Software Requirements Specifications

for

Question Paper Generator & Evaluator

Version 1.0

Prepared by

Group : COE2

Aditya Kansal	(101603017)	aditya14569@gmail
Amneet Kaur	(101603029)	amneetaks10work@gmail
Aiana Verma	(101610003)	aianaverma5@gmail

Instructor: *Dr. Sanmeet Bhatia*

Course: Software Engineering (UCS503)

Lab Section: COE2

Teaching Assistant: Ms Deepali Bhagat

Date: 4th September 2018

Contents

1 Introduction

1.1 Document Purpose

We are talking about QUESTION PAPER GENERATOR AND EVALUATOR. The purpose of the product is to generate question papers and do their evaluation in an automated way. It will save the time and effort of the teachers. It will also help students to practice and revise regularly. Also, it is a secure process and it will prevent the leaking of question papers in the institutions.

1.2 Product Scope

This project makes easy for both the students and teachers to access the questions anytime. Students can practice anytime, any subject. It will help the institution in improving their overall results by letting the students practice more and more in an efficient way. Also an online database is maintained which will help the teachers to analyze the previous results and make improvement in the current education procedure.

This system also evaluate the student's capability and skills efficiently. It is fully automated system which fast results. In this system there is no need of transporting paper through police/security vans to all colleges. This system provide unbiased result. The usage of this system reduces human effort and save time and resources to an extent

1.3 Intended Audience and Document Overview

The current faculty of the institutions and the students studying there.

1.4 Definitions, Acronyms and Abbreviations

Admin	Person responsible for upkeep configuration and reliable operation of the system.
CPU	Central Processing Unit
JPEG	Joint Photographic Experts Group
SRS	Software requirements specifications
USB	Universal Serial Bus

1.5 Document Conventions

This document follows the IEEE formatting requirements. We have used Arial font size throughout the document for text and of size 12. We have used italics for comments. Document text is single spaced and we have maintained the 1" margins. Headings are bold and of size 14. General numbering convention is used.

1.6 References and Acknowledgments

- http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=6449828&queryText%3DQuestion+paper+generator+system
- http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=6779494&queryText%3DQuestion+paper+generator
- http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=4756830&queryText%3DQuestion+paper+generator+system

2 Overall Description

2.1 Product Overview

In this system we present a smart question paper generating system for universities. It is made to allow universities to generate question papers with random but even questions to cover most chapters of subject with difficulty level within seconds and mail them to colleges instantly. In our system we allow administrator to input a set of questions and respective answers for option ticking. We also allow admin to provide weight age and complexity for each of these questions. After this the questions are stored in database along with their weight age. Now on question paper generating time the admin just has to select the percentage of difficulty. On this selection the system selects questions randomly in a way that their weightage makes up for 100 marks and according to difficulty that admin chooses the questions based on their complexity level. The questions are also added for various difficulty levels so that as soon admin chooses the type of paper difficulty (Easy, medium, difficult) the system automatically generates paper, prepares doc file as per selected paper format. Also emails it to other colleges.

Evaluation is also done for the tests that the students give online. A database of marks is maintained for teachers and students.

2.2 Product Functionality

In our product the teachers inputs the questions along with the weightage, difficulty level and subject code before the time stated by the administrator. The system then filters the questions and maintains the database of these questions.

The process in done in two ways : online and offline

Online-

In this further, two cases are generated:

Case 1: The student requests the question bank from the system and practices the questions without any time limit and also, he/she can check on his/her own answers by requesting for the same.

Case 2: The student requests for the question paper by entering the subject code for a mock test/online test. Proper time limit is set and the answers are evaluated in the end at the backend by any teacher. The result is sent to the student by mail and stored in the database as well.

Offline-

The teachers requests for the question paper on the basis of difficulty level and weightage.They get it printed and the exam is conducted offline.Evaluation is done offline by the teachers and the marks are uploaded which can later be viewed by the students.

2.3 Design and Implementation Constraints

- -Questions provided by teachers might not be accurate. A possible solution for this is to cross-check the dataset supplied by the user using our model and if the probability output by our software contradicts then we don't accept that data.
- -The product must have a user friendly interface that is simple enough for all types of users to understand.
- -The central database server and backup database servers should be updated regularly. This updating and replication of data from central database server to the backup database server can introduce additional latency in the working of the system.
- -Response time for loading the software and for checking should both be lesser than 5 seconds.
- -Model architecture should be such that it gives both good accuracy and response.

2.4 Assumptions and Dependencies

- The project was started with the assumption that we would be given the necessary support in the form of hardware and software resources. Our project depends a lot on the inputs.

- User has the basic knowledge of the system environment.

- The users have sufficient knowledge of the system interface.

- Invigilation is provided in every such examination hall as our product does not guarantee checking of unfair means on its own.

- In case of any discrepancies, the current state of the quiz will be stored as such.

3 Specific Requirements

3.1 External Interface Requirements

3.1.1 Hardware Interfaces

The following list presents the external interface requirements:

● The product just a simple keypad and mouse for taking the user input.

● The product does not require usage of sound or animation.

CPU Requirements:

- Processor – i3
- Hard Disk – 5 GB
- Memory – 1GB RAM

3.1.2 Software Interfaces

Software Requirements:

- Windows XP, Windows 7(ultimate, enterprise)
- Visual studio 2010
- Sql 2008

3.2 Functional Requirements

3.2.1 F1: The system shall generate questions for the required subjects automatically and according to the difficulty level.

3.2.2 Features:

- **Admin Login:** Admin would be having a login account. He can add questions in the system and their respective answers. The answers are stored as a base for reference for AI to use while checking answers.
- **Question Insertion:** User may insert questions as per weight age, difficulty, subject and chapter in the system.
- **Difficulty Choosing:** User may then choose the paper difficulty level
- **Random Paper generation:** The system now automatically chooses random questions as per the selected difficulty level.

