**INTRODUCTION**

The chief idea of this project is to develop a full-stack web application which would function as an online test management system for academic institutions. The application would function in a closed network in an institution where it is used to conduct examinations. It would allow students and teachers to sign-up to the website and create an account to rejoice all the conveniences of an online test that the application offers. The format of the test that the application allows to conduct is of multiple-choice nature, through which auto-correction and instantaneous evaluation becomes possible.

The teacher gets all the control and conveniences he/she would require to conduct a test in order to examine the students. He/she gets a dashboard of his/her own on registering whereby all the basic information of the teacher is displayed. This acts as a mini-profile for the teacher which could be updated by him/her anytime. He/she may organize a test anytime by simply setting the question paper and submitting it. On making the submission, the newly created test would be stored and displayed in the list of tests organized by the teacher previously, every time he/she logs-in. The newly created test can be initiated by the teacher any moment as per his/her wish and convenience. After the initialization of the test, the application efficiently takes care of smooth and successful completion of it. The application recognizes the students who are supposed to appear in the test and makes the question paper accessible to them. It keeps track of the duration and the remaining time so that no conflicts or disruptions occur in the examination process. Based on the data provided by the teacher while setting the question paper, the application automatically evaluates the responses of each student who appeared in the test and generates individual mark-sheets for them instantly. The teacher on the other hand receives the collective performa of the test which enlists the performance of each student. The performa would be downloadable as a file as well.

Every registered student gets the facility to log-in to his/her account and appear in the test instantly from a device connected to the network. Unlike pen-and-paper examinations, the student here is allowed to change his/her response to a particular question as many times he/she wishes, but before making final submission and within the duration of the examination. On the completion of the test, he/she not only gets his/her evaluated mark-sheet but do also receive a downloadable file of the answer keys of the question paper he/she attempted. Moreover, a student, on logging-in, gets details of all the upcoming tests which he/she is supposed to appear as well as that of the previously-appeared tests and his/her performance in those tests. The student gets a pretty interactive interface while responding to a question paper where all relevant controls are present right at the screen and highlighted so that he/she doesn’t lose time learning to use the tools and finding the controls. Additionally, a particular question can be bookmarked by the student and such questions are allotted a separate space on the screen for the student, so that he could make a response to it at a later period of time during the test hours.

The application is not only an online medium of organizing examinations but is a strong and effective alternative to the conventional pen-and-paper mode of test and examinations which currently prevails in almost every educational institution as it resolves various drawbacks of the standard method and makes the entire procedure of examination a whole lot simpler, easier and manageable. It nullifies the scope of wrong-evaluation of answer-scripts which is always likely to happen while manual checking due to lack of concentration and hence it also sweeps out the burden of scrutinizing the answer-scripts of each student by the teacher saving a lot of time. Moreover, with the mark-sheet and the answer keys being in a state of ready-to-access for the student all the time, he/she can always refer back to these materials to analyse himself/herself, clear the topics covered in the question paper as well as prepare better for future examinations. The application gives its users entire control over the data they create or the activity they do. For a teacher, he/she may delete a test from his/her dashboard whether conducted or not.

**FRONT-END DEVELOPMENT TOOLS**

The front-end development of “Quick Test” has been accomplished through the utilization of primary front-end technologies such as HTML, CSS and JavaScript on one side and other various secondary tools such as Skeleton (a CSS framework), jQuery (a JavaScript library), jQuery UI (jQuery’s sub-ordinate for more complicated user-interface development) and AJAX (client-side script for data exchange with the server). These tools and technologies have been discussed in vivid details below:

1. **Hypertext Mark-up Language (HTML):** The view of the web-pages of the project has been developed upon HTML-5. HTML-5 (or HTML in general) acts as the skeleton to the web-pages whereby we defined the basic structure of it. In this well-defined structure, content is placed in various states and based on all the actions and events that occur through the client.
2. **Cascading Style Sheets(CSS):** CSS is the tool that is incorporated into a web-page to enhance its appeal. CSS takes care of all the decorative part of the pages so that they are appealing and eye-catchy to the user. CSS also makes the web-pages a whole lot interactive for the user based on his/her actions on various static elements on the page through animation, shadows and varying style definitions based on various states of the elements such as hover, active and focus just to name a few. The project uses CSS-3 to stylize the content of the web-pages as seen by the client.
3. **Skeleton:** Skeleton is a CSS framework that consists of pre-defined classes for specific HTML elements. These classes have their own set of CSS properties defined. A web-page may be linked to the skeleton style-sheets and then all the classes which exist in skeleton gets the properties associated with it automatically. The framework thus reduces the job of styling the pages to some extent. Moreover, one big advantage of skeleton over other CSS frameworks is that it is very light-weight and hence doesn’t adds much to the entire size of the application. Besides, the skeleton style-sheet is quite simpler to read and understand for other developers and hence gives a better idea of what exactly happens on including different skeleton classes.

Quick-test uses the latest version of skeleton i.e., Skeleton-2.0.4 and exploits its grid system in the project.

