

THE MAHARAJA SAYAJIRAO UNIVERSITY OF  
BARODA  
FACULTY OF SCIENCE



A PROJECT REPORT  
on  
**Admission Portal**  
for  
**FYBSc, Faculty of Science, MS University, Baroda**

*Submitted by*

**Project No. 22**

*In fulfillment for the course of Semester-VI*

**(BCA1601)**

*of*

**BACHELOR OF COMPUTER APPLICATIONS**

*in*

**The Department of Computer Applications**

May 18, 2020.

**Guided by:**

**Mrs. HETA  
PAREKH**

**Team members – PRN:**

**Parth Vaswani – 2018033800104434  
Sharanam Chotai – 2018033800104732  
Milit Bhatt – 2018033800104747**

# Project Completion Certificate from The Company



## Department of Computer Applications

Faculty of Science

The Maharaja Sayajirao University of Baroda

Vadodara - 390 002 Gujarat, India

Ph: (+91-0265) 2781328

26<sup>th</sup> May 2021

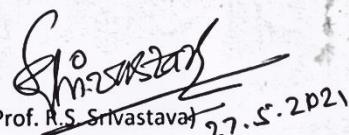
### CERTIFICATE OF COMPLETION

This is to certify that the below mentioned final year students of Bachelor of Computer Applications at Department of Computer Applications, Faculty of Science, The Maharaja Sayajirao University of Baroda have successfully completed the pilot project initiated for "**Online admission process of First year BSc.**"

- |                    |   |                  |
|--------------------|---|------------------|
| 1. Parth Vaswani   | - | 2018033800104434 |
| 2. Sharanam Chotai | - | 2018033800104732 |
| 3. Milit Bhatt     | - | 2018033800104747 |

They have accomplished all the project related work satisfactorily from February 1, 2021 to May 15, 2021.

During the project tenure they were sincere and highly dedicated to the work. Their performance during the project was commendable. And I wish them all the best for their future endeavours.

  
(Prof. R.S. Srivastava)

27.5.2021

Head



# College Certificates

The Maharaja Sayajirao University of Baroda  
Faculty of Science

Department of Computer Applications



BCA Programme

## Certificate

*This is to certify that Mr./Ms. Sharanam Maheshbhai Chotai a student of BCA Programme (Seat No. \_\_\_\_\_) of The Maharaja Sayajirao University of Baroda, Vadodara, has successfully completed the project / Internship prescribed for the sixth semester of the programme during the period from 01<sup>st</sup> January 2021 to 30<sup>th</sup> April 2021 towards the partial fulfillment for the degree of Bachelor of Computer Applications.*

*Date:*

*Name & Signature of (Prof. H.R. Kataria) (Prof. R.S. Srivastava)  
the Guide Director Head*

*\* → The format of the certificate was given from the college, but due to the corona pandemic, these certificates were not signed and/or stamped. Therefore, even without signatures, these certificates will be considered completely valid.*

The Maharaja Sayajirao University of Baroda  
Faculty of Science

Department of Computer Applications



BCA Programme

## Certificate

This is to certify that Mr./Ms. Parth Vaswani  
a student of BCA Programme (Seat No. \_\_\_\_\_) of The  
Maharaja Sayajirao University of Baroda, Vadodara, has  
successfully completed the project / Internship  
prescribed for the sixth semester of the programme  
during the period from 01<sup>st</sup> January 2021 to 30<sup>th</sup> April  
2021 towards the partial fulfillment for the degree of  
Bachelor of Computer Applications.

Date:

Name & Signature of (Prof. H.R. Kataria) (Prof. R.S. Srivastava)  
the Guide Director Head

The Maharaja Sayajirao University of Baroda  
Faculty of Science

Department of Computer Applications



BCA Programme

## Certificate

This is to certify that Mr./Ms. Milit Bhatt  
a student of BCA Programme (Seat No. \_\_\_\_\_) of The  
Maharaja Sayajirao University of Baroda, Vadodara, has  
successfully completed the project / Internship  
prescribed for the sixth semester of the programme  
during the period from 01<sup>st</sup> January 2021 to 30<sup>th</sup> April  
2021 towards the partial fulfillment for the degree of  
Bachelor of Computer Applications.

Date:

Name & Signature of (Prof. H.R. Kataria) (Prof. R.S. Srivastava)  
the Guide Director Head

## **Abstract**

This document is intended to report final year project of pupils of Bachelor of Computer Applications – i.e., ours. This report discusses our experience as a software developer in a project from January 2021 to May 2021.

In this project, we have created an admission-portal for those students who want to take admission in BSc and have applied in the ‘digital university system’ of the university. BSc is one of the most popular courses of Faculty of Science, M.S. University, Baroda. People associated with admission procedures of BSc must have to interact with our application, i.e., candidates and staff of academic section. Both of them have special role in our application.

## Table of Contents

Project Completion Certificate from The Company .....	2
College Certificates.....	3
Abstract .....	6
Table of Contents.....	7
PREFACE.....	8
Acknowledgement .....	9
About Project .....	10
System Requirements.....	13
Tools, Technologies & Libraries .....	14
User Stories & Product Backlogs .....	18
Sprint Backlogs.....	21
Sprint Burndown Charts .....	25
Data Modelling .....	27
Data Dictionary .....	28
Screenshots & Explanations .....	30
Restful APIs .....	86
Testing.....	88
Security and Other Features.....	96
Coding Standard.....	102
Limitations and Future Enhancements.....	103
References and Bibliography .....	104
Conclusion .....	105

## PREFACE

After completing 5 semesters in Department of Computer Applications successfully, we got to imply our knowledge of coding / programming, teamwork, communication skills, tools, strategies, and creativity into this project.

This project is main subject of final semester (i.e., 6<sup>th</sup> semester) of Bachelor of Computer Applications whose subject code is ‘BCA1601’. It must be completed by group of students. And this group can be of 2 to 4 students in size.

“Admission portal” is project of the group “P22”, which includes 3 members (us): Parth Vaswani, Sharanam Chotai and Milit Bhatt.

This project is designed to digitize the admission process of First Year of Bachelor of Science, Faculty of Science, Maharaja Sayajirao University, Baroda, under the guidance of Mrs. Heta Parekh.

As the term of semester 6 is coming to an end, we are submitting this document to faculties of BCA, which includes diagrams, strategies, data dictionary, features, and limitations of application.

In this document, we tried to make everything i.e., sections and diagrams with explanations, consistent, subsequent, and easy to interpret by anyone who knows at least basics of software engineering.

As this document will not be read by only faculties but also by junior students of our succeeding batch to document their project like this. With this concern in mind, we are putting strategies and methods along with appropriate graphics, which we have considered while working.

## **Acknowledgement**

Foremost, we would like to express our sincere gratitude...

To Dean of the Faculty of Science, Prof. Haribhai R. Kataria and our Head of the Department, Prof. Rakesh Srivastava for providing us an opportunity to show our knowledge, which we have gained in previous semesters; And for sharing their personal experiences about system and software development on-field, too.

To our passionate guide, Mrs. Heta Parekh for advising us throughout the project with her personal experience and professional ideas, who did not only recognize our mistakes but helped to solve them.

To Mrs. Priti Patel, for offering and allowing us to work on this project. Otherwise, it was not possible for us or anyone else to get this amazing opportunity to work on. We feel blessed because of her.

To Mr. Kshitij Tripathi, for guiding us throughout the most difficult module -- i.e., “Seat Allocation Strategy”. He explained the system well and helped us through his explanatory skills and experience.

To Mr. Krishna N. Rastogi, for giving us his precious time and experience to help us, especially for the “Merit List Generation” module.

To Mrs. Pooja Patel, for assisting, recognizing & encouraging our work with this matter.

To all the faculties of Department of Computer Applications for putting efforts and managing situations even due to pandemic-lockdown. They showered their rain of knowledge on us when we had no idea about so many things and almost everything was new for us.

To our beloved parents, for encouraging and supporting us throughout whole semester by providing enough physical computer resources and peaceful environment at home.

## About Project

### What was the need of this application?

Since M.S. University has its own portal to apply for admission online which works with every available course in each faculty, but that portal does not specifically solve the problems encountered during the admission process of BSc.

University's portal, named "Digital University System", provides functionality to upload documents of applicants, on-screen verification of those documents and online fees payment.

Admission portal, which is made by us, was supposed to automate the process of admission which was being done using one's hand and provide following functionalities:

- Login credentials should be sent via SMS and Email to all applicants, whose data is being provided by digital university system.
- Necessary notifications or announcements for the candidates or visitors should be shown on the portal. All previous notifications should be available on the portal.
- The participants' current application status, and logs of previous application statuses, should be shown to participants.
- Participants must be asked to confirm their participation in an admission round.
- Participants must have an interface to modify choice-filling.
- Participants should have the facility to cancel their admission in the middle of the process.
- System should prepare merit list based on various criteria/parameters.
- The system should call participants for admission rounds via SMS and email, where the number of participants in the admission rounds needs to be calculated as given in the 'requirement document'.
- System should allocate seats to candidate as per the guidelines automatically after receiving confirmation from academic section.

Because digital university system was not providing these facilities, distinct admission portal was required.

### How did we define the project? And what is the scope & purpose of it?

Generally, A project is defined as an effort to create or modify a specific product or service. Projects are temporary work efforts with a clear beginning and end.

Projects can be completely contained within a specific unit / department or include other organizations and vendors. A work effort may be considered a project if it meets the criteria established by the organization.

The sole purpose of this project is to simplify the way students take admission in BSc, which is very difficult to handle everything manually.

As we have had requirements documentation, we analyzed it and concluded that:

- The project can be considered a high-profile project for the faculty of science.
- The project will involve every department of BSc.
- The project has dependencies on other project (i.e., digital university system).
- The project impacts a large number of users (i.e., faculties of academic section and students).
- The project would take 4 to 5 months, which would be suitable for the duration of our semester.

### **How did we decide team size?**

According to the analysis, optimal team size was around four members. And according to scrum guidelines, teams of larger sizes require too much effort for communication; And teams of smaller sizes may not have sufficient qualifications or the capacity to complete work in a timely manner. (if you are wondering, why we followed the scrum guidelines !? it is discussed in the subsequent section.)