- **Wide Chapter Coverage:** The system tries to cover as many chapters as possible for that subject.
- **Doc File Creation:** The system now exports the well formatted question paper in a doc file.
- **Emailing:** The file is now emailed to various users ie college/exam
- **Evaluation:** Done in offline and online modes.

3.3 Use Case Model

3.3.1

Logging in

Purpose- Login the users having correct credentials

Priority- High

Actors- Teacher,Student,Admin

Includes-Access question bank, make question paper,enter course name,provide question bank,results,profile,answer question

Flow of events

1. User enter login credentials

2. If the credentials are correct,user is redirected to user page.

3. In case of error,error message is displayed

3.3.2

Access question bank

Purpose- Logged in users access questions

Priority- Moderate

Actors- Admin

Includes- make question paper

Flow of events

1. Admin logs in.

2. Admin accesses the question bank to make the required question paper

3.3.3

Makes question paper

Purpose- Make valid question papers for the students

Priority- Moderate

Actors- Admin

Extend- MCQ's/ one word

Flow of events

1. Admin logs in.

2. Admin access the question bank.

3. Admin generates the question paper.

3.3.4

Provides question bank

Purpose- Provide valid questions to be tested in the quiz.

Priority- Moderate

Actors- Teacher

Flow of events

1. Teacher logs in.

2. Teacher makes the questions and adds them to the database.

3.3.5

Results

Purpose- Logged in users access results

Priority- Moderate

Actors- Teachers,Students

Flow of events

1. User logs in.

2. User views the results.

4. Other Non-functional Requirements

4.1. Performance Requirements

The actual quizzes are managed through internet connection and this makes performance measures crucial.

For desired performance, speed of connection, response time, processing speed must be considered.

System should work real-time which means there should be an acceptable time delay such as max 4-5 seconds between request and response.

4.2. Safety and Security Requirements

The computer that runs the program will have its own security. Only the System Admin will log in to the main system with his/her username and password.

In case of malfunction, system should shutdown itself and reboot in order to prevent unpredicted results.

System should store user data on database securely and set access permissions to the data carefully

4.3. Software Quality Attributes

Interface is accessible from at least one or more convenient places.

This includes all of the rooms of the institution, hostels and residences. If the system interface is not readily accessible, then it will not be as easy to control and will offer little convenience.

Interface and system must be properly connected to the sensors.

It is essential to have secure wiring, preferably inside the walls, for reliable information to guard against any false generation and evaluation of questions.

System must have a backup power supply.

The system needs to have a constant power supply while the quiz is going on.
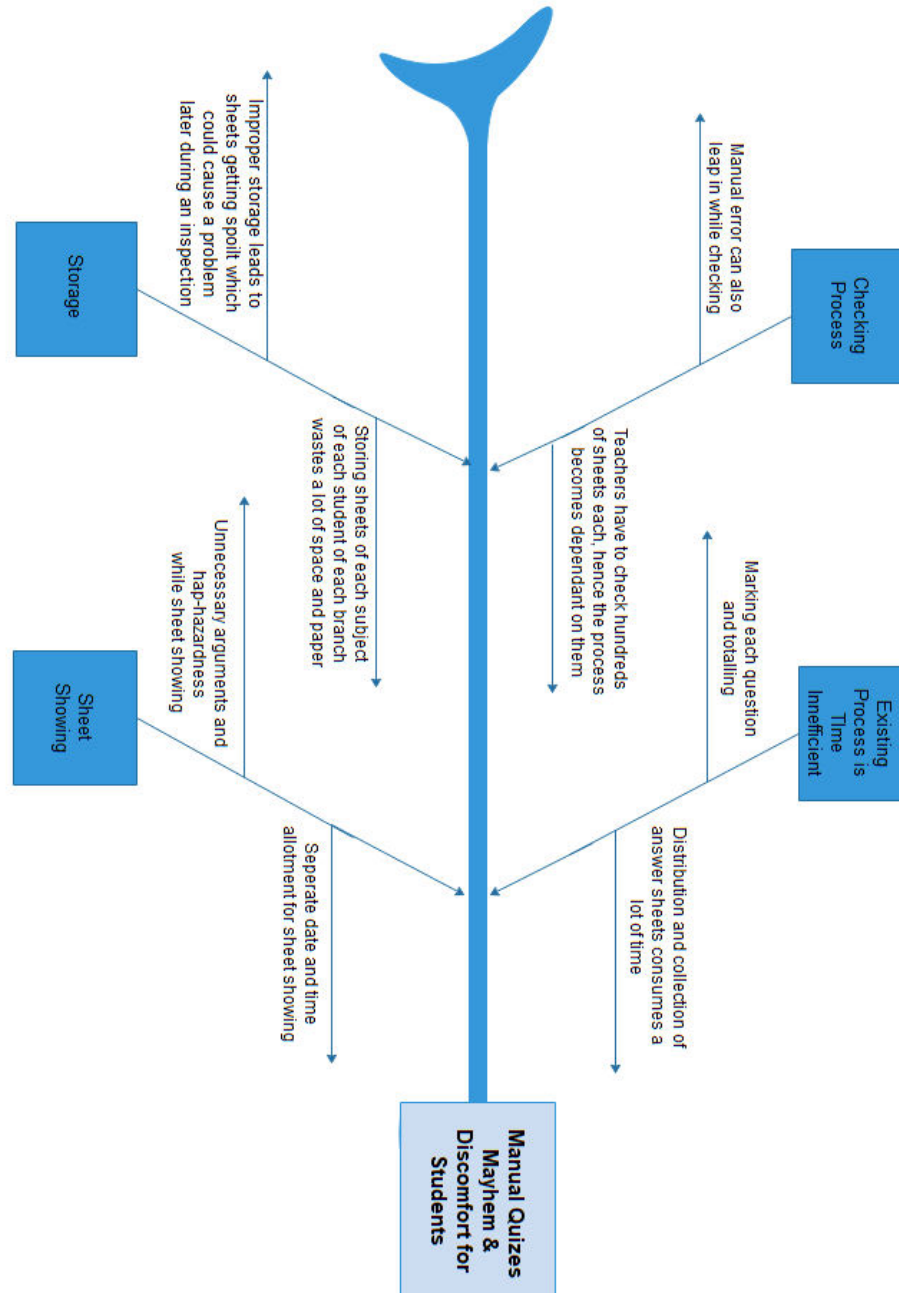
5. Test Cases