1. **JavaScript(JS):** JavaScript or ECMAScript is the native scripting language of almost every web browser today. Thus, the programming logic of a web page and its inner content is written through JavaScript. These codes are then run by the browser according to the logic defined in the script. The various events created by the user are handled through JavaScript event-handlers and programming codes. In other words, JavaScript brings life to the entire content on the page. JavaScript has been used extensively to produce relatively complexed responses against user actions such as rendering new content or modifying the present content on the page on button presses without making a page refresh. All this happens as JavaScript keeps track of the events (whichever it had been asked to do) and runs the code for that event and produces the results which is then reflected in the DOM(document-object model) of the page and hence is made visible for the client.

“Quick-test” is based on the ECMAScript-5(ES5) standards and definitions.

1. **jQuery**: jQuery is a popular JavaScript library that eases-out the method of selecting elements from the DOM of the web-page and makes changes various forms of changes to its content, attribute values and properties associated with it. jQuery also comes with several predefined methods that allows to us improve the UI/UX of the page through styling and animations.

“Quick-test” uses the minified script of the latest version of jQuery i.e., jquery-3.2.0 to rejoice all that jQuery offers and yet check the size of the app.

1. **jQuery UI**: The project uses jQuery-ui-1.2.1. jQuery UI is a sub-ordinate of the jQuery library i.e., focussed upon the methods that aid in improving the UI/UX.

In “Quick-test”, jQuery UI has been used to produce the slide effects in the new test form as well as in the test-page generated for the student to reduce the scrollable content and yet make everything available for accessing and modifying whenever the need arises for the client.

1. **Asynchronous JavaScript and XML(AJAX)**: AJAX is a client-side script that communicates to and from a server/database without the need for a post-back or a complete page refresh.

“Quick-test” uses AJAX calls extensively to make the application much more efficient and enhance the UX. All form submissions of the application are executed via AJAX calls whereby, an asynchronous post request is made to the server and the entire data provided by the user is passed along to it without halting the page or any of the response that is supposed to occur on other events made by the client in the meantime. Also, several other button actions have been achieved via AJAX requests. On the confirmation from the teacher that a particular test is to be deleted from his/her dashboard, it is through AJAX that the information regarding that test is sent to the server and necessary codes are executed in order to delete the test data from the database as well as remove it from the view forever. Another crucial use of the AJAX calls and requests was in the initialization of the test. As soon as the teacher attempts to start a test, an AJAX request is made to the server for sending the time allotted to the test. This time is then displayed in a timer on the start-test dialog box that pops up on the screen. On receiving a confirmation from the teacher for the test to be initialized by running the timer, teacher’s control over the page is curtailed to a large extent as the test is in progress. As the timer completes its count, the test is considered to be complete and all controls are returned back to the teacher and performa for the test just completed is distributed to each student as well as the teacher. While the test is in progress, if a teacher logs-out and then logs-in back within the duration, he/she yet would get the timer running on the dashboard, with the time elapsed being counted even though he/she logged-out.

**BACK-END DEVELOPMENT TOOLS**

**Back End Technology Stack:** The backend technology stack primarily uses the following mentioned technologies for the website.

* Java: As primary backend language.
* Java EE: The Web application API and Libraries being part of it EE.
* Java Servlets: Performing the task of route and controller in our complete architecture of the project.
* Java Server Pages (JSP): Use as the templating language that allows for dynamic generation of pages based on request.
* JSTL (JavaServer Pages Standard Tag Library): Used for shortening the java code inside JSP.
* JDBC: For establishing connection to the MySQL database and executing queries and extracting data.
* MySQL: As Database for the website.
* Apache Tomcat Server

**Tools used for backend purposes:** Netbeans with support for Java EE and SE libraries.

\*Note that with Java into the picture the development cycle changes to

Code -> Compile -> Deploy

As contrary to modern web application that are used by startups this is not favorable since they need rapid development along with deployment of service. Hence the cycle mostly being as the following:

Run testing server ->Code ->Change as necessary

**Description of use of each technology aforementioned:**

**Java:** Chosen for the project based on the syllabus and also as an opportunity to learn about the Java EE development. Also Java is known for its strong OOPS implementation and also scalability factor in expanding a project adding to the reason of choosing this language.

**Java EE:** Words from the official docs

‘*The Java EE platform is built on top of the Java SE platform. The Java EE platform provides an API and runtime environment for developing and running large-scale, multi-tiered, scalable, reliable, and secure network applications*.’

The Servlet API is a part of Java EE which is in the core of Apache Tomcat’s runtime environment.

**Java Servlets:** From the official docs

‘A servlet is a Java programming language class that is used to extend the capabilities of servers that host applications accessed by means of a request-response programming model. Although servlets can respond to any type of request, they are commonly used to extend the applications hosted by web servers. For such applications, Java Servlet technology defines HTTP-specific servlet classes.

The javax.servlet and javax.servlet.http packages provide interfaces and classes for writing servlets. All servlets must implement the Servlet interface, which defines life-cycle methods. When implementing a generic service, you can use or extend the GenericServlet class provided with the Java Servlet API. The HttpServlet class provides methods, such as doGet and doPost, for handling HTTP-specific services.’