In the final analysis, we agreed to have 3 members in team:

1. Parth Vaswani
2. Sharanam Chotai
3. Milit Bhatt

### **What is agile and how our strategies were corelating with it?**

Agile is an umbrella term for a set of frameworks and practices. It advocates adaptive planning, evolutionary development, early delivery, and continual improvement, and it encourages flexible responses to change.

One of the differences between agile software development methods and traditional methods is the approach to quality and testing. In the traditional method, work moves through Software Development Lifecycle (SDLC) phases—with one phase being completed before another can start—hence the testing phase is separate and follows a build phase. However, in agile software development testing is completed in the same iteration (sprint) as programming.

Because testing is done in every iteration—which develops a small piece of the software—users can frequently use those new pieces of software and validate the value. After the users know the real value of the updated piece of software, they can make better decisions about the software's future. Having a value retrospective and software

re-planning session in each iteration—Scrum typically has iterations of just two weeks which is also known as sprint—helps the team continuously adapt its plans to maximize the value it delivers. This follows a pattern like the Plan-Do-Check-Act (PDCA) cycle, as the work is planned, done, checked (in the review and retrospective), and any changes agreed are acted upon.

This iterative approach supports a product rather than a project mindset. This provides greater flexibility throughout the development process; whereas on projects the requirements are defined and locked down from the very beginning, making it difficult to change them later. Iterative product development allows the software to evolve in response to changes in business environment or market requirements.

In agile, we were using scrum framework specifically (in which sprints take place). Every sprint is of 2 to 4 weeks.

### How was the work distributed and done?

→ Following table shows how team members contributed throughout project's life.

Team Member	LOC (Lines of code)	Assigned work (in hours)	Work done (in hours)
Parth Vaswani	5179	154	168
Sharanam Chotai	7951	123	137
Milit Bhatt	281	30	2
<b>Total</b>	<b>13411</b>	<b>307</b>	<b>307</b>

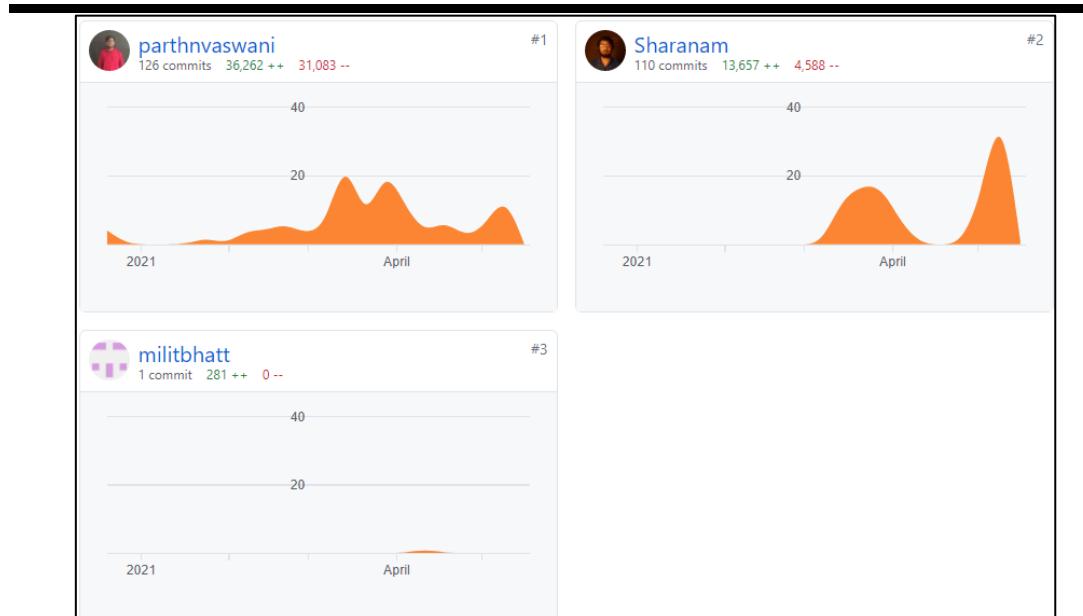


Fig: report generated by GitHub. Shows the contributions of individuals.

## System Requirements

**What are the things required for the application to run properly on the server side?**

➔ For backend:

Hardware:

- ✓ CPU(s): x86-64 (Intel is preferred); 4 cores minimum (16 or more recommended).
- ✓ Low-voltage CPUs (i.e., Intel Atom) are strongly not recommended due to poor performance.
- ✓ RAM: 8GB minimum, 16GB or more recommended.
- ✓ Disk: 64 GB or more of usable storage for one-server installation, including:
  - 30 GB minimum for OS,
  - 35 GB for the infrastructure,

Software:

- ✓ Latest stable version of following tools must be installed:
  - Node JS,
  - Mongo DB

**And minimum technical requirements for the end-user are...**

- ✓ Any of the following Operating System:
  - Android 5 and newer
  - iOS 10 and newer
  - Microsoft Windows 7 or higher
  - Any Linux distribution (with recommended web browser)
- ✓ Any of the following Web Browser (latest stable version recommended):
  - Chrome Browser
  - Android Browser
  - Edge Browser
  - Internet Explorer Browser (from version 11)
  - Firefox Browser
  - Opera Browser
  - Safari Browser
  - iPhone/iPad Browser (from version 6)

## Tools, Technologies & Libraries

We used the following technologies and libraries to develop this web application:

➤ Front-end:

- Technology:
  - Vue *// open-source model–view–viewmodel front end JavaScript framework for building user interfaces and single-page applications*
- Libraries:
  - Dependencies:
    - axios *// promise based HTTP client*
    - core-js *//provides polyfills for older-browser*
    - jwt-decode *// JWT decoder*
    - register-service-worker *// simplify SW registration*
    - vue *// building user interfaces*
    - vue-json-csv *// export Json Data into CSV*
    - vue-json-viewer *//simple JSON viewer*
    - vue-router *//part of vue ecosystem*
    - vuetify *//provides functional components*
    - vuex *//centralized store*
  - Dev-dependencies *// Development dependencies are intended as development-only packages, that are unneeded in production.*
    - @vue/cli-plugin-babel
    - @vue/cli-plugin-eslint
    - @vue/cli-plugin-pwa
    - @vue/cli-plugin-router
    - @vue/cli-plugin-vuex
    - @vue/cli-service
    - @vue/eslint-config-prettier
    - babel-eslint
    - eslint *//identify and reporting patterns*
    - eslint-plugin-prettier
    - eslint-plugin-vue
    - prettier *// code formatter*
    - sass
    - sass-loader
    - vue-cli-plugin-vuetify
    - vue-template-compiler
    - vuetify-loader

➤ Back-end:

- Technologies:
  - Mongo DB // NoSQL Database
  - Express JS // Back-end framework
  - NodeJS // JavaScript runtime
- Libraries:
  - Dependencies:
    - bcryptjs // hashes passwords using keys
    - body-parser // parses incoming data from the request
    - compression // provides http compressions
    - cors // allows APIs to interact from cross origin
    - csv-parser // parses csv into js objects
    - express // non-opinionated, minimalist web framework
    - express-fileupload // file uploader, middleware
    - express-mongo-sanitize // protects agt nosql injections
    - express-rate-limit //protects agt brute force and DoS
    - flat // flatten JS objects.
    - helmet // provides security by setting http headers
    - hpp // protects agt http param pollution
    - jsonwebtoken // used to sign and verify JWT
    - mongoose // ODM for mongoDB
    - morgan // logger
    - node-schedule // schedules cron jobs
    - nodemailer // email API provider
    - passport // authenticates users
    - passport-jwt // strategy for passport using JWT
    - rate-limit-mongo // middleware for express-rate-limit
    - validator // provides basic validations
    - xss-clean // middleware to sanitize user input
  - Dev-dependencies:
    - eslint
    - eslint-config-airbnb-base
    - eslint-config-prettier
    - eslint-plugin-import
    - Nodemon // auto-restarts node app when file changes

These technologies as a whole make stack, which is known as MEVN stack. The acronym MEVN stands for MongoDB, Express.js, VueJS, Node.js.

MEVN stack is the open-source JavaScript software stack that has emerged as a new and evolving way to build powerful and dynamic web applications. Its software

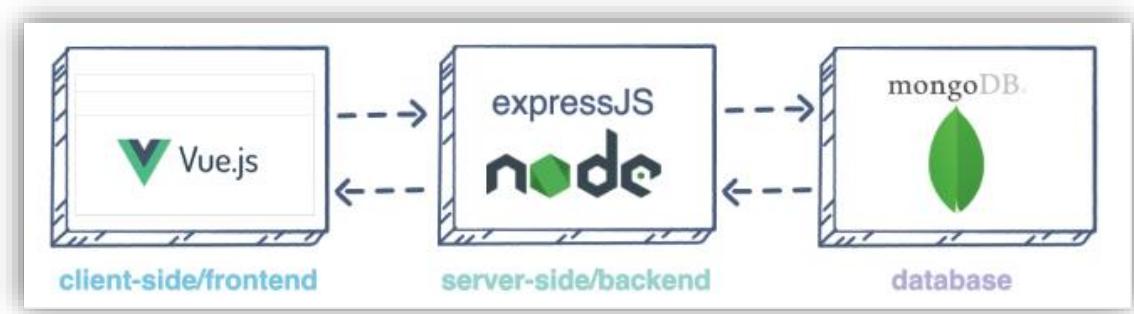
components can be used to effectively design frontend and backend development and improve the functionality of your website or app.

MongoDB: A document-oriented, No-SQL database used to store the application data.

ExpressJS: A framework layered on top of NodeJS, used to build the backend of a site using NodeJS functions and structures. Since NodeJS was primarily developed to run JavaScript on a machine instead of making websites, ExpressJS was created for the latter purpose.

Vue JS: VueJS is referred to as a client-side framework and is especially used in front-end web development. It has two-way data binding that allows seamless frontend development along with MVC capability and interactive server-side applications.

NodeJS: The JavaScript runtime environment. It is used to run JavaScript on a machine rather than in a browser.



