		
HSC/A-level	<input type="text"/>																																		
Bachelor level	<input type="text"/>																																		
<b>Footer</b>																																			

Figure 131: External (Admission - Session going - Admission form - Submit)

<b>Home</b>	<b>Sign in</b>						
<b>Header</b>							
<b>Welcome</b>							
<table> <tr><td><b>Admission</b></td><td><b>Create Account</b></td></tr> <tr><td><b>Go</b></td><td><b>Go</b></td></tr> </table>		<b>Admission</b>	<b>Create Account</b>	<b>Go</b>	<b>Go</b>		
<b>Admission</b>	<b>Create Account</b>						
<b>Go</b>	<b>Go</b>						
<b>Footer</b>							
<b>Home</b>	<b>Sign in</b>						
<b>Header</b>							
<b>Admit Card</b> <table> <tr><td>Application ID</td><td><input type="text"/></td></tr> <tr><td>Date of Birth</td><td><input type="text"/></td></tr> <tr><td colspan="2"><b>Download Admit card</b></td></tr> </table>		Application ID	<input type="text"/>	Date of Birth	<input type="text"/>	<b>Download Admit card</b>	
Application ID	<input type="text"/>						
Date of Birth	<input type="text"/>						
<b>Download Admit card</b>							
<b>Footer</b>							

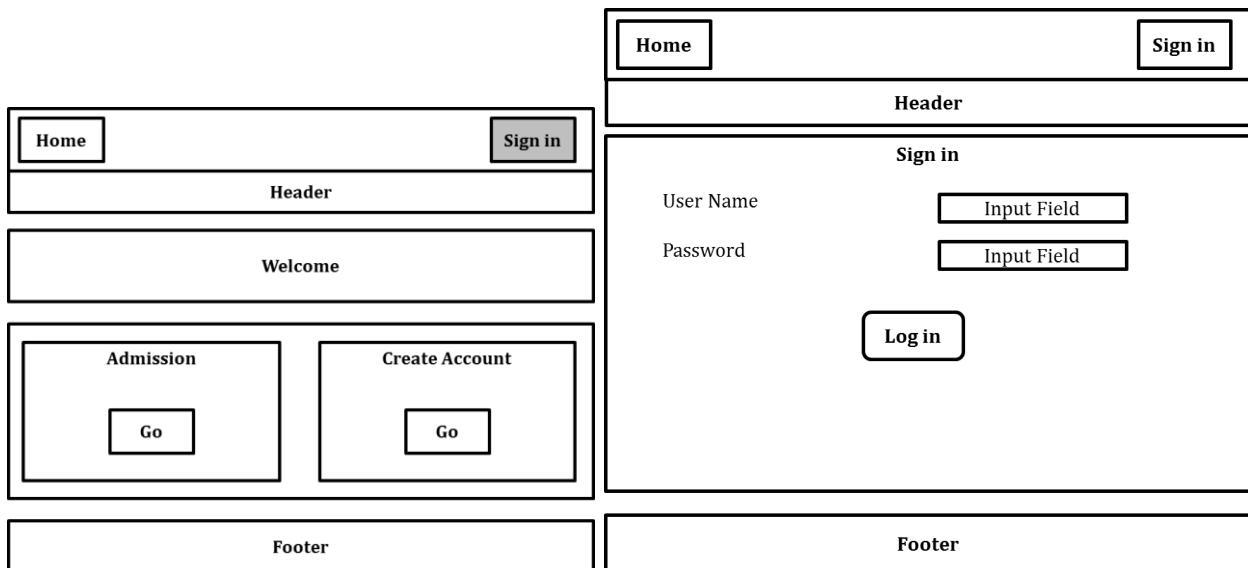
Figure 132: External (Admission – Session going – Admit card - Available)



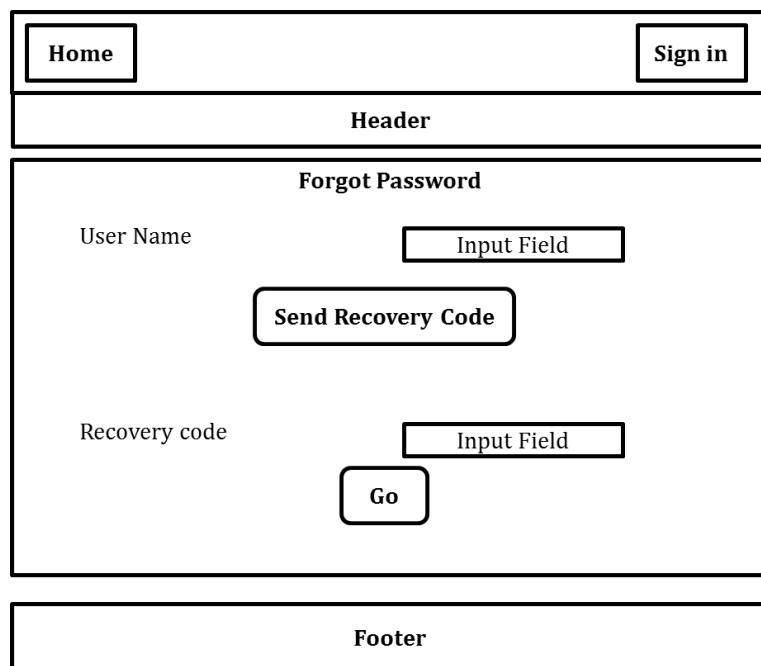
**Figure 133: External (Admission – Session going – Admit card – Not Available)**

The diagram illustrates a web page structure. At the top is a header bar containing 'Home' and 'Sign in' buttons. Below this is a large central content area labeled 'Header' at the top. Inside this area, the text 'Registration Form (Only for Teachers)' is displayed. To the left of the form fields, there is a vertical list of field labels: 'Full Name', 'Username', 'Father's name', 'Mother's name', 'Email', 'Mobile Number', 'Teacher Registration Number', and 'Present Address'. To the right of each label is a corresponding 'Input Field' box. At the bottom right of the form area is a 'Submit' button. On the far left, there is a sidebar with 'Home' and 'Sign in' buttons, a 'Header' section, a 'Welcome' message, and two buttons: 'Admission' (with a 'Go' button) and 'Create Account' (with a 'Go' button). At the bottom of the page are two footer bars.

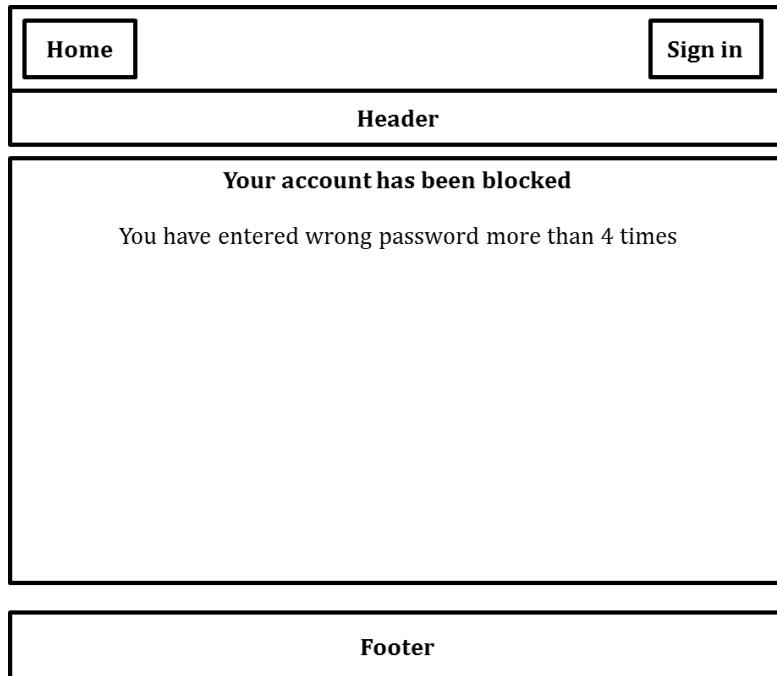
**Figure 134: External (Create Account)**



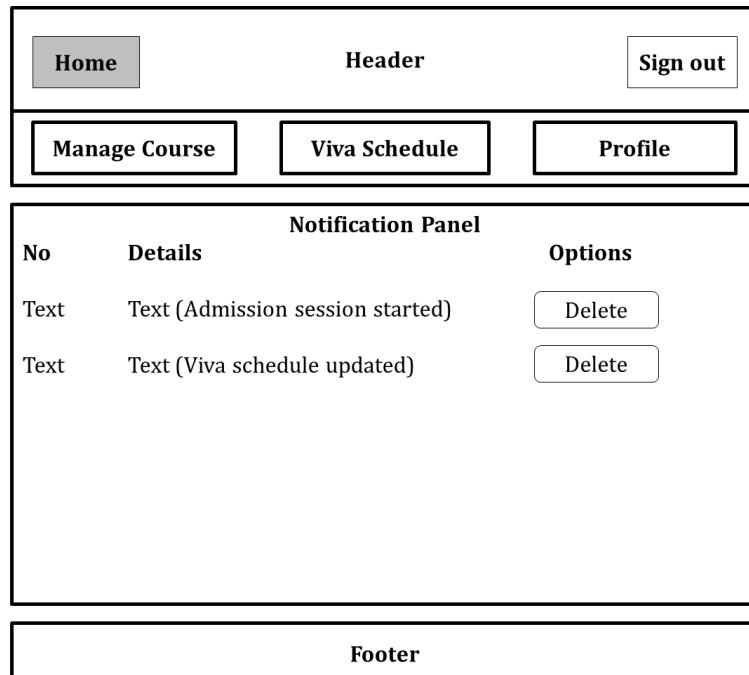
**Figure 135: External (Sing in)**



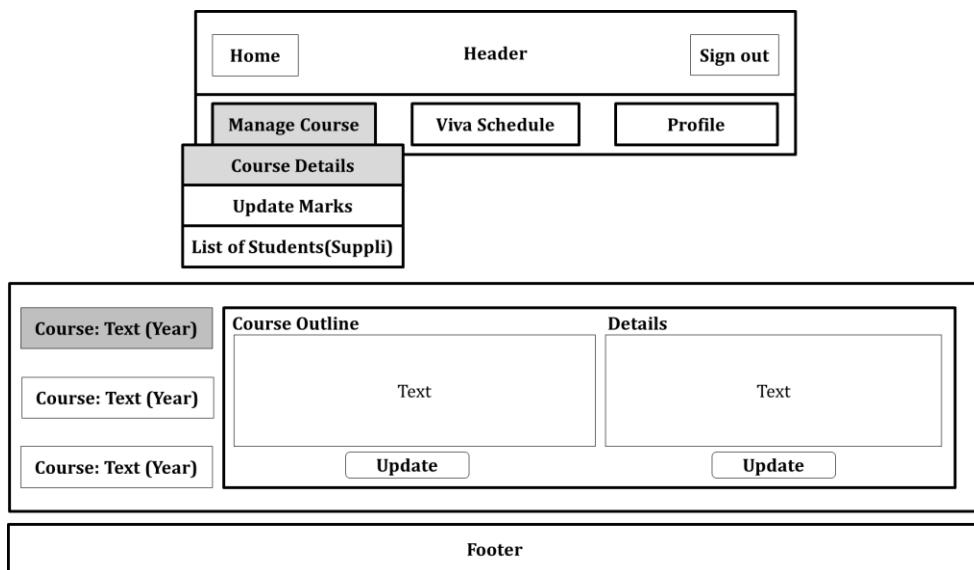
**Figure 136: External (Sing in-Forgot password)**



