RESEARCH ARTICLE OPEN ACCESS

Automation of Examination Registration Process

Romik Amipara [1],Rutvik Kokate [2], Srividya Inampudi[3], Shweta Tripathi[4], Shagufta Rajguru[5]

[1 to 3] UG Students
[4 to 5] Assistant Professor
Computer Engineering Department, Fr. C. Rodrigues Institute of Technology, Navi Mumbai 400703, Maharashtra
email: Romik.amipara@gmail.com

--************************----------------------------------

Abstract:

With vast advancements in technology most for our day to day activities which would require manual and arduous work have been simplified. These processes have been automated with help of machines and technology. The outdated way of generating hall tickets is to make each student fill the exam form, and then the admin manually enters tons of data into the database to produce hall tickets is a tedious and time-consuming process. A method has been proposed to change this mundane process, an online portal will be created which will be accessible to students for uploading their details, and the admin can generate the hall tickets, verify student data, update time table with a single click.This system can increase efficiency by reducing the human errors and time consumed and aiding the concept of automation of manual tasks such as registration process, hall ticket generation process etc.

Keywords **—**Website Development, Django, Reactjs

--************************----------------------------------

I. INTRODUCTION

In an era of booming technologies, we still follow an antiquated work style. Web-Sites have become the most integral part of our day-to-day life; moreover, it has become a necessity. The global search engine has become an essence of a holy book that contains answers to all the life questions, so it's no wonder web development has seen such tremendous growth in recent years. Every work we know which used to be done manually is now completely done via websites.

Computers, technology and software have a significant impact on our day to day lives. Automation of systems and reducing manual work is the call of the hour. Similarly, colleges, universities need to automate many manual systems. A centralised automated exam cell system can be used to perform verification of details, hall ticket generation, time table updating, online exam form filling etc [1].

The current process of generating hall tickets by the exam cell is done manually, which is time-consuming. In the current system, the students have to fill all the forms offline and submit the hard copy to the exam cell; once the form is verified, it has to be submitted to the office. Then the staff members of the exam cell have to fill in the details of the students who have submitted the form into the system. The details of the students are stored to keep records of the department, semester, examination subjects and the backlog subjects [2].

Due to all these reasons, a better system has to be developed which will overcome all these problems quickly and efficiently; hence we have developed a website which would cater the exam cell and would reduce the amount of manual work done in the whole examination process [3].

The exam cell portal is a user-friendly website which would allow the students to fill their vital information necessary for generating hall tickets. The admin will be able to analyse the entries, verify them and finally be able to generate hall tickets. The admin also has various functionalities such as updating of time table, entry of examination scheme, verification and deletion of student records.

II. LITERATURE REVIEW

Rao, Ganesh et al., implemented a project which can replace the current exam cell activities which are done mostly on paper and manually. The current system produces a large amount of data which makes it a tedious task for analysis. They implemented a centralised system, acting as an intermediary between staff and students. This system would make the exam cell-related activities more efficient and easily accessible. Their final product aimed to replicate the offline examination cell process constituted of a computerised model. Their work had various functionalities on student and admin: login, form filling and hall ticket generation options for students. Seat number generation and generation of results and analysis functionalities for admin are available. Analysis of results can also be done through their product efficiently [4].

SejalD'mello et al.,proposed a methodology and implemented a project which could reduce the work done by the exam cell on paper and during the examination process. Their proposed system has three main modules, namely Admin, Staff and Students. The role of the Admin is to Add / Update / Delete / View staff members in a database, View Student Registration in Database and Generate Marksheet Exam Cell Automation System. The role of the Staff members is to View Students Registration status and Add Student Marks into the system. The part of the student is to Register into the System, enrolment of Examination which is online, Filling the Exam Form online then Download Hall Ticket and finally Viewing Marks. Hence, they automated the offline mode of the exam cell of their institute to increase efficiency and maintain productivity [5].

