

e-Abhyaas

ई-अभ्यास



Summer Internship 2012, IIT Bombay

Under Prof. Deepak B. Phatak

Contents

Abstract.....	4
Team Members.....	5
Acknowledgements.....	6
Introduction	7
Purpose	7
Scope.....	7
Project Requirements	8
Software.....	8
Hardware	8
Invigilator Interface & Networking	9
Introduction	9
Steps.....	9
Technology.....	9
e-Abhyaas Desktop Application	10
Class Diagram.....	11
Activity Diagram.....	12
Screenshots.....	13
Web Interface	16
Introduction	16
Sub System Overview.....	16
Explicitly Declared CSS Classes.....	16
Classification	16
Activity Diagram.....	28
Sequence Diagram.....	30
Content Module.....	31
Accessibility Flow	33
Disabling Android UI & Encryption	34
Disabling Android UI	34
PDF Reader.....	34
Encryption and decryption.....	35
Description of the encryption algorithm.....	36
Class diagram	38

Answer sheet	39
How it works	39
Various types of questions.....	41
How answers are captured.....	44
End of exam	45
Future Scope	46

Abstract

Technology has become an integral part of our lives. The world has seen rapid strides in technology over the past couple of centuries. Letters have been replaced by emails, books have become electronic books, gramophones have become electronic MP3 players and soon all this is moving to the cloud. One area which hasn't seen such penetration is the field of education. Online textbooks are hardly used, exams are written on paper and submitted and much time is spent in correction where is it not necessary like objective-type papers. Education with technology is a powerful combination which can revolutionise our society. And with the Aakash tablet slated to be in the hands of every Indian student, new avenues were opened.

Keeping this in mind, our app “e-Abhyaas | ई-अभ्यास”, developed under the guidance of Prof. Deepak B. Phatak and Dr. Avinash Awate aims to bridge the gap existing between technology and education. e-Abhyaas is an application for the Aakash tablet which brings testing right to the tablet. Online testing has existed from before but none of them provide such flexibility which e-Abhyaas provides, that too on a portable 7-inch tablet which can be used by the student for other educational and recreational purposes as well. Read on to know more details on how this application is made, what languages are used and all other aspects.

Team Members

Pratyush Nalam: First Year Computer Science Engineering Student at IIT Bombay. Worked on developing the answer sheet which allows the student to answer the question paper.

Akash Priyadarshi: Third Year Computer Science Engineering Student at NIT Jamshedpur. Worked on the interface for the invigilator during the conduct of the examination and developed the networking functionality for the entire application.

Prabhanshu Agrawal: Second Year Computer Science Engineering Student at KIIT Bhubaneswar. Worked on disabling the Android UI elements during conduct of the exam, creating a PDF reader used to display the question paper and encrypting the question paper to avoid snooping.

Krattika Mishra: Third Year Computer Science Engineering Student at MANIT Bhopal. Worked on developing the web interface for the teacher to submit the question paper and for the invigilator to download the requisite files before the start of the exam.

Aastikta Sharma: Second Year Computer Science Engineering Student at NIT Puducherry. Worked on developing the web interface for the teacher to submit the question paper and for the invigilator to download the requisite files before the start of the exam.

Acknowledgements

This e-Abhyaas application could have never been done by us alone. We have received immense help from various people in the summer internship team.

We would first like to whole-heartedly thank Prof. Deepak B. Phatak, Subrao M. Nilekani Chair Professor, Department of Computer Science and Engineering, IIT Bombay for selecting us as a part of Ekalavya Summer Internship Programme 2012 and giving us such a wonderful opportunity.

Dr. Avinash Awate, the Program Director and our mentor for this project also gets our deepest regards, thanks and acknowledgements for personally guiding us throughout the project and helping us to solve our problems.

We are also thankful to Mr. Dileep Singh, Mr. Rajavel D. and Mr. Ninad Chilap for helping us debug the numerous errors we encountered. Without their invaluable help and guidance, this project would have been much more difficult to complete.

We are also grateful to Mr. Bikas Chhatri for helping us with the various administrative formalities and ensuring that we aren't distracted by petty logistical issues which could have cropped up.

There are many other people whom we would like to thank but haven't mentioned their names simply because so many people are there. To all of them, we say a big "Thank You"!

Introduction

The power of technology needs to be leveraged to its true potential in the education sector. Present testing solutions are highly inflexible – most of them allow just one type of question, the single correct answer. e-Abhyaas has been designed to give this flexibility to the teacher. The teacher can choose from any of five types of objective questions and the answer sheet intelligently detects each question type and generates the answer sheet.

There are 5 main modules in e-Abhyaas: the interface for the invigilator during the conduct of the exam, the web interface for the paper-setter and the invigilator, disabling the Android UI during the conduct of the exam, encryption and decryption of the question paper and display of the answer sheet. These 5 modules communicate with each other through the core networking component which runs throughout the project.

Purpose

The purpose of this document is to explain how the various modules work and how they communicate with each other for the successful working of the application. It elaborates how Java socket programming helped in the network communication between various modules; how HTML, CSS, JavaScript and JSP worked together to make a solid web interface for the teacher to upload the question paper and the key and for the invigilator to download the question paper and list of students for the conduct of the exam; how the graphic elements of Java were utilised to make the interface for the invigilator during the conduct of the exam; how the question paper is encrypted to prevent snooping by unauthorised people and decrypted again on the tablet; how the various components of the Android UI are disabled during the conduct of the exam so that the student writes only the exam and does nothing else and how the answer sheet was developed on Android and how it is generated dynamically according to the specifications of the question paper.

Scope

Our application, e-Abhyaas has a large scope in the educational testing sector. Given its flexibility to handle 5 different types of objective-type questions, it is a great boon to online testing. Teachers will have increased freedom to ask students more types of questions without being restricted to the single correct answer type as is the case most of the time now with regards to e-testing.

Project Requirements

Software

We used the following software for this project:

- a. All of us used Microsoft® Windows® 7 as our operating system.
- b. For Java socket programming and to make the invigilator interface (again in Java), we used NetBeans with Java SE Development Kit. The end-user needs to have Java SE Runtime Environment installed.
- c. To make the web interface in JSP, NetBeans along with Apache Tomcat was used. The HTML and CSS part was designed using Adobe® Dreamweaver®. And, this interface was tested on Internet Explorer® 9, Mozilla Firefox 11 (and above) and Google Chrome 17 (and above).
- d. To make the student interface on the Aakash tablet, we used Eclipse.
- e. We used MySQL as the database to store the results of the test.
- f. Connectify is required to run the hotspot on the invigilator's laptop.

Hardware

The following hardware configuration is required to run the various softwares for this project:

- Processor: Intel® Core™ i3 CPU
- Memory: 4GB RAM
- Storage required: Maximum of 1GB
- Graphics card: Not needed
- Wi-Fi: Any standard driver

Invigilator Interface & Networking

Introduction

The invigilator front end is designed to provide a user friendly GUI interface for non-technical end users. The aim of this interface is to make it child's play. It is designed in such a way that it provides maximum efficiency with minimum number of CPU cycles.

The interface displays a list of student names, along with their photos, who are going to attend the test. A start-stop button at right upper corner allows the invigilator to start the test and stop any more new connections after the desired time.

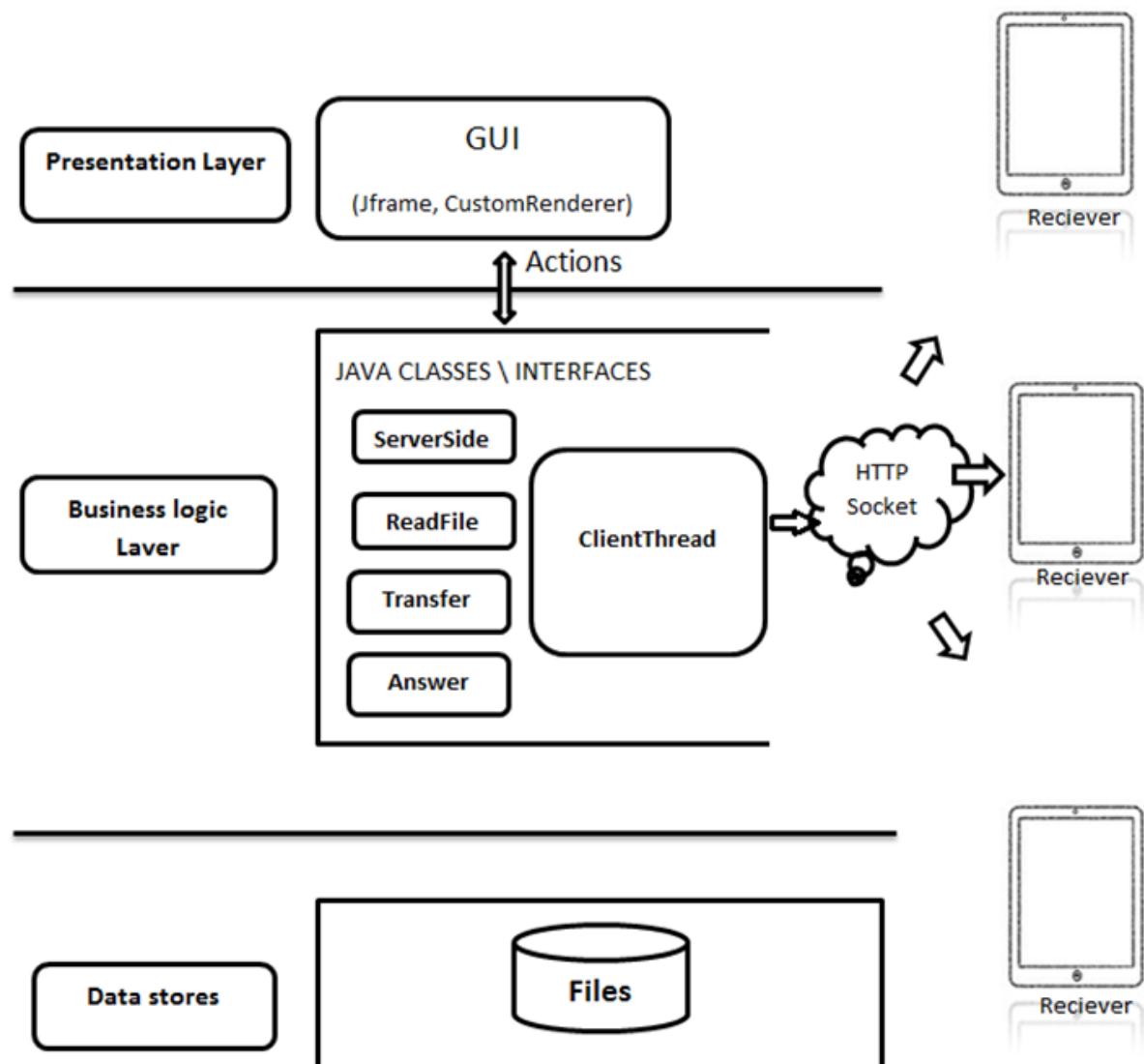
Steps

1. Invigilator has to download the zip folder from e-Abhyaas website.
2. Invigilator has to unzip the folder and start Invigilator Software.
3. Invigilator has to start the test at pre-defined time by clicking on "Start" button.
4. After 10-15 minutes invigilator should stop any new students from connecting by clicking on "Stop" button.
5. After end of exam Invigilator have to upload Answer.csv to the e-Abhyaas website.