Servlets are in the core of our projects architecture since they act as the primary controller for all routes managing and loading/unloading of Sessions. When the web application is accessed by a client then the Servlets are responsible for responding with the necessary data and they are the ones that call for the database activities. The following are the servlets along with functionalities in our Project

* loginController: Post method responsible for calling authentication mechanism and redirecting to user dashboard
* registerController: Post method use for registering users and Get method for updating and redirecting to user dashboard.
* testInteract: Post method for adding new content by user. Get method for getting the current running time of the test . Delete method for deleting any test
* startTest: Post method for starting the test
* logout: logs out the user by clearing his/her session.
* performa: generates the performa of the student in post and serves performa for student. Get routes serves the performa to the student.

**Java Server Pages (JSP):** JavaServer Pages (JSP) technology allows you to easily create web content that has both static and dynamic components. JSP technology makes available all the dynamic capabilities of Java Servlet technology but provides a more natural approach to creating static content.

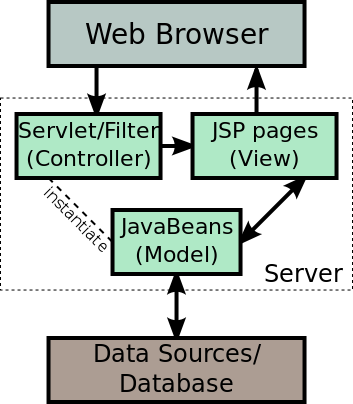


Fig: Standard architecture of a JSP based Web Server

The main features of JSP technology are as follows:

* A language for developing JSP pages, which are text-based documents that describe how to process a request and construct a response
* An expression language for accessing server-side objects
* Mechanisms for defining extensions to the JSP language

JSP is used as the means of defining the web pages that are served in the web application in our project. The JSP in reality is just a Java Servlet only. When the JSP is compiled the resulting form is a Java Servlet with a humongous amount of document print statement which contains the HTML content.

Static resources are also served with the HTML content namely JavaScript files and CSS files for the web page.

The following are the standard JSP that are getting served.

* Login page that also have the register option and HTML inside it.
* Teacher Dashboard for controlling exams
* Student Dashboard for giving exams and reviewing old records.
* Test Page for giving the exam.
* Performa page for students.

**JSTL:**  The standard Java Server pages Standard Tag Library are basically html esque tags that can be created and customized and can be attached to any certain Java class program that will create data for the enclosing tags that call for the controller. The use of JSTL is kept to the minimum in this project since most of the logic is achieved using JSP and Java only.

**JDBC (JAVA DATABASE CONNECTIVITY):** From official docs:

**‘**Java Database Connectivity (JDBC) is an application programming interface (API) for the programming language Java, which defines how a client may access a database. It is part of the Java Standard Edition platform, from Oracle Corporation. It provides methods to query and update data in a database, and is oriented towards relational databases. A JDBC-to-ODBC bridge enables connections to any ODBC-accessible data source in the Java virtual machine (JVM) host environment.’

The JDBC allows us to interact with MySQL server that we are using for our web application. The connection is done with thread safe method i.e the connection is done through a JDBC Connection Pool.

Overview of Pooling in JDBC:

Establishing JDBC connections is resource-expensive, especially when the JDBC API is used in a middle-tier server environment, such as when DataDirect Connect for JDBC or DataDirectSequeLink for JDBC is running on a Java-enabled web server. In this type of environment, performance can be improved significantly when connection pooling is used. Connection pooling means that connections are reused rather than created each time a connection is requested. To facilitate connection reuse, a memory cache of database connections, called a connection pool, is maintained by a connection pooling module as a layer on top of any standard JDBC driver product.

Connection pooling is performed in the background and does not affect how an application is coded; however, the application must use a DataSource object (an object implementing the DataSource interface) to obtain a connection instead of using the DriverManager class. A class implementing the DataSource interface may or may not provide connection pooling. A DataSource object registers with a JNDI naming service. Once a DataSource object is registered, the application retrieves it from the JNDI naming service in the standard way.

More on pooling: https://www.progress.com/tutorials/jdbc/jdbc-jdbc-connection-pooling

**MySQL:** MySQL is a freely available open source Relational Database Management System (RDBMS) that uses Structured Query Language (SQL). SQL is the most popular language for adding, accessing and managing content in a database. It is most noted for its quick processing, proven reliability, ease and flexibility of use.

An RDBMS was chose for this project due to its factor of scalability along with its ease of query language and data management which often goes out of hands when data is to heterogeneous in Non-relational.

The MySQL needs to be hosted on a separate server from the main Application Server.

For our project the MySQL operations are encapsulated under a class called dbManager. This encapsulation allows us to achieve a JavaBean kind of property where we deal with data with a single point of contact instead of writing queries here and there.

**APACHE TOMCAT:** Tomcat acts as the server for our Java Web Application. Any other server could also be chosen like Glassfish. The inner working of tomcat is as follows

Tomcat mainly has a Classloader Hierarchy and a Thread Pool. When a web application is deployed into tomcat, tomcat scans the Webapp, reads its deployment descriptor (web.xml or the equivalent) and decides that Servlets (and JSPs) need to be deployed and be made available. Servlets and JSPs (which are eventually servlets) are class loaded based on a class loader hierarchy, also any listeners and filters relevant are loaded and made available. When a request comes it is delegated to the appropriate Servlet class (which is run in a separate thread), the servlet does what is necessary to process the request, responds, then the thread is free and will be used to serve the next request).