Patki, Nair et al., proposed a system to reduce the considerable amount of data created manually due to various forms such as exam form, K.T form, revaluation form and photocopy forms. The system will allow students to access the website for filling the different forms, to access the time table, exam related notices etc. The system is designed in a manner such that the records are updated efficiently. The student data can be accessed and verified by staff anytime. The project has two main modules, the Admin Panel and the Student Panel. The Admin Panel has various functionalities for Admin such as Admin Login, Manage Department, Manage Semester, Manage Subject, Manage Notice, Manage, Manage Exam Timetable, Manage Exam Form (Regular K.T., Revaluation, Photocopy. The Student Panel has various functionalities for Students such as Student Registration, Student Login, Filling of exam forms, View Exam Timetable, View Hall, View Marks and View Profile [6].

Dhembareet al., proposed a system to automate exam-cell activities. This system would aid in making the department activities more efficient and accurate. The whole system would be controlled by the admin as proposed in the paper. The model proposed had various functionalities at the admin level and student level. Admin has access to edit the timetable, verify student data and generate hall tickets. The students can fill the examination form, KT form from the portal and can download the hall ticket. The result analysis could be done on the admin side after the examinations. Apart from these functionalities, they increased the security of the system. This security aspect was implemented so that the personal data of the student is secure in their approach. This proposed application would help in making the examination cell process more organised [7].

III. ARCHITECTURE AND DESIGN

The design of the project is simple and user friendly, divided into two main divisions, i.e. the admin end, and the student end as shown in Fig 1.

Students will be able to fill their vital information required for the hall ticket. The important thing about the information is whether the student is appearing for the KT exam or not. For this, the student will be able to select whether he belongs to the regular or KT category. If he/she is a KT student then besides the regular details he will be able to select his backlog subjects. Once the admin releases the print link, the students will be able to download their respective hall tickets.

The most important part of the project is the admin side, the main aim is to reduce the admins' tedious and manual work by automating the system.The admin will be able to modify or upload the timetable of the examinations, moreover, he will be able to validate the forms of students and once everything is filled and validated, he will release the link for the students to download their hall tickets.

This web application is implemented using the combination of ReactJS at the front-end for fetching and rendering data from the back-end built with Django and Django REST Framework.ReactJS is a JavaScript library for building user interfaces. Redux is a predictable state container for JavaScript apps [8].

Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design. Django is used for building the database models, defining the login and registration parameters and maintaining security of the entire web application [9]. Django REST Framework (DRF) is a powerful and flexible toolkit for building Web APIs. The main objective of DRF in this application is to build an API from where the front-end can fetch data. DRF serialises, i.e. converts the data from Django databases to native python data types which can be rendered into JSON, XML format [10].