➤ Tools & Platforms:

Visual Studio Code	// Code Editor
Postman	// For API Testing & Exclusive API-Documentation
Heroku	// Production - deployment
MongoDB Atlas	// Deployment of Database
Mongo DB compass and shell	// local development (database)
Git / Github	// Version control System & source code sharing
Chrome Dev Tools	// For debugging
Figma	// Prototypes of web pages and vector arts
Trello	// organized work with teammates
MS Office Visio	// Data Modelling (Diagrams)
Squoosh	//Image compressor (for serving image faster)
Adobe Photoshop	// Minor editing in images
MS Office PowerPoint	// User manuals (Handbooks)

## User Stories & Product Backlogs

### What are user stories and product backlog?

In software development and product management, a user story is an informal, natural language description of features of a software system. They are written from the perspective of an end user or user of a system, and may be recorded on index cards, Post-it notes, or digitally in project management software.

Here are some of our user stories which we made earlier:

As an admin, I want a login page which can only be accessible with 2 factor authentications so that I can access portal authoritatively and securely.

As an admin I want a dashboard so that I can upload CSV files into system which can be seeded into database afterwards.

As an admin I want an interface for sending multiple emails to candidates so that candidates can have credentials for accessing portal.

As an admin I want an interface for sharing text-based messages and files so that I can announce important stuff and publish merit lists.

As a visitor I want an interface to read and download stuff so that I can get updated with important stuff published by authorities.

As a visitor I want a search box above announcements so that I can filter-out content of my interest.

As a candidate I want to receive email for activating and accessing my account.

As a candidate I want a login page so that I can access dashboard through credentials given to me by authorities.

As a student, I want a dashboard so that I can see my current and previous statuses (logs).

As a student I want to choose subject groups of my interest so that faculty will know my priorities and which subject I want to be in.

As an admin I want an interface for adding candidates into database so that candidates who missed deadline to fill registration form can also be welcomed afterwards.

As an admin I want an interface for removing candidates so that I can remove candidates who gave incorrect information during registration.

As an admin I want to edit details of candidates so that candidates who gave wrong details can be rectified.

As an admin I want to get filters upon caste, group, and state so that I can generate various merit lists to be considered in admission rounds.

As a candidate, I should be able to change my interest about the admission at BSc.

As an admin, I should be able to generate admission rounds and access the details on the dashboard as well as in CSV format for admission purpose.

As an admin, I should be able to update admission success and fee payment's status at the end of admission rounds.

As an admin, I should be able to lock or unlock or schedule a lock on the choice filling, so that I can stop candidates from changing their choices.

As an admin, I should be able to upload the seat matrix, to seed the database and also update it afterwards.

As an admin, I should get notified if anything goes wrong in the system by any of my action so that I can retrospect and correct or solve them.

As an admin, I should have manual for the whole application so that I know my role and how I can do certain things within the system.

As a candidate, I should have manual for the whole application so that I know my role and how I can do certain things within the system.

A product backlog is a list of the new features, changes to existing features, and other activities that a team may deliver to achieve a specific outcome. The product backlog is the single authoritative source for things that a team works on.

Priority	Item	Description			Est	By	Sprint
<b>High</b>							
	1 Create reusable utils (backend)		8	p	1		
	2 Database		6	p	1		
	3 prototyping		7	S	5		
	4 Authentication and authorization(backend)		8	p	2		
	Log in						
	Notify						
	Delete						
	5 Homepage		8	S	2		
	6 Data seeding		16	P	2		
	7 Choice filling		6	SP	2		
	8 API documentation		12	P	3		
	Docs						
	9 Admin panel		10	S	3		
	10 Student profile		4	S	3		
	11 Modify and create required schema for admission rounds		1	P	4		
	12 Upload and save seat matrix (ADMIN) (backend)		3	S	4		
	13 Implement special category seat allocation		6	P	5		
<b>Medium</b>							
	14 Create boilerplate for project (backend)		4	P	1		
	15 Database relations and data dictionary		12	SP	1		
	16 Create boilerplate for project (Frontend)		4	P	1		
	17 Security		8	P	1		
	XSS(cross site scripting)						
	Regex DOS						
	DOS						
	NoSQL injection						
	hpp(http parameter pollution)						
	18 Manage announcements		8	SP	1		
	19 Authentication and authorization(frontend)		8	S	2		
	Log in						
	20 Merit list generation (backend)		8	P	3		
	21 Manage users (backend)		16	P	3		
	22 Add dynamic route imports (frontend)		1	S	3		
	23 one-time-database-setup		8	S	3		
	upload csv						
	Fetch CSV						
	Seed data						
	24 Reset password page		1	M	3		
	25 Frontend Optimization		3	P	3		
	26 Seat allocation		6	P	4		
	27 Add possibility of creating and managing multiple admins		3	P	4		
	28 Generate list for admission rounds		8	p	4		
	29 Get seat matrix		0.5	P	4		
	30 Logout Bug		2	S	4		
	31 Notify candidate about status in rounds		1	P	4		
	32 Api for updating seat matrix		0.5	P	4		
	33 Api for updating fee payment status		2	P	4		
	34 Api for accepting or rejecting invitation for admission round		1	S	4		
	35 Change Interest(Candidate)		1	S	5		
	36 Upload seat matrix		1	S	5		
	37 Show seat matrix		1	S	5		
	38 Update seat matrix		1	S	5		
	39 Generate list for admission round		1	S	5		
	40 Get Admission Round		1	S	5		
	41 Update student admission status		1	S	5		
	42 Update fee payment status		1	S	5		
	43 Accept or reject call for admission round		1	S	5		
<b>Low</b>							
	44 Create logos		2	S	1		
	45 Alpha testing(Frontend)		16	S	4		
	46 Alpha testing(backend)		16	p	4		
	47 Deployment and testing		4	p	5		
	48 Handbook for admin		3	S	4		
	49 Handbook for student		3	S	5		

Fig: Scrum-product-backlog of our project

# Sprint Backlogs

## What is sprint backlog?

The sprint backlog is a list of tasks identified by the scrum team to be completed during the scrum sprint. During the sprint planning meeting, the team selects some number of product backlog items, usually in the form of user stories and identifies the tasks necessary to complete each user story. Teams also estimate how many hours each task will take someone on the team to complete.

Below figures shows our sprint backlogs, that we worked on. Here, the tasks are written in the ‘sprint backlog’ column. In ‘assigned to’ column, characters denote first letter of the team member’s name. In ‘original estimated hours’ column, fractional number denotes how many hours it can take.

Sprint #1 Backlog		Assigned to	Status	Initial hours	04-Feb	05-Feb	06-Feb	07-Feb	08-Feb	09-Feb	10-Feb	Thursday	Friday	Saturday	Sunday	Monday	Tuesday	Wednesday	Sprint Review		
<b>Backend</b>																					
Create boilerplate for project																					
Create folder structure																					
P Completed 0.2 0.2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0																					
Create package.json																					
P Completed 0.5 0.5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0																					
Add dependencies																					
P Completed 0.2 0.2 0.1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0																					
Setup boilerplate code																					
P Completed 0.5 0.5 0.5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0																					
Setup config files																					
P Completed 1 0.8 0.8 0.5 0.5 0.2 0.5 0 0 0 0 0 0 0 0 0 0 0																					
Setup git																					
P Completed 0.2 0.2 0.2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0																					
Setup git hub repository																					
P Completed 0.2 0.2 0.2 0.3 0.3 0.3 0.3 0 0 0 0 0 0 0 0 0 0																					
Setup eslint																					
P Completed 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0 0 0 0 0 0 0 0 0 0																					
Setup prettier																					
P Completed 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0																					
Setup development and production scripts.																					
P Completed 1 1 1 0.5 0.5 0 0 0 0 0 0 0 0 0 0 0 0 0																					
Database relations and data dictionary		SP	Completed	12 12 10 11 11 8 9 6 4 4.5 3 3 2 0 0.5 0																	
Database				3 3 3 3 3 3 2 1 0 0.5 0 0 0 0 0 0 0 0																	
Validate data before saving it in database		P	Completed	3 3 3 2 2 1.5 0 0.6 0 0 0 0 0 0 0 0 0 0																	
Create database models				3 3 3 2 2 1.5 0 0.6 0 0 0 0 0 0 0 0 0 0																	
Security		S	Completed	XSS(cross site scripting)																	
Add validation for user generated data				1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0																	
Add xss-clean as a middleware				1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0																	
Add helmet as a middleware				1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0																	
Regex DOS				Clean and escape regular expressions before using them																	
DOS				P Completed 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0																	
Add express-mongo-sanitize as a middleware				P Completed 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0																	
Setup rate-limit-mongo				P Completed 1 1 1 1 1 1 1 1 1 0.8 0 0 0 0 0 0 0 0																	
NoSQL injection				Add express-mongo-sanitize as a middleware																	
hpp(http parameter pollution)				P Completed 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0																	
Add hpp as a middleware				P Completed 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0																	
Create reusable utils				8 8 8 8 8 8 8 8 8 7 5 4.4 3 4.7 2 0.5 0																	
Manage announcements				8 8 8 8 8 8 8 8 8 8 6 5.5 2 0 0 0 0 0																	
Create logos				2 2 2 2 2 1.5 1 1.2 0 0.5 0 0 0 0 0 0 0																	
Estimated Total Hours				64.5 64.3 60.9 55.4 55.3 41.2 37.6 32.8 22.8 18.5 16.9 15.5 9.7 2 1 0																	
Actual Total Hours				66 65 61 57 50 40 33 30 23 19 17 15 10 3 2 0																	

Fig: Sprint backlogs for 1<sup>st</sup> iteration of our product.

Fig: Sprint backlogs for 2<sup>nd</sup> iteration of our product.