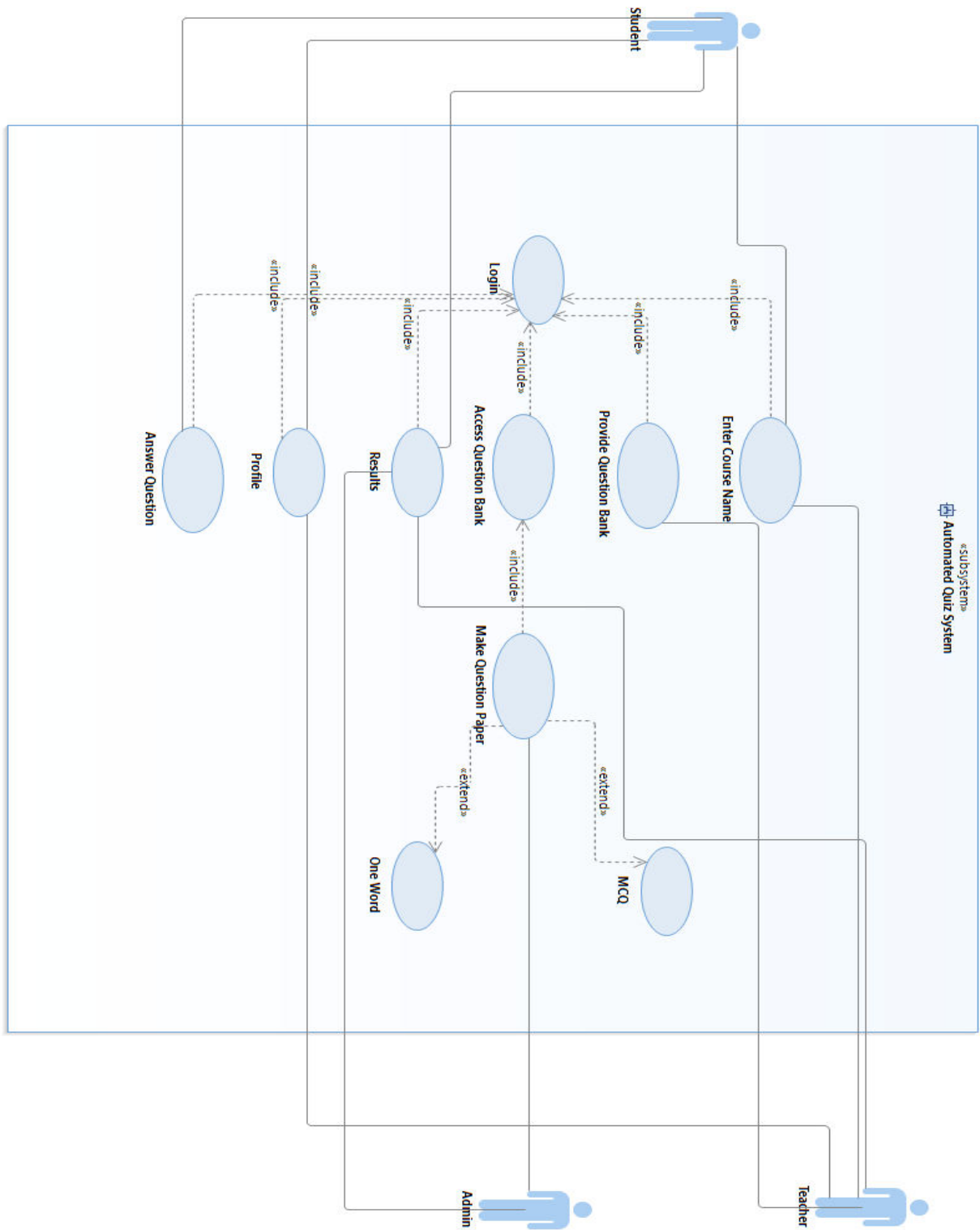| Test Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
TU01	Check Student Login with empty e-mail	1. Go to Login Screen 2. Enter Roll Number 3. Enter Password 4. Click Log in	e-mail = Password = 4567	Application should ask for email	As Expected	Pass
TU02	Check Student Login with empty password	1. Go to Login Screen 2. Enter e-mail 3. Enter Password 4. Click Log in	e-mail = aditya14569@gmail.com Password =	Application should ask for password	As Expected	Pass
TU03	Check timer starts synchronously with the quiz	1. Go to quiz page 2. Enter User ID 3. Click Next	User Id:- 101603017	Timer should start as soon as next is clicked	As Expected	Pass
TU04	Check test	1. Go to quiz	Quiz answers	Test Results	As	Pass

	results	page		should be displayed	Expected	
		2. Enter User ID				
		3. Click Next				
		4. Give the quiz				
		5. Submit the quiz				

6. Diagrams

1- Ishikawa Diagram

Storage

Checking Process

Sheet Showing

Existing Process is Time Innefficient

Improper storage leads to sheets getting spoilt which could cause a problem later during an inspection

Manual error can also leap in while checking

Storing sheets of each subject of each student of each branch wastes a lot of space and paper