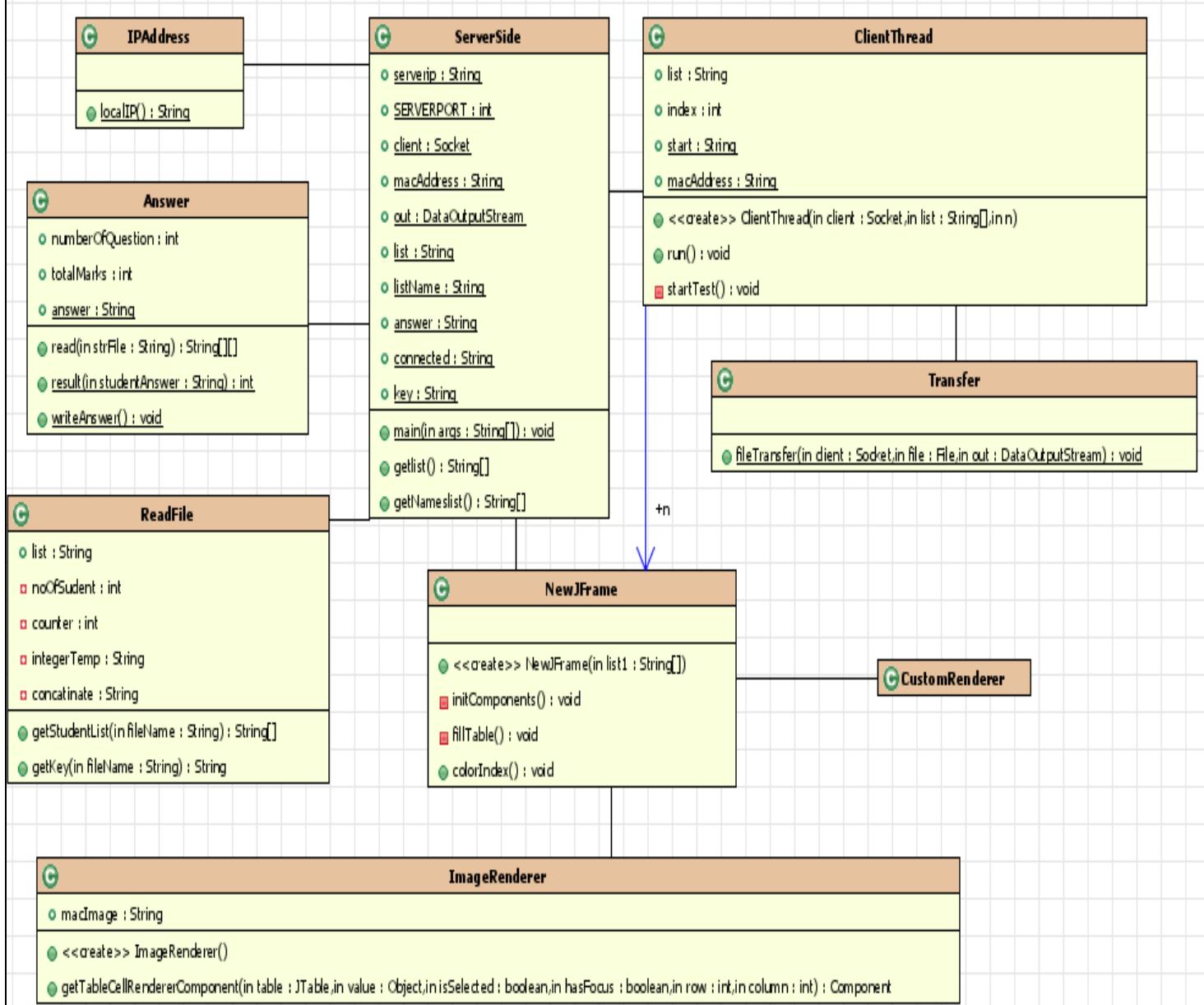
Technology

1. The front end of the application is designed by using Java Swing.
2. Invigilator-Student connection is implemented with the help of Java Socket Programming.
3. Invigilator side evaluation is done with help of Java.

e-Abhyaas Desktop Application

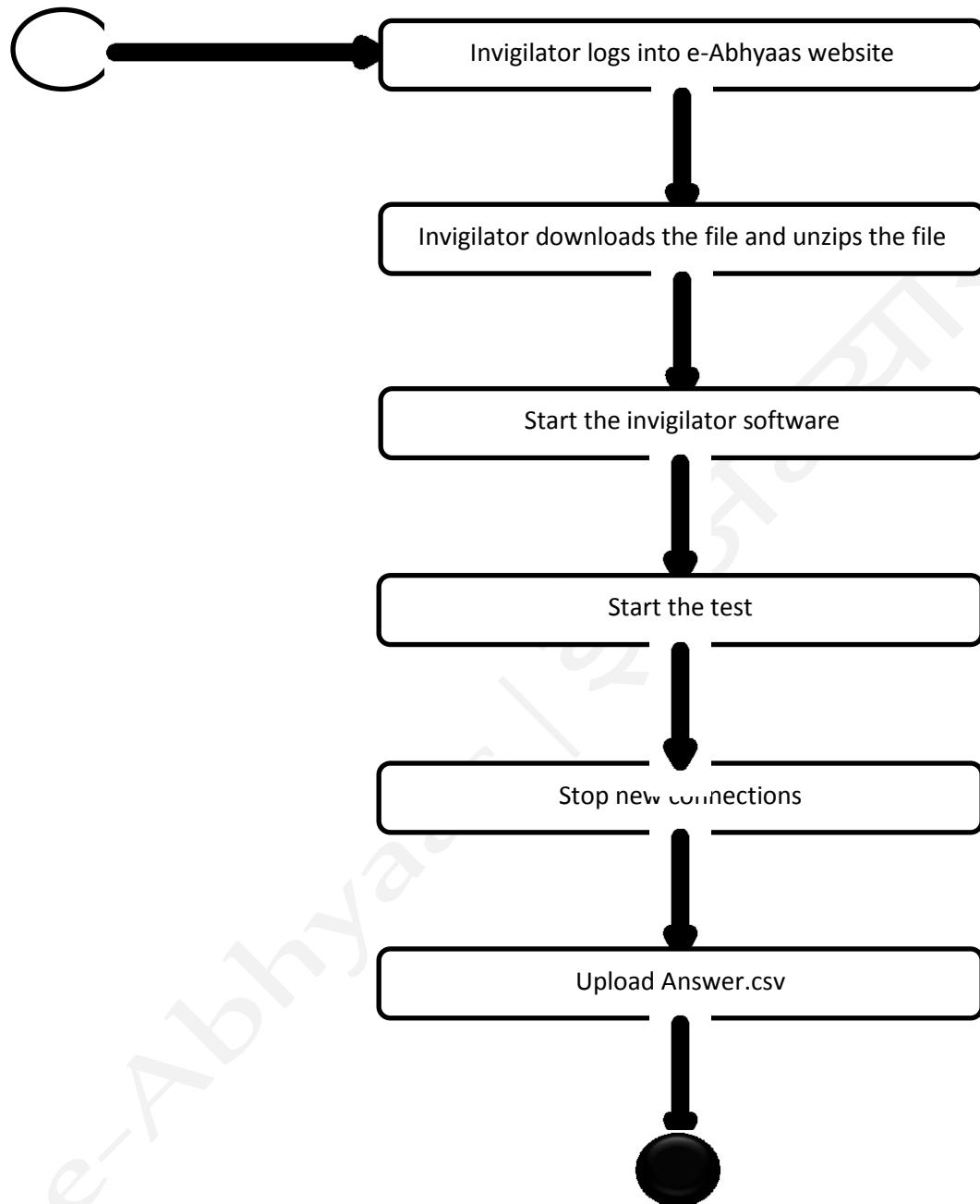


Class Diagram



Activity Diagram

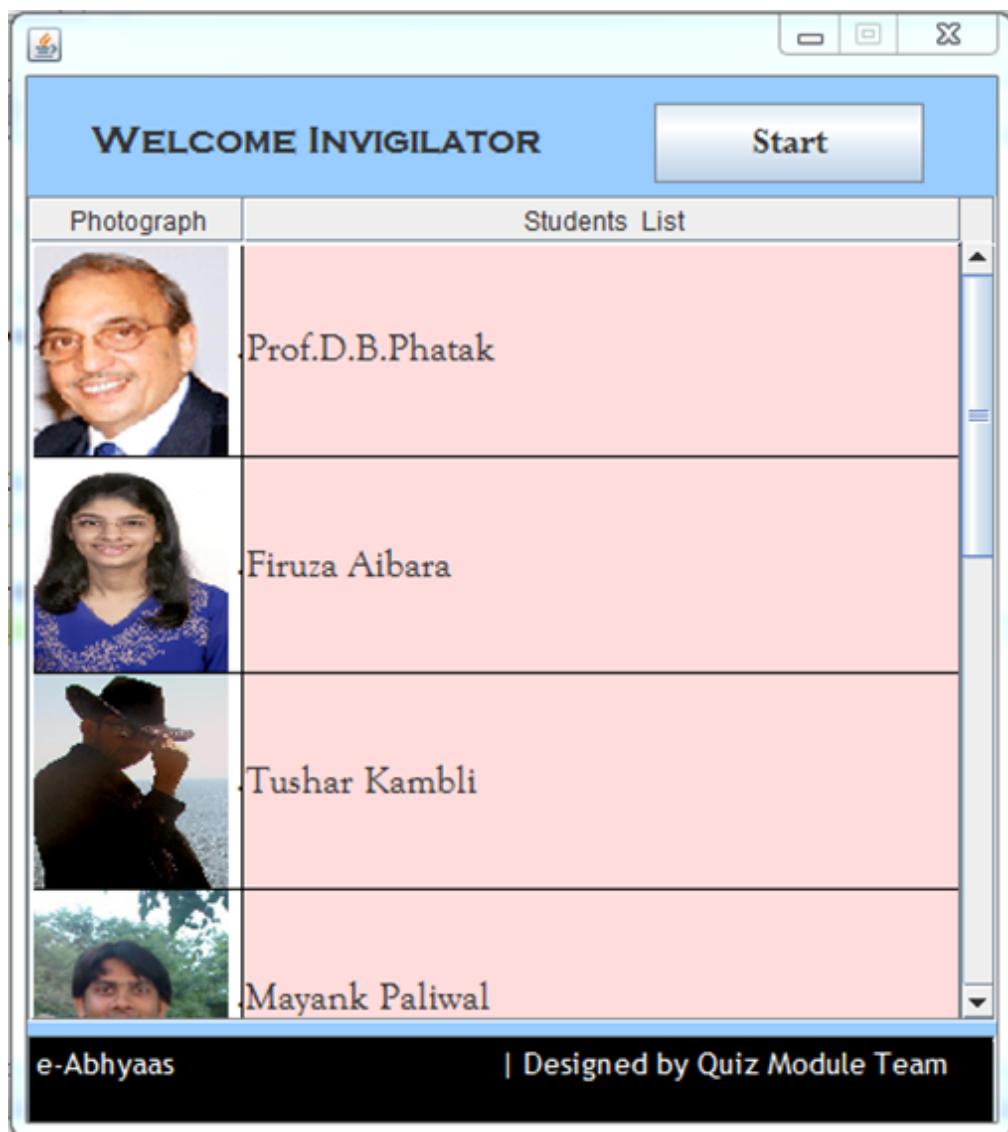
Invigilator:



Screenshots

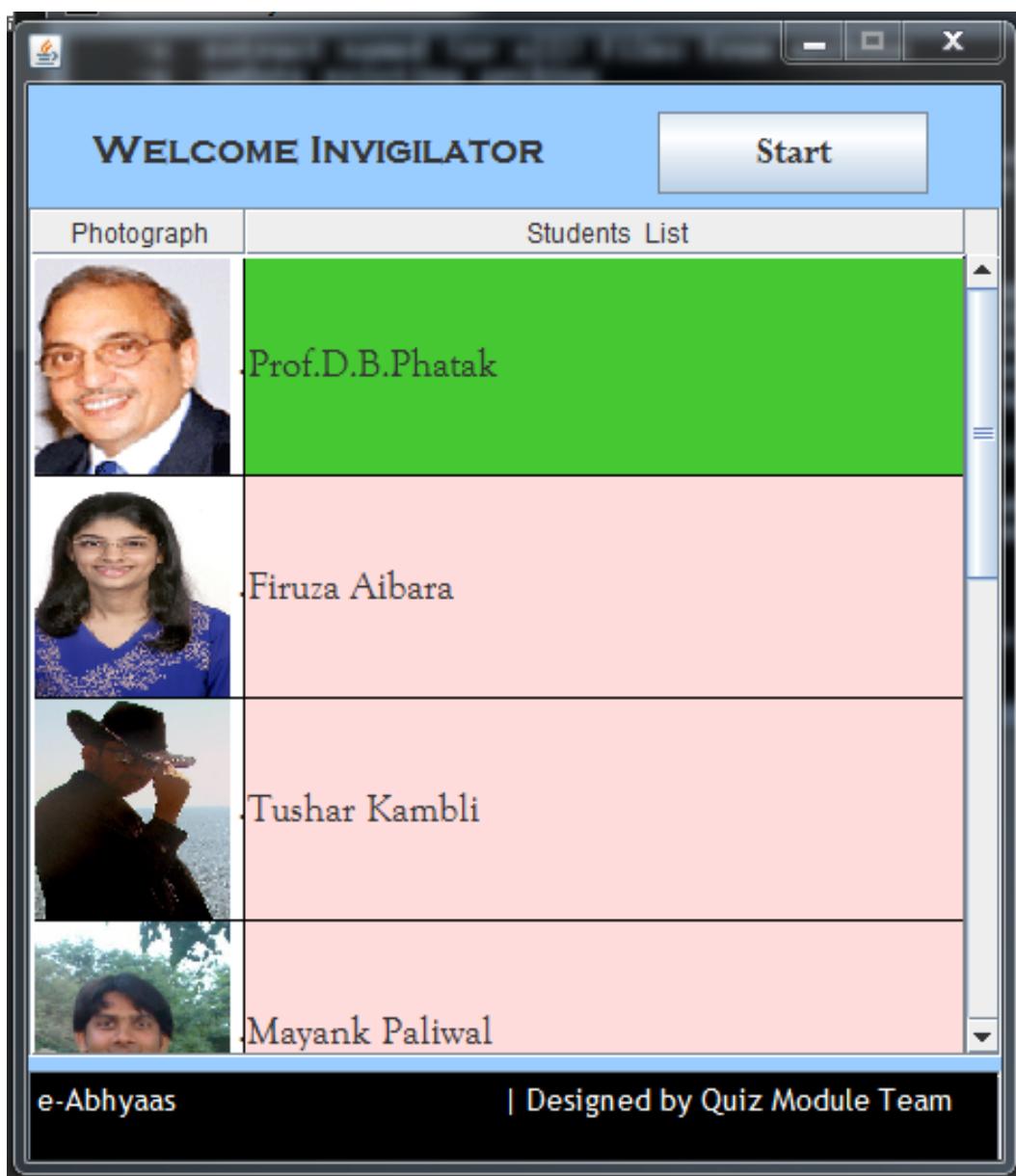
Invigilator Screen:

The invigilator gets this screen when he starts the application. At the beginning nobody is connected to the invigilator.



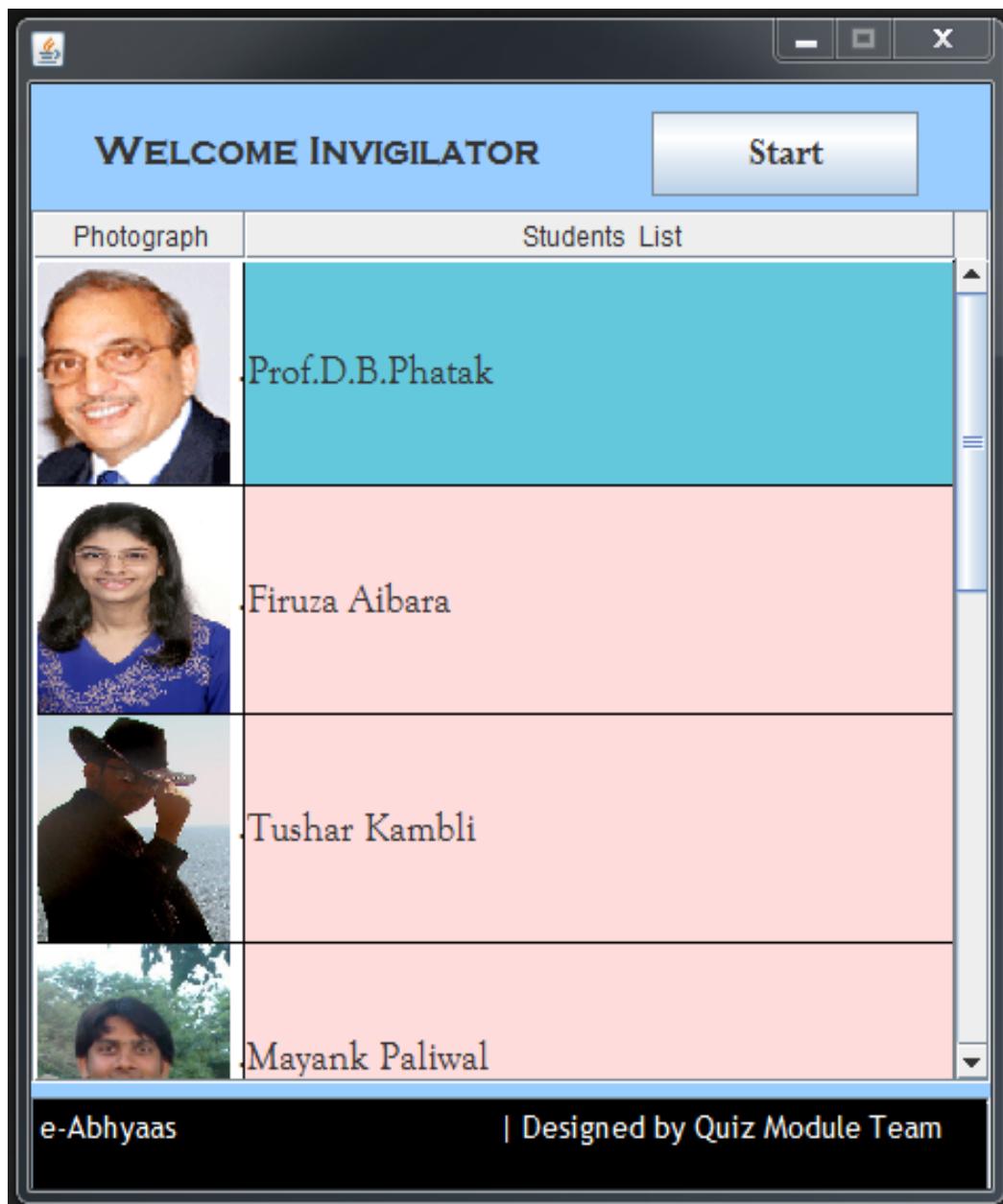
Start-up screen

This is how the application looks when files are being transferred to a particular student. It shows green colour as long as the files are being transferred.



Files being transferred

This is how the application looks after the files are transferred to the student. The particular student turns blue.



Files transferred to student

Web Interface

Introduction

The frontend can be described as the part of the website which gets published. It is what the owner of the website wants the public or selected groups to see when they open the site in their internet browser.

The goal of user interface design is to make the user's interaction as simple and efficient as possible i.e. user-centred design. Good user interface design facilitates finishing the task at hand without drawing unnecessary attention to it.

Sub System Overview

- Designed and edited CSS to match with all modules.
- Website CSS is in compliance with Web 2.0 standards and has explicit use of style-sheet, JSP pages and JavaScript files.
- Front-end design of e-Abhyaas is inspired from BITSAT (Birla Institute of Technology and Science Aptitude Test).

Explicitly Declared CSS Classes

Forms were to have a common design template; to achieve this “.fieldsetFormat” CSS class selector was used. For aligning label, input and their span (which incorporate errors) specific classes were used for each of them. For styling all the web pages with various colours and styles, different classes were used. The styles are applied to the various “<div>” tags used on web pages and various headings and buttons.

Classification

Web Pages are classified according to User Types:

- Public Pages:
 1. Home Page:

This is the welcome page of e-Abhyaas. It provides an introduction about the purpose and functionality of the website.

It provides a registration form for the Teachers and Invigilators so that they can register themselves for the various Quiz Submissions and Tests. It also provides the About, Announcements and Contact Us pages for further assistance. The Home Page is provided with the Login Page for the registered teachers and invigilators to login.

Home Page

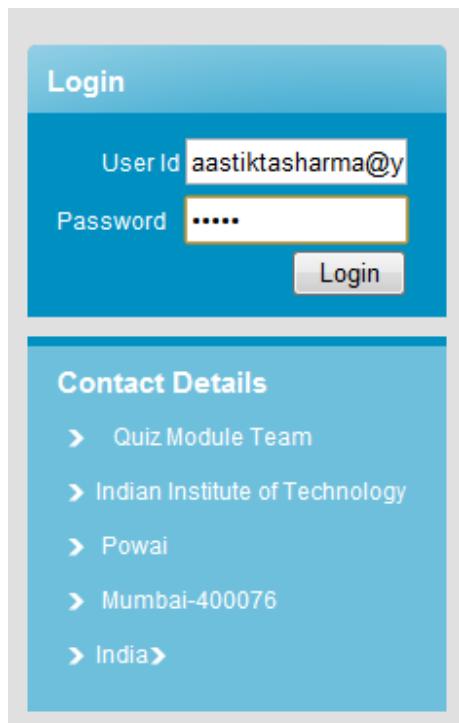
2. Registration Form:

It is the form for the Teachers and Invigilators to register themselves for e-Abhyas.

Registration

3. User Login:

The Teacher/Invigilator who is already registered logs in here.



Login

- Pages Visible To The Teacher:

1. Teacher Home Page:

The Teacher sees this web page after logging in.

Welcome Aastika !!

Log Out

Welcome to our website

An Introduction to the Quiz Module which has been developed. The benefits of the Quiz Module which will be enjoyed by the Teachers and Invigilators. And also an introduction to the various types of Quizzes that are available on the module. The various responsibilities shared among the teachers and the invigilators.

Announcements

- 1.The Teachers are asked to upload the question and answers in CSV format one week before the Test.
- 2.The Invigilators are asked to download the Seating Arrangement and the Centre allotted to them now.
3. All Teachers and Invigilators are requested to check their respective emails and confirm their registration

Let us sacrifice our today so that our children can have a better tomorrow.