**DEMONSTRATION OF THE PROJECT**

The whole project has been segregated into two sections- portal for teacher and portal for student. On being connected to the same Local Area Network, one can access the website. The landing page of the website is the login/register page where a student or a teacher can either register if he/she is new or one can login into their dashboard if he/she had already registered. If the user is new, the following page opens and one needs to fulfil his/her details which gets directly stored into the database.

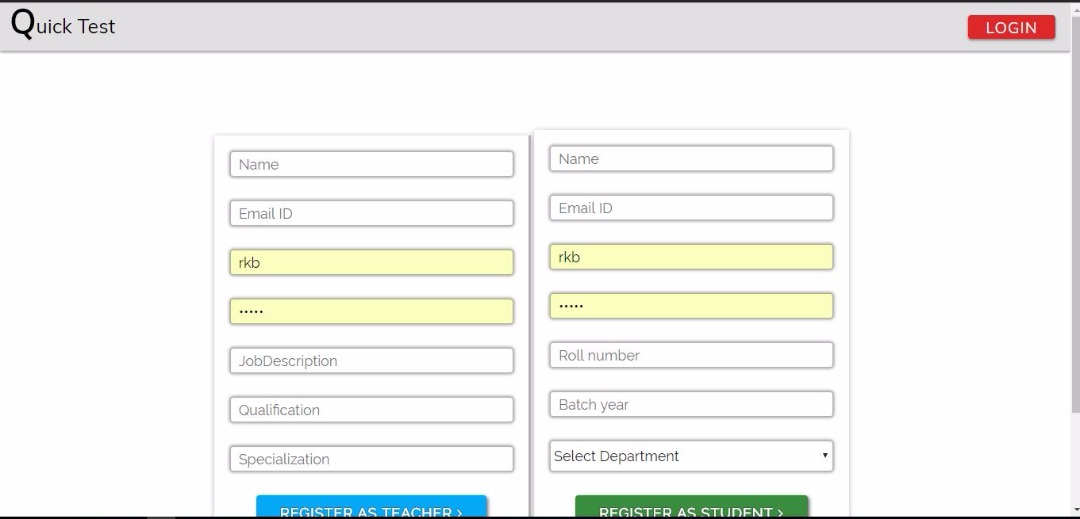


Fig 2: The Register page

If the student/teacher had already registered before, he/she just needs to login by entering the user ID and the authenticating password. If the user enters a wrong password or user ID, he/she is redirected to the same page for entering the correct user ID along with the appropriate password or is pinged to register if the user is not registered.

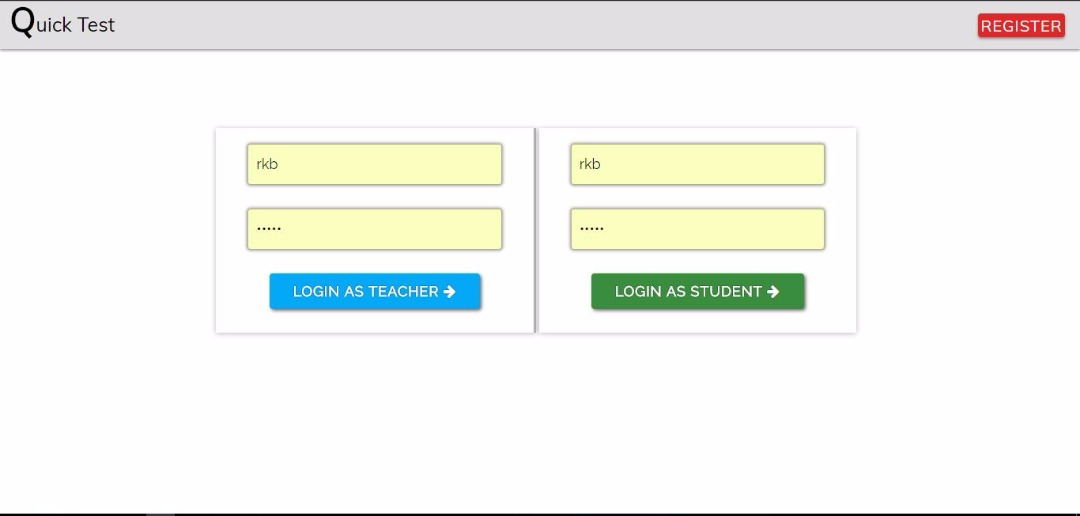


Fig 3: The Login Page

After being authenticated, the user is directed to their individual dashboards. The description about the working of the different portals of teacher and student is given below.

**Portal for teacher:**

The dashboard page of the teacher is as shown below. All the details of the teacher- name, designation, College, etc, are shown according to the form filled up by him/her during registration. The teacher also has the ability to update his/her profile as and when he/she feels necessary. During his/ her updation, the new data gets stored in place of the old ones and are visible to the teacher.