Teachers have to check hundreds of sheets each, hence the process becomes dependant on them

Unnecessary arguments and hap-hazardness while sheet showing

Marking each question and totalling

Seperate date and time allotment for sheet showing

Distribution and collection of answer sheets consumes a lot of time

Manual Quizes Mayhem & Discomfort for Students

2. Use Case Diagram

«subsystem» Automated Quiz System

Actors: Student, Teacher, Admin

Use Cases: Login, Access Question Bank, Provide Question Bank, Enter Course Name, Answer Question, Profile, Results, Make Question Paper, One Word, MCQ
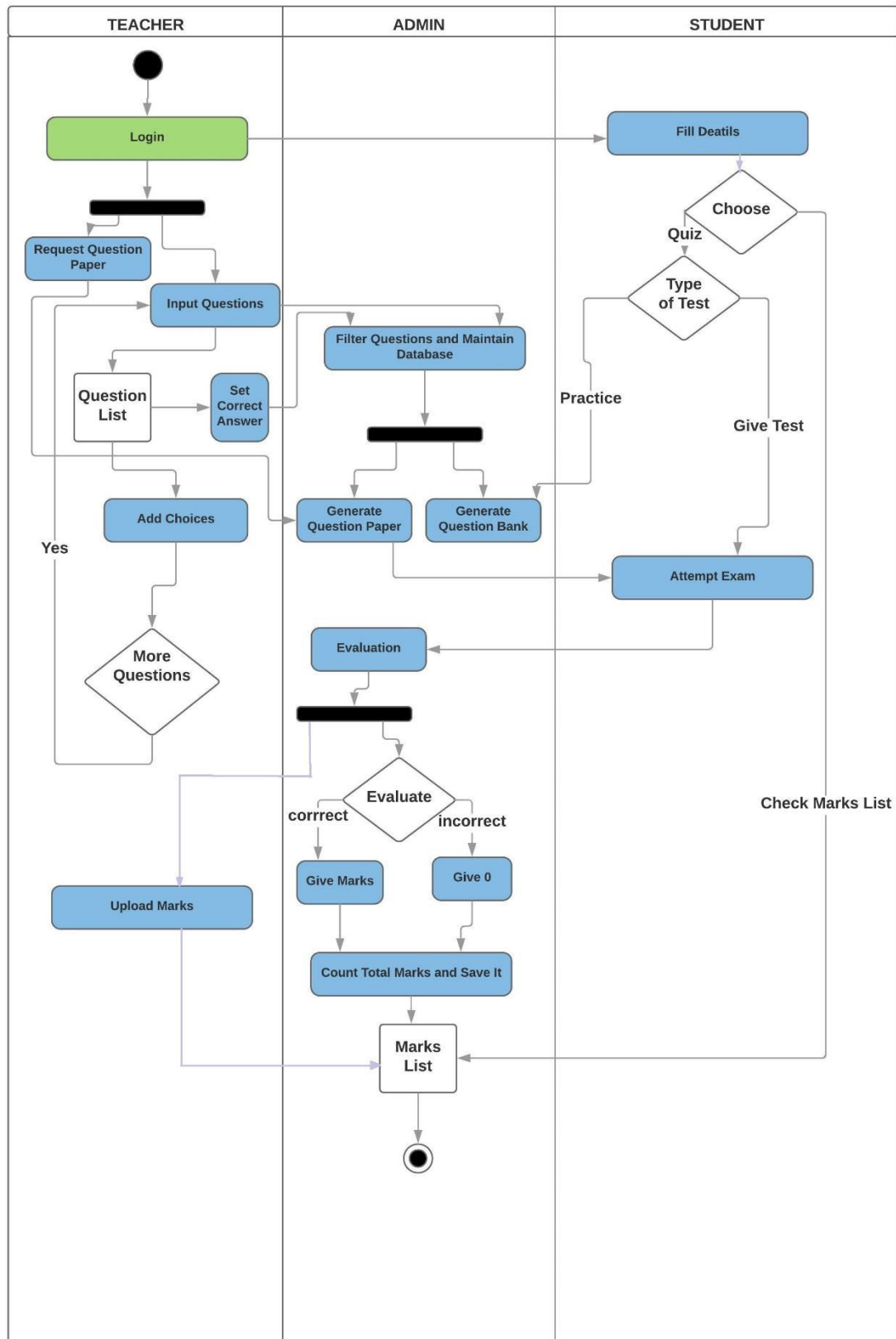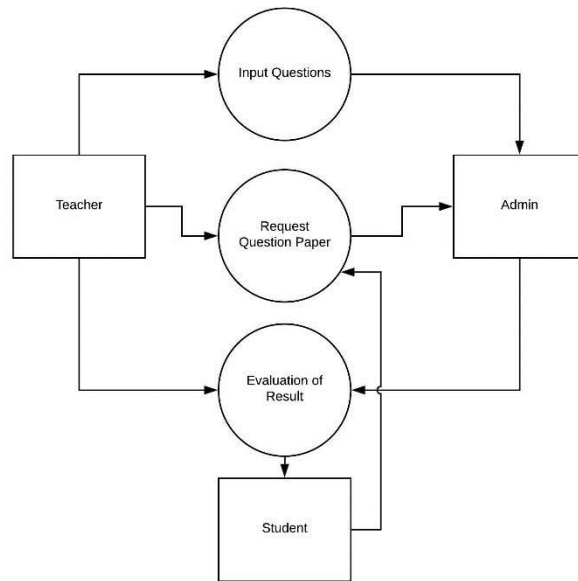
Relationships: «include», «extend»