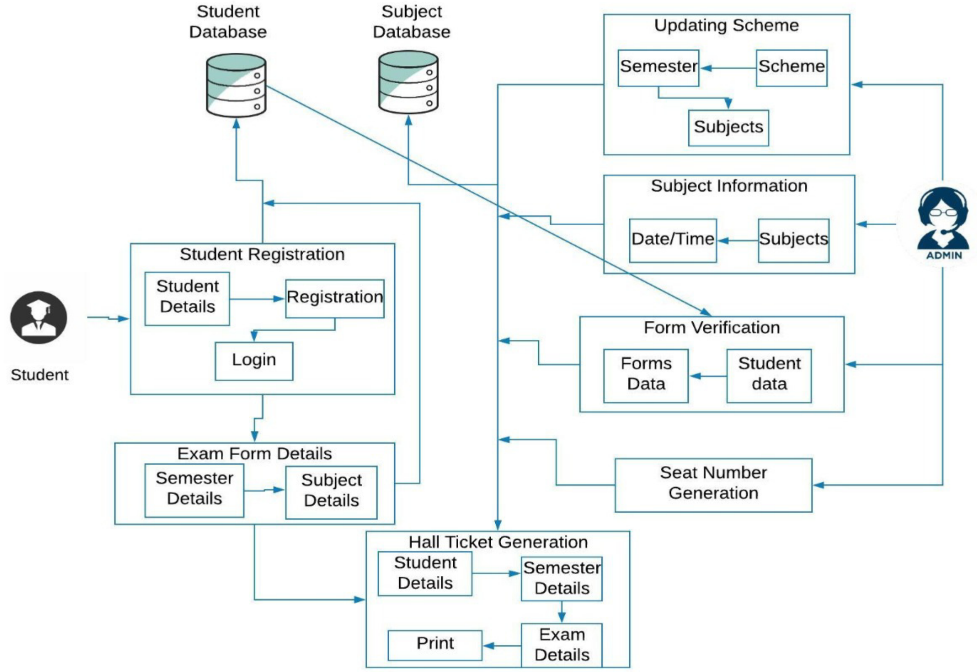


Fig 1. Architecture of the proposed system

IV. METHODOLOGY

A. Login and Registration

The login and registration portal provides the user (Admin or Student) to maintain a separate account of their own which ensures the security of data such as personal information of the student.This process is implemented using Django models. In Django, a model is the single, definitive source of information. It contains the essential fields and behaviours of data you are storing [11]. After registration and login, the student dashboard is visible as shown in Fig 2.

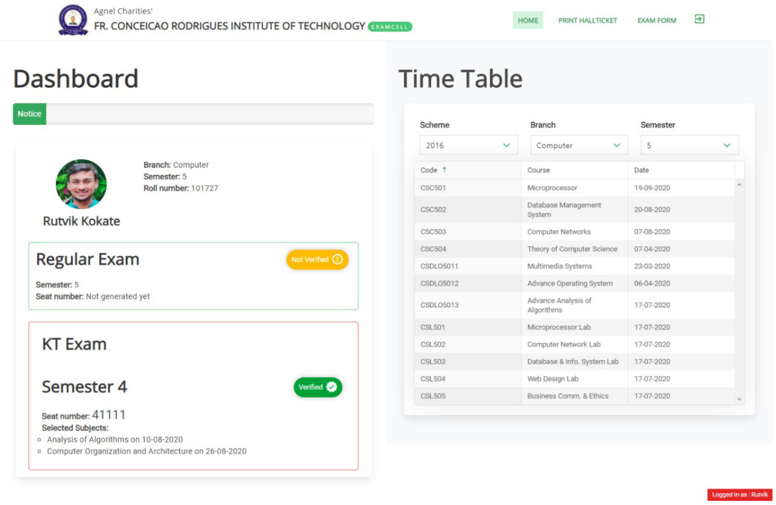
A view is a class which takes a Web request and returns a Web response.Django takes security to the next level. It has built-in libraries which protect against several web attacks like Cross-site request forgery (CSRF). Django uses the PBKDF2 algorithm with SHA256 hashing, salting to store the passwords.All the concepts, as mentioned earlier, are used in the following manner:

1) Back-end:

- Models are created in Django. These models contain fields such as username, email and password.
- Here DRF comes into play. From the models, the data is serialised. In this process, different validation functions ensure - passwords entered are correct, if the user exists or not, if the email address has already been used for registration etc.
- After serialising the data, it is passed on to views. Views take care of handling the HTTP requests such as creating, updating, deleting, reading requests. It processes the input request (i.e. username and password the user enters) and then sends a response to calling the HTTP request.
- If all the data is validated, it sends a response, i.e. a token. This token allows the user to login. If there is a token mismatch, the user will not get logged in. Similarly, after logging out, the token gets destroyed automatically. This step ensures session privacy and authentication.

2) Front-end:

- React fetches the token generated and stores the state of that token with the help of Redux state management. This token is then used in future for checking if the same user is logged in.
- In case of errors, such as invalid data entries, data not found respective errors are too stored in states. These errors are then displayed to the user.