2. Test Specifications:

The Teacher submits the test specifications here. He gives the test name which he is going to upload along with the total time and the total marks for the test. Then he gives the various specifications for the questions i.e. question number, question type, number of qualifiers and sub qualifiers, marks allotted for correct answer and marks deducted for wrong answers.

Welcome Aastikta !!

Log Out

Instructions :

- 1.Multiple Choice Single Correct Answer : Enter the number of Options in column 2 and enter ZERO in column 3.
- 2.Multiple Choice Multiple Correct Answer : Enter the number of Options in column 2 and enter ZERO in column 3.
- 3.Numerical Answer : Enter the total number of Digits before decimal in column 2 and enter number of digits after decimal in column 3.
- 4.Matrix Type : Enter the number of rows in column 2 and number of columns in column 3.
- 5.Match Columns : Enter the number of Rows in Column 2 and the number of Columns in Column3.
- 6.Enter Test Name , Total marks for the test and the Total time

Test Name-

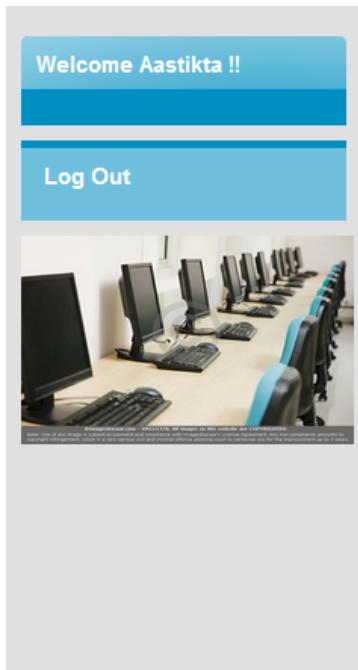
Total Marks- Total Time-

1	Multiple Choice Single Correct	<input type="button" value="▼"/>	1	<input type="button" value="▼"/>	0	<input type="button" value="▼"/>	Marks Allotted-	<input checked="" type="checkbox"/>	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
---	--------------------------------	----------------------------------	---	----------------------------------	---	----------------------------------	-----------------	-------------------------------------	----------------------	--------------------------	----------------------

Add Done Reset Submit

3. Answer Sheet:

After he submits the Question Specifications, he is directed to the answer sheet page where he marks the answers for the given question specifications. It is an OMR sheet where he marks the answers.



OMR

1 (A) (B) (C)

2 (A) (B) (C) (D)

3	(0)	(0)	(0)	(0)
	(1)	(1)	(1)	(1)
	(2)	(2)	(2)	(2)
	(3)	(3)	(3)	(3)
	(4)	(4)	(4)	(4)
	(5)	(5)	(5)	(5)
	(6)	(6)	(6)	(6)
	(7)	(7)	(7)	(7)
	(8)	(8)	(8)	(8)
	(9)	(9)	(9)	(9)
	(•)	(•)	(•)	(•)

4

P (A) (B) (C)

Q (A) (B) (C)

R (A) (B) (C)

After marking answers the Answer Sheet looks like this.

1	<input checked="" type="radio"/>	(B)	(C)	
2	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>	(D)
3	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>
	0	1	1	1
	2	2	2	2
	3	3	3	3
	4	4	4	4
	5	5	5	<input checked="" type="radio"/>
	6	6	6	6
	7	7	7	7
	8	8	8	8
	9	9	9	9
	.	.	.	<input checked="" type="radio"/>
4				
P	<input checked="" type="radio"/>	(B)	(C)	
Q	<input checked="" type="radio"/>	<input checked="" type="radio"/>	(C)	
R	<input checked="" type="radio"/>	(B)	<input checked="" type="radio"/>	

After marking the Answers he is directed to the last step i.e. uploading the Question Paper in .pdf format.

e-Abhyaas

Welcome Aastikta!!

Log Out

Home Quiz Specifications Results

Upload Question Paper(in pdf format):

Choose File No file chosen

Submit

Let us sacrifice our today so that our children can have a better tomorrow.

4. Results:

The Teacher can click on this page to see the results of various tests submitted by him that has been conducted.

Welcome Aastikta !!

Log Out

Home Quiz Specifications Result

Select test to view results-

Physics Test
G.K. Test

Let us sacrifice our today so that our children can have a better

Welcome Prabhanshu !!

Log Out

Home Quiz Specifications Result

Test Name:- GK

MacID	Student Name	Marks Scored	Total Marks	Questions Attempted	Number of Right Answers	Number of Wrong Answers	Status
00:0f:cf:02:24:39	Prof.D.B.Phatak	10	10	7	7	0	PASS
00:0f:cf:02:29:98	Anup Naik	2	10	7	4	3	FAIL

- Pages Visible to The Invigilator:

1. Invigilator Home Page:

The Invigilator sees this web page after logging in.

Welcome Krattika!!

Log Out

Instructions :

1. Go To Download Section and Download the CONNECTIFY INSTALLER.
2. Install it by double clicking on it.
3. Install it to take full control over your Hotspot.
4. Download the Question Paper in pdf format.
5. Download the Questions and Answers (CSV format).
5. Download the Photographs of Students(ZIP folder).
5. Download the Key for encryption (CSV format).

Let us sacrifice our today so that our children can have a better tomorrow.

2. Centre Allotment:

The Invigilator clicks on this web page to get the name of the centre allotted to him/her for the test.

Welcome Krattika!!

Log Out

Centre Allotment

MANIT Bhopal

Let us sacrifice our today so that our children can have a better tomorrow.

3. Seating Arrangement:

The Invigilator is informed about the students present under his/her invigilation.

Mac ID	Student Name
00:0f:cf:02:24:30	Firuza Aibara
00:0f:cf:02:24:11	Mayank Paliwal
00:0f:cf:02:24:38	Parag Tiwari
00:0f:cf:02:24:36	Deepak Jayanth
00:0f:cf:02:24:31	Ajay Babar

Welcome Krattika!!

Log Out

Seating Arrangement

Let us sacrifice our today so that our children can have a better tomorrow.

4. Downloads:

In this section invigilator downloads the ZIP folder which contains CSV files of Questions, Questions and Answers, Key for encryption of .pdf file and Student Mac IDs list. It also contains the setup for Connectify and photographs of the Students enrolled for the test under him/her.

Welcome Krattika!!

Log Out

Downloads

Download (ZIP folder)

Do you want to open or save **Invigilator.zip** (6.44 MB) from **localhost**?

Open **Save** **Cancel** **X**

Let us sacrifice our today so that our children

5. Result:

Here the Invigilator uploads the CSV file of the marks obtained by the students who gave the test under him.

The screenshot shows the e-Abhyaas Admin Home Page. The top navigation bar includes links for Home, Centre Allotment, Seating Arrangement, Downloads, and Result. The main content area has a heading 'Upload Results in CSV format:' followed by a file upload input field labeled 'Browse...', a 'Submit' button, and a small image of a student writing on a chalkboard. A sidebar on the left displays a 'Welcome Krattika!!' message and a 'Log Out' link. A photograph of a row of computer workstations is also visible. On the right, there is a quote: 'Let us sacrifice our today so that our children can have a better tomorrow.'

- Pages visible to the Admin:
 1. Admin Home Page:

The screenshot shows the e-Abhyaas Admin Home Page. The top navigation bar includes links for Home, Test Allotment, Centre Allotment, Student, and Seating Plan. The main content area has a heading 'Welcome to our website' followed by a paragraph of text about the Quiz Module and its benefits. Below this is a section titled 'Announcements' containing three numbered points: 1. Teachers upload questions and answers in CSV format. 2. Invigilators download seating arrangements and centre allotments. 3. Teachers and invigilators check their emails for registration confirmation. A photograph of a student writing on a chalkboard is on the right, along with the same quote: 'Let us sacrifice our today so that our children can have a better tomorrow.'

2. Test Allotment:

The Admin allots the test which has to be conducted.

Test No.	Test Name	Select
1	prabhanshu.agrawal2010@gmail.com;General Knowledge.pdf	<input checked="" type="radio"/>
2	prabhanshu.agrawal2010@gmail.com;G.K.pdf	<input type="radio"/>
3	prabhanshu.agrawal2010@gmail.com;.pdf	<input type="radio"/>
4	aastiktasharma@ymail.com;qwe.pdf	<input type="radio"/>
5	aastiktasharma@ymail.com;sdf.pdf	<input type="radio"/>
6	prabhanshu.agrawal2010@gmail.com;G.K..pdf	<input type="radio"/>
7	aastiktasharma@ymail.com;phy.pdf	<input type="radio"/>
8	aastiktasharma@ymail.com;ch.pdf	<input type="radio"/>
9	prabhanshu.agrawal2010@gmail.com;gk1.pdf	<input type="radio"/>

The selected test gets updated in the Database.

Test No.	Test Selected
1	prabhanshu.agrawal2010@gmail.com;gk1.pdf

Update

3. Centre Allotment:

Admin allots the Centre to the various invigilators for the test.

First Name	Last Name	Gender	State	City	Email	Centre Alloted	Test Allotted	Select
Krattika	Mishra	Female	Madhya Pradesh	Bhopal	mkrattika@gmail.com	MANIT Bhopal	prabhanshu.agrawal2010@gmail.com;gk1.pdf	Edit
akash	priyadarshi	Male	Bihar	Muzaffarpur	akash39.nitjsr@gmail.com	IIT Bombay	prabhanshu.agrawal2010@gmail.com;gk1.pdf	Edit

The screenshot shows the 'Student' tab selected in the navigation bar. A table displays a single student record:

First Name	Last Name	Date Of Birth	Gender	State	City	Mobile	Email	Centre Allotted	Select
akash	priyadarshi	10/03/1992	Male	Bihar	Muzaffarpur	9905902156	akash39.nitjsr@gmail.com	IIT Bombay	<button>Update</button>

On the left sidebar, there is a 'Welcome admin!!' message and a 'Log Out' button.

4. Student:

Admin gives the details of the students who are enrolled for the tests.

The screenshot shows the 'Student' tab selected in the navigation bar. A form allows adding student records:

Total Students-

1	Student Name- <input type="text"/>	<input type="text"/>	Mac IDs Alloted- <input type="text"/>
---	------------------------------------	----------------------	---------------------------------------

Add Done Reset

On the left sidebar, there is a 'Welcome admin!!' message and a 'Log Out' button.

5. Seating Plan:

The admin allots the invigilator to the students.

The screenshot shows the 'Seating Plan' tab selected in the navigation bar. A table lists student records with allotments:

Mac ID	Student Name	Centre Allotted	Invigilator ID Alloted	Select
00:0f:cf:02:24.39	Prof.D.B.Phatak	IIT Bombay	akash39.nitjsr@gmail.com	<button>Edit</button>
00:0f:cf:02:24.30	Firuza Aibara	NIT Py	mkrattika@gmail.com	<button>Edit</button>
00:0f:cf:02:24.40	Tushar Kamble	NIT Py	NA	<button>Edit</button>
00:0f:cf:02:24.11	Mayank Paliwal	NIT Py	mkrattika@gmail.com	<button>Edit</button>
00:0f:cf:02:24.59	Rajni Kant Jangir	NA	Na	<button>Edit</button>
00:0f:cf:02:29.98	Anup Naik	NA	NA	<button>Edit</button>
00:0f:cf:02:24.38	Parag Tiwari	NIT Py	mkrattika@gmail.com	<button>Edit</button>
00:0f:cf:02:24.36	Deepak Jayanth	NIT Py	mkrattika@gmail.com	<button>Edit</button>
00:0f:cf:02:24.31	Ajay Babar	NIT Py	mkrattika@gmail.com	<button>Edit</button>

On the left sidebar, there is a 'Welcome admin!!' message and a 'Log Out' button.