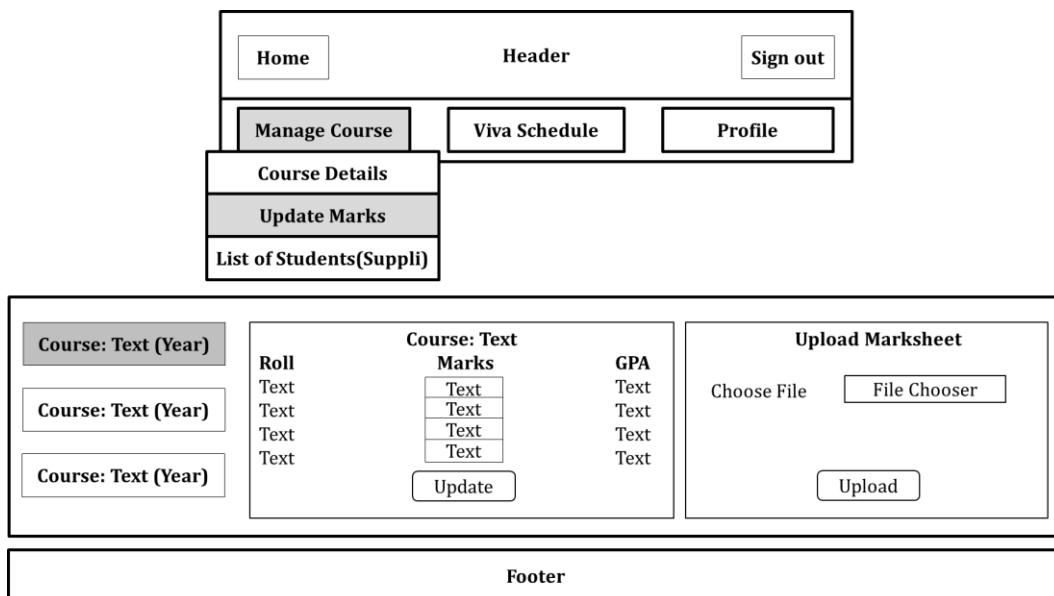
**Figure 137: External (Sing in - Blocked)**



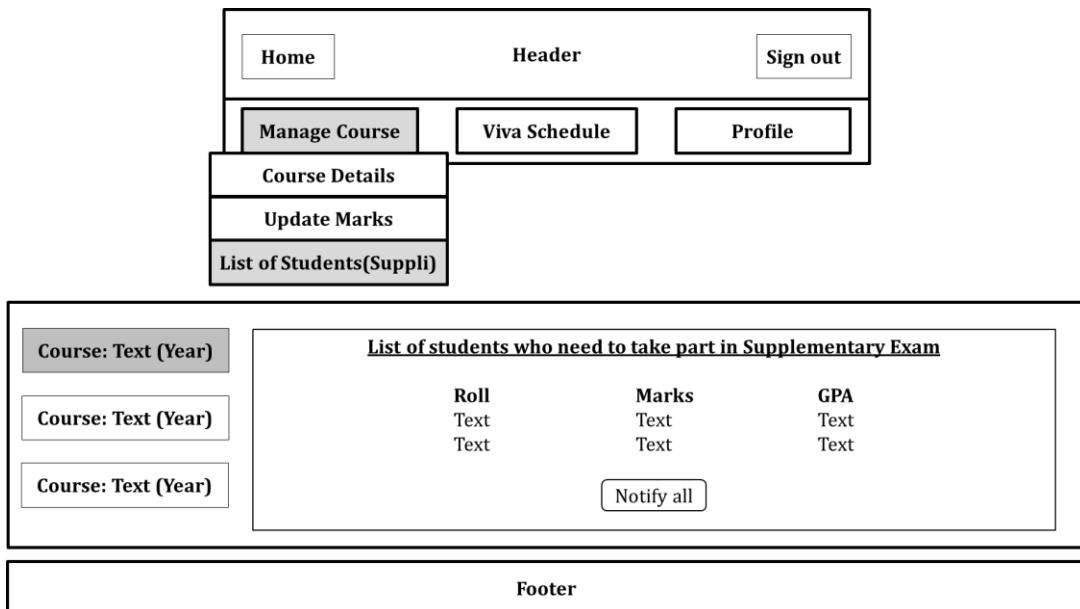
**Figure 138: Internal (Teacher: Home)**



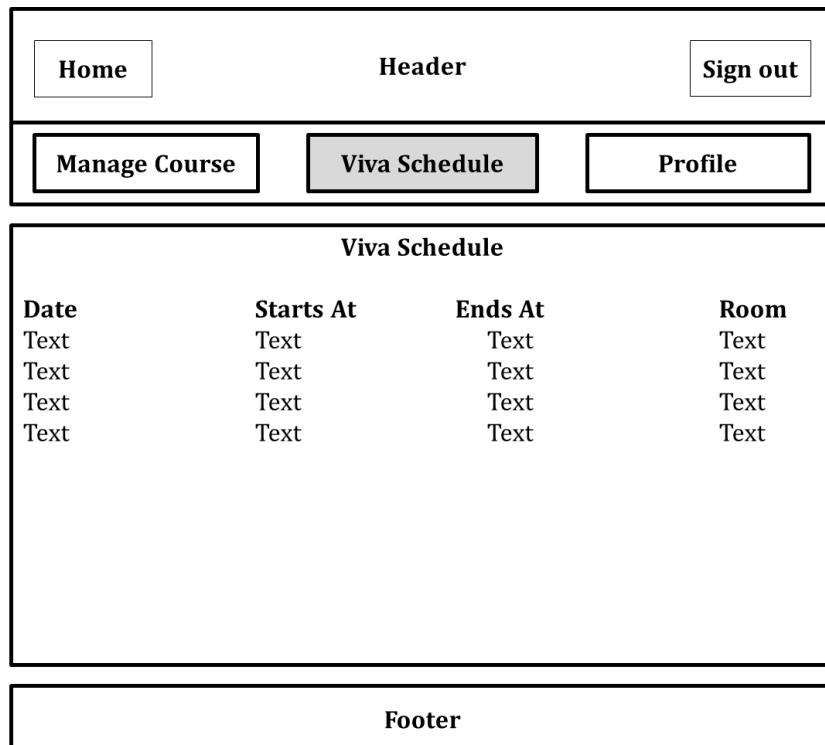
**Figure 139: Internal (Teacher: Manage Course - Course Details)**



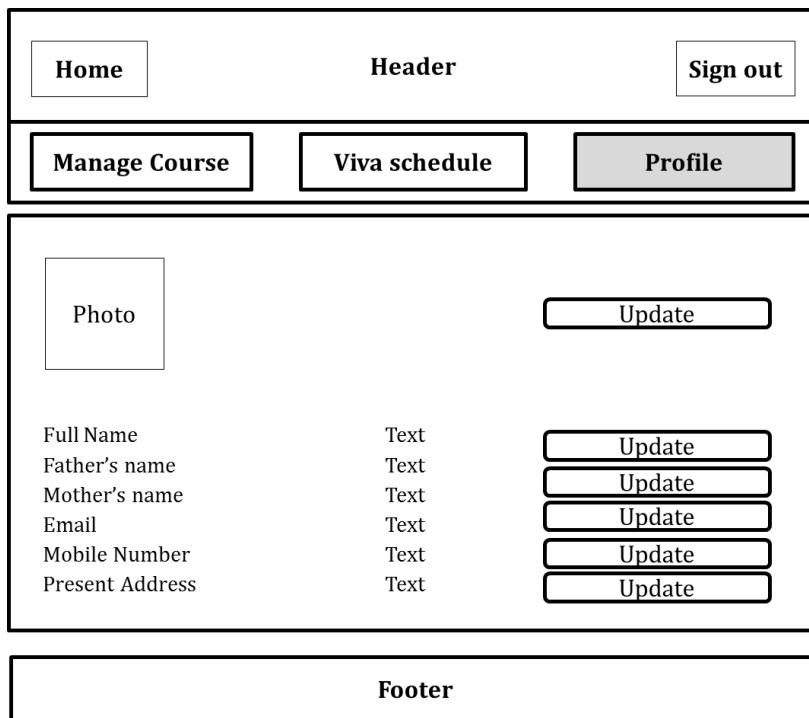
**Figure 140: Internal (Teacher: Manage Course - Update Marks)**



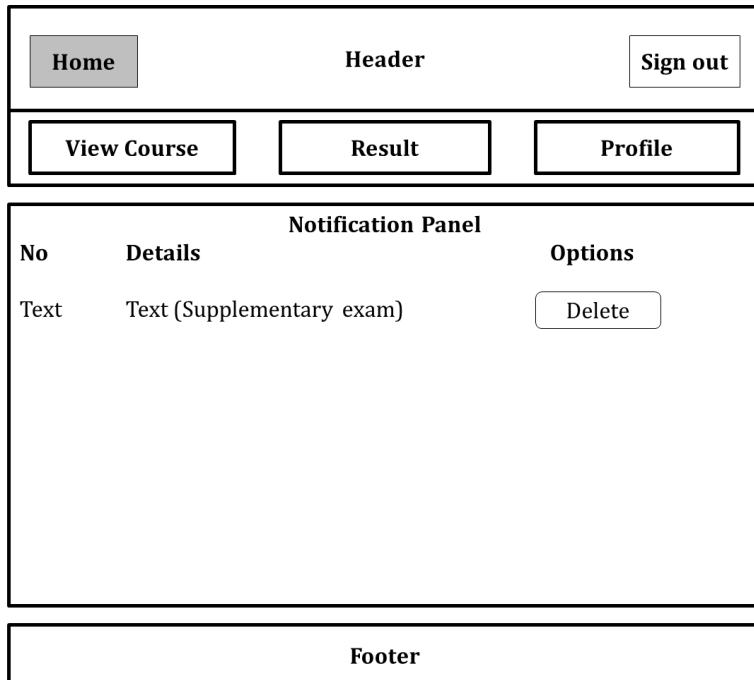
**Figure 141: Internal (Teacher: Manage Course - List of Students (Suppli))**



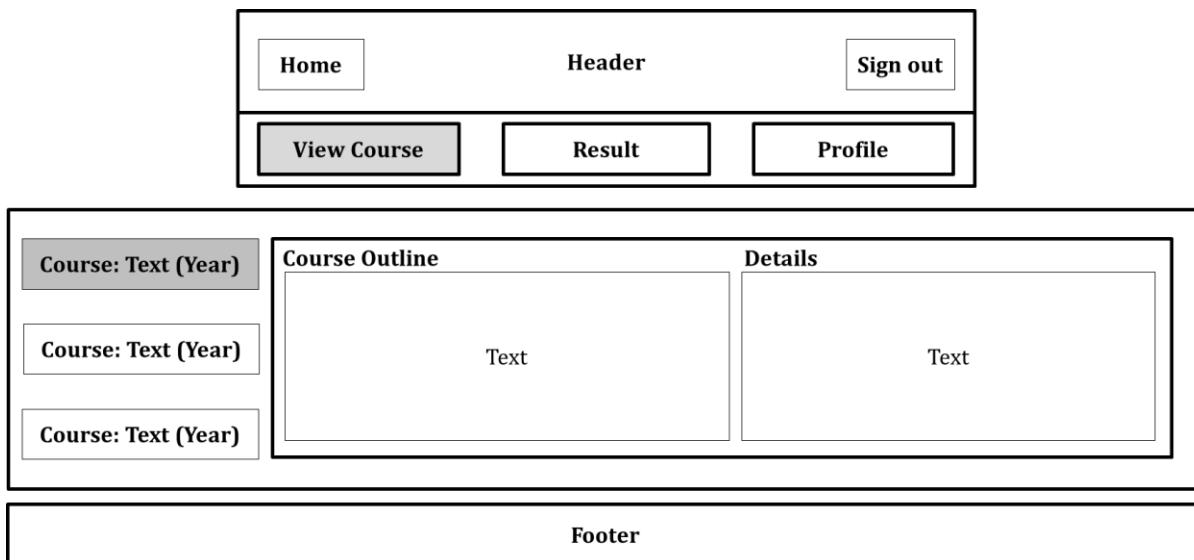
**Figure 142: Internal (Teacher: Viva Schedule)**