3. Activity Diagram

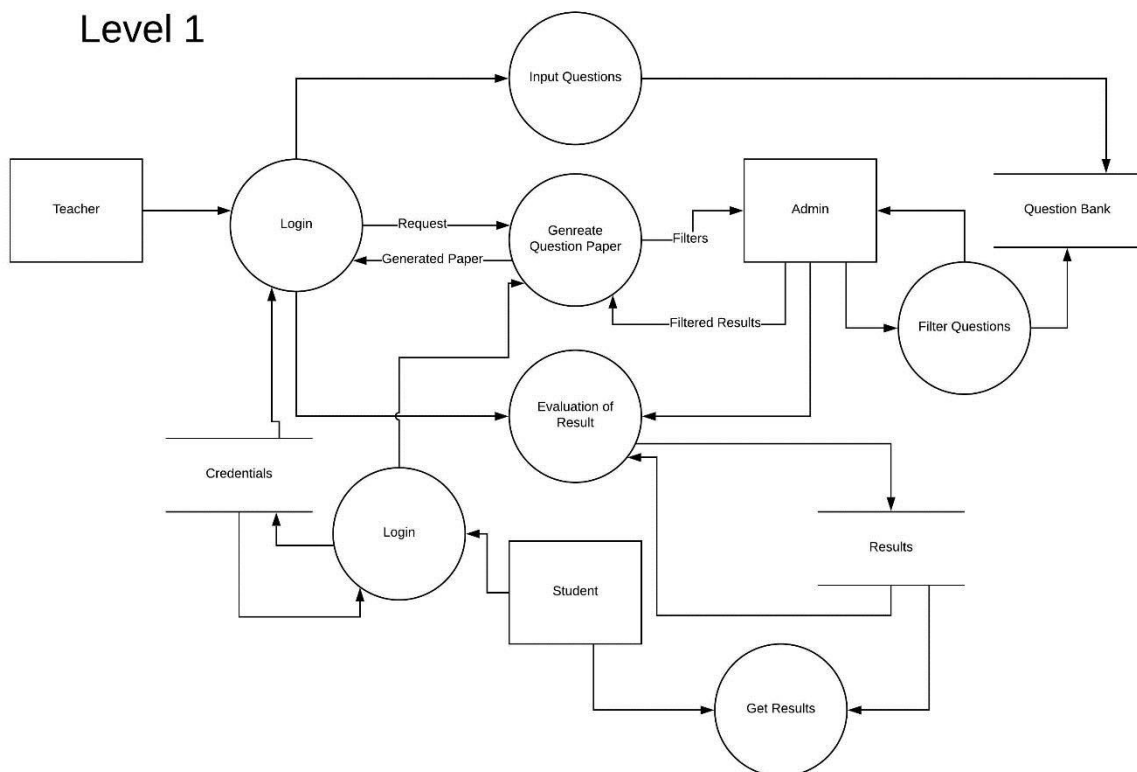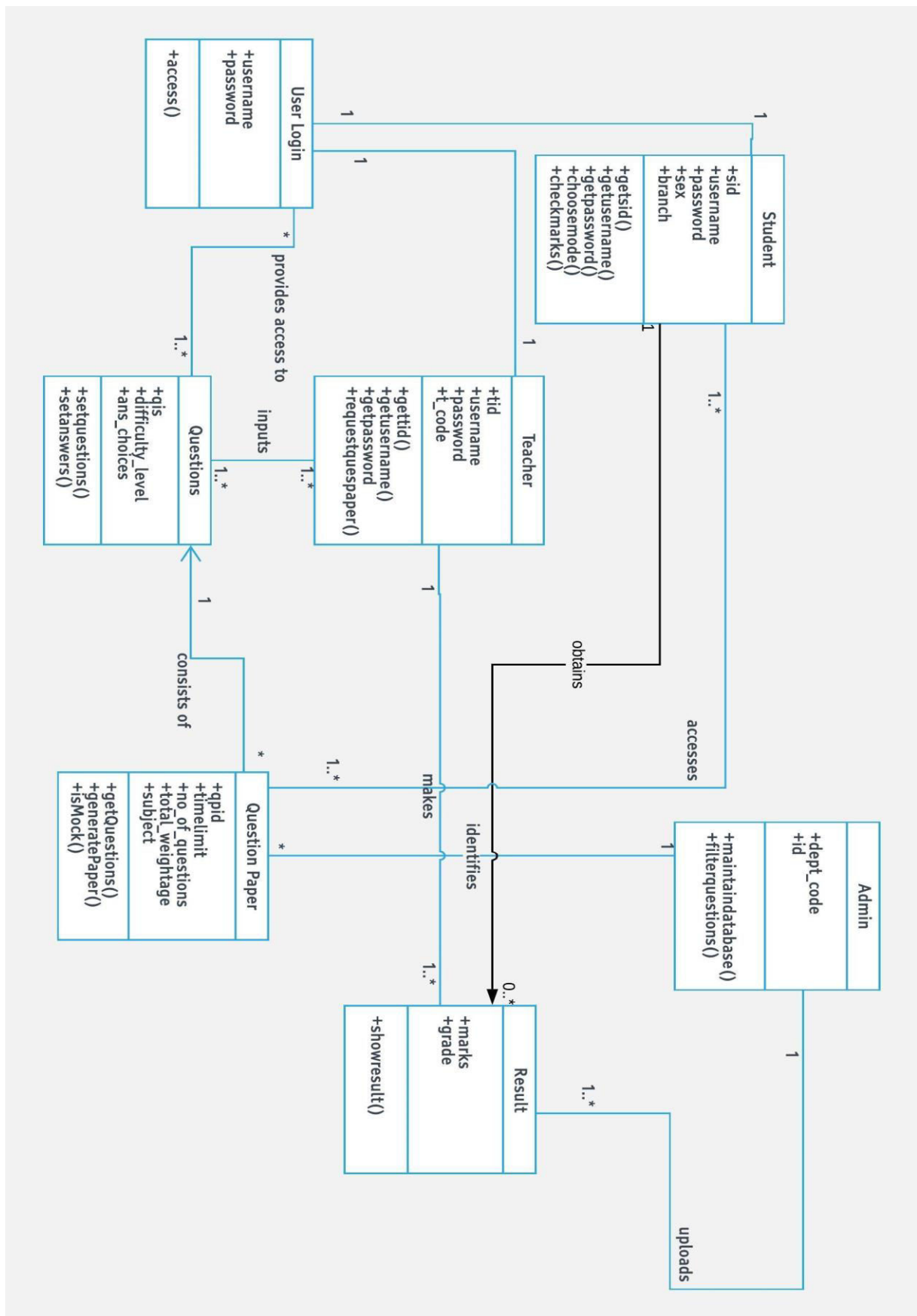AUTOMATED QUIZ ACTIVITY DIAGRAM