Dashboard

Rutvik Kokate

Branch: Computer
Semester: 5
Roll number: 101727

Regular Exam

Semester: 5
Seat number: Not generated yet

KT Exam

Semester 4

Seat number: 41111
Selected Subjects:

- Analysis of Algorithms on 10-08-2020
- Computer Organization and Architecture on 26-08-2020

Time Table

Scheme: 2016
Branch: Computer
Semester: 5

Code ↑	Course	Date
CSC501	Microprocessor	19-09-2020
CSC502	Database Management System	20-08-2020
CSC503	Computer Networks	07-08-2020
CSC504	Theory of Computer Science	07-04-2020
CSDLO5011	Multimedia Systems	23-03-2020
CSDLO5012	Advance Operating System	06-04-2020
CSDLO5013	Advance Analysis of Algorithms	17-07-2020
CSL501	Microprocessor Lab	17-07-2020
CSL502	Computer Network Lab	17-07-2020
CSL503	Database & Info. System Lab	17-07-2020
CSL504	Web Design Lab	17-07-2020
CSL505	Business Comm. & Ethics	17-07-2020

Logged in as : Rutvik

Fig 2. Student Dashboard after Registration and login

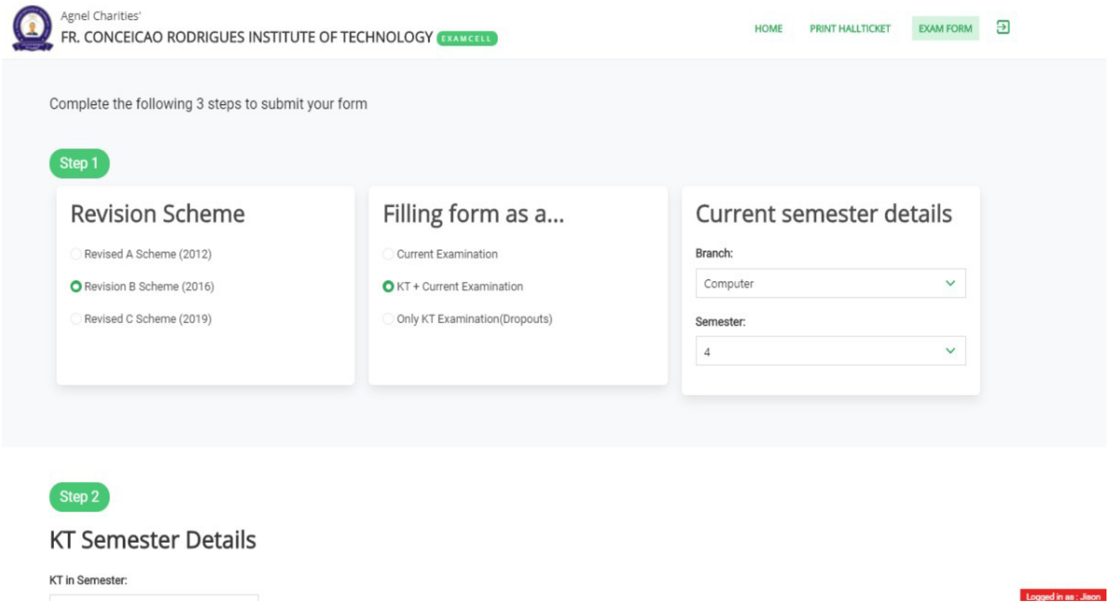
B. Time table entry

As the admin selects the branch and semester for which the timetable is to be updated, simultaneously requests are sent to the API with the selected data. The response is received from the API and rendered into tables. While editing dates, an onItemChange() function continuously updates the data as the input field gets changed. Additionally, the admin can enter department electives in the same manner.

Here the "PATCH" request method is used to update a single instance while the "POST" method is used to create a new data, i.e. add a new elective.