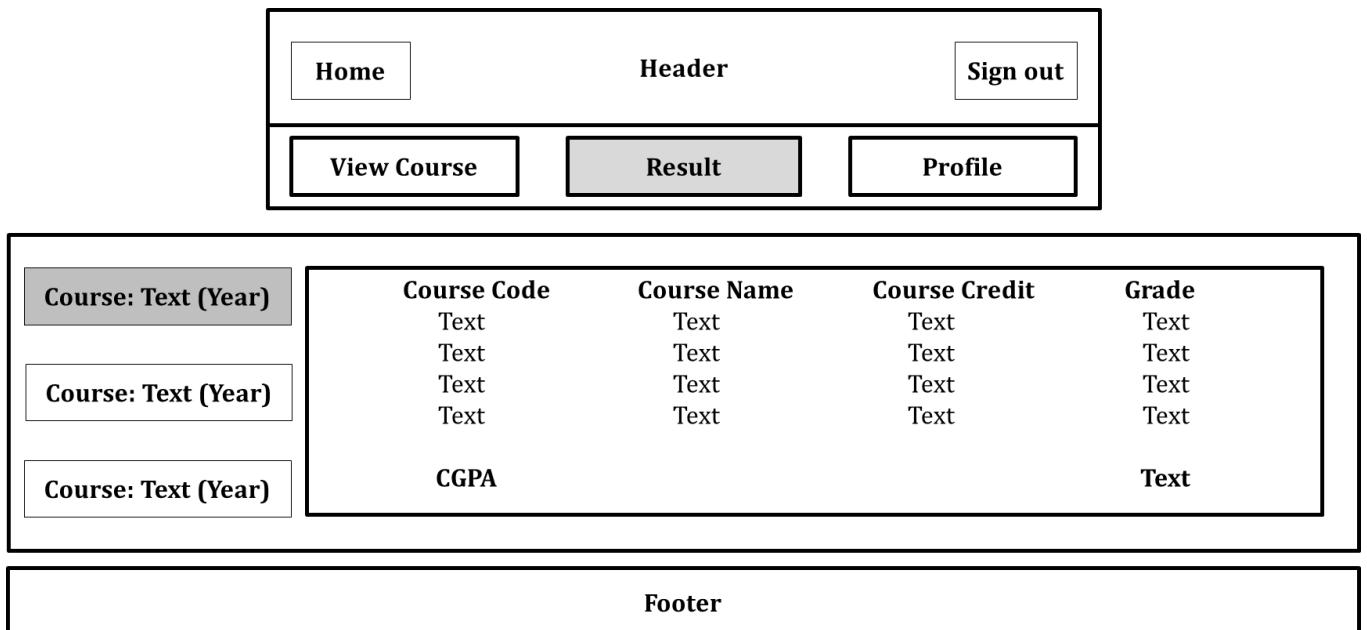
**Figure 143: Internal (Member: Profile)**



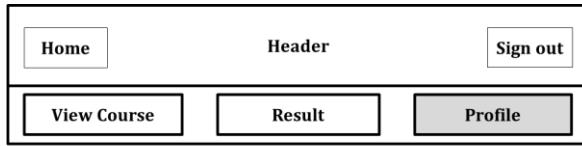
**Figure 144: Internal (Student: Home)**



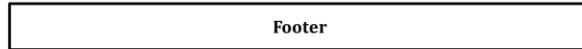
**Figure 145: Internal (Student: View Course)**



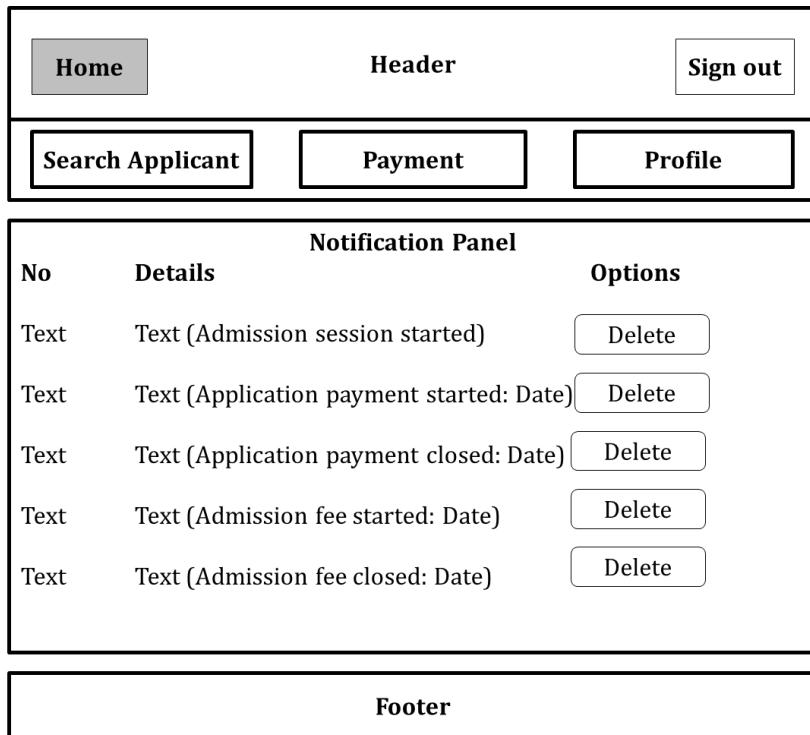
**Figure 146: Internal (Student: View Course)**



**Same as Teacher: Profile**



**Figure 147: Internal (Student: Profile)**



**Figure 148: Internal (Receptionist: Home)**

Home	<b>Header</b>	Sign out																
<input type="button" value="Search Applicant"/> <input type="button" value="Payment"/> <input type="button" value="Profile"/>																		
<input type="button" value="Search by name"/>																		
<input type="button" value="Search by mobile no"/>																		
<p>Applicant name:</p> <div style="border: 1px solid black; padding: 2px; width: 150px; height: 20px; margin-bottom: 5px;"></div> <input type="button" value="Input Field"/> <input type="button" value="Search"/>	<p style="text-align: center;"><b><u>Applicant Information</u></b></p> <table border="0"> <tbody> <tr> <td>Application ID</td> <td>Text</td> </tr> <tr> <td>Name of Applicant</td> <td>Text</td> </tr> <tr> <td>Father's name</td> <td>Text</td> </tr> <tr> <td>Mother's name</td> <td>Text</td> </tr> <tr> <td>Email</td> <td>Text</td> </tr> <tr> <td>Mobile Number</td> <td>Text</td> </tr> <tr> <td>Present Address</td> <td>Text</td> </tr> <tr> <td>Date of Birth</td> <td>Text</td> </tr> </tbody> </table> <div style="border: 1px solid black; width: 150px; height: 50px; float: right; margin-top: 10px; margin-left: 10px;"></div> <p style="text-align: center;">Photo</p>		Application ID	Text	Name of Applicant	Text	Father's name	Text	Mother's name	Text	Email	Text	Mobile Number	Text	Present Address	Text	Date of Birth	Text
Application ID	Text																	
Name of Applicant	Text																	
Father's name	Text																	
Mother's name	Text																	
Email	Text																	
Mobile Number	Text																	
Present Address	Text																	
Date of Birth	Text																	
<b>Footer</b>																		

**Figure 149: Internal (Receptionist: Search Applicant-By name)**

Home	<b>Header</b>	Sign out																
<input type="button" value="Search Applicant"/> <input type="button" value="Payment"/> <input type="button" value="Profile"/>																		
<input type="button" value="Search by name"/>																		
<input type="button" value="Search by mobile no"/>																		
<p>Applicant's Mobile No:</p> <div style="border: 1px solid black; padding: 2px; width: 150px; height: 20px; margin-bottom: 5px;"></div> <input type="button" value="Input Field"/> <input type="button" value="Search"/>	<p style="text-align: center;"><b><u>Applicant Information</u></b></p> <table border="0"> <tbody> <tr> <td>Application ID</td> <td>Text</td> </tr> <tr> <td>Name of Applicant</td> <td>Text</td> </tr> <tr> <td>Father's name</td> <td>Text</td> </tr> <tr> <td>Mother's name</td> <td>Text</td> </tr> <tr> <td>Email</td> <td>Text</td> </tr> <tr> <td>Mobile Number</td> <td>Text</td> </tr> <tr> <td>Present Address</td> <td>Text</td> </tr> <tr> <td>Date of Birth</td> <td>Text</td> </tr> </tbody> </table> <div style="border: 1px solid black; width: 150px; height: 50px; float: right; margin-top: 10px; margin-left: 10px;"></div> <p style="text-align: center;">Photo</p>		Application ID	Text	Name of Applicant	Text	Father's name	Text	Mother's name	Text	Email	Text	Mobile Number	Text	Present Address	Text	Date of Birth	Text
Application ID	Text																	
Name of Applicant	Text																	
Father's name	Text																	
Mother's name	Text																	
Email	Text																	
Mobile Number	Text																	
Present Address	Text																	
Date of Birth	Text																	
<b>Footer</b>																		

**Figure 150: Internal (Receptionist: Search Applicant-By mobile no)**

Home	Header	Sign out																												
Search Applicant	Payment	Profile																												
Application payment																														
Admission fees																														
<p>Application ID:</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;">Input Field</div> <div style="border: 1px solid black; padding: 5px; width: fit-content;">Paid</div> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; padding: 5px;">No</th> <th style="text-align: left; padding: 5px;">Application ID</th> <th style="text-align: left; padding: 5px;">Name</th> <th style="text-align: left; padding: 5px;">Contact No</th> </tr> </thead> <tbody> <tr><td style="padding: 5px;">Text</td><td style="padding: 5px;">Text</td><td style="padding: 5px;">Text</td><td style="padding: 5px;">Text</td></tr> <tr><td style="padding: 5px;">Text</td><td style="padding: 5px;">Text</td><td style="padding: 5px;">Text</td><td style="padding: 5px;">Text</td></tr> <tr><td style="padding: 5px;">Text</td><td style="padding: 5px;">Text</td><td style="padding: 5px;">Text</td><td style="padding: 5px;">Text</td></tr> <tr><td style="padding: 5px;">Text</td><td style="padding: 5px;">Text</td><td style="padding: 5px;">Text</td><td style="padding: 5px;">Text</td></tr> <tr><td style="padding: 5px;">Text</td><td style="padding: 5px;">Text</td><td style="padding: 5px;">Text</td><td style="padding: 5px;">Text</td></tr> <tr><td style="padding: 5px;">Text</td><td style="padding: 5px;">Text</td><td style="padding: 5px;">Text</td><td style="padding: 5px;">Text</td></tr> </tbody> </table>			No	Application ID	Name	Contact No	Text																							
No	Application ID	Name	Contact No																											
Text	Text	Text	Text																											
Text	Text	Text	Text																											
Text	Text	Text	Text																											
Text	Text	Text	Text																											
Text	Text	Text	Text																											
Text	Text	Text	Text																											
Footer																														

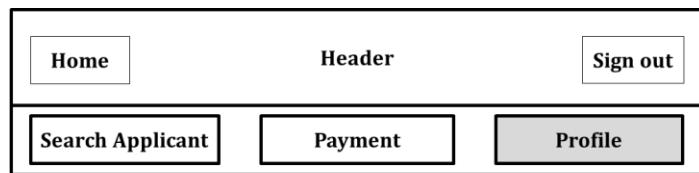
Home	Header	Sign out
Search Applicant	Payment	Profile
Application payment		
Admission fees		
Payment Session is closed		
Footer		

**Figure 151: Internal (Receptionist: Payment-Application payment)**

Home	Header	Sign out																																
Search Applicant	Payment	Profile																																
Application payment																																		
Admission fees																																		
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 20%;">Application ID:</td> <td colspan="3" style="text-align: center;"><b><u>List of Students who have paid Admission fees</u></b></td> </tr> <tr> <td style="text-align: center;">Input Field</td> <td style="text-align: center;">No</td> <td style="text-align: center;">Application ID</td> <td style="text-align: center;">Name</td> </tr> <tr> <td></td> <td>Text</td> <td>Text</td> <td>Text</td> </tr> </table>			Application ID:	<b><u>List of Students who have paid Admission fees</u></b>			Input Field	No	Application ID	Name		Text	Text	Text																				
Application ID:	<b><u>List of Students who have paid Admission fees</u></b>																																	
Input Field	No	Application ID	Name																															
	Text	Text	Text																															
	Text	Text	Text																															
	Text	Text	Text																															
	Text	Text	Text																															
	Text	Text	Text																															
	Text	Text	Text																															
Paid																																		
Footer																																		

Home	Header	Sign out
Search Applicant	Payment	Profile
Application payment		
Admission fees		
Payment Session is closed		
Footer		