Sprint #3 Backlog		Assigned to	Status	Initial hours	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday	Monday	Tuesday	Wednesday	Thursday
Backend																					
API documentation																					
Docs																					
Setup collection in postman	P	Completed		2	2	2	2	2	2	1	0.5	0	0	0	0	0	0	0	0	0	0
Create all endpoints	P	Completed		2	2	2	2	2	2	1.5	0	0	0	0	0	0	0	0	0	0	0
Add description to endpoints	P	Completed		8	8	8	8	8	8	7.5	7	6	5.5	4	2.5	2.5	1	0	0.5	0	0
Merit list generation	P	Completed		8	7.5	6	6	6.5	5.5	5.5	5	5	4.5	3.5	3	3	1.5	2	1	0	0
Manage users	P	Completed		16	15	14.5	14.5	13	11	11	9	8.5	7	6	6.5	5	5	3	2.5	1	0.5
Frontend																					
Add dynamic route imports																					
admin																					
get user and user details	S	Completed		0.2	0.2	0.2	0.2	0.2	0.2	0.2	0	0	0	0	0	0	0	0	0	0	0
get, set and seed file into or from database	S	Completed		0.2	0.2	0.2	0.2	0.2	0.2	0.2	0	0	0	0	0	0	0	0	0	0	0
lock, unlock, schedule and get lock information	S	Completed		0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0	0	0	0	0	0	0	0	0
get merit lists	S	Completed		0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.3	0.3	0	0	0	0	0
notify user(s)	S	Completed		0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0	0	0	0
prototyping																					
Student																					
logout page	S	Completed		0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0	0	0	0
Profile	S	Completed		0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0	0	0	0	0	0	0	0	0	0
update in choice filling	S	Completed		0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0	0	0	0	0	0	0	0	0	0
save new password	S	Completed		0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0	0	0	0	0	0	0	0	0	0
Admin																					
set database	S	Completed		0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.2	0	0	0	0	0	0	0	0	0	0
notify users	S	Completed		0.5	0.5	0.5	0.5	0.5	0.5	0.5	0	0	0	0	0	0	0	0	0	0	0
new candidate	S	Completed		0.5	0.5	0.5	0.5	0.5	0.5	0.5	0	0	0	0	0	0	0	0	0	0	0
merit lists	S	Completed		0.5	0.5	0.5	0.5	0.5	0.5	0.5	0	0	0	0	0	0	0	0	0	0	0
lock choices	S	Completed		0.5	0.5	0.5	0.5	0.5	0.5	0.5	0	0	0	0	0	0	0	0	0	0	0
one-time-database-setup																					
upload csv																					
file uploader with specific mime types	S	Completed		1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1	1	1	0.5	0	0	0	0	0	0
Fetch CSV																					
Fetch uploaded csv from backend	S	Completed		2	2	2	2	2	2	2	2	2					0				
csv-parser function in javascript	S	Completed		1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1	0.5	0	0	0	0	0	0	0
Seed data																					
Interface to start seeding process with forewarning	S	Completed		1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1	0	0	0	0	0	0	0
Reset password link	M	Completed		1	1	1	1	1	1	1	1	1	1	1	1	1	1	0.7	0.7	0.5	0.3
Admin panel	S	Completed		10	10	10	10	10	10	10	7.5	8	6.5	4	4.5	4	4	3	3.5	1	0.5
Student profile	S	Completed		10	10	10	10	10	10	10	9.5	8	6	5	3.5	2	2	1	0.5	0	0
Frontend Optimization	P	Completed		10	10	10	10	10	10	10	10	10	10	10	8	7	7	5.5	4	2.5	1
Estimated Total Hours				79	77.5	75.5	75.5	74.5	71.5	71.5	62	57.1	46.1	39.4	32.2	25.5	25.5	15.9	13.2	6.5	2.3
Actual Total Hours				80	79.5	78	77	75	74	73	60	54	47	40	33	28	26	16	13.5	7	2.5

Fig: Sprint backlogs for 3<sup>rd</sup> iteration of our product.

Sprint #4 Backlog		Assigned to	Status	Initial hours	01-Apr	02-Apr	03-Apr	04-Apr	05-Apr	06-Apr	07-Apr	08-Apr	09-Apr	10-Apr	11-Apr	12-Apr	13-Apr	14-Apr	Sprint Review
<b>Backend</b>																			
Add possibility of creating and managing multiple admins	P	Completed	3	2.5	2	1	0	0.5	0	0	0	0	0	0	0	0	0	0	
Deployment and testing	P	Completed	4	4	2	0	0	0	0	0	0	0	0	0	0	0	0	0	
Modify and create required schema for admission rounds	P	Completed	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Upload and save seat matrix	S	Completed	3	3	3	3	3	3	2	0.5	0	0	0	0	0	0	0	0	
Generate list for admission rounds	P	Completed	8	8	8	8	6.5	4	2	0	0	0	0	0	0	0	0	0	
Get seat matrix	P	Completed	0.5	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Notify candidate about status in rounds	P	Completed	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Api for updating seat matrix	P	Completed	0.5	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Api for updating fee payment status	P	Completed	2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
Api for accepting or rejecting invitation for admission round	S	Completed	1	1	1	1	1	1	0.5	0.5	0	0	0	0.5	0	0	0	0	
<b>Frontend</b>																			
Handbook for admin	S	Pending	3	3	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	15	
Logout Bug	S	Completed	2	1	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	
prototyping	S	Completed	2	2	2	2	1.5	1.5	1	1	0	0	0	0.5	0	0	0	0	
Upload Status files	M	Pending	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	1	
Upload seat matrix	M	Pending	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0.5	
Estimated Total Hours																			
Actual Total Hours		34	32	26	21	18.5	16	10.5	8.5	7	6	5	7.5	6	4	2	1	2.5	

Fig: Sprint backlogs for 4<sup>th</sup> iteration of our product. (90° in anticlockwise direction)

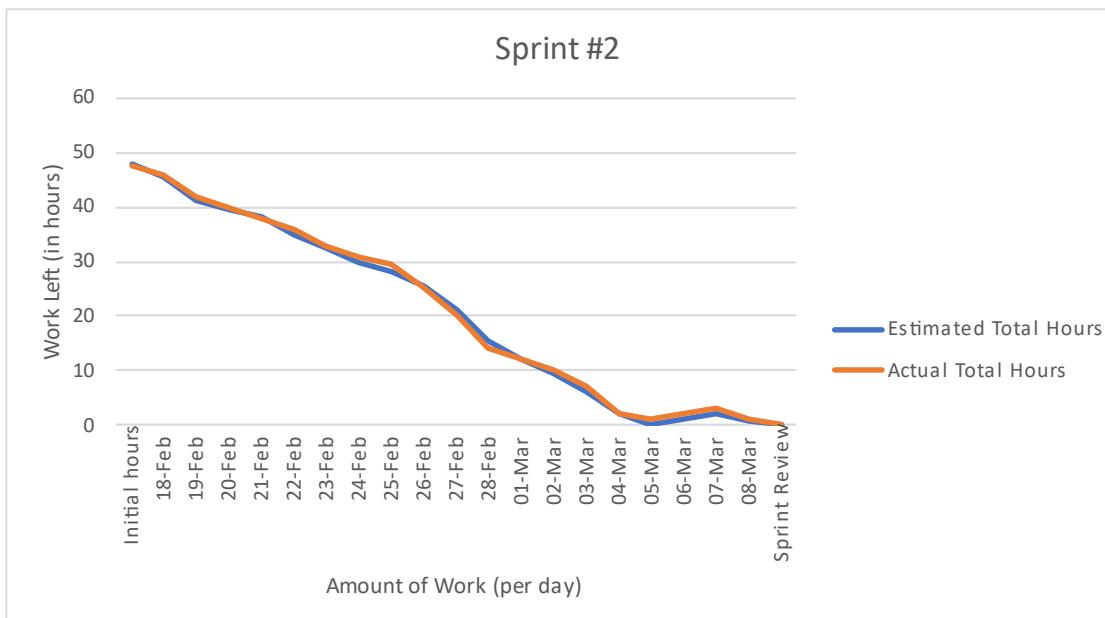
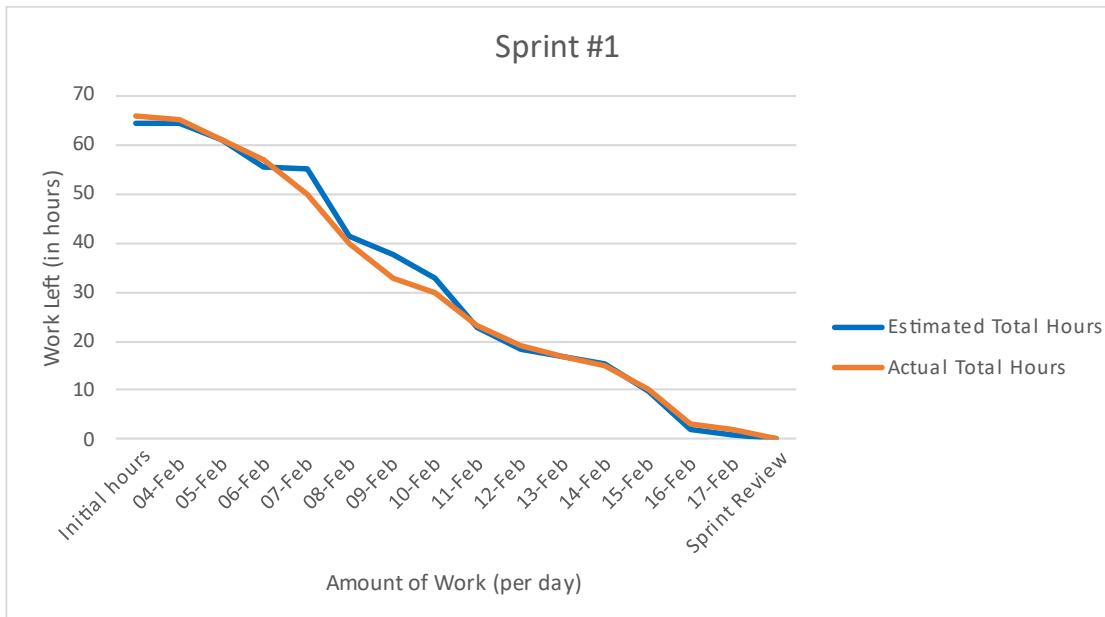
During last sprint, non-uniform updates in requirements and exceptions occurred, therefore we needed to make changes in sprint life cycles. So that we can implement the changes and test them well.