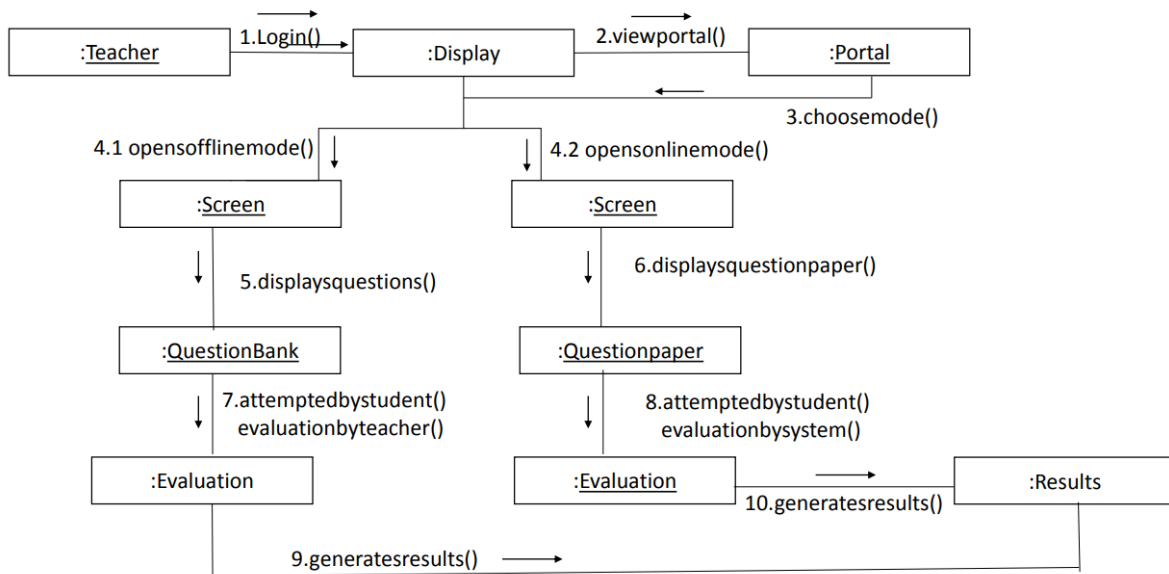## 4. Data flow diagram

Level 0

Level 1

5. Class Diagram

6. Collaboration Diagram

Collaboration Diagram (Student)

| :Teacher | 1.Login() → | :Display | 2.viewportal() → | :Portal |

3.choosemode()

4.1 opensofflinemode() ↓

:Screen

5.displaysquestions()

:QuestionBank

7.attemptedbystudent()
evaluationbyteacher()

:Evaluation

9.generatesresults() →

4.2 opensonlinemode()

:Screen

6.displaysquestionpaper()

:Questionpaper

8.attemptedbystudent()
evaluationbysystem()

:Evaluation

10.generatesresults()

:Results

Collaboration Diagram (Teacher)

| :Teacher | 1.Login() → | :Display |

4.filterquestions()

2.viewportal()

:Admin — 3.inputquestions() ← :Portal
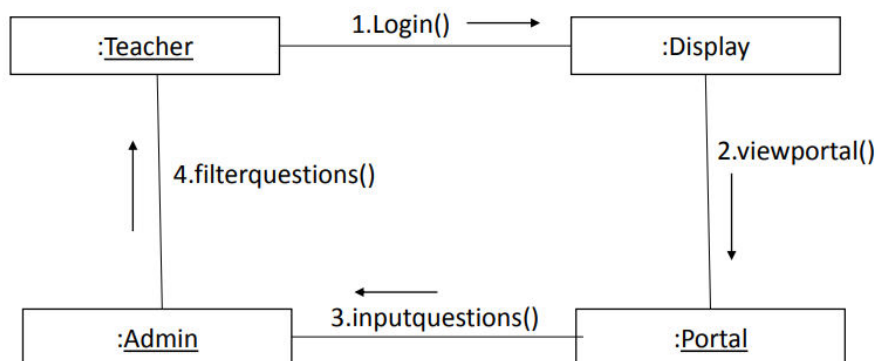
7. Sequence Diagram