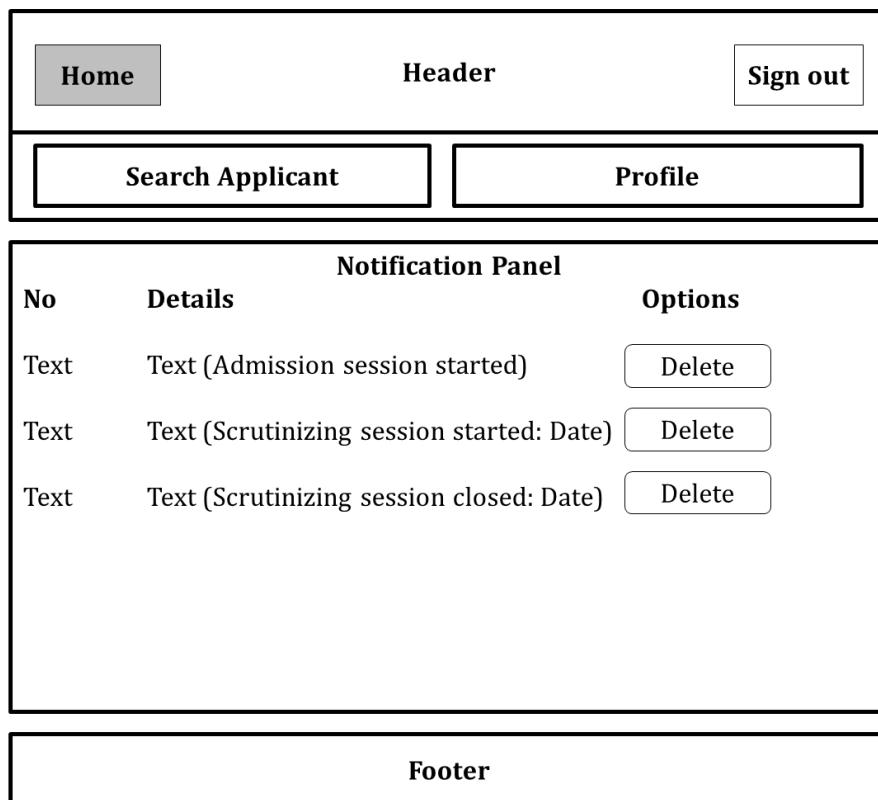
**Figure 152: Internal (Receptionist: Payment-Admission fees)**



**Same as Teacher: Profile**

Footer

**Figure 153: Internal (Receptionist: Profile)**



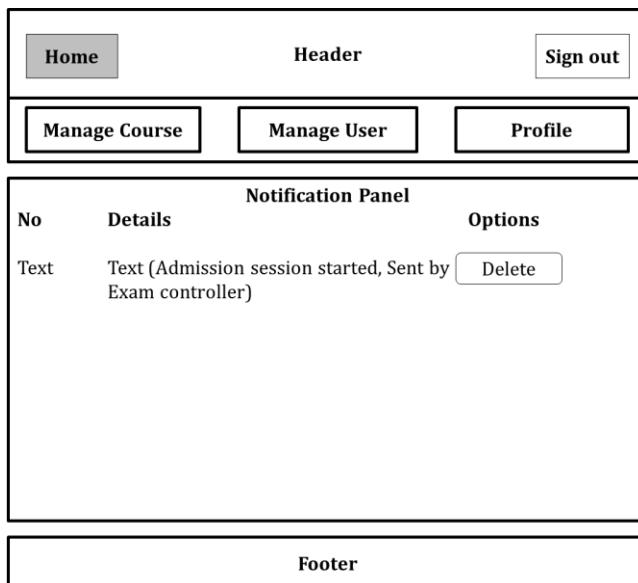
**Figure 154: Internal (Scrutinizer: Home)**

Home	Header	Sign out																																		
<input type="button" value="Search Applicant"/> <input type="button" value="Profile"/>																																				
Application ID: <input type="text" value="Input Field"/>  <input type="button" value="Search"/>  <input type="button" value="Download all"/>	<p><b>Applicant Information</b></p> <table border="0"> <tr><td>Application ID</td><td>Text</td></tr> <tr><td>Name</td><td>Text</td></tr> <tr><td>Father's name</td><td>Text</td></tr> <tr><td>Mother's name</td><td>Text</td></tr> <tr><td>Email</td><td>Text</td></tr> <tr><td>Contact No</td><td>Text</td></tr> <tr><td>Photo</td><td>Photo</td></tr> <tr><td>Date of birth</td><td>Date</td></tr> <tr><td>Nationality</td><td>Text</td></tr> <tr><td>Permanent Address</td><td>Text</td></tr> <tr><td colspan="2"><b>Academic career</b></td></tr> <tr><td>SSC/O-level</td><td>Text</td></tr> <tr><td>HSC/A-level</td><td>Text</td></tr> <tr><td>Bachelor level</td><td>Text</td></tr> <tr><td colspan="2"><b>Marks in mathematics</b></td></tr> <tr><td>HSC/A-level</td><td>Text</td></tr> <tr><td>Bachelor level</td><td>Text</td></tr> </table> <div style="text-align: right; margin-top: 20px;"> <input type="button" value="Eligible"/> </div>	Application ID	Text	Name	Text	Father's name	Text	Mother's name	Text	Email	Text	Contact No	Text	Photo	Photo	Date of birth	Date	Nationality	Text	Permanent Address	Text	<b>Academic career</b>		SSC/O-level	Text	HSC/A-level	Text	Bachelor level	Text	<b>Marks in mathematics</b>		HSC/A-level	Text	Bachelor level	Text	
Application ID	Text																																			
Name	Text																																			
Father's name	Text																																			
Mother's name	Text																																			
Email	Text																																			
Contact No	Text																																			
Photo	Photo																																			
Date of birth	Date																																			
Nationality	Text																																			
Permanent Address	Text																																			
<b>Academic career</b>																																				
SSC/O-level	Text																																			
HSC/A-level	Text																																			
Bachelor level	Text																																			
<b>Marks in mathematics</b>																																				
HSC/A-level	Text																																			
Bachelor level	Text																																			
<b>Footer</b>																																				

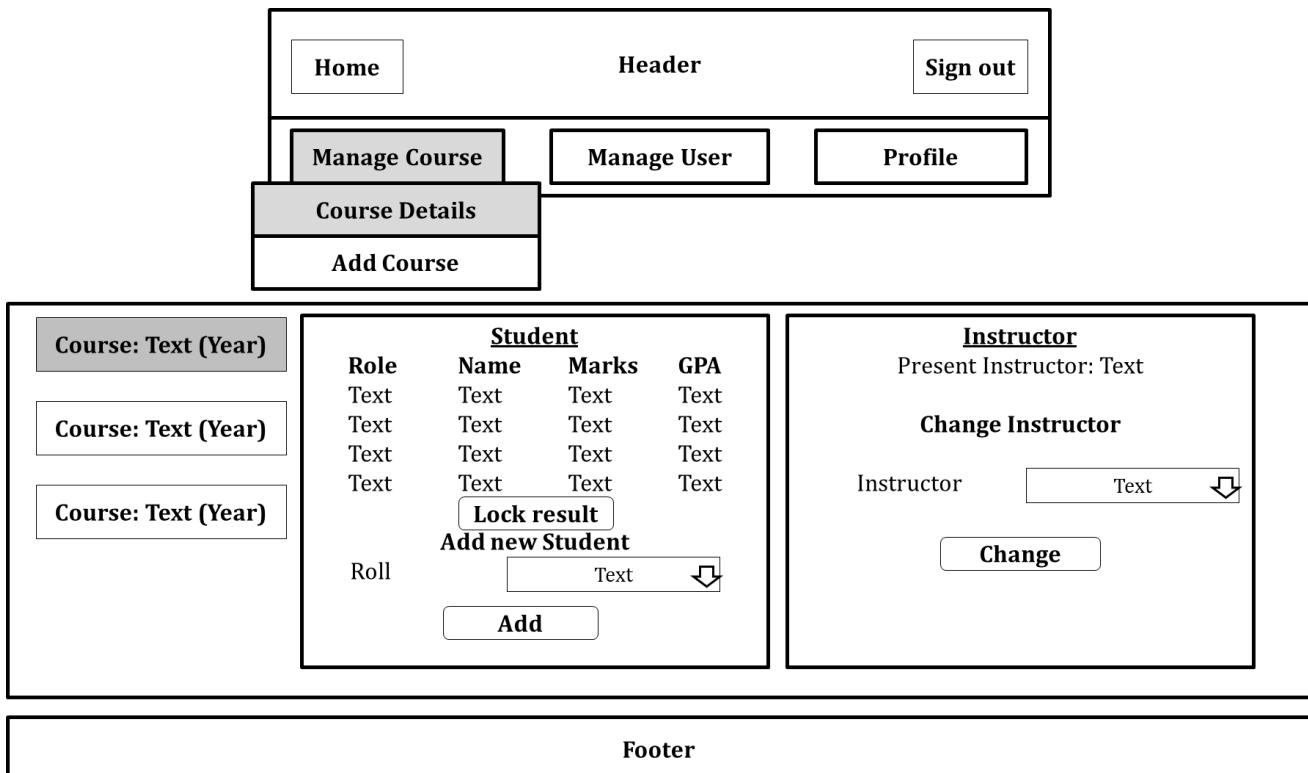
**Figure 155: Internal (Scrutinizer: Search Applicant)**

Home	Header	Sign out
<input type="button" value="Search Applicant"/> <input type="button" value="Profile"/>		
<p><b>Same as Teacher: Profile</b></p>		
<b>Footer</b>		

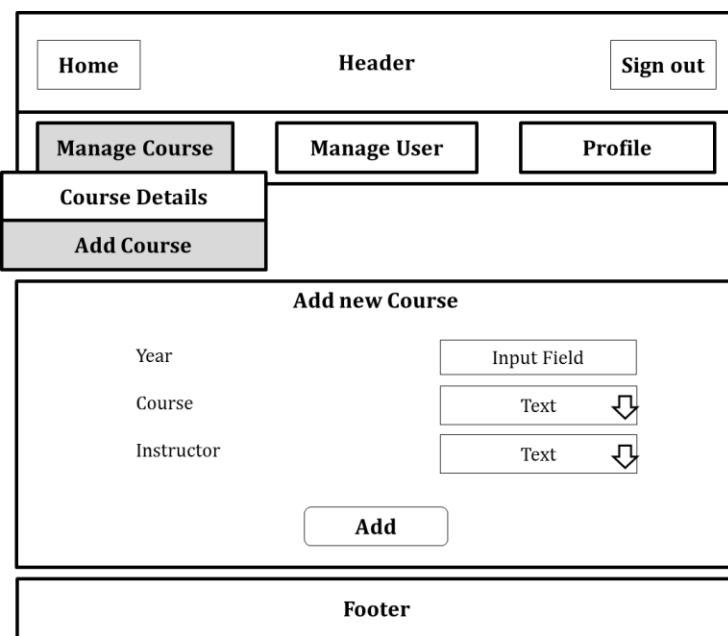
**Figure 156: Internal (Scrutinizer: Profile)**



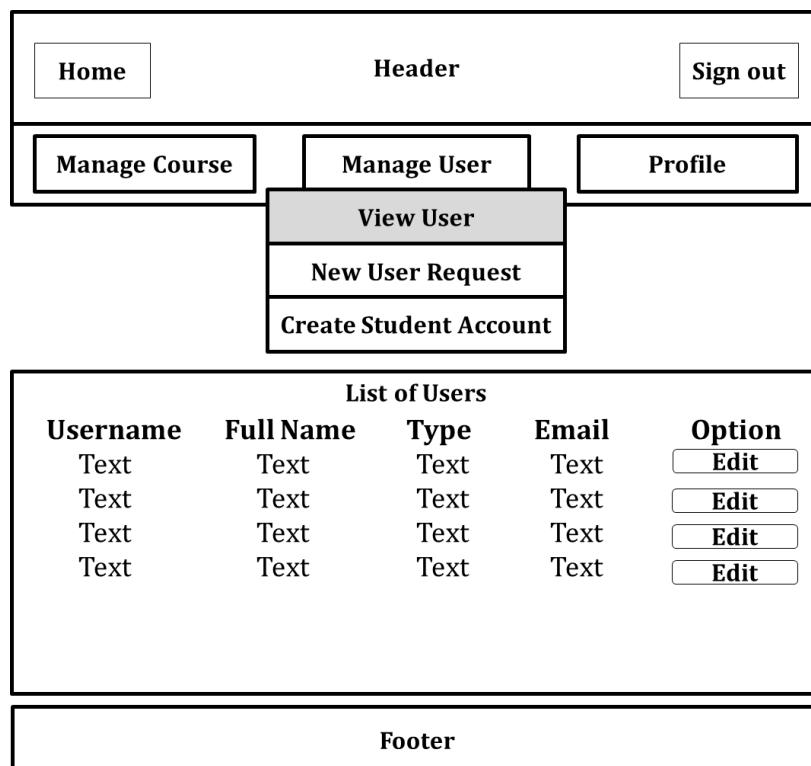
**Figure 157: Internal (Program chairperson: Home)**