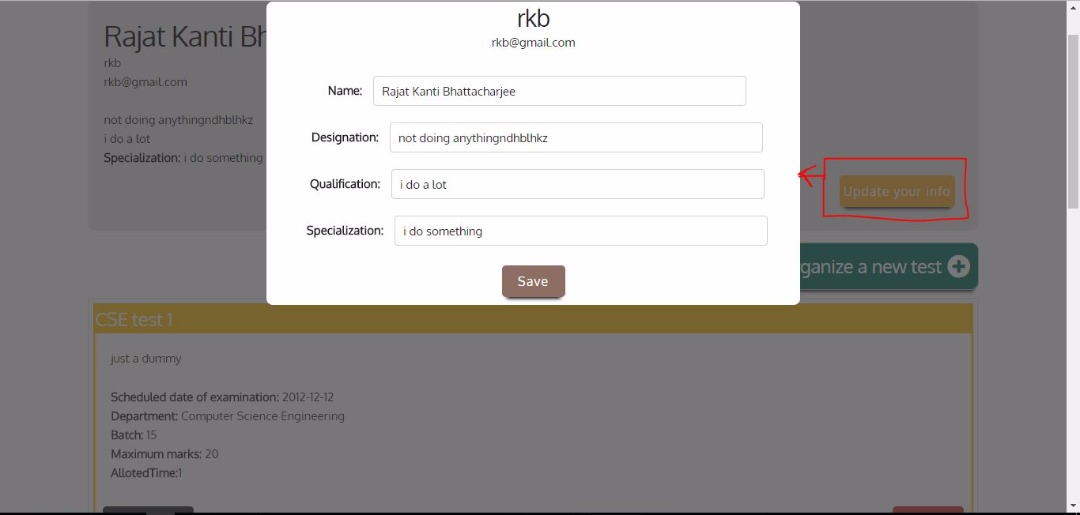
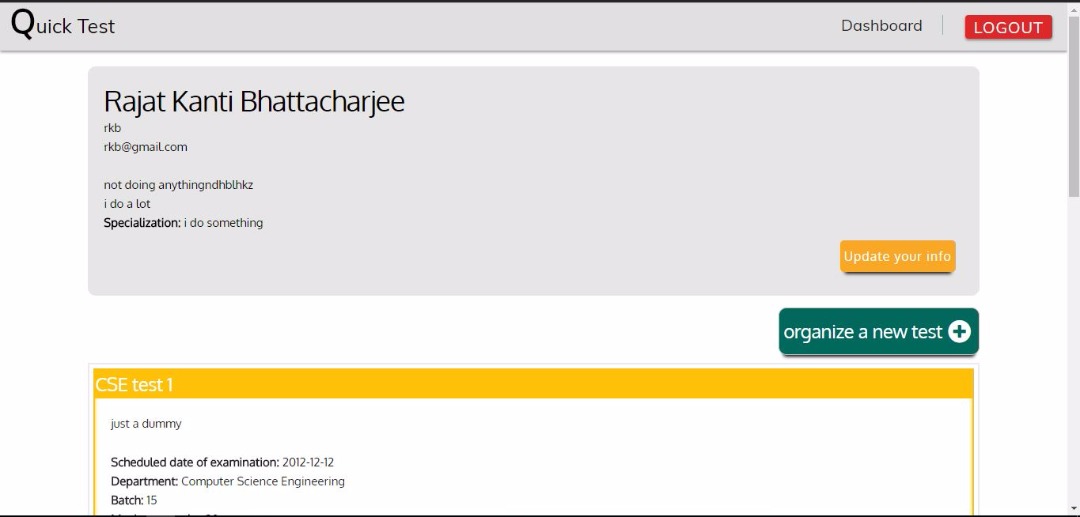


Fig 4: The Teacher dashboard Fig 5: The update Page for teacher

For conducting exams, the teacher can create new tests. For this, he/she needs to click on the “Organise a new test” button. On clicking, a new page comes up where the teacher needs to fill up the details of the test like the title of the test, description, full marks, pass marks, date of examination and time to be allotted for the test. The teacher must strictly follow the format of inputting the details; else an error message pops up for the teacher to rectify the allotted format. Then he gets the facility to input the questions for the conduction of the test. Among the questions, if the teacher wishes to search or go to some question, he/she has the search bar which shall direct him/her to the question he wants to check which nullifies his/her difficulty to scroll through the questions.