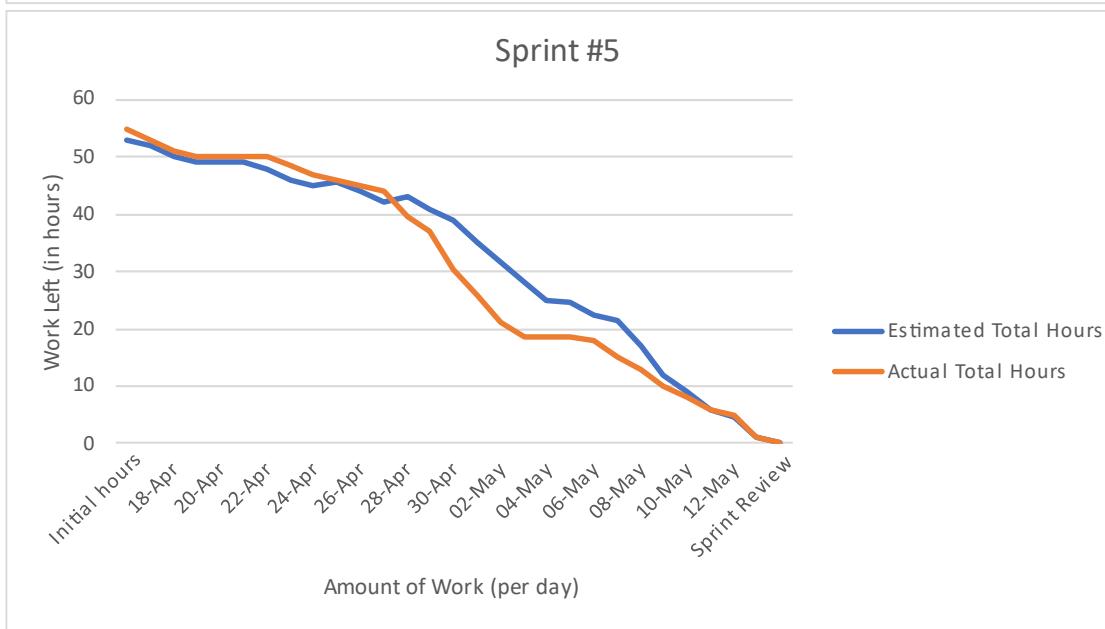
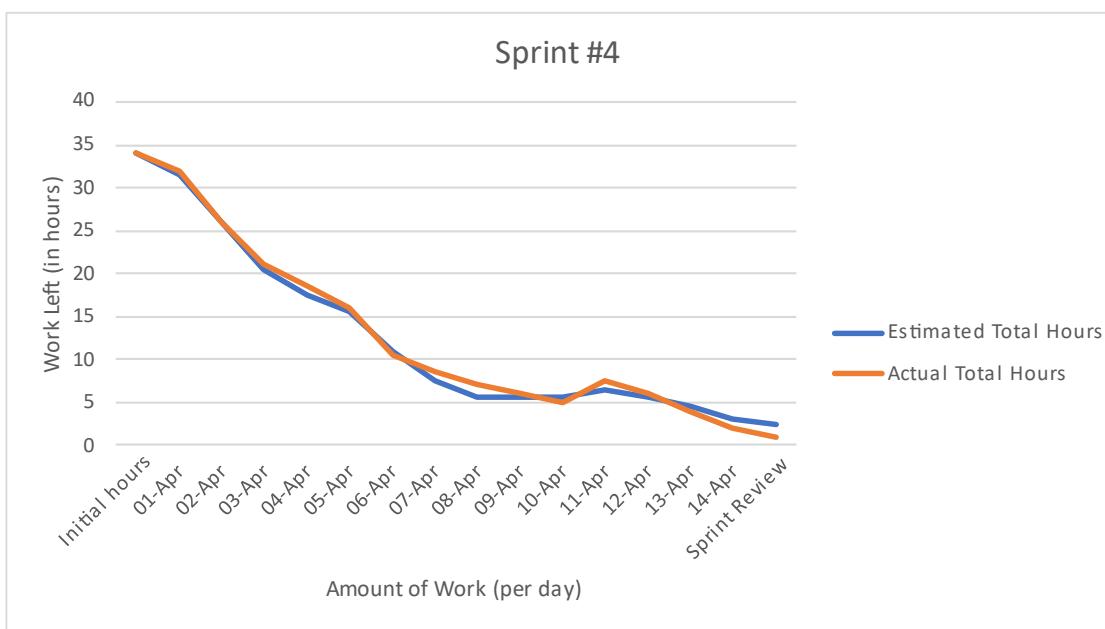
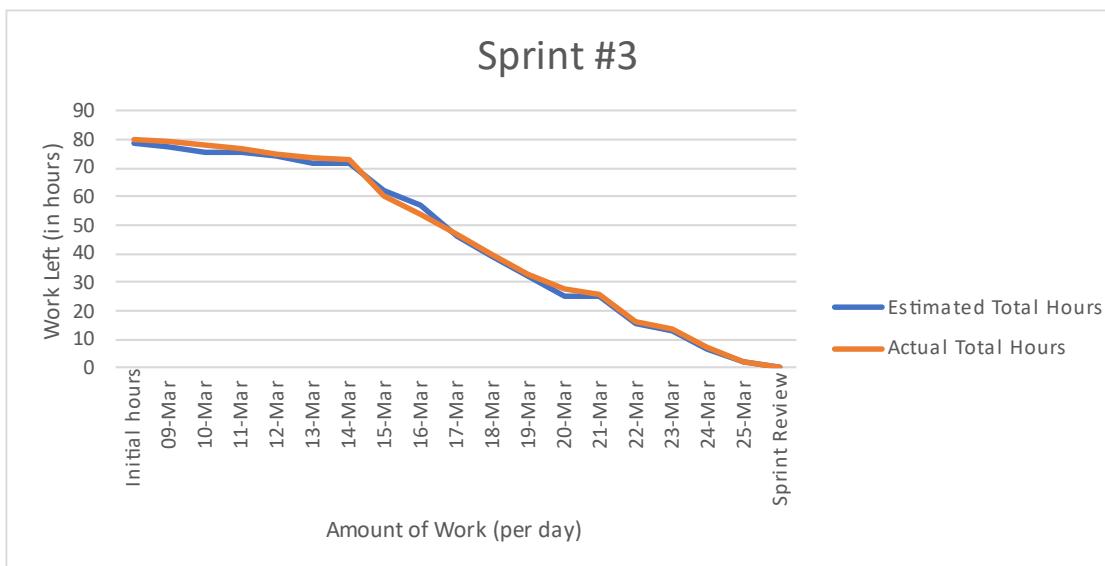
Fig: Sprint backlogs for 5<sup>th</sup> iteration of our product. (90° in anticlockwise direction)

# Sprint Burndown Charts

## What is sprint burndown charts?

A burndown chart is a tool used by Agile teams to gather information about both, the work they have completed on a project and the work that is yet to be done within a Sprint. It shows the total effort against the amount of work for each sprint. The chart slopes downward over Sprint duration.





# Data Modelling

Following diagram describes the schema of database of our application, which is implemented in MongoDB.

MongoDB is a cross-platform document-oriented database program and classified as a NoSQL database program. It uses JSON-like documents (BSON) with optional schemas.

Instead of entity-relationship diagram, crow's foot notations are used here for ease.

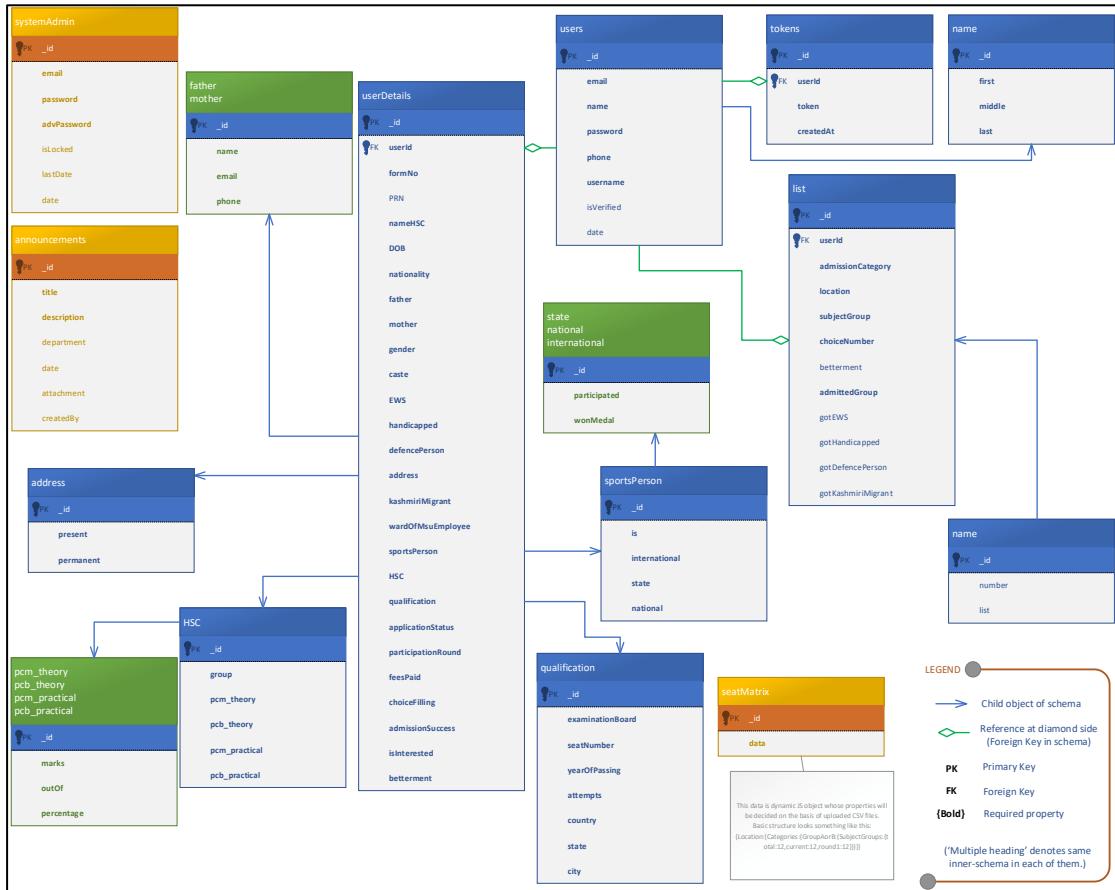


Fig: data diagram also known as data modelling.

# Data Dictionary

It is a tabular description of database schema, by which developers can imply database and/or understand its schema.