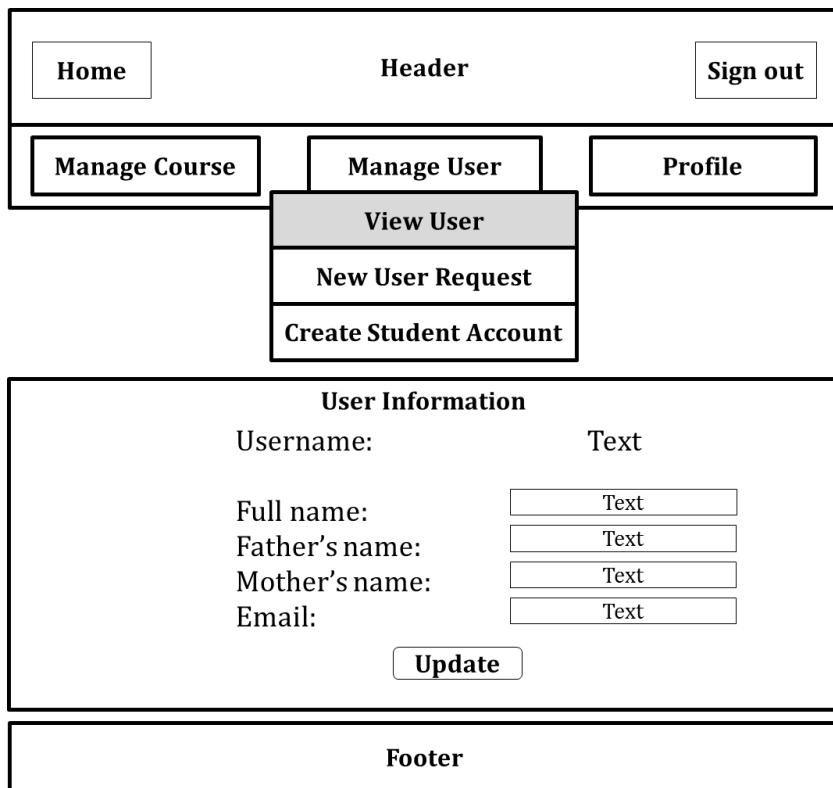
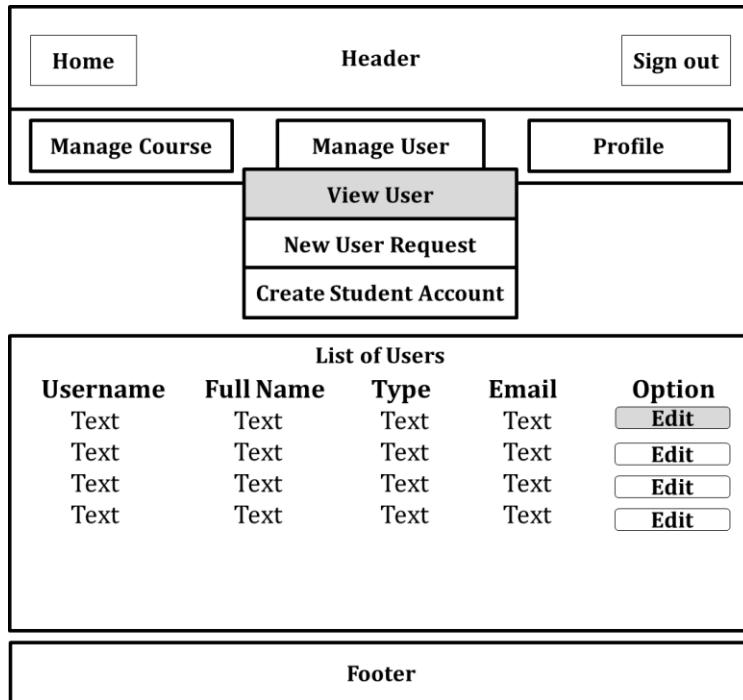
**Figure 158: Internal (Program chairperson: Manage Course- Details)**



**Figure 159: Internal (Program chairperson: Manage Course- Add)**



**Figure 160: Internal (Program chairperson: Manage User- View)**



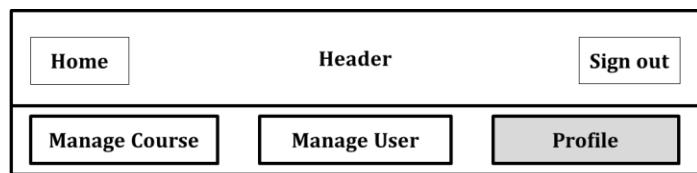
**Figure 161: Internal (Program chairperson: Manage User- View - Edit)**

Home	Header			Sign out
	Manage Course	Manage User	Profile	
		View User		
		New User Request		
		Create Student Account		
Request: 1 Request: 2 Request: 3	Full name Text Father's name Text Mother's name Text Email Text Mobile Number Text Present Address Text Date of Request Text Teacher ID Text Expected Username Text <div style="text-align: right; margin-top: 10px;"> <a href="#">Deny</a>      <a href="#">Approve</a> </div>			
Footer				

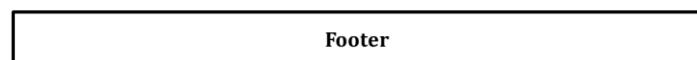
**Figure 162: Internal (Program chairperson: Manage User- New User Request)**

Home	Header			Sign out
	Manage Course	Manage User	Profile	
		View User		
		New User Request		
		Create Student Account		
Name Text Father Text Mother Text Contact Text Email Text Address Text Option Select	Name Text Father Text Mother Text Contact Text Email Text Address Text Option Select	Name Text Father Text Mother Text Contact Text Email Text Address Text Option Select	Name Text Father Text Mother Text Contact Text Email Text Address Text Option Select	Name Text Father Text Mother Text Contact Text Email Text Address Text Option Select
Create account and send credentials through email				
Footer				

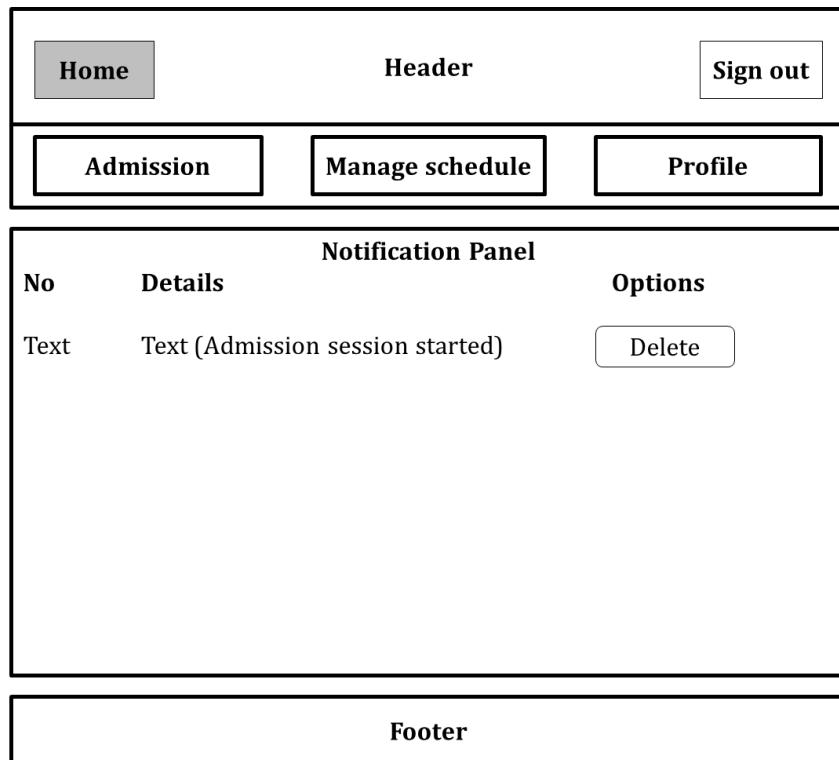
**Figure 163: Internal (Program chairperson: Manage User- Create Student Account)**



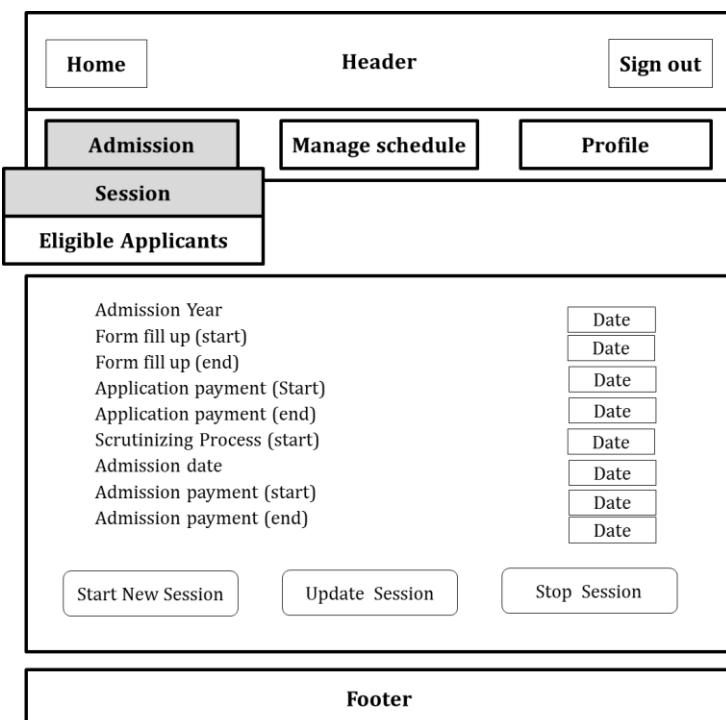
**Same as Teacher: Profile**



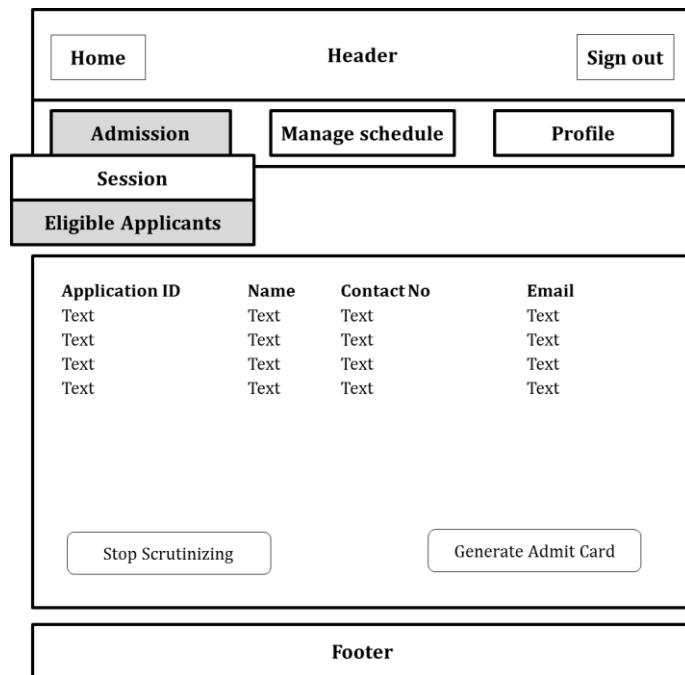
**Figure 164: Internal (Program Chairman: Profile)**



**Figure 165: Internal (Exam controller: Home)**



**Figure 166: Internal (Exam controller: Admission-Session)**



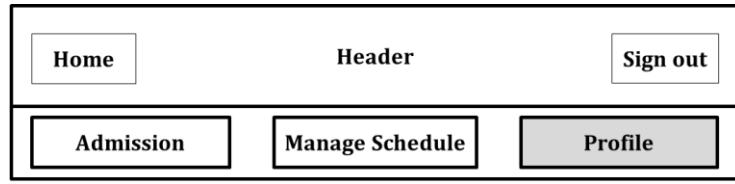
**Figure 167: Internal (Exam controller: Admission-Eligible Applicants)**