SEQUENCE DIAGRAM FOR TEACHER

TEACHER   SCREEN   ADMIN

1. Login

1.1 Open Portal

2. Input Questions

2.1 Filter Questions
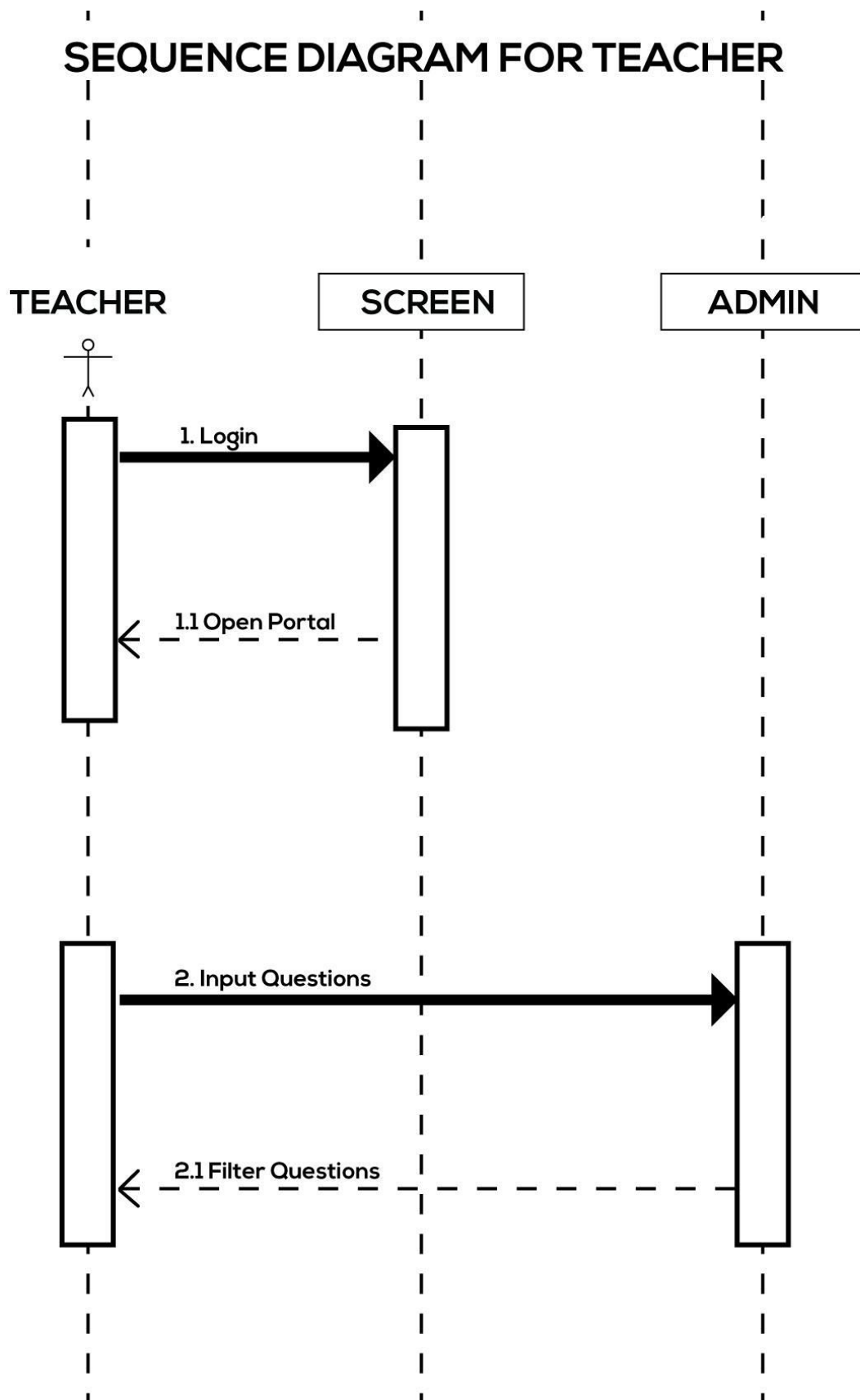
SEQUENCE DIAGRAM FOR STUDENT

STUDENT   DISPLAY   SYSTEM   TEACHER

1. Login

1.1 Display Portal

2. Request Question Paper

2.1 Generated Question Paper

3. Choose Offline/ Online Mode

3.1 Display Chosen Mode

Alternative

[Mode = Online]

4. Attempts Questions

4.1 System Evaluates Answers

[Else]