C. Student form

Students can fill their exam-related and personal details through this section. Students can fill the form as a regular student as well as a KT student as shown in Fig 3. Each student can select their choice of department elective. The same process is applied here for sending the student data to the back-end and validation. If a student with the same roll number is already registered, an error will be raised.



Complete the following 3 steps to submit your form

Revision Scheme

○ Revised A Scheme (2012)
● Revision B Scheme (2016)
○ Revised C Scheme (2019)

Filling form as a...

○ Current Examination
● KT + Current Examination
○ Only KT Examination(Dropouts)

Current semester details

Branch: Computer

Semester: 4

KT Semester Details

KT in Semester:

Logged in as : Jison

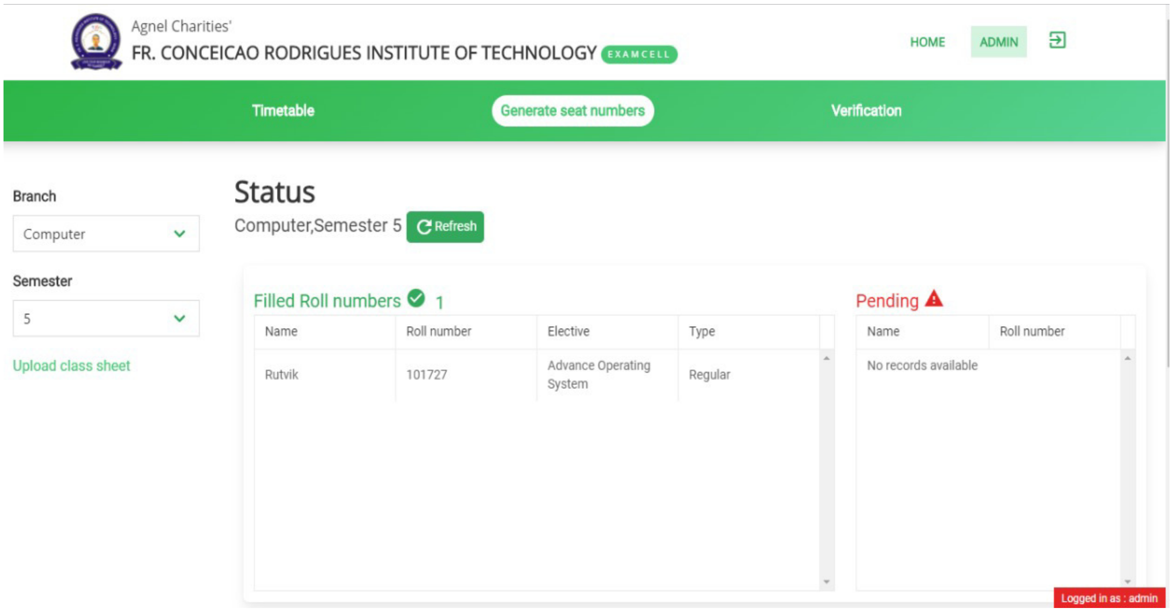
Fig 3. Online Form filling for the student

D. Checking filled/pending students

To check how many students have filled the form, a list of all the students of a particular semester and branch will be displayed. This list of students is checked against the attendance sheet of the class, which is in a .csv format. handleData() function reads the CSV file contents and then filters the students according to their filing status as shown in Fig 4.

E. Seat number generation

After all the students have filled the exam form, a list will be generated as explained in the above part.This list of students will be assigned a seat number in a pre-defined manner. Students first will be sorted as Regular and then KT students. Then they will be sorted according to their Elective chosen and then according to their roll number. The newly assigned seat number is updated to the respective student instance.