Admin selects the invigilator for the selected student from the table of invigilators.

The screenshot shows the e-Abhyaas application interface. At the top, there is a navigation bar with links for Home, Centre Allotment, Student, and Seating Plan. Below the navigation bar, the title "Invigilator List (Allot Invigilator to each Student) :" is displayed. A table lists two entries:

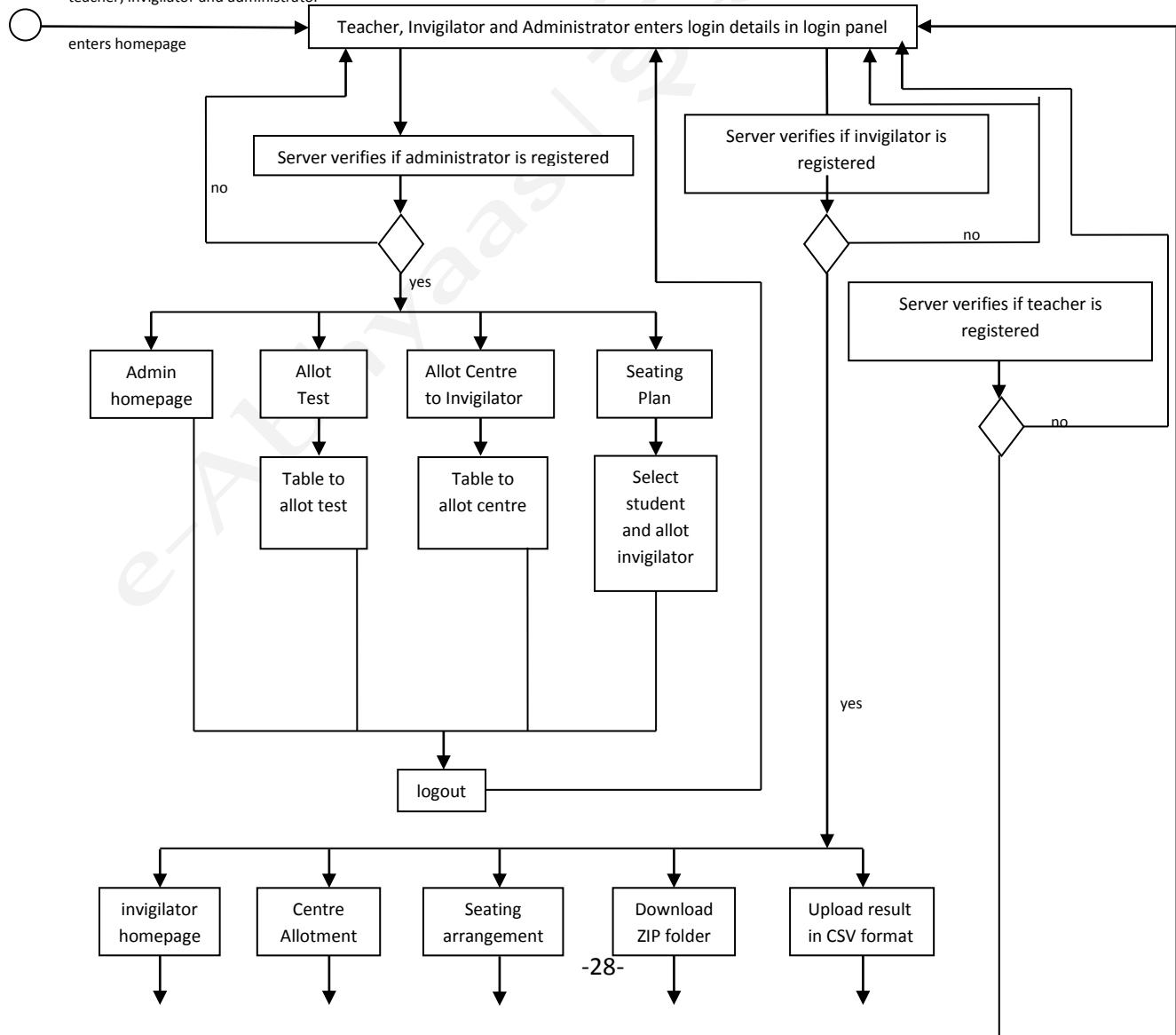
First Name	Last Name	Gender	State	City	Email	Centre Allotted	Test Alloted
Krattika	Mishra	Female	Madhya Pradesh	Bhopal	mkrattika@gmail.com	MANIT Bhopal	prabhanshu.agrawal2010@gmail.com;gk1.pdf
akash	priyadarshi	Male	Bihar	Muzaffarpur	akash39.nitjsr@gmail.com	IIT Bombay	prabhanshu.agrawal2010@gmail.com;gk1.pdf

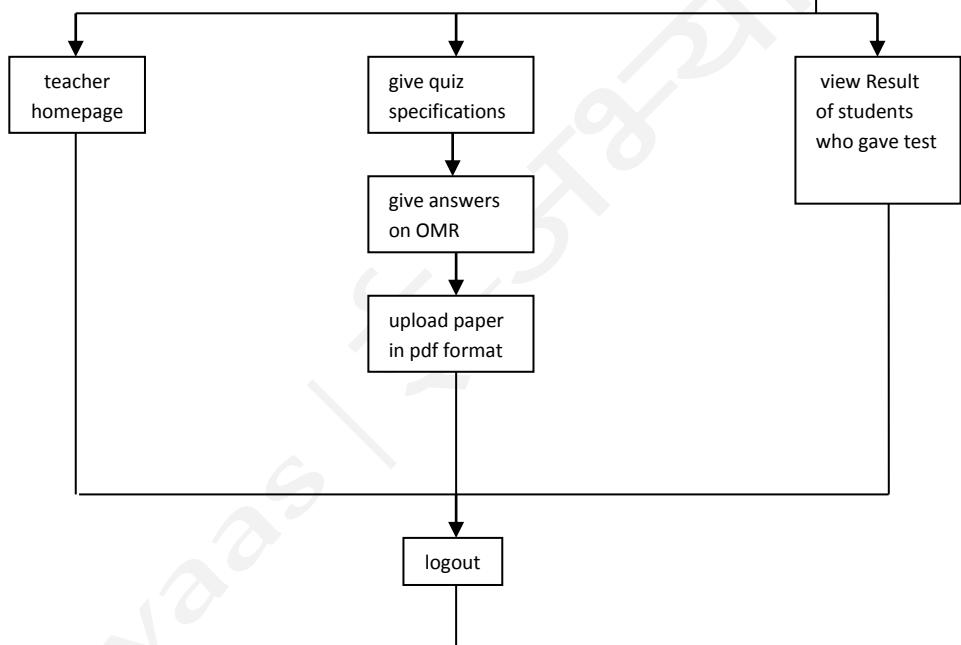
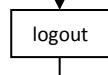
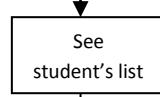
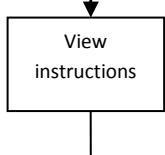
Below this, the title "Student List :" is shown, followed by another table:

Mac ID	Student Name	Centre Allotted	Invigilator Alloted	Select
00:0f:cf:02:24:30	Firuza Albara	NIT Py	mkrattika@gmail.c	<input type="button" value="Update"/>

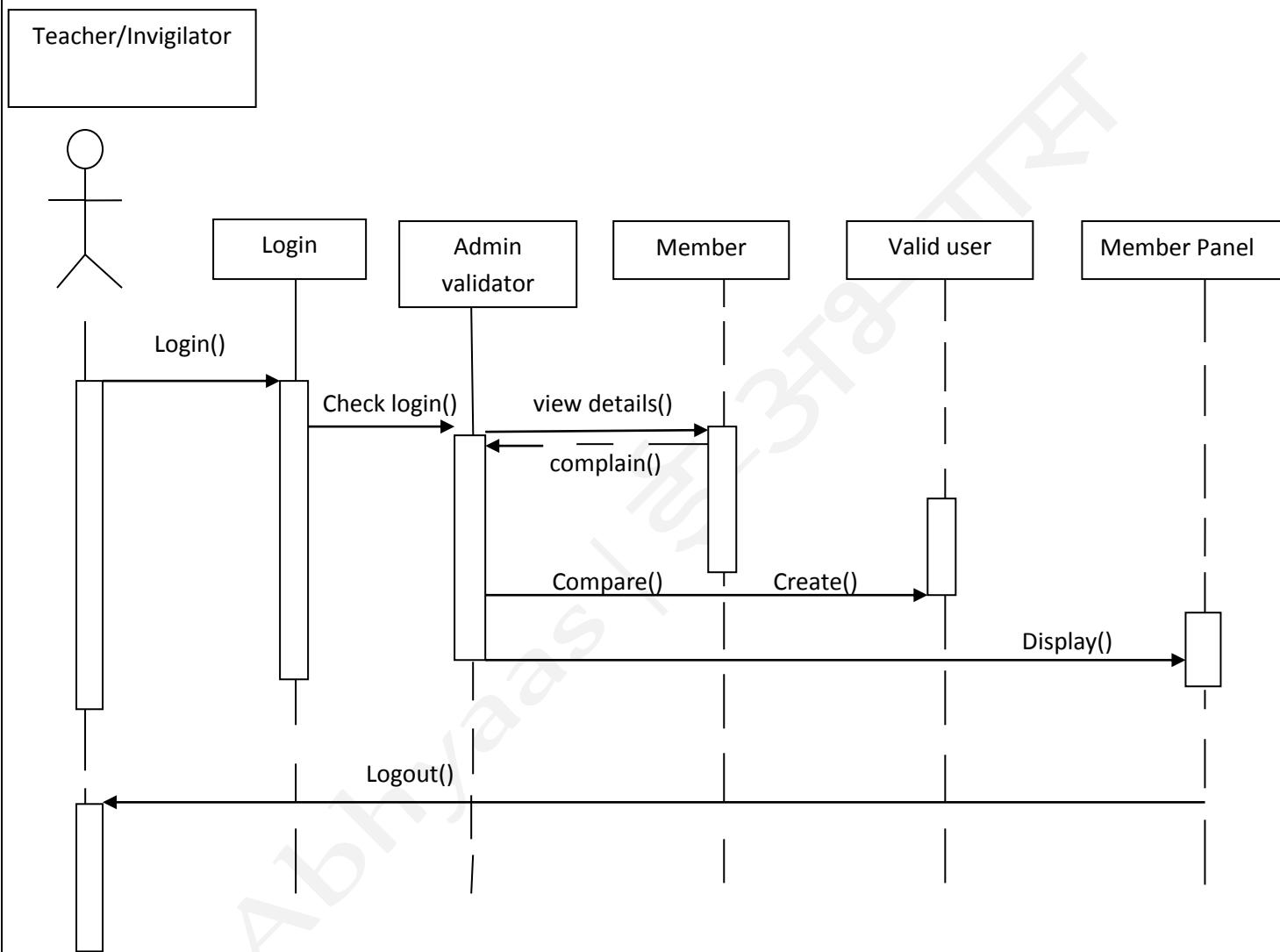
Activity Diagram

teacher, invigilator and administrator





Sequence Diagram



Content Module

- Stages of Content Flow in Quiz module:
Teacher → Quiz Upload → Invigilator → Download → Test
- Question Types Supported:
 - Multiple Choice Single Correct
 - Multiple Choice Multiple Correct
 - Numerical Answer
 - Match the Column
 - Matrix Match
- Directory Structure for the uploaded files :
Content/uploads/emailid; test_name.pdf
- Database Details:

Database:- Quiz _ModuleDatabase

- Table Schemas:

1. QuizDatabase:

Field	Type
TestName	varchar (50)

2. Registration:

Field	Type
Role	varchar (20)
FirstName	varchar (25)
LastName	varchar (25)
DOB	varchar (25)
Gender	varchar (25)
State	varchar (25)
City	varchar (25)
Mobile	varchar (25)

Email	varchar (50)
Password	varchar (25)
CentreAllotted	varchar (100)
TestAllotted	varchar (100)

3. Student:

Field	Type
MacID	varchar (100)
StudentName	varchar (100)
CentreAllotted	varchar (100)
InvigilatorAllotted	varchar (100)

- o Relevant Table Details:

1. QuizDatabase:

This table stores the names of the question paper that have been uploaded by the teacher. The name of the uploaded file includes the email id of the teacher and the test name given by him/her in pdf format.