Admin Schema							
Object/ Column	Type	Reference	Required	Unique	Default	Remarks	Read it as
email	String		TRUE	TRUE		lowercase	email is a required, unique, string object, which must be in lower-case.
password	String		TRUE				password is required, string object
advPassword	String		TRUE				advance password is required, string object
lastDate	String						last date contains timestamp in string format
isLocked	Boolean				FALSE		by default, it is not locked, boolean object
date	Date				Current date		when admin was created
Announcement Schema							
Object/ Column	Type	Reference	Required	Unique	Default	Remarks	Read it as
title	String		TRUE			6 < x < 30	length of title compulsorily needs to be in range of 6 to 30
description	String		TRUE			12 < x < 512	length of description compulsorily needs to be in range of 6 to 30
department	String						optional for specific department
attachment	String						optional for specific announcement, contains path of file
createdBy	String						Created by whom (admin's id)
date	Date				Current date		date when admin announces something
User Schema							
Object/ Column	Type	Reference	Required	Unique	Default	Remarks	Read it as
name	Set		TRUE				contains name of candidate
email	String		TRUE			lowercase	email of candidate is required
password	String		FALSE				password of candidate can be blank
phone	String		TRUE				phone number of candidate is required
username	String		TRUE	TRUE			username is required
isVerified	Boolean				FALSE		candidate is verified or not, by default no
date	Date				Current Date		date when student activate account.
'name' in User Schema							
Object/ Column	Type	Reference	Required	Unique	Default	Remarks	Read it as
first	String		TRUE				first name of candidate as String is required.
middle	String		TRUE				middle name of candidate as String is required.
last	String		TRUE				last name of candidate as String is required.
Tokens Schema							
Object/ Column	Type	Reference	Required	Unique	Default	Remarks	Read it as
userId	ObjectID	users	TRUE				User Id is foreign key from 'users', type is objectID of schema, required.
token	String		TRUE				Token is String, required.
createdAt	Date		TRUE		Current Date	expires in 3 days	Creation time is required. Token will be expire after 3 days
User Details Schema							
Object/ Column	Type	Reference	Required	Unique	Default	Remarks	Read it as
userId	ObjectID	users	TRUE			immutable	User Id is foreign key from 'users', type is objectID of schema, required and immutable
formNo	String		TRUE	TRUE			form no is stored as string. It is required and unique
PRN	Number		FALSE				PRN is number and is not required.
nameHSC	String		TRUE				nameHSC is string and required
DOB	String		TRUE				DOB is string and is required.
nationality	String		TRUE				nationality is required
gender	String		TRUE				gender is required.
father	Set		TRUE				set is containing father's details
mother	Set		TRUE				set is containing mother's details
address	Set		TRUE				set is containing address
caste	String		TRUE				caste is required
EWS	Boolean						whether candidate is in category of EWS
handicapped	Boolean						whether candidate is in category of handicapped
defencePerson	Boolean						whether candidate is in category of defence person
kashmiriMigrant	Boolean						whether candidate is in category of kashmiri migrant
wardOfMsuEmployee	Boolean						whether candidate is in category of ward of msu employee
sportsPerson	Set						contains details of sports
qualification	Set		TRUE				contains details of qualification
HSC	Set						contains details of HSC
applicationStatus	[String]						array of application statuses (historical)
participationRound	Type.Mixed			{}			participation round is mixed type schema property, by default blank object
feesPaid	Boolean				FALSE		checks whether fees is paid or not
choiceFilling	[Number]						array of choices made by candidate
admissionSuccess	Boolean				FALSE		checks whether admission is done or not
isInterested	Boolean				TRUE		whether candidate is interested or not in admission
betterment	Boolean				FALSE		whether candidate wants betterment or not
'father' in User Schema							
Object/ Column	Type	Reference	Required	Unique	Default	Remarks	Read it as
name	String		TRUE				Full name of father is required
email	String		TRUE			lowercase	email of father is required
phone	String		TRUE				phone number of father is required
'mother' in User Schema							
Object/ Column	Type	Reference	Required	Unique	Default	Remarks	Read it as
name	String		TRUE				Full name of mother is required
email	String		TRUE			lowercase	email of mother is required
phone	String		TRUE				phone number of mother is required
'address' in User Schema							
Object/ Column	Type	Reference	Required	Unique	Default	Remarks	Read it as
present	String		TRUE				present address is required
permanent	String		TRUE				permanent address is required

'sportsPerson' in User Schema							
Object/ Column	Type	Reference	Required	Unique	Default	Remarks	Read it as
is	Boolean						is candidate sports person or not.
international	Set						contains data of international match
state	Set						contains data of state level match
national	Set						contains data of national level match
'international' in 'sportsPerson' in User Schema							
Object/ Column	Type	Reference	Required	Unique	Default	Remarks	Read it as
participated	Boolean						did candidate participated in international level competition
wonMedal	Boolean						did candidate won medal in international level competition
'national' in 'sportsPerson' in User Schema							
Object/ Column	Type	Reference	Required	Unique	Default	Remarks	Read it as
participated	Boolean						did candidate participated in national level competition
wonMedal	Boolean						did candidate won medal in national level competition
'state' in 'sportsPerson' in User Schema							
Object/ Column	Type	Reference	Required	Unique	Default	Remarks	Read it as
participated	Boolean						did candidate participated in state level competition
wonMedal	Boolean						did candidate won medal in state level competition
'qualification' in User Schema							
Object/ Column	Type	Reference	Required	Unique	Default	Remarks	Read it as
examinationBoard	String		TRUE				Name of examination board is required
seatNumber	String		TRUE				seat number of candidate in exam is required
yearOfPassing	Number		TRUE				yearofpassing as number is required
attempts	Number		TRUE		minimum 1		attempts in passing exam as number is required
country	String		TRUE				country where exam was given is required
state	String		TRUE				state where exam was given is required
city	String		TRUE		Uppercase		city where exam was given is required, is uppercase
'HSC' in User Schema							
Object/ Column	Type	Reference	Required	Unique	Default	Remarks	Read it as
pcm_theory	Set						Contains details of physics, chemistry,maths theory
pcb_theory	Set						Contains details of physics, chemistry,biology theory
pcm_practical	Set						Contains details of physics, chemistry,maths practical
pcb_practical	Set						Contains details of physics, chemistry,biology practical
group	String		TRUE				HSC group is required
'pcb_theory' in 'sportsPerson' in User Schema							
Object/ Column	Type	Reference	Required	Unique	Default	Remarks	Read it as
marks	Number				0		Obtained marks in number
outOf	Number				0		Total marks in number
percentage	Number				0		percentage in number format
'pcm_theory' in 'sportsPerson' in User Schema							
Object/ Column	Type	Reference	Required	Unique	Default	Remarks	Read it as
marks	Number				0		Obtained marks in number
outOf	Number				0		Total marks in number
percentage	Number				0		percentage in number format
'pcm_practical' in 'sportsPerson' in User Schema							
Object/ Column	Type	Reference	Required	Unique	Default	Remarks	Read it as
marks	Number				0		Obtained marks in number
outOf	Number				0		Total marks in number
percentage	Number				0		percentage in number format
'pcb_practical' in 'sportsPerson' in User Schema							
Object/ Column	Type	Reference	Required	Unique	Default	Remarks	Read it as
marks	Number				0		Obtained marks in number
outOf	Number				0		Total marks in number
percentage	Number				0		percentage in number format
Admission Round Schema							
Object/ Column	Type	Reference	Required	Unique	Default	Remarks	Read it as
number	Number			TRUE			current round number
list	[Object]						list of candidates, with details
'list' in Admission Round Schema							
Object/ Column	Type	Reference	Required	Unique	Default	Remarks	Read it as
userId	ObjectID	userDetails	TRUE				User Id is foreign key from 'users', type is objectID of schema, required
admissionCategory	String		TRUE				admission category is string and required
location	String		TRUE				location is string and required
subjectGroup	String		TRUE				subject group is string and required
choiceNumber	Number		TRUE				choice number consist integer and is required
betterment	Boolean				FALSE		betterment is needed or not. It is boolean and by default no (false)
admittedGroup	String		TRUE				admitted group is string and required
gotEWS	Boolean				FALSE		candidate got admission in EWS seat, by default false
gotHandicapped	Boolean				FALSE		candidate got admission in handicapped's seat, by default false
gotDefencePerson	Boolean				FALSE		candidate got admission in defence person's seat, by default false
gotKashmiriMigrant	Boolean				FALSE		candidate got admission in kashmiri migrant's seat, by default false
Seat Matrix Schema							
Object/ Column	Type	Reference	Required	Unique	Default	Remarks	Read it as
data	Object						Data uploaded via CSVs, to JS Object

# Screenshots & Explanations

## Prototypes:

At very first, we started this project by making prototypes in figma.

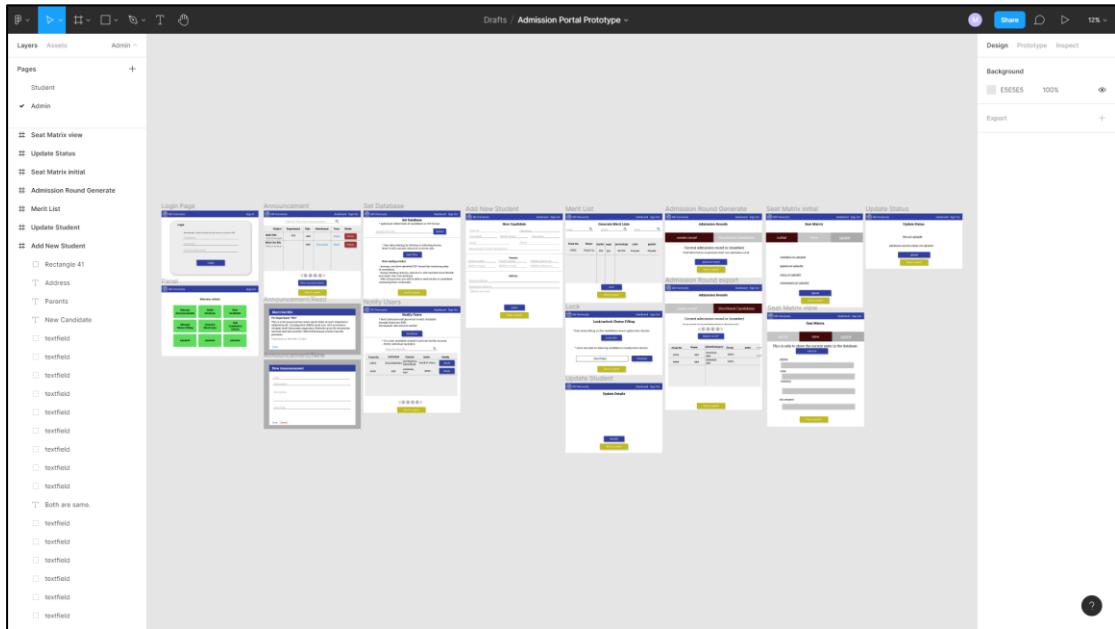


Fig: Front page of figma application, where we used to make prototypes of web pages.