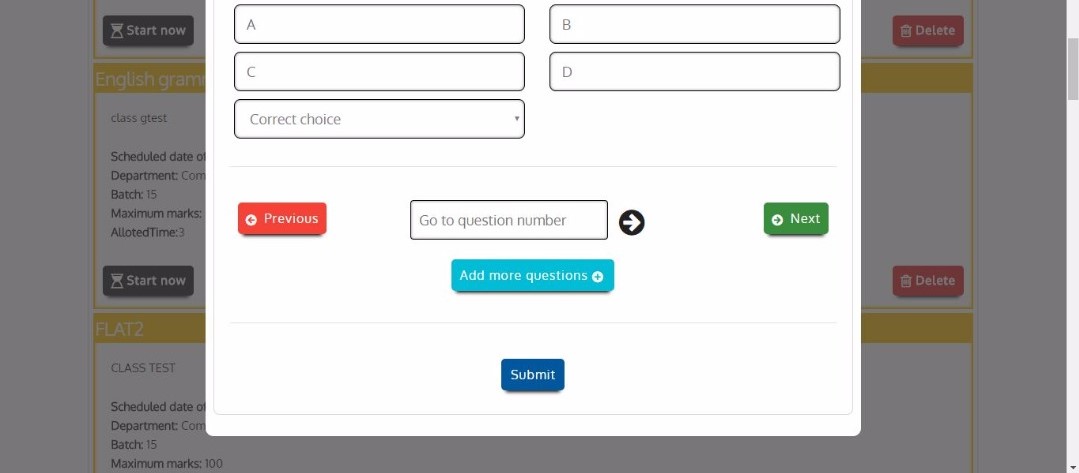
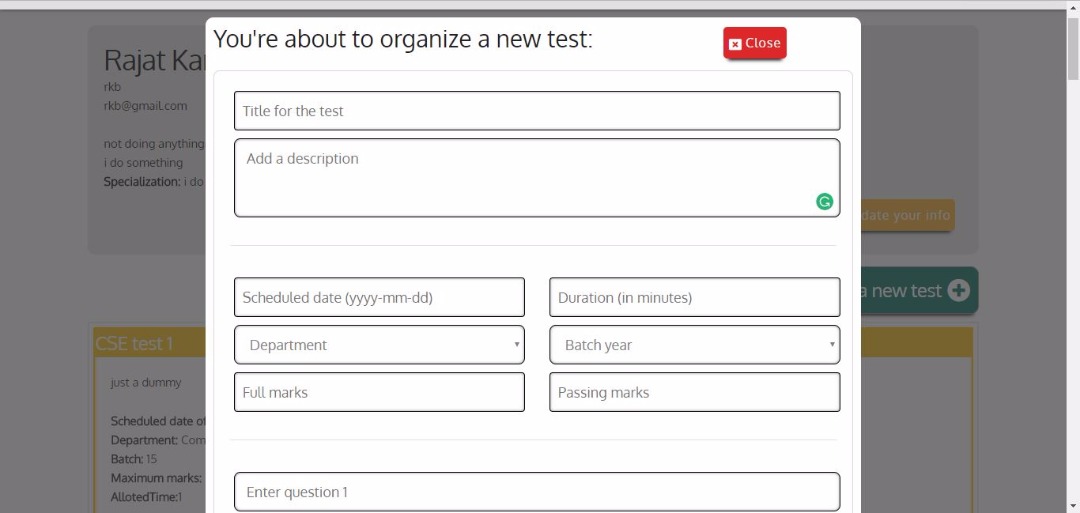


Fig 6: For the teacher to organise a test

After inputting all the test questions, he/she shall submit the form by clicking on the “submit” button. Then the teacher is redirected to the dashboard where he/she can see the test created. As soon as the test is created, it is displayed on the students’ dashboards whose department and batch match with the details of the test created by the teacher. However, if he/she had already created some tests before, those shall be displayed on the dashboard for him/her to scrutinize and conduct the test at some later time. Also, the teacher can delete a test from the dashboard by clicking on the “delete” button. A confirmation page shall appear before deleting the test. On a test being deleted, the test shall not be shown on the dashboard.

For conduction of the test the teacher, in the process of being authenticated and logged in, can conduct the test by clicking on the “Start Test” button. A page appears over the dashboard informing him/her about the details of the test to be started. After confirming the test, he/she can start the timer of the test with the “Start” button. Else if he/she wishes to close it can cancel the test, after which the dashboard shall appear again.

Fig 7: Start of timer for the conduction of the test

On completion of the test, i.e., when the timer closes the teacher shall receive the responses of all the students as “Performa Page”.

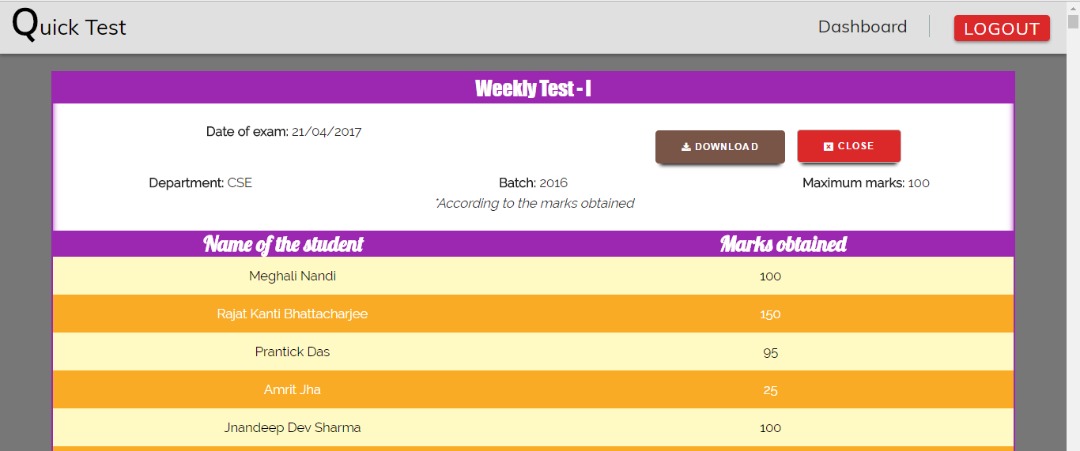


Fig 8: Performa Page of all the students

On completion of creation or conduction of tests, the teacher can logout from his/her profile on clicking on the “Logout” button, after which he/she shall again be redirected to the landing page of the website.