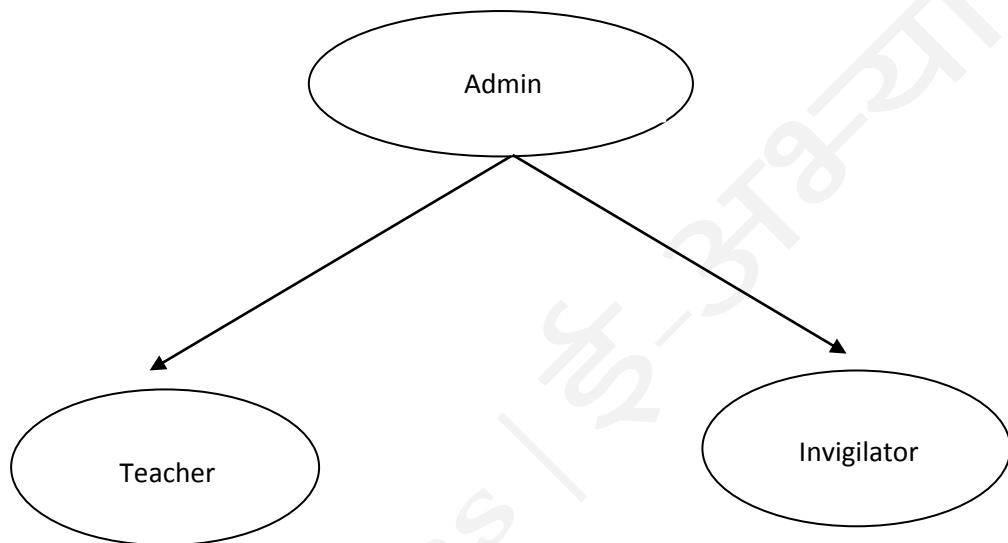
2. Registration:

This table stores the details of the people registering themselves for the role of teacher or invigilator. The details include role of the person registering, first name, last name, date of birth (under constraints i.e. between 1900 and 2200), gender, state, city, mobile (10 digit number), Email (of format xyz@pqr.abc), password (given by the person so that he/she can login using this password), Centre allotted (it will be assigned during the allotment of centre to the invigilator and in case of teacher remains NA),test allotment (it will be assigned during the allotment of test by the admin).

3. Student:

This table stores the student database. The details of Students include their Mac Ids (IDs of the machines allotted to them), Student name, Centre allotment (it is assigned by the person giving the database), Test allotment (it is assigned by the admin while allotting students to a particular invigilator).

Accessibility Flow



Disabling Android UI & Encryption

Disabling Android UI

The most important security feature for an exam is that the student should not be allowed to cheat. So, the main concern of the application is not to allow the student to switch to other apps during the exam.

For that, the “BACK” button, the “HOME” button and the “NOTIFICATION BAR” are disabled so that a student will have no chance to switch to other apps. But still for security we have set an alarm which gets activated whenever a student will switch to other app by any other means (like using shortcut methods using the External Keyboard).

And as the device has a facility of phone calls too, if this application is running then it is taken care that this application has the highest priority i.e. no phone calls (incoming or outgoing) will be allowed till the application is running or in other words the student is giving the exam i.e. the application has a programmatic approach to reject the incoming calls without user intervention. This approach uses JAVA reflection to call methods of an internal class of Android Telephony Framework. The application tries to get the instance of the hidden class of Android Telephony Framework. This can be said as a service activity which runs throughout the application.

PDF Reader

The application comes with an embedded PDF reader. The device need not have a pre-installed PDF reader for it because the application comes with an inbuilt PDF reader. As soon as the PDF reader activity is fired it reads the PDF file and displays it on the screen with “BACK” button, “HOME” button, “Notification Bar” and phone calls disabled in the background.

Elapsed Time

Questions Attempted

Max Marks: 20

5 questions

Duration: 1 minute

1) World Computer Literacy Day is celebrated on

A. 17th November
B. 2nd December

2) A Close bottle containing water at room temperature is taken to the moon and then the lid is opened. The water will

A. Boil
B. Freeze
C. Decompose into oxygen and hydrogen
D. None

Answer sheet

50

Notification Bar hidden

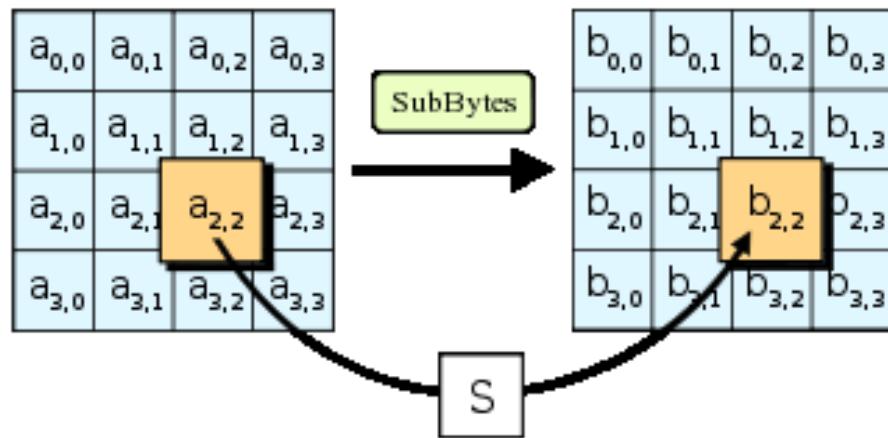
An opened PDF File

Encryption and decryption

As the question paper is sent to student using Wi-Fi so there are chances of breaching the question paper for the security the paper is encrypted during the paper upload. As the teacher clicks on upload button in the module the encryption starts and an object file or dot 'o' (.o) file is created. And the encryption technique used is "AES" technique which generates a key at the type of encryption which is required using decryption. So, when the invigilators click on start button to start the exam the key (also encrypted) is sent and the decryption starts at the student side and then the PDF activity is called which displays the PDF.

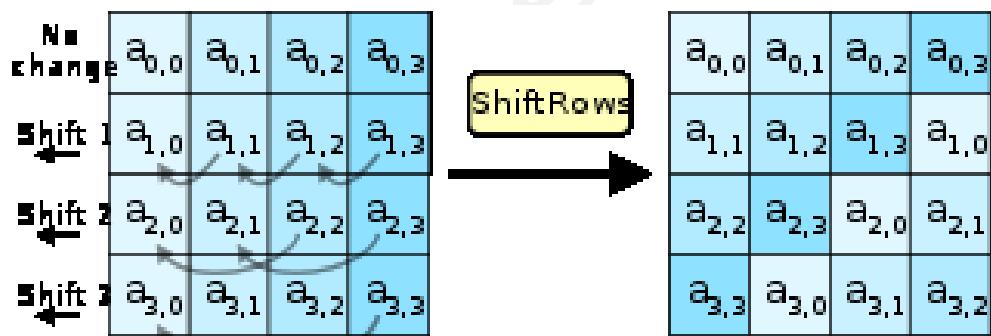
Description of the encryption algorithm

- Key Expansion—round keys are derived from the cipher key.



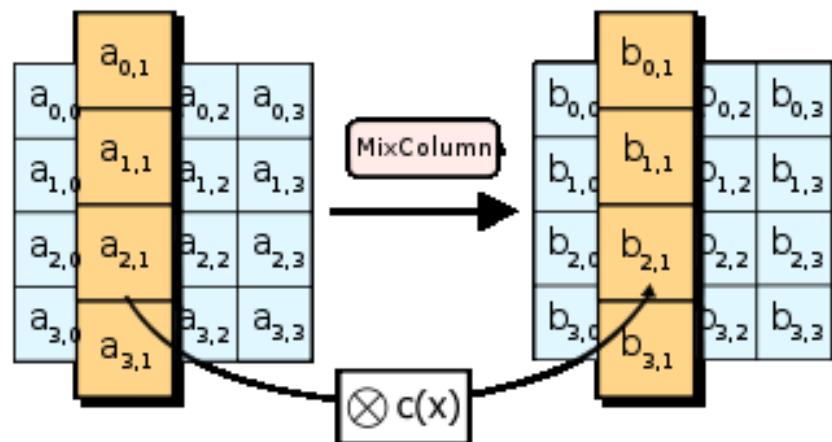
- Initial Round

- Add Round Key—each byte of the state is combined with the round key using bitwise xor.