Branch: Computer

Semester: 5

Status

Computer,Semester 5

Filled Roll numbers ✓ 1

Name	Roll number	Elective	Type
Rutvik	101727	Advance Operating System	Regular

Pending ⚠

Name	Roll number
No records available	

Fig 4. Admin Checking Status of the forms and Generation of Hall Tickets

F. Generation of Hall Ticket

A component called as PDFexport converts a specified element to the Pdf format which is directly downloadable. All the details of students, i.e. exam details, seat no is displayed in a table. This table is wrapped in the PDFexport component. Everything rendered inside the PDFexport component is converted into Pdf format. All the student data is fetched from the server in the same manner as explained earlier.

G. Admin Portal Security

Keeping in mind the sensitive and vital data of students, securing the admin portal is an integral part. The admin login portal requires Two Factor authentication for a successful login. A npm package - Speakeasy; is used to achieve TOTP (Time-based One-Time Password) authentication. This package uses Google Authenticator app(iOS/Android) which creates an account and displays a 6-digit code, expiring after every 30 seconds. New accounts can be generated from within the admin portal, thus making the previous account and its codes obsolete.

V. CONCLUSION

By eliminating the tedious and mundane tasks of manual form-filling, an efficient solution is deployed, ensuring optimal productivity and efficient use of time. Various modules such as admin validation and verification, form filling, hall ticket generation, verification, seat number generation were implemented and were subjected to in-depth analysis and testing. During the testing phase, various edge cases were tested, and subsequent changes were made. Implementing this system will primarily benefit the institute and precipitate the automation of several different activities, which will render the organization task-friendly and enhance student experience.

ACKNOWLEDGMENT

Authors would like to express their sincere thanks to Mr. Mathewlal Thomas Associate

Professor, and Praison Sir, Fr C Rodrigues Institute of Technology, Vashi for their constant support, guidance and encouragement. We also thank Dr. Lata Ragha, HOD Computer Science Department and Dr. S.M. Khot Principal Fcrit for providing the facilities.

REFERENCES

[1] Vamsi Krishna Yepuri, Gopi Chand Pamu, Naveen Kodali, “Examination Management Automation System”, International Research Journal of Engineering and Technology (IRJET), April 2018

[2] BondreRutujaAvinash, Durgi Varsha Vijaykumar, Mohit Pradnesh Rajeev and Paekar Vishal VT, “Automated Examination Support System”, International Journal of Current Engineering and Technology, Vol.5, No.2 (April-2015)

[3] F. Ansari, U. M. Ahmad, K. Saqlain and S. M. Sohail, "E-exam cell," 2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), Coimbatore, 2017, pp. 1-4, doi: 10.1109/ICIIECS.2017.8275941.

[4] Aditya Rao, Abhishek Ganesh and Stuti Ahuja, “Online Exam Cell and Result Analysis Automation”, April 2015 (IJEAS).

[5] SejalDmello, Ashraf Pinjari, Harishchandra Damre, Aishwarya Sankhe, Vrushali Ambre, ” Exam Cell Automation System”, IOSR Journal of Engineering (IOSRJEN) march 2018.

[6] RakshitPatki, Saranya Nair, Indira joshi, “Exam Cell Automation”, International Research Journal of Engineering and Technology (IRJET), Volume: 06 Issue: 03 | Mar 2019

[7] Monika Dhembare, Archana Gorad, Prof. S. H. Darekar, “Automated Exam Cell with Result Analysis”, International Journal of Engineering Science and Computing (IJESC), Volume 7 Issue No.4

[8] Arghandabi, Hidayat. “Redux with React Is Simple.” Medium. JavaScript In Plain English, May 1, 2020.https://medium.com/javascript-in-plain-english/redux-with-react-is-simple-3e3480a83432.

[9] Django. Accessed September 4, 2020. https://www.djangoproject.com/.

[10] Christie, Tom. “Django REST Framework.” Home - Django REST framework. Accessed September5, 2020.https://www.django-rest-framework.org/.

[11] Arora, Naveen. “Django Basic App Model - Makemigrations and Migrate.” GeeksforGeeks, February11, 2020.https://www.geeksforgeeks.org/django-basic-app-model-makemigrations-and-migrate/.