Home	Header	Sign out																																																			
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 33%;">Admission</td> <td style="width: 33%;">Manage schedule</td> <td style="width: 33%;">Profile</td> </tr> <tr> <td colspan="3" style="text-align: center; background-color: #cccccc;"><b>Viva Schedule</b></td> </tr> <tr> <td colspan="3" style="text-align: center; background-color: #cccccc;"><b>Update Room</b></td> </tr> </table>			Admission	Manage schedule	Profile	<b>Viva Schedule</b>			<b>Update Room</b>																																												
Admission	Manage schedule	Profile																																																			
<b>Viva Schedule</b>																																																					
<b>Update Room</b>																																																					
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 40%; vertical-align: top;"> <b>Viva Schedule</b> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>Name</td><td>Date</td><td>Starts</td><td>Ends</td><td>Room No</td></tr> <tr><td>Text</td><td>Text</td><td>Text</td><td>Text</td><td>Text</td></tr> <tr><td>Text</td><td>Text</td><td>Text</td><td>Text</td><td>Text</td></tr> <tr><td>Text</td><td>Text</td><td>Text</td><td>Text</td><td>Text</td></tr> <tr><td>Text</td><td>Text</td><td>Text</td><td>Text</td><td>Text</td></tr> </table> </td> <td style="width: 20%; vertical-align: top;"> <b>New Schedule</b> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>Teacher:</td><td>Text</td><td>Select</td></tr> <tr><td>Name:</td><td>Text</td><td>Select</td></tr> <tr><td>Name:</td><td>Text</td><td>Select</td></tr> <tr><td>Date of Viva:</td><td>Input</td><td></td></tr> <tr><td>Starts from:</td><td>Input</td><td></td></tr> <tr><td>Ends at:</td><td>Input</td><td></td></tr> <tr><td>Room No</td><td>Text</td><td>↓</td></tr> <tr><td colspan="2" style="text-align: right;"><b>Add</b></td><td></td></tr> </table> </td> </tr> </table>			<b>Viva Schedule</b> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>Name</td><td>Date</td><td>Starts</td><td>Ends</td><td>Room No</td></tr> <tr><td>Text</td><td>Text</td><td>Text</td><td>Text</td><td>Text</td></tr> <tr><td>Text</td><td>Text</td><td>Text</td><td>Text</td><td>Text</td></tr> <tr><td>Text</td><td>Text</td><td>Text</td><td>Text</td><td>Text</td></tr> <tr><td>Text</td><td>Text</td><td>Text</td><td>Text</td><td>Text</td></tr> </table>	Name	Date	Starts	Ends	Room No	Text	Text	Text	Text	Text	Text	Text	Text	Text	Text	Text	Text	Text	Text	Text	Text	Text	Text	Text	Text	<b>New Schedule</b> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>Teacher:</td><td>Text</td><td>Select</td></tr> <tr><td>Name:</td><td>Text</td><td>Select</td></tr> <tr><td>Name:</td><td>Text</td><td>Select</td></tr> <tr><td>Date of Viva:</td><td>Input</td><td></td></tr> <tr><td>Starts from:</td><td>Input</td><td></td></tr> <tr><td>Ends at:</td><td>Input</td><td></td></tr> <tr><td>Room No</td><td>Text</td><td>↓</td></tr> <tr><td colspan="2" style="text-align: right;"><b>Add</b></td><td></td></tr> </table>	Teacher:	Text	Select	Name:	Text	Select	Name:	Text	Select	Date of Viva:	Input		Starts from:	Input		Ends at:	Input		Room No	Text	↓	<b>Add</b>		
<b>Viva Schedule</b> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>Name</td><td>Date</td><td>Starts</td><td>Ends</td><td>Room No</td></tr> <tr><td>Text</td><td>Text</td><td>Text</td><td>Text</td><td>Text</td></tr> <tr><td>Text</td><td>Text</td><td>Text</td><td>Text</td><td>Text</td></tr> <tr><td>Text</td><td>Text</td><td>Text</td><td>Text</td><td>Text</td></tr> <tr><td>Text</td><td>Text</td><td>Text</td><td>Text</td><td>Text</td></tr> </table>	Name	Date	Starts	Ends	Room No	Text	Text	Text	Text	Text	Text	Text	Text	Text	Text	Text	Text	Text	Text	Text	Text	Text	Text	Text	Text	<b>New Schedule</b> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>Teacher:</td><td>Text</td><td>Select</td></tr> <tr><td>Name:</td><td>Text</td><td>Select</td></tr> <tr><td>Name:</td><td>Text</td><td>Select</td></tr> <tr><td>Date of Viva:</td><td>Input</td><td></td></tr> <tr><td>Starts from:</td><td>Input</td><td></td></tr> <tr><td>Ends at:</td><td>Input</td><td></td></tr> <tr><td>Room No</td><td>Text</td><td>↓</td></tr> <tr><td colspan="2" style="text-align: right;"><b>Add</b></td><td></td></tr> </table>	Teacher:	Text	Select	Name:	Text	Select	Name:	Text	Select	Date of Viva:	Input		Starts from:	Input		Ends at:	Input		Room No	Text	↓	<b>Add</b>					
Name	Date	Starts	Ends	Room No																																																	
Text	Text	Text	Text	Text																																																	
Text	Text	Text	Text	Text																																																	
Text	Text	Text	Text	Text																																																	
Text	Text	Text	Text	Text																																																	
Teacher:	Text	Select																																																			
Name:	Text	Select																																																			
Name:	Text	Select																																																			
Date of Viva:	Input																																																				
Starts from:	Input																																																				
Ends at:	Input																																																				
Room No	Text	↓																																																			
<b>Add</b>																																																					
Footer																																																					

**Figure 168: Internal (Exam controller: Manage Schedule-Viva)**

Home	Header	Sign out																							
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 33%;">Admission</td> <td style="width: 33%;">Manage schedule</td> <td style="width: 33%;">Profile</td> </tr> <tr> <td colspan="3" style="text-align: center; background-color: #cccccc;"><b>Viva Schedule</b></td> </tr> <tr> <td colspan="3" style="text-align: center; background-color: #cccccc;"><b>Update Room</b></td> </tr> </table>			Admission	Manage schedule	Profile	<b>Viva Schedule</b>			<b>Update Room</b>																
Admission	Manage schedule	Profile																							
<b>Viva Schedule</b>																									
<b>Update Room</b>																									
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%; vertical-align: top;"> <b>Room Information</b> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>Room No</td><td>Capacity</td><td>Option</td></tr> <tr><td>Text</td><td>Text</td><td>Update</td></tr> <tr><td>Text</td><td>Text</td><td>Update</td></tr> <tr><td>Text</td><td>Text</td><td>Update</td></tr> <tr><td>Text</td><td>Text</td><td>Update</td></tr> </table> </td> <td style="width: 50%; vertical-align: top;"> <b>Add new room</b> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>Room No:</td><td>Input</td></tr> <tr><td>Capacity:</td><td>Input</td></tr> <tr><td colspan="2" style="text-align: right;"><b>Add</b></td></tr> </table> </td> </tr> </table>			<b>Room Information</b> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>Room No</td><td>Capacity</td><td>Option</td></tr> <tr><td>Text</td><td>Text</td><td>Update</td></tr> <tr><td>Text</td><td>Text</td><td>Update</td></tr> <tr><td>Text</td><td>Text</td><td>Update</td></tr> <tr><td>Text</td><td>Text</td><td>Update</td></tr> </table>	Room No	Capacity	Option	Text	Text	Update	Text	Text	Update	Text	Text	Update	Text	Text	Update	<b>Add new room</b> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>Room No:</td><td>Input</td></tr> <tr><td>Capacity:</td><td>Input</td></tr> <tr><td colspan="2" style="text-align: right;"><b>Add</b></td></tr> </table>	Room No:	Input	Capacity:	Input	<b>Add</b>	
<b>Room Information</b> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>Room No</td><td>Capacity</td><td>Option</td></tr> <tr><td>Text</td><td>Text</td><td>Update</td></tr> <tr><td>Text</td><td>Text</td><td>Update</td></tr> <tr><td>Text</td><td>Text</td><td>Update</td></tr> <tr><td>Text</td><td>Text</td><td>Update</td></tr> </table>	Room No	Capacity	Option	Text	Text	Update	Text	Text	Update	Text	Text	Update	Text	Text	Update	<b>Add new room</b> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>Room No:</td><td>Input</td></tr> <tr><td>Capacity:</td><td>Input</td></tr> <tr><td colspan="2" style="text-align: right;"><b>Add</b></td></tr> </table>	Room No:	Input	Capacity:	Input	<b>Add</b>				
Room No	Capacity	Option																							
Text	Text	Update																							
Text	Text	Update																							
Text	Text	Update																							
Text	Text	Update																							
Room No:	Input																								
Capacity:	Input																								
<b>Add</b>																									
Footer																									

**Figure 169: Internal (Exam controller: Manage Schedule-Update room)**



**Same as Teacher: Profile**

The diagram shows a large, empty rectangular box centered on the page. At the bottom center of this box, the word "Footer" is written in a small, bold, black font.

**Figure 170: Internal (Exam controller: Profile)**

## DEFINE EVENTS THAT WILL CAUSE THE STATE OF THE USER INTERFACE TO CHANGE

### 1. Admission

#### 1.1. when admission session is ended:

Ex. Home → Admission Go → Admission Session Ended

#### 1.2. When admission is going

Ex. Home → Admission Go → Fill up form  
form info. → Submit → Show

### 2. Admit Card

#### 2.1. When card is available

Ex. Home → Admission Go → Download

#### 2.2. When card is unavailable

Ex. Home → Admission Go → Download → Not available

### 3. Create Account (Only for teacher)

Ex. Home → Account Go → Fill up form → Submit

### 4. Sign In

Ex. Home → Sign in → Fill up input → Log in → Home

### 5. Profile

Home → Profile → Fill up input → Update

### 6. Update Courses Details

Home → Manage course → Course details → Fill up input → Update

### 7. Update Marks

Home → Manage course → Update marks → Fill up input → Update

### 8. Notify Supplementary Student

Home → Manage course → List of students → Notify All

**9. Show Viva Schedule**

Home → Viva schedule

**10. View Courses**

Home → View courses

**11. View Result**

Home → View result

**12. Search Applicant By Name**

Home → Search applicant → by phone n → Search  
Fill up input

**13. Search Applicant by Phone No.**

Home → Search applicant → by phone no. → Search  
Fill up input

**14. Verify Admission Payment**

Home → Payment → Admission payment → Paid  
Fill up input

**15. Verify Semester Fees**

Home → Payment → Application fee → Paid  
Fill up input

**16. Download All Application**

Home → Search applicant → Download all

**17. Verify Applications**

Home → Search applicant → Search → Eligible  
Fill up input

**18. Add Student to Course**

Home → Manage course → Details → Add  
Fill up input

**19. Change Course Teacher**

Home → Manage course → Details → Change  
Fill up input

**20. Add Course****21. Update User Information****22. Approve/Deny User Request****23. Create User****24. Start Admission Session****25. Stop/Update Session****26. Generate Admit Card****27. Manage Viva Schedule****28. Update Schedule Room****29. Log out**