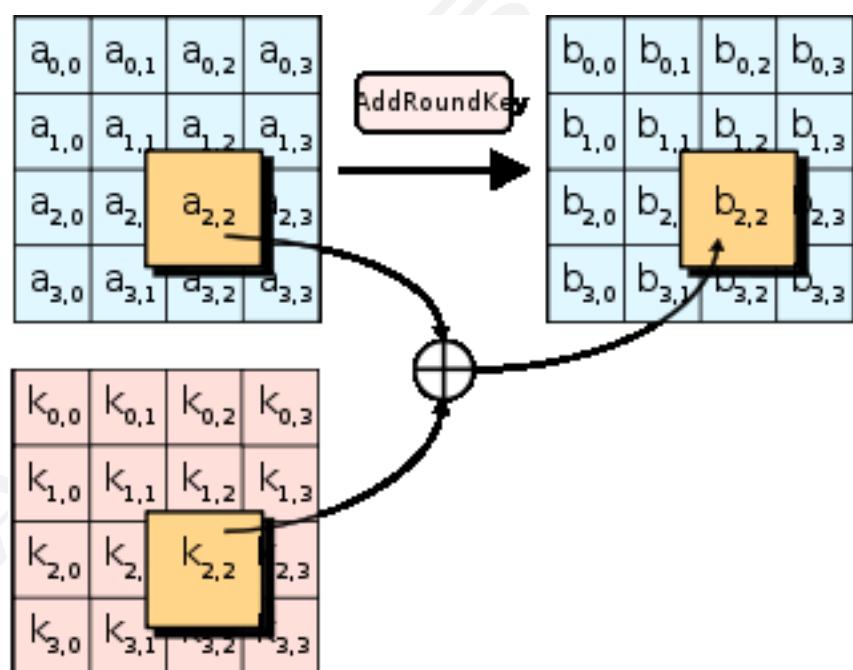
- Rounds

- Sub Bytes—a non-linear substitution step where each byte is replaced with another.
- Shift Rows—a transposition step where each row of the state is shifted cyclically a certain number of steps.
- Mix Columns—a mixing operation which operates on the columns of the state, combining the four bytes in each column.
- Add Round Key

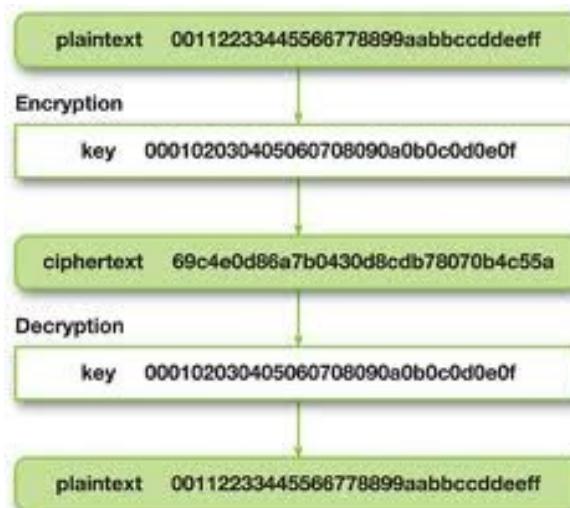


4. Final Round (no Mix Columns)

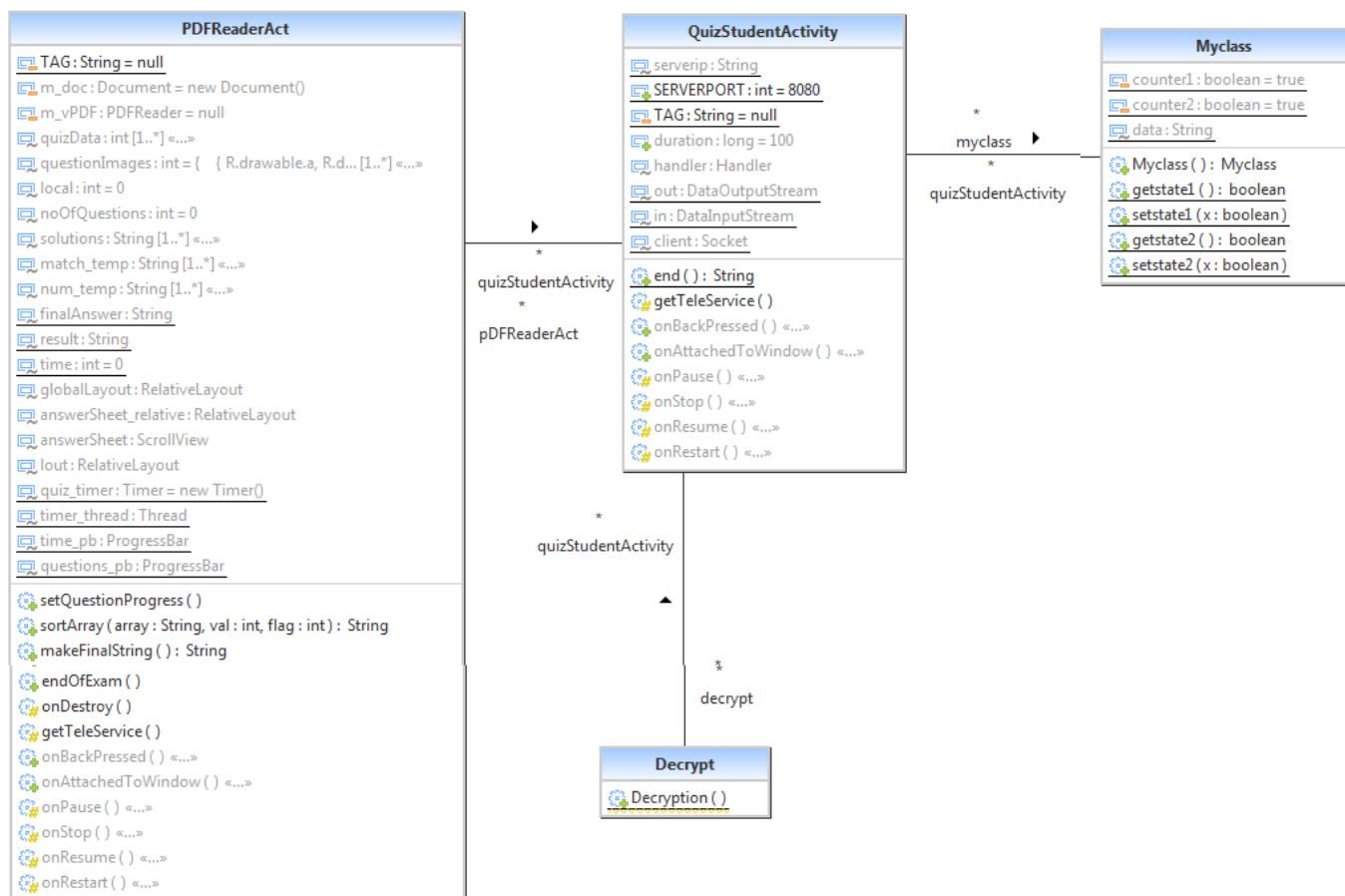
1. Sub Bytes
2. Shift Rows
3. Add Round Key



An example of AES encryption of a simple plain text is shown below in the diagram:-



Class diagram



Answer sheet

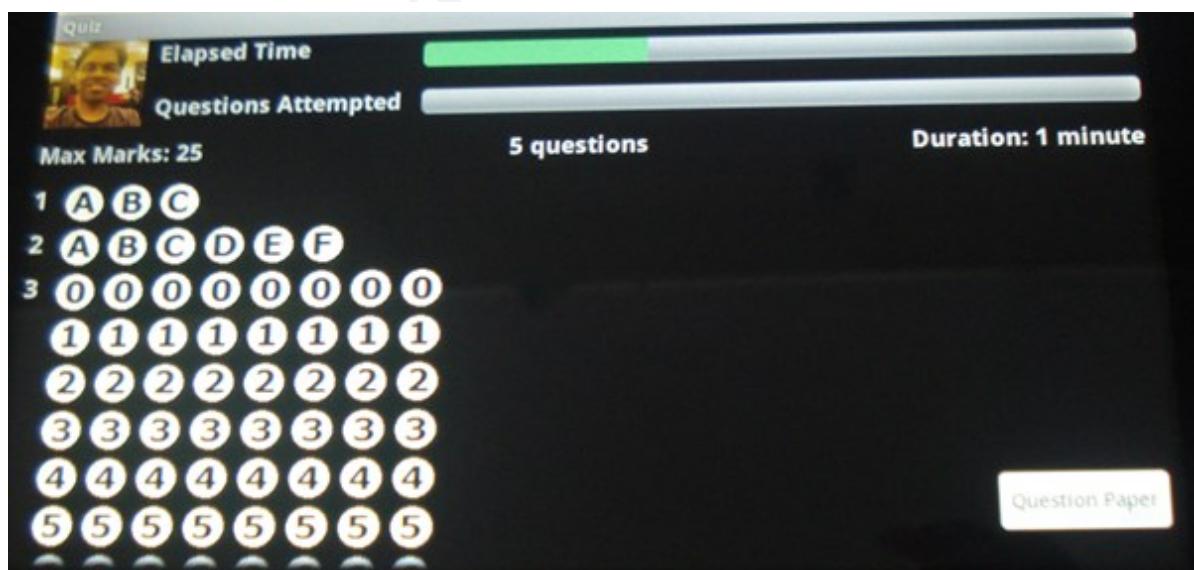
The answer sheet displayed to the student is a dynamically generated answer sheet which is decided according to the specifications of the question paper. Questions can be one of five types: single choice correct, multiple choice correct, numerical answer, match the following and matrix match.

How it works

- First, the CSV file containing the quiz specifications is read. The first line of the CSV file contains the number of questions in the test, the duration of the test and the maximum marks of the test. The subsequent lines contain the specifications for each question – question number, question type, qualifier, sub-qualifier, marks for correct answer and marks for incorrect answer.

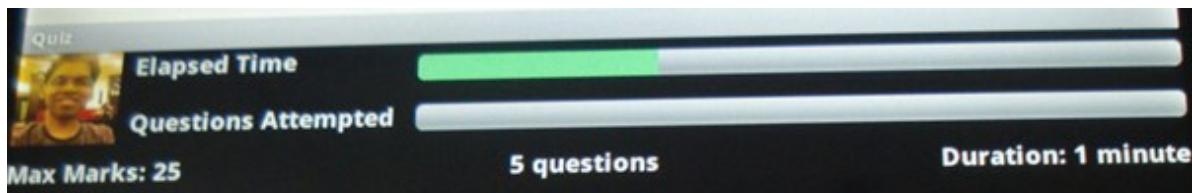
Time in minutes	
Number of questions	← 5,1,25, → Maximum marks of the test
	1,1,3,0,3,1,
	2,2,6,2,4,0,
	3,3,8,0,5,2,
	4,4,5,5,12,0,
	5,5,4,6,1,0,

- Now the answer sheet is prepared. The answer sheet consists of two parts – an upper fixed view (“essentials”) and a bottom scrollable view.

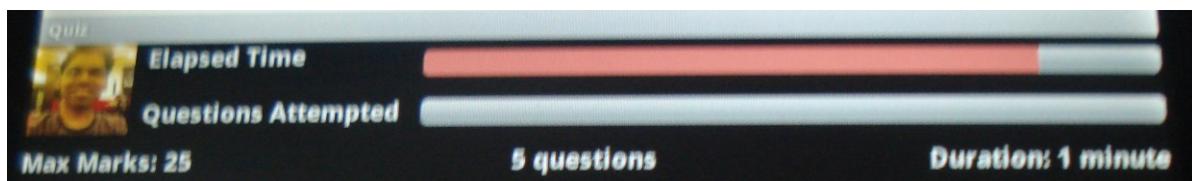


A sample answer sheet

3. Essentials consists of the student's photograph, a progress bar which shows the elapsed time, a progress bar which shows the number of questions attempted, the maximum marks of the test, the number of questions and the total duration of the test.