4.2 Attempts Questions

4.3 Teacher Evaluates Answers

5. Ask for Result

5.1 Shows Result
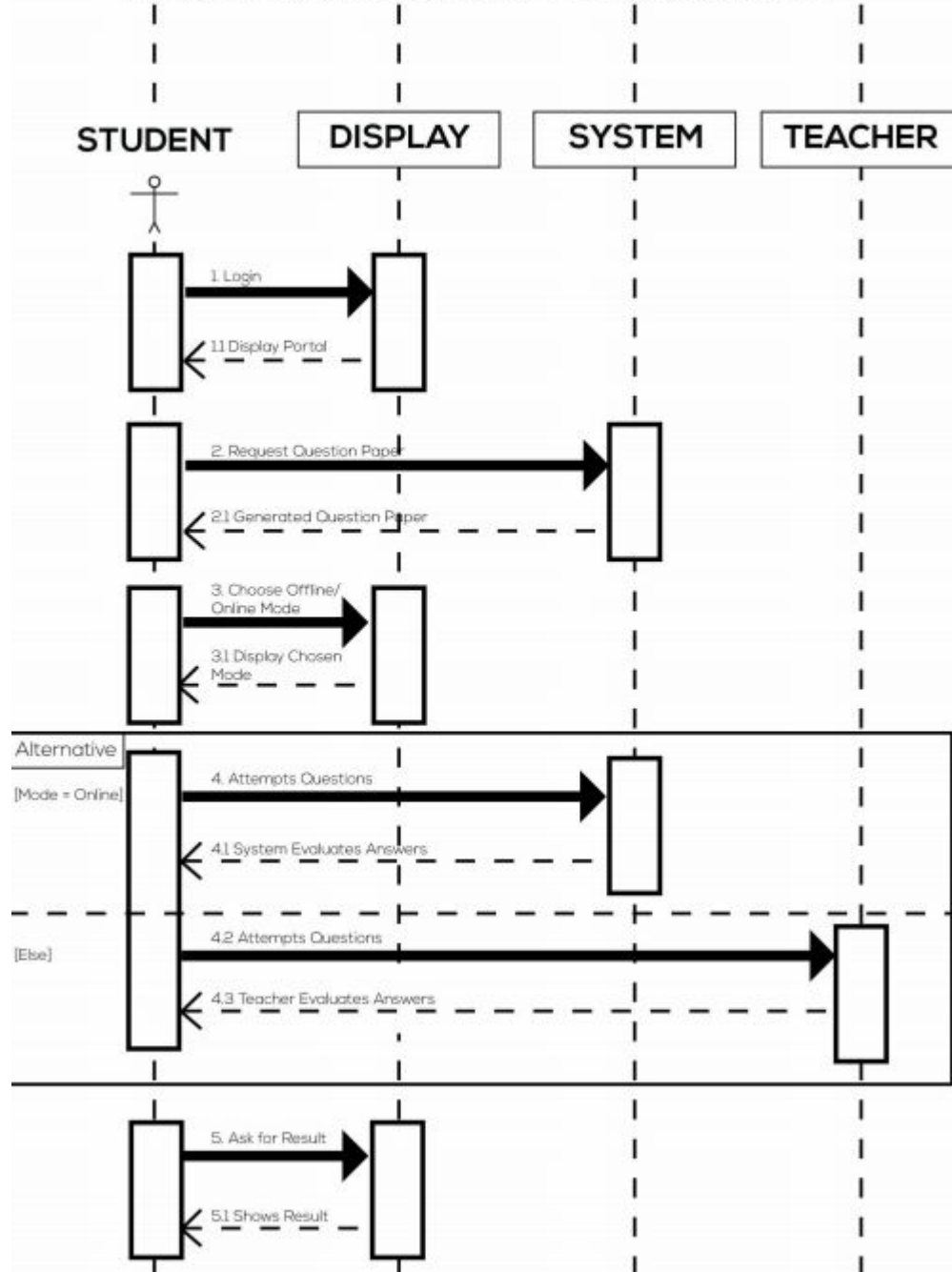
8. State Chart Diagram
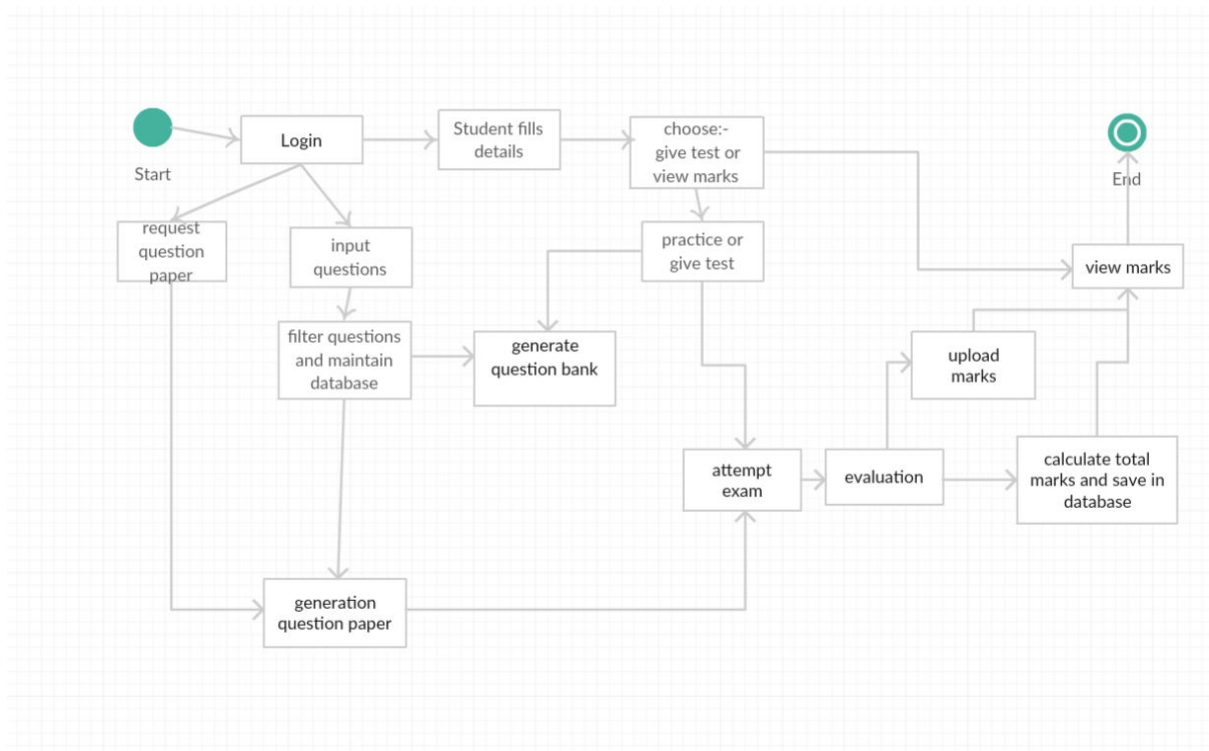
9. Component Diagram

10. Deployment Diagram

JSP Server `<<device>>`

Java Environment `<<execution environment>>`

`<<artifact>>` quizzy/main.jsp

`<<artifact>>` give_t.jsp

`<<artifact>>` set_q.jsp

`<<manifest>>`

`<<manifest>>`

`<<component>>` Question Paper

`<<component>>` Question Bank

`<<device>>` User 1

TCP/IP Protocol

`<<device>>` User 2

TCP/IP Protocol

`<<device>>` User 3

TCP/IP Protocol

TCP/IP Protocol

Database Server `<<device>>`

Microsoft Sql Server 2008 `<<database>>`

`<<schema>>` Student

`<<schema>>` Teacher

`<<schema>>` Question Bank

`<<schema>>` Question Paper