## DEPICT EACH INTERFACE STATE AS IT LOOK TO END USER

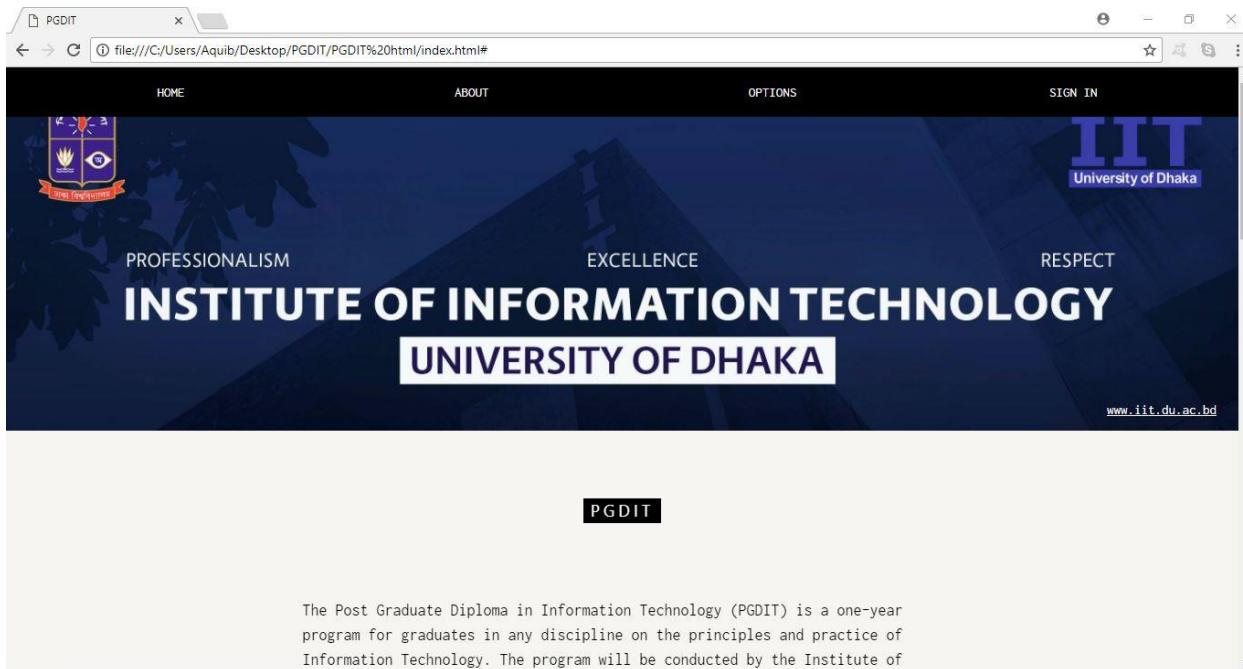


Figure 171: Home

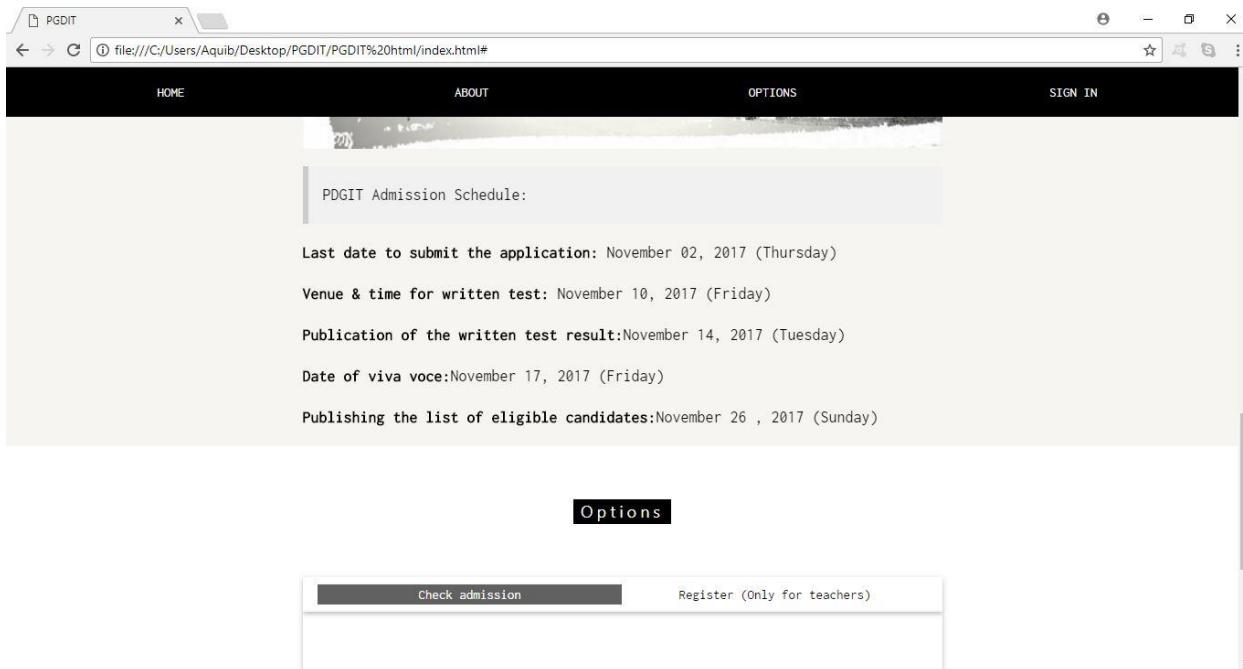
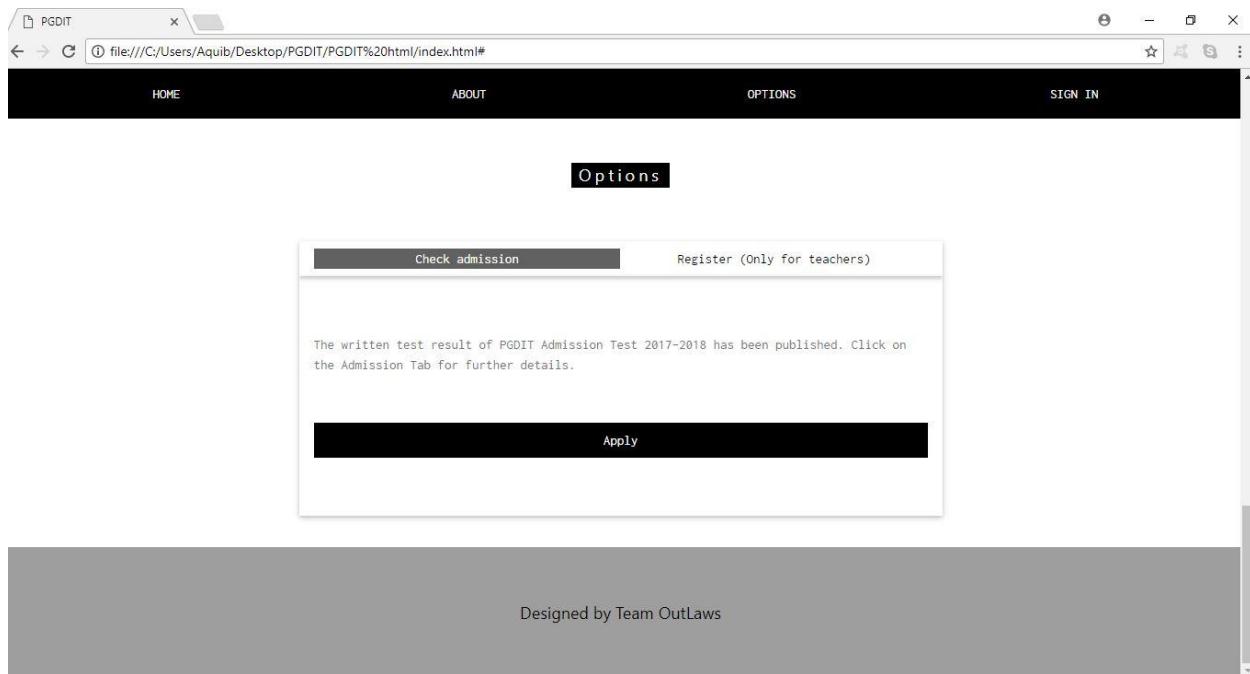


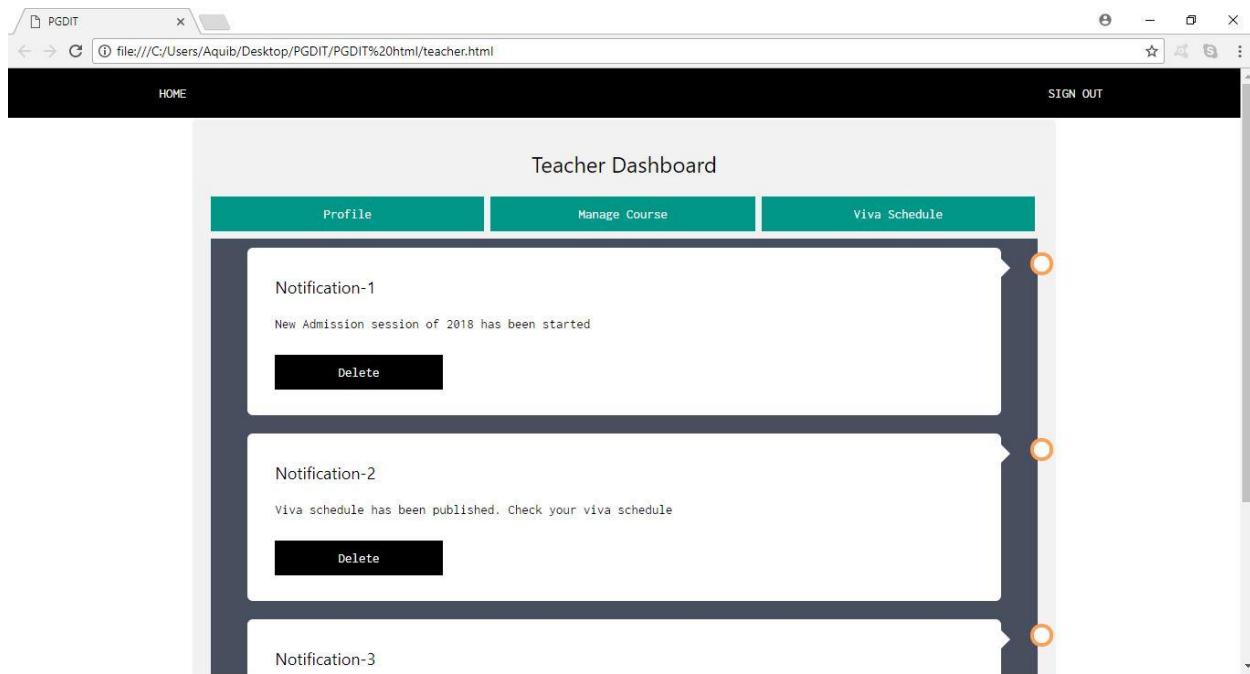
Figure 172: Notices



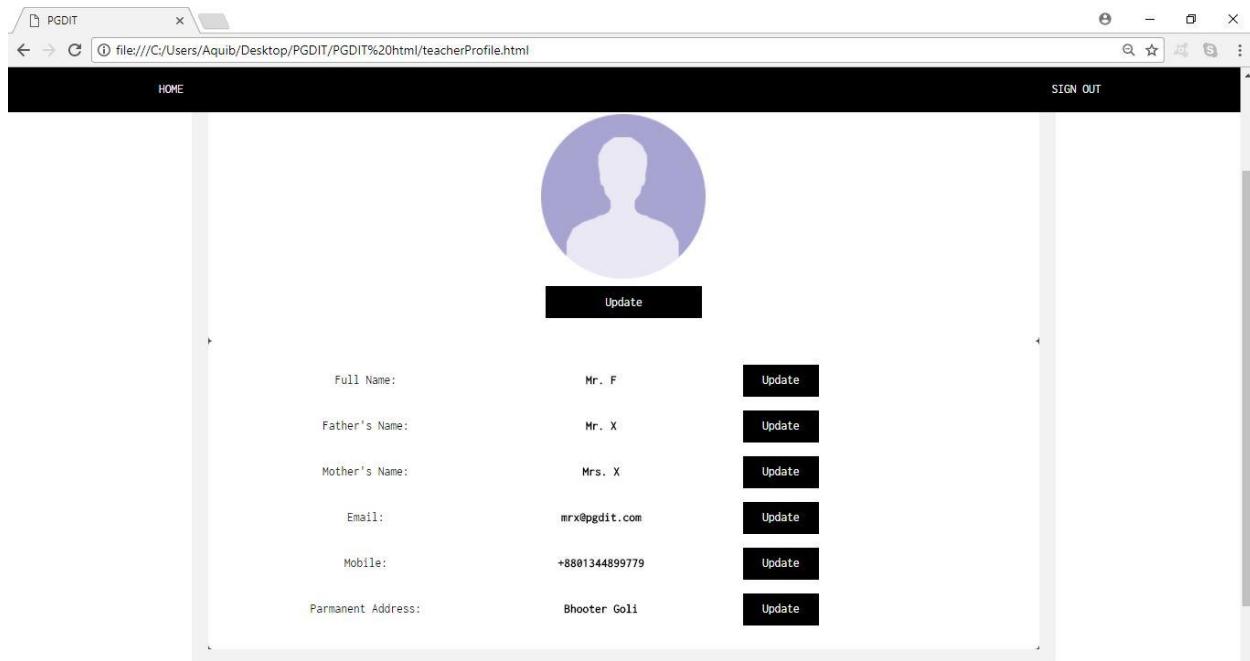
**Figure 173: Options**

The screenshot shows a web browser window titled 'PGDIT'. The address bar displays 'file:///C:/Users/Aquib/Desktop/PGDIT/PGDIT%20html/applicationForm.html'. The page title is 'PGDIT Admission Form'. The form is divided into sections: 'Personal Information' and 'Academic Career'. The 'Personal Information' section contains fields for Name of Applicant, Father's Name, Mother's Name, E-mail Address, Mobile Number, Date of Birth, and Nationality. The 'Academic Career' section contains tables for SSC/O-Level and HSC/A-Level education details, each with columns for School/College Name, Board, Group, Year, and Marks/GPA.