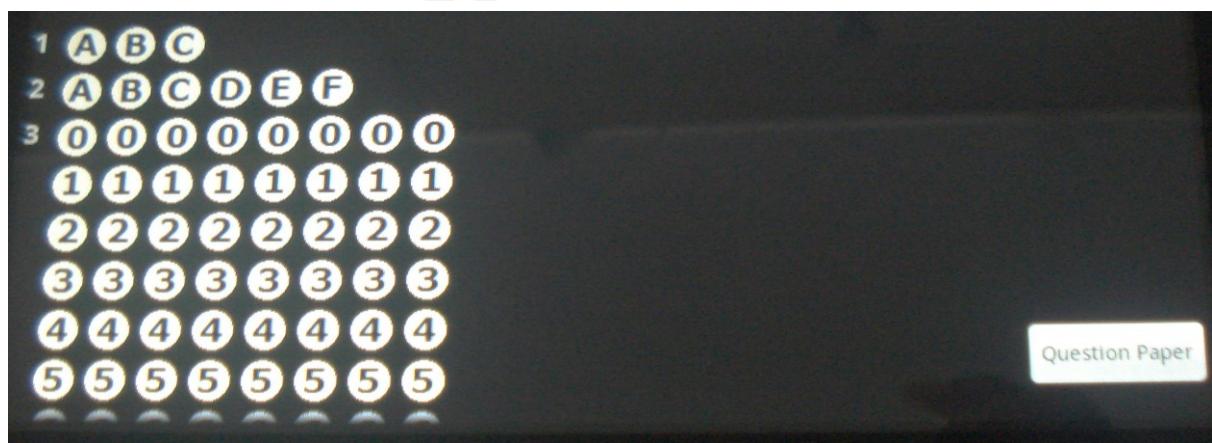


The top view – essentials. This contains the progress bar for elapsed time, questions attempted and displays the maximum marks for the test, number of questions and the duration of the test

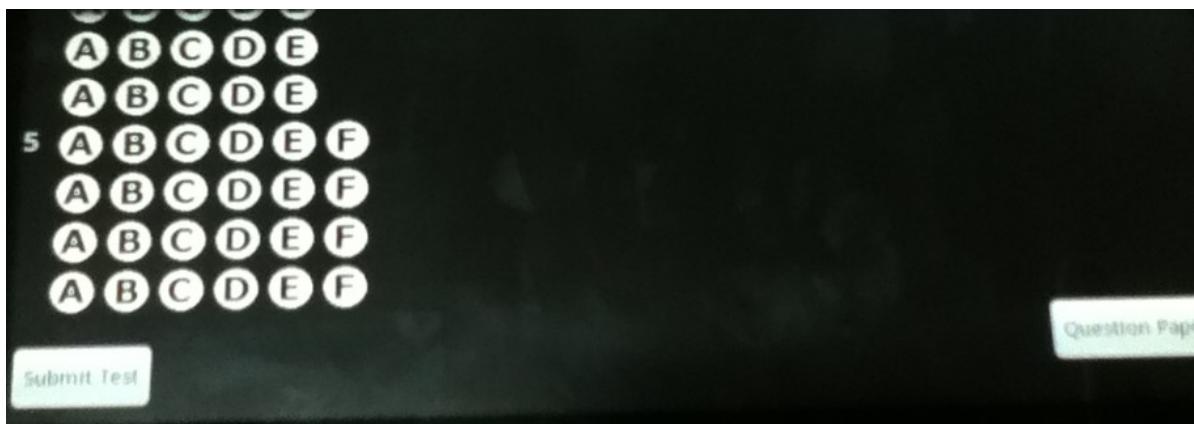


In the last 20% of the time, the progress bar turns red.

4. The bottom scrollable view contains the answer sheet. Clicking once on an option selects it and clicking again deselects it. Clicking on the “Question Paper” button takes the student back to the question paper. Clicking on “Submit” submits the paper and shows the marks.



The bottom scrollable view. This contains the answer sheet, the “Submit” button (shown in the next picture) and a button to show the question paper.

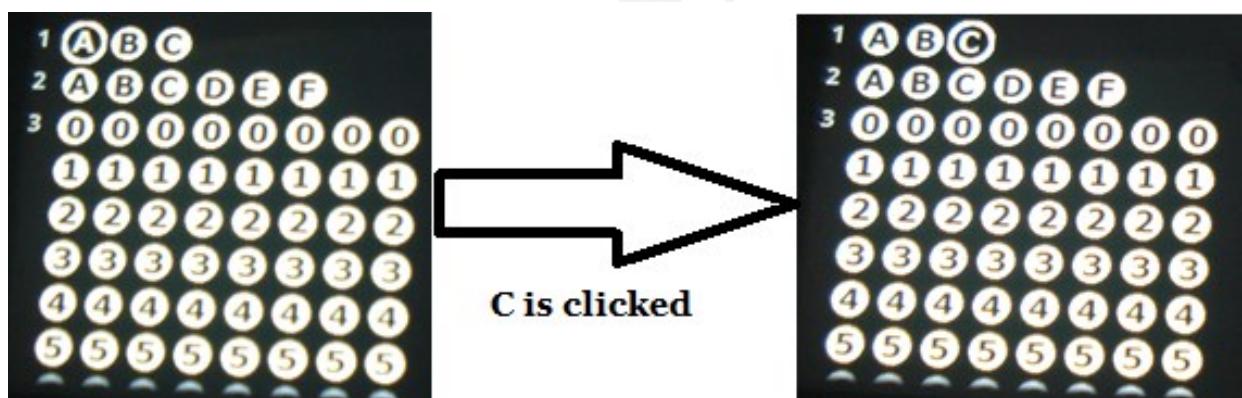


Answer sheet showing the “Submit” button

Various types of questions

1. Single choice correct:

This is the most common type of question. In this case, each question has just one correct answer. In such type of questions, a student can never select more than one answer for a question. If a student selects another option, the previous option gets de-selected and the new option gets selected.

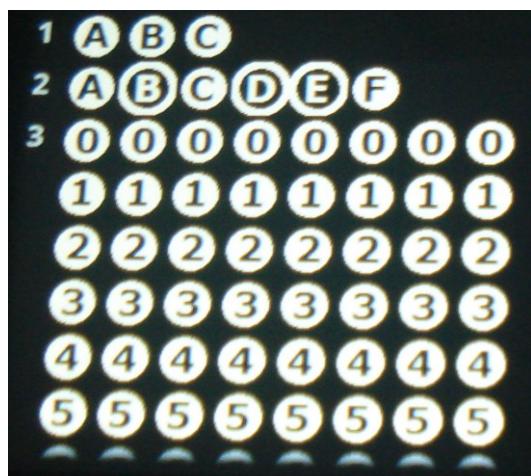


A is selected now

**A gets deselected and
C gets selected**

2. Multiple choice correct:

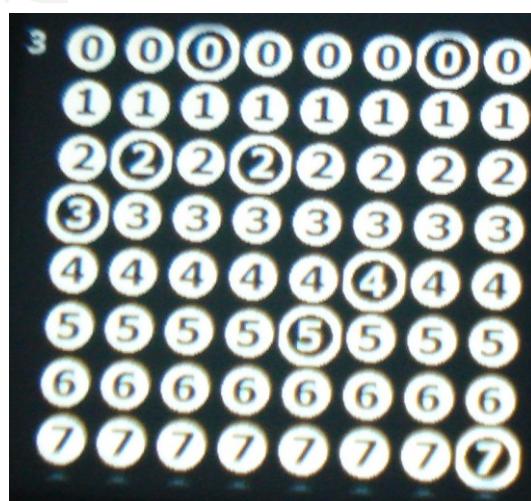
In this type, any number of answers can be correct for a question. Hence, a student can select any number of answers – how many ever he deems to be correct – for a particular question.



Multiple options selected for second question

3. Numerical answer:

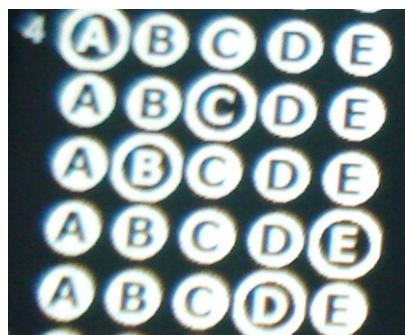
Some questions require the student to give a numerical type answer rather than any specific option. e-Abhyas allows teachers to set questions which demand a numerical answer. The number of digits has to be specified and the layout shown below is generated. The student selects the answers by tapping the required digits which make up his answer.



Numerical type question – each column represents a placeholder for a digit

4. Match the following:

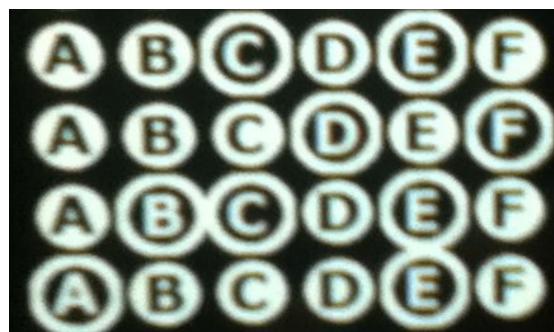
Sometimes, the teacher wants the student to match 2 columns. e-Abhyaas provides this facility too. In match the following, each row can have just one correct option. Again, the student taps once to select the option and taps again to deselect. And, for each option, he can't select more than one option like in single correct.



This is how match the following looks

5. Matrix match:

This is similar to match the following type questions except that each row can have more than one correct option.



Matrix match – where each row can have more than one answer

How answers are captured

An array of answers is maintained where the index number stores the answer of question “index number + 1” i.e. `answers[0]` will store the answer of question number 1. When a student clicks on any option on the answer sheet, that particular entry in the array gets updated. It essentially gets re-written. This is easy for single choice correct answers, but gets more challenging for other types of questions.

1. Single choice correct:

For single choice correct questions, when a student clicks on any option as his answer, that particular option is written into the array. And when an option is deselected i.e. the student doesn't want to attempt the question, an empty string is stored.

2. Multiple choice correct:

Multiple choice correct questions pose a greater challenge than single choice correct questions because as each option is selected or deselected, the array must update for the entire question. So each time a student selects or deselects an option, all the options are checked to see which ones are selected and those selected are put into the array as a string. And if the question is not attempted, an empty string is stored.

3. Numerical type

In numerical type questions, when a number is selected, a loop is run and in each column, the selected number is appended into a string. Again, in each column, the student can select only one column. And, as usual, questions which are not attempted are stored as an empty string.

4. Match the following

In match the following, each row can have only correct answer. Arrays are maintained for match the following questions and each cell corresponding to each row stores the answer. When an option is selected or deselected, the corresponding entry in the array is updated. Then, a loop is run and all the answers are made into a single string and stored in the answers array. As usual, questions which have been left by the student are stored as an empty string.

5. Matrix match

Matrix match questions use a similar logic to match the following type questions to read the answers. An array is maintained where each row's answers are stored. Now, when the final string is prepared, the answers for each row are separated by a period so that it is known which row is which answer. This was not done in match the following as each row had only one answer. Since in matrix match, each row can have multiple answers, they are separated by a period. And, questions which are not attempted are stored as an empty string.

End of exam

The exam can end due to 2 causes: either the student clicks “Submit” or the time is over. When the exam ends, a function is called which reads the answers array and makes an answers string which is passed to the invigilator for evaluation. In the answers string, the answers are separated by a semi-colon. To guard against unforeseen circumstances, this function is called every time an option is selected.

Future Scope

This is version 1.0 of e-Abhyaas. As with any first version, there are always certain limitations which we were forced to accept. The first thing is that the invigilator is forced to install Connectify and all the students must only connect to the hot spot created by that software on the invigilator's laptop. Any routers which the school already has are of no use. In the next version, students and invigilators should be able to connect to the school routers (if present) and conduct the test.

The next limitation is that on the server, the system administrator has to manually create a folder called "uploads" so that the files uploaded by the invigilator go into the correct folder. This can be updated in the future where the folder can be automatically created if not present.

Another constraint is that a student who uses a keyboard with the tablet can potentially exit the app to browse the web for answers. Version 1.0 of e-Abhyaas allows students, who use keyboards, to exit the app. In a forthcoming version, we must be able to stop the student from exiting the app even with any kinds of hardware accessories.