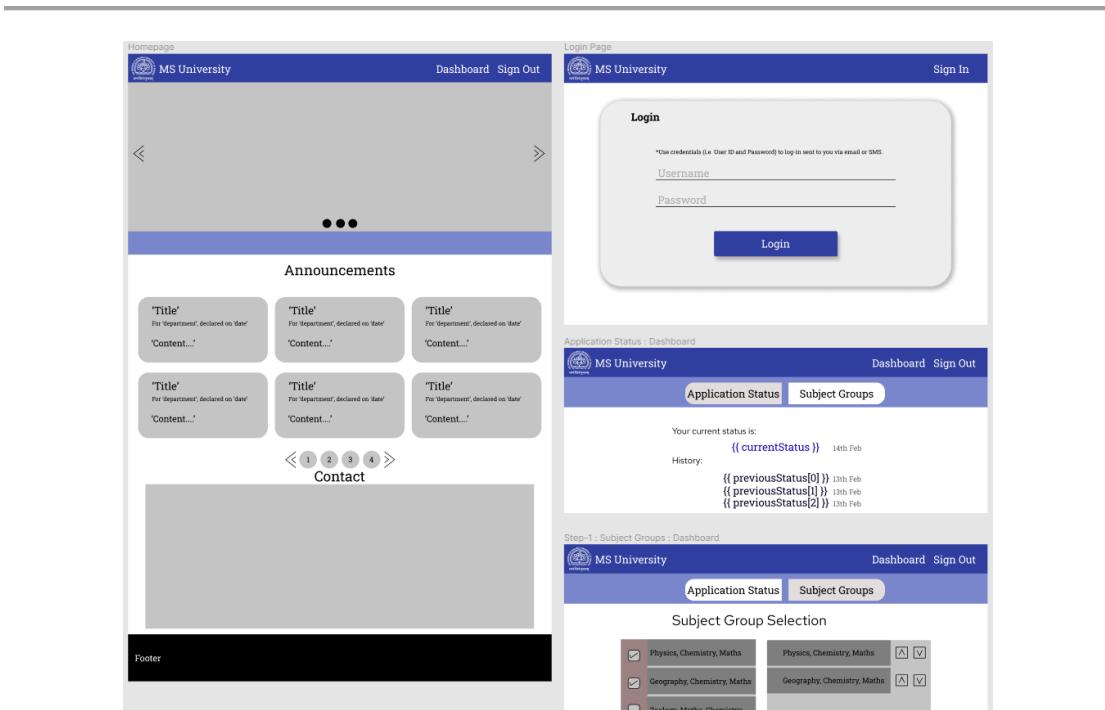


Fig: prototypes of homepage, login page and application status for candidates.

Profile

MS University

**Personal**

Name  
Gender  
Place of Birth  
Date of Birth  
Mother's Name  
Blood Group  
Marital Status

**Contact**

Email  
Phone

**Address**

Line 1:  
Village:  
District:  
Pincode:

**Reservation**

Category:  
Sports:  
Ward of MSU:

**Education**

SSC  
HSC Theory: 250/300 Practical: 145/150

Footer

MS University

Dashboard Sign Out

Application Status Subject Groups

**Subject Group Selection**

<input checked="" type="checkbox"/> Physics, Chemistry, Maths <input checked="" type="checkbox"/> Geography, Chemistry, Maths <input type="checkbox"/> Zoology, Maths, Chemistry	<input checked="" type="checkbox"/> Physics, Chemistry, Maths <input checked="" type="checkbox"/> Geography, Chemistry, Maths
--	--

**Clear All**

Note: You will have to confirm your selected groups in next page

**Proceed**

Step-2 Confirm subjects : Subject Groups : Dashboard

MS University

Dashboard Sign Out

Application Status Subject Groups

**Subject Group Selection**

Selected subject groups are in following order:

1. Physics, Chemistry, Maths 2. Geography, Chemistry, Maths 3. Zoology, Maths, Chemistry
--

I want to modify choices.

**Modify Choices**

Fig: prototypes of profile and choice filling (candidate)

The image displays four wireframe prototypes for a university management system:

- Login Page:** A blue header bar with 'Login Page' and 'MS University' on the left, and 'Sign In' on the right. Below is a white 'Login' form with fields for 'Username', 'Password', and 'Advance Password', and a 'Login' button.
- Admin Panel:** A blue header bar with 'Panel' and 'MS University' on the left, and 'Sign Out' on the right. Below is a 'Welcome, Admin!' message and a 3x3 grid of green buttons with labels: 'Manage Announcements', 'Build Database', 'New Candidate' (top row); 'Manage Choice Filling', 'Generate Merit Lists', 'Edit Candidate's Details' (middle row); and '#####', '#####', '#####' (bottom row).
- Manage Announcement Page:** A blue header bar with 'Announcement' and 'MS University' on the left, and 'Dashboard' and 'Sign Out' on the right. It features a search bar 'Filter by Title of an Announcement' with a magnifying glass icon. Below is a table with columns: Subject, Department, Date, Attachment, View, and Delete. Two rows are shown:
 

Subject	Department	Date	Attachment	View	Delete
<b>Bold Title</b> Thin Descript...	###	###		<a href="#">Read</a>	<a href="#">Delete</a>
<b>Merit list BSc</b> This is to be a ...		###	<a href="#">Download</a>	<a href="#">Read</a>	<a href="#">Delete</a>

 Navigation buttons '«« 1 2 3 4 »»', 'New Announcement' (blue), and 'Back to panel' (yellow) are at the bottom.
- Read-Announcement Dialog Box:** A blue header bar with 'Announcement/Read' on the left. Below is a box titled 'Merit list BSc' with the text: 'For Department: "BSc" This is to be announced as Lorem ipsum dolor sit amet consectetur adipisicing elit. Consequuntur debitis quod, iure, velit accusamus voluptas modi laboriosam aspernatur distinctio quos illo recusandae nesciunt aperiam maiores? Officia doloremque eveniet maxime possimus.' Published on: Wed Mar 10 2021. A 'Close' button is at the bottom.

Fig: prototypes of login page (admin), admin panel (dashboard), manage-announcement page, read-announcement dialog box.

The image shows a 'New Announcement' dialog box with the following fields:

- Title:** An input field for the announcement title.
- Department:** An input field for the announcement department.
- Description:** A text area for the announcement description.
- Attach file:** A file input field for attaching documents.

At the bottom are 'Save' and 'Cancel' buttons.

Fig.: Prototype of new announcement dialog box

Set Database
MS University
Dashboard
Sign Out

### Set Database

\* upload pre-refined data of candidates in CSV format.

---

\* Start data seeding, by clicking on following button.  
Note: It will consume resources in server side.

**How seeding works?**

- Assume, you have uploaded CSV format file containing data of candidates.
- During 'Seeding' process, web server will read data from that file and insert rows into database.
- After this process, you will be able to send emails to candidates containing their credentials.

### Notify Users

MS University
Dashboard
Sign Out

#### Notify Users

\* Send Username and password to each candidate through Email and SMS.  
Prerequisite: data must be seeded.

Filter by Form No. 
🔍

\* If in case, candidate missed to activate his/her account,  
- Notify individual candidate.

Form No.	username	Contact	name	Notify
14062	D20CJ0002922	parth@kma... 6353195124	Parth N Vasw...	<input style="background-color: #002060; color: white; border: 1px solid #ccc; padding: 2px 10px; border-radius: 5px;" type="button" value="Notify"/>
####	###	#####... ###	####...	<input style="background-color: #002060; color: white; border: 1px solid #ccc; padding: 2px 10px; border-radius: 5px;" type="button" value="Notify"/>

<<
1
2
3
4 >>

Fig: prototypes for uploading student data csv file, seeding data into database, Manage candidate (here, manage candidate was modified afterwards, therefore prototype was not updated)

Department of Computer Application, M.S. University, Baroda

33 | Page

Add New Student
MS University
Dashboard
Sign Out

### New Candidate

**Parents**

### Address

Both are same.

Insert

Back to panel

Update Student
MS University
Dashboard
Sign Out

### Update Details

note: same form component as new candidate

Modify

Back to panel

Fig: prototypes for adding and updating candidate directly into database through admin-dashboard. (as we were using VueJS, components remain same for similar use cases)

Merit List

MS University [Dashboard](#) [Sign Out](#)

**Generate Merit Lists**

caste   group   state  

Form No.	Name	marks	total	percentage	caste	gender
14062	Parth Va..	250	300	83.33%	General	Female

**save**

**Back to panel**

Lock

MS University [Dashboard](#) [Sign Out](#)

**Lock/unlock Choice Filling**

\* lock choice filling, so that candidates cannot update their choices.

**Lock Now**

\* select last date for allowing candidates to modify their choices.

**Date Picker** **Schedule**

**Back to panel**

Fig: prototypes for generating merit lists and managing choice filling



Fig: prototypes for uploading and viewing seat matrix files and last module of admin Update status, (update fees payment status and admission success status)

Here, component for default file uploader was set before this prototype was made. Thus, we were simply writing which component would be there, and deciding in which route it will be shown.