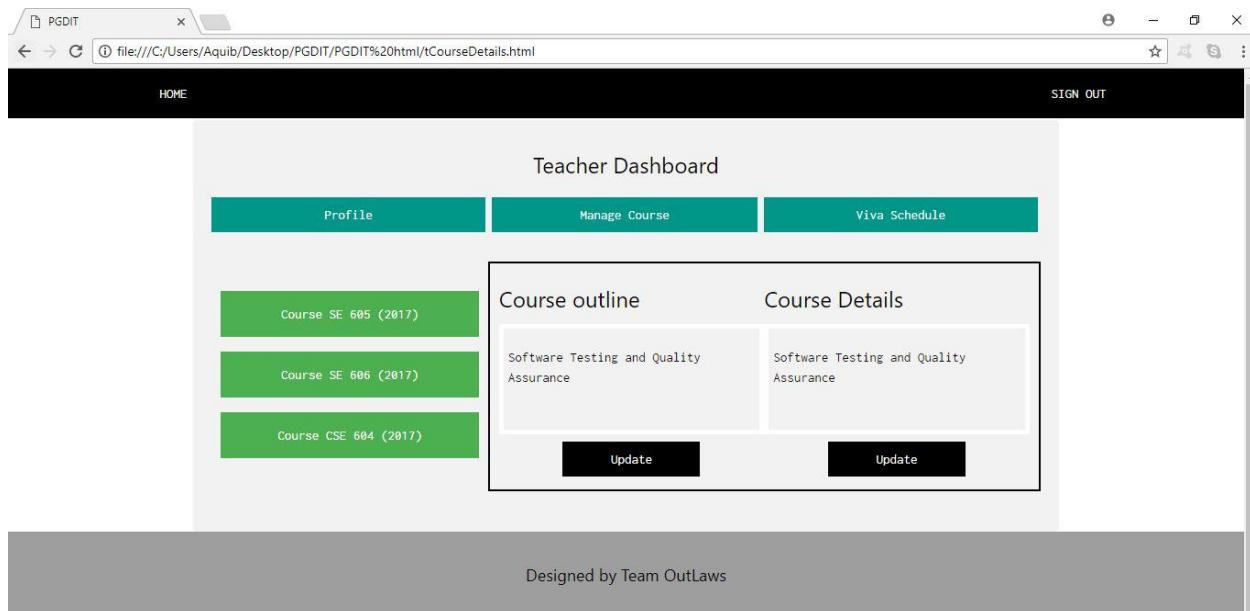
**Figure 174: Admission form**



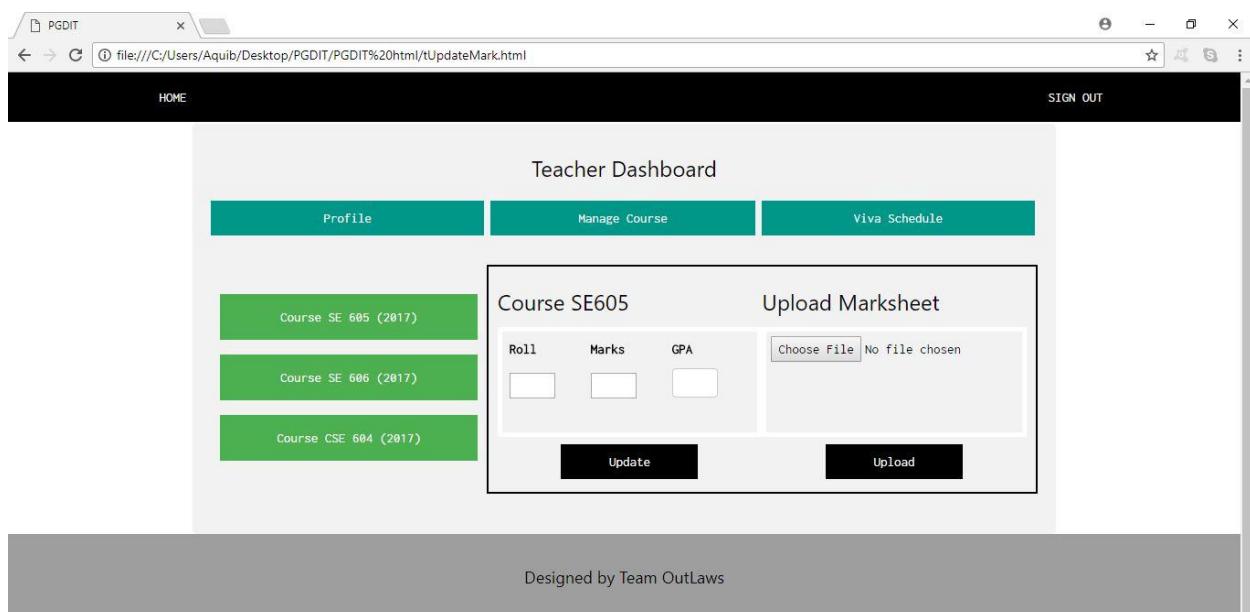
**Figure 175: Dashboard of Teacher**



**Figure 176: Profile**



**Figure 177: Course details**



**Figure 178: Update marks**

The screenshot shows a web-based application titled "Teacher Dashboard". At the top, there are three navigation tabs: "Profile", "Manage Course", and "Viva Schedule". Below these tabs, there is a section titled "List of student who need to take part in supplementary exam". This section contains a table with three rows, each representing a student's roll number, marks, and GPA. A "Notify All" button is located at the bottom right of this section. On the left side, there are three green rectangular boxes labeled "Course SE 605 (2017)", "Course SE 606 (2017)", and "Course CSE 604 (2017)". At the bottom of the page, a grey footer bar states "Designed by Team OutLaws".

Roll	Marks	GPA
07**	99	F
07**	99	F
07**	99	F

**Figure 179: Supplementary exam list**

The screenshot shows a web-based application titled "Teacher Dashboard". At the top, there are three navigation tabs: "Profile", "Manage Course", and "Viva Schedule". Below these tabs, there is a section titled "Viva Schedule". This section contains a table with four columns: "Date", "start at", "end at", and "room No.". There are four rows of data, all corresponding to the date "16 November 2017", starting at "10:00 AM" and ending at "12:00 PM", and all assigned to "room No. 402". At the bottom of the page, a grey footer bar states "Designed by Team OutLaws".

Date	start at	end at	room No.
16 November 2017	10:00 AM	12:00 PM	402
16 November 2017	10:00 AM	12:00 PM	402
16 November 2017	10:00 AM	12:00 PM	402
16 November 2017	10:00 AM	12:00 PM	402

**Figure 180: Viva schedule**

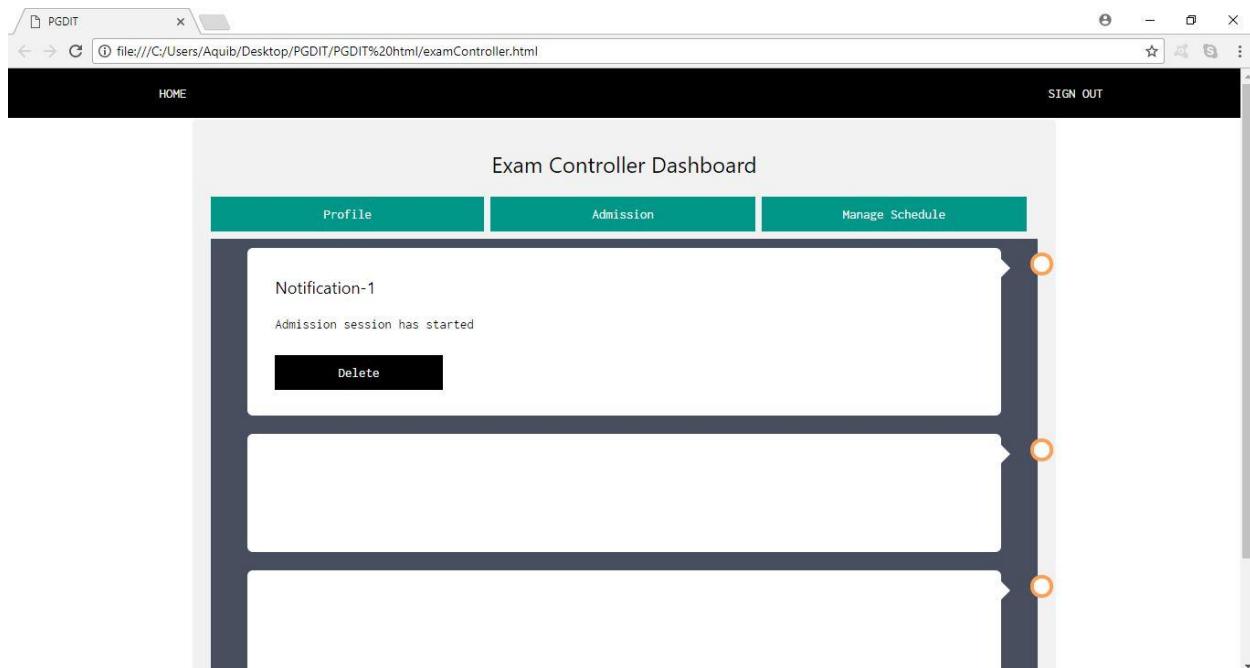


Figure 181: Dashboard of exam controller

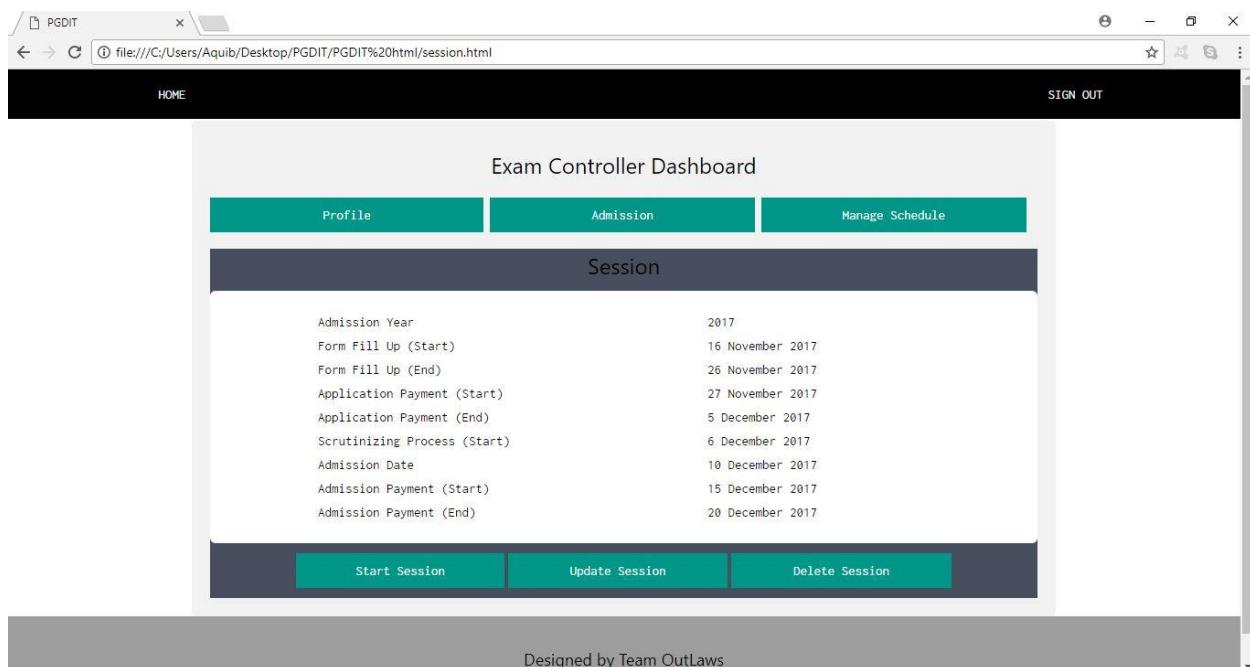


Figure 182: Admission session

PGDIT

file:///C:/Users/Aquib/Desktop/PGDIT/PGDIT%20html/vivaSchedule.html

HOME SIGN OUT

Profile Admission Manage Schedule

Viva Schedule

Name	Date	Start at	End at	Room No.
Mr. X	16 November 2017	10:00 AM	12:00 PM	402
Mr. A	16 November 2017	10:00 AM	12:00 PM	402
Ms. F	16 November 2017	10:00 AM	12:00 PM	402
Mr. C	16 November 2017	10:00 AM	12:00 PM	402

New Schedule

Teacher:

Name: A	Select a teacher ▾
Name: B	Select a teacher ▾
Name: C	Select a teacher ▾

Date of viva

Starts from

End at

Room no.

Add

Designed by Team OutLaws

**Figure 183: Viva schedule manage**

PGDIT

file:///C:/Users/Aquib/Desktop/PGDIT/PGDIT%20html/updateRoom.html

HOME SIGN OUT

Profile Admission Manage Schedule

Exam Controller Dashboard

Room Information

Room No.	Capacity	Option
402	40	Update
406	45	Update
107	30	Update
104	29	Update

Add New Room

Room No.

Capacity

Add

Designed by Team OutLaws

**Figure 184: Update room**