That is how a teacher can interact with the students and conduct the tests through the website.

**Portal for students:**

The dashboard page for the students is as shown below. The page consists of the “Dashboard” link on the navigation bar, which on being hovered, opens up the details of the student like the Name, Semester, Batch and Branch of study.

The page is sub divided into two sections- the “Upcoming Tests” and the “Previous Tests”. The “Upcoming Tests” section consists of all the tests which are yet to be conducted and has been created by the teacher. The “Previous Tests” section consists of all the tests which the student had already appeared. On clicking on any test from the “Previous Tests” section, the student can see his/her performance in the test.

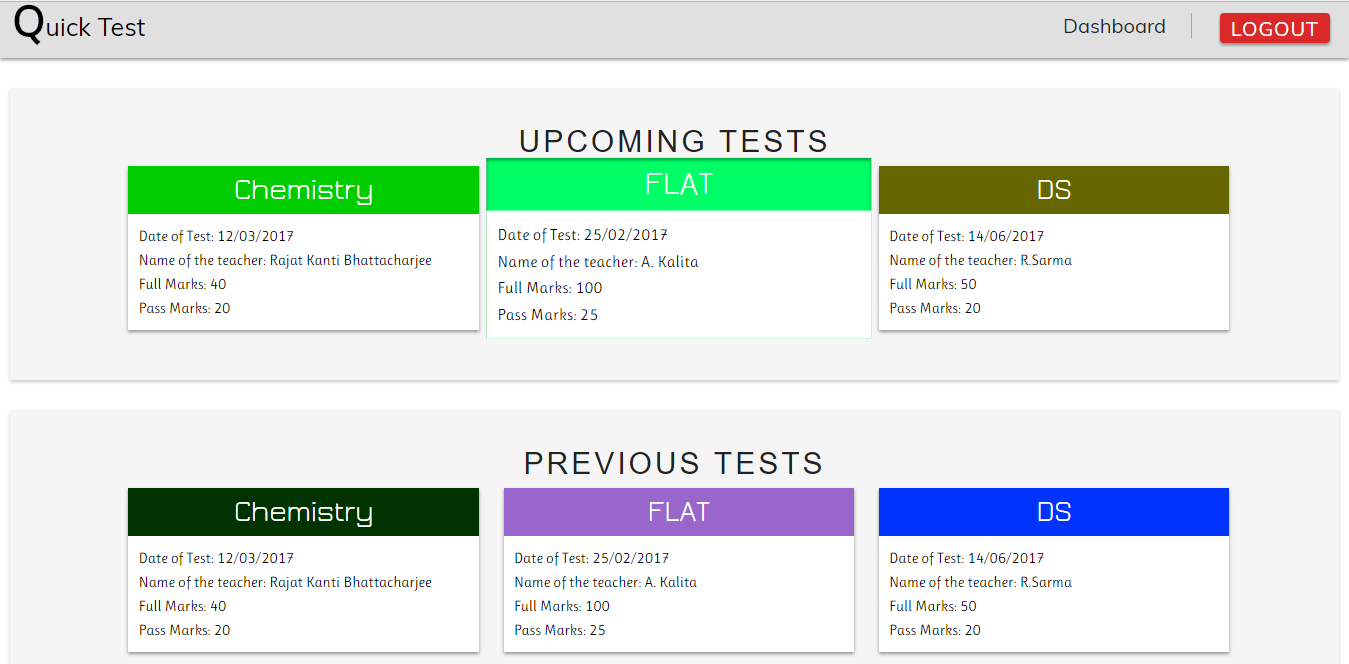


Fig 8: Dashboard of the student

On clicking on any test which has not been started by the teacher, the student shall be directed to the page where he/she is informed that the test is not yet started. On the other hand, if the respective test has been started by the teacher, the student can click on the test and appear for the exam. As soon as he/she enters the exam page, the timer loads and the student can then start to give the exam within the stipulated time. In the test page, on the right hand side, the student is allowed to confirm his identity with his/her name, branch, Roll No. and Semester being displayed along with the name of the test he is appearing. At a time, only one question is displayed for the student. For answering the next or previous question, he/she can click on the right and left arrows which shall make the respective questions appear. Moreover, a bookmark label is being provided in each question for the student to label the question if he/she wishes to revisit the question during the conduction of the test. On clicking the bookmark, the colour changes which shall act as a sign of reminder that the question had been bookmarked. Among all, if a student wishes to visit some question, he/she can type the question number in the search box.

If the student completes the exam before the end of the timer, he/she can submit the answers by clicking on the “Submit” button. Soon, he will be directed to the marksheet page where he/she can see the correct answers along with his/her response against the respective question numbers.

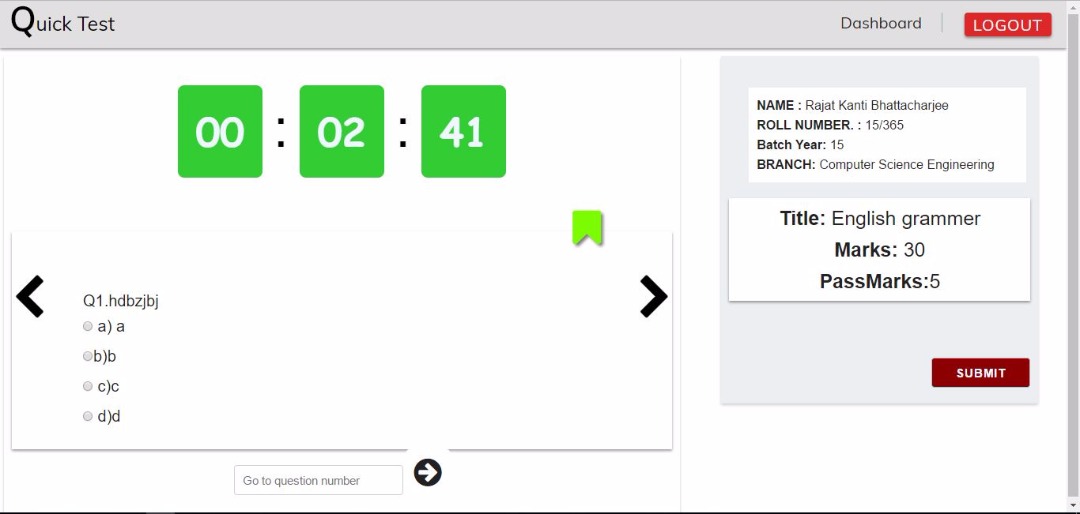


Fig 9: The page for the students to give his/her exam

In the meantime, if a student fails to complete the test or forgets to submit his/her answers, then along with the completion of the countdown of the timer, the answers shall automatically be recorded for the production of the marksheet. For the benefit of the students, the marksheets can be downloaded as PDF on clicking on the “Save as PDF” button.

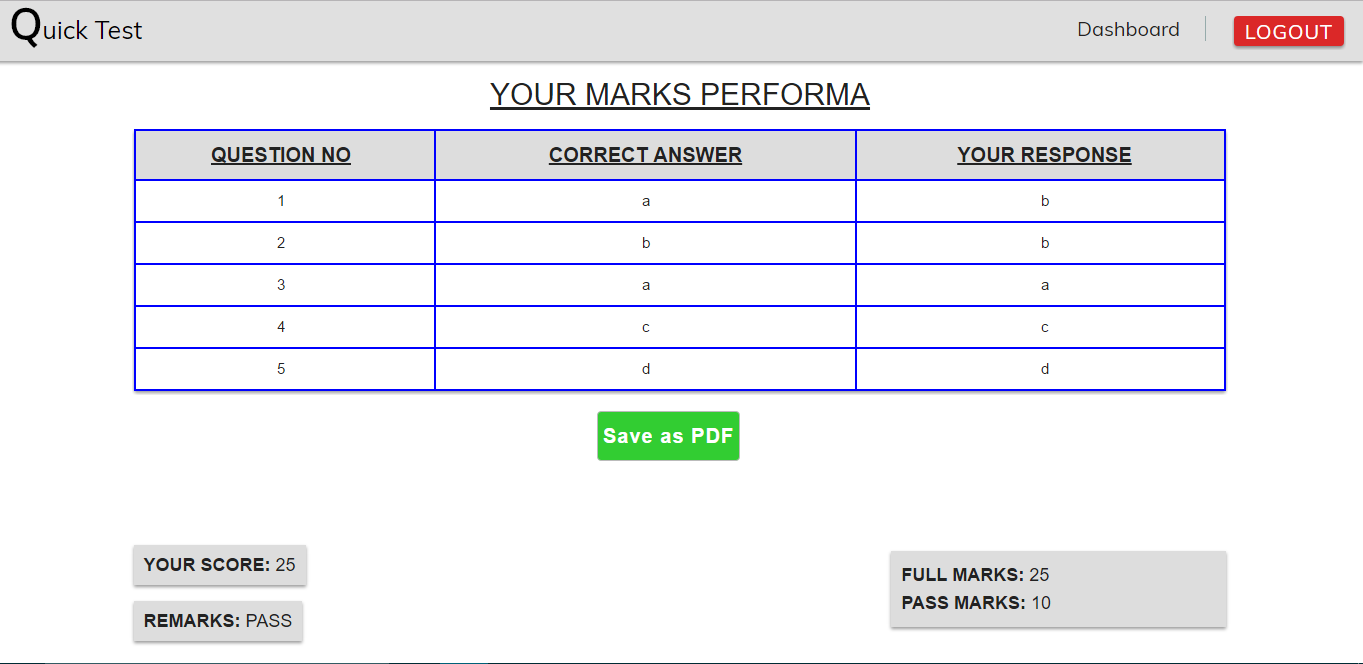


Fig 10: Marksheet of the student

This is how a student can appear for the tests and scrutinize himself/herself based on the records of the previous tests.

**CONCLUSION**

The project was completed and the application with the desired functionalities was built successfully. The application is now efficient in handling tests conducted through it. As targeted beforehand, the application allows the teachers to sign-up/log-in and organize the tests whereas the students can sign-up/log-in and appear in the tests.

We would like to conclude on the hope that the experience and knowledge gathered by working on this project would surely prove beneficial to us in future as well.