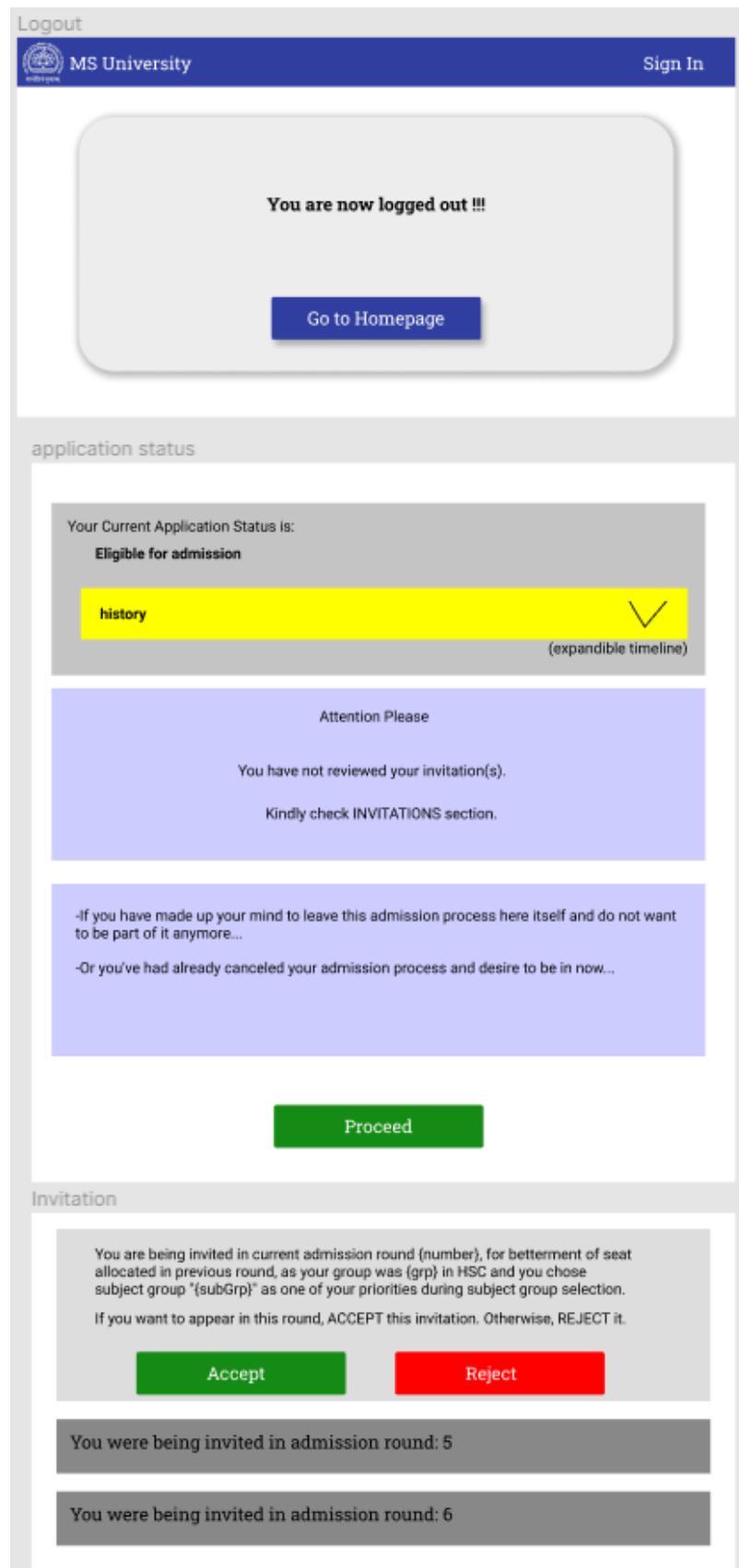
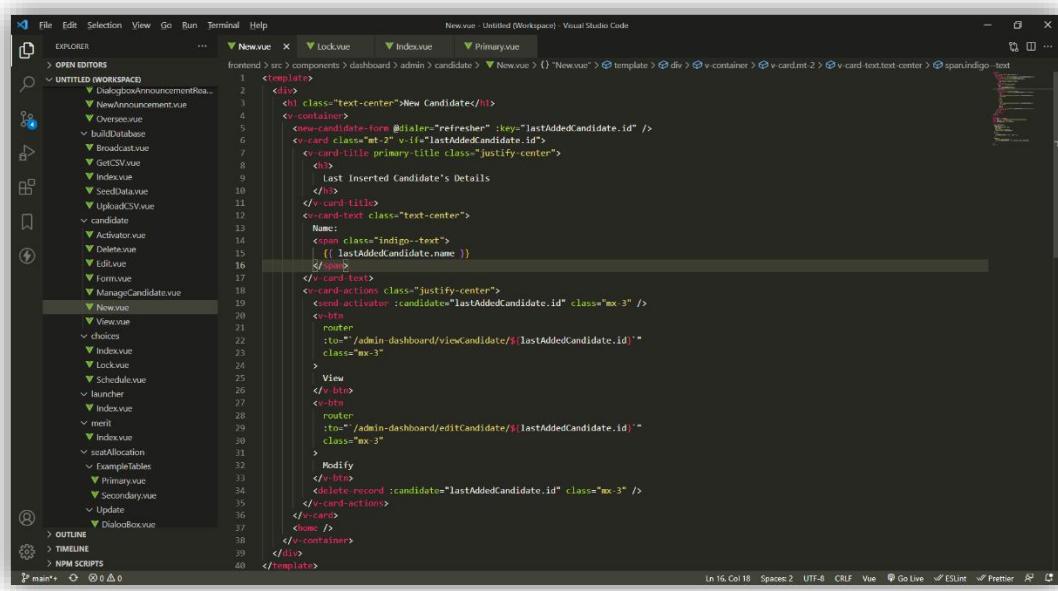


Fig: prototypes for candidate dashboard, logout page, and invitation page.

## Development-tools:

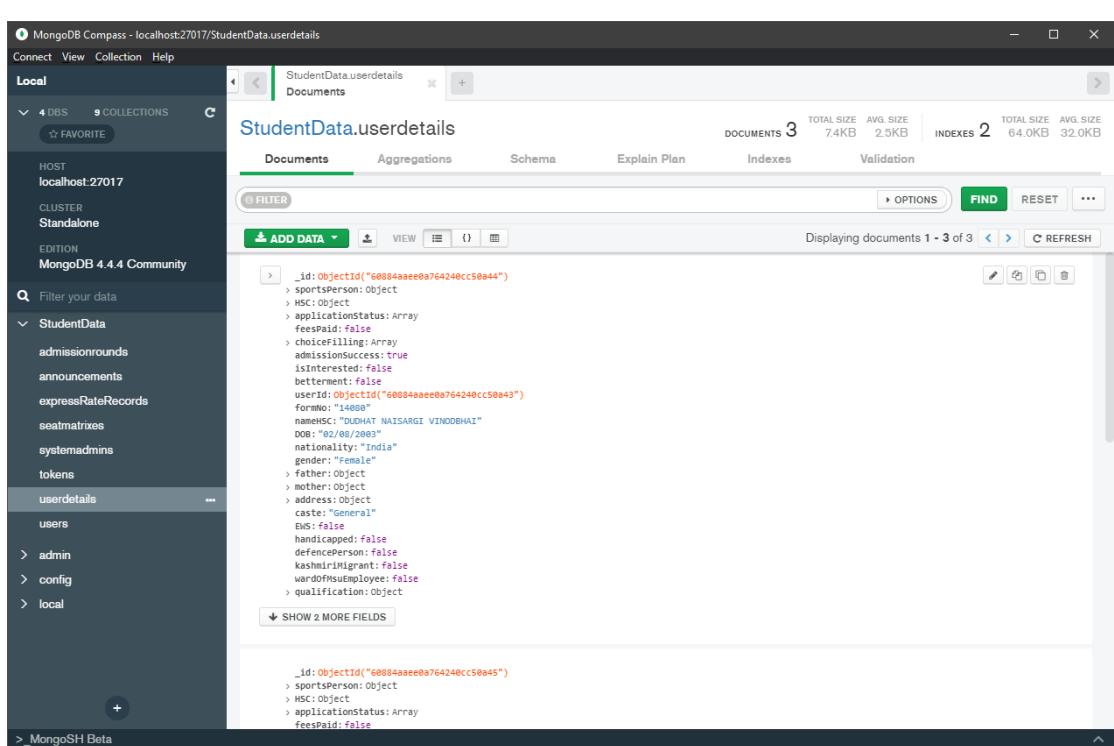


```

<template>
  <div>
    <h1>New Candidate</h1>
    <div>
      <form>
        <input type="text" v-model="name" placeholder="Name" />
        <input type="text" v-model="email" placeholder="Email" />
        <input type="text" v-model="phone" placeholder="Phone" />
        <input type="text" v-model="address" placeholder="Address" />
        <input type="text" v-model="city" placeholder="City" />
        <input type="text" v-model="state" placeholder="State" />
        <input type="text" v-model="pincode" placeholder="Pincode" />
        <input type="text" v-model="country" placeholder="Country" />
        <input type="text" v-model="qualification" placeholder="Qualification" />
        <input type="text" v-model="experience" placeholder="Experience" />
        <input type="text" v-model="skills" placeholder="Skills" />
        <input type="text" v-model="hobbies" placeholder="Hobbies" />
        <input type="text" v-model="languages" placeholder="Languages" />
        <input type="text" v-model="interests" placeholder="Interests" />
        <input type="text" v-model="achievements" placeholder="Achievements" />
        <input type="text" v-model="certifications" placeholder="Certifications" />
        <input type="text" v-model="projects" placeholder="Projects" />
        <input type="text" v-model="resume" placeholder="Resume" />
        <input type="text" v-model="portfolio" placeholder="Portfolio" />
        <input type="text" v-model="references" placeholder="References" />
        <input type="text" v-model="password" placeholder="Password" />
        <input type="text" v-model="confirmPassword" placeholder="Confirm Password" />
        <input type="button" value="Submit" />
      </form>
    </div>
  </div>

```

Fig: visual studio code: primary code editor (IDE)



```

{
  "_id": "60884aaee0a764240cc50844",
  "sportsPerson": Object,
  "HSC": Object,
  "applicationStatus": Array,
  "feesPaid": false,
  "choiceFilling": Array,
  "admissionSuccess": true,
  "isInterested": false,
  "betterment": false,
  "userId": "60884aaee0a764240cc50843",
  "form": "12th pass",
  "nameHSC": "DUDHAT NAISARGI VINOOBHAI",
  "DOB": "01/01/2000",
  "nationality": "India",
  "gender": "Female",
  "father": Object,
  "mother": Object,
  "address": Object,
  "caste": "General",
  "EWS": false,
  "handicapped": false,
  "defencePerson": false,
  "kashmiriMigrant": false,
  "wardOffsumEmployee": false,
  "qualification": Object
}

{
  "_id": "60884aaee0a764240cc50845",
  "sportsPerson": Object,
  "HSC": Object,
  "applicationStatus": Array,
  "feesPaid": false
}

```

Fig: MongoDB compass on local machine. To check and access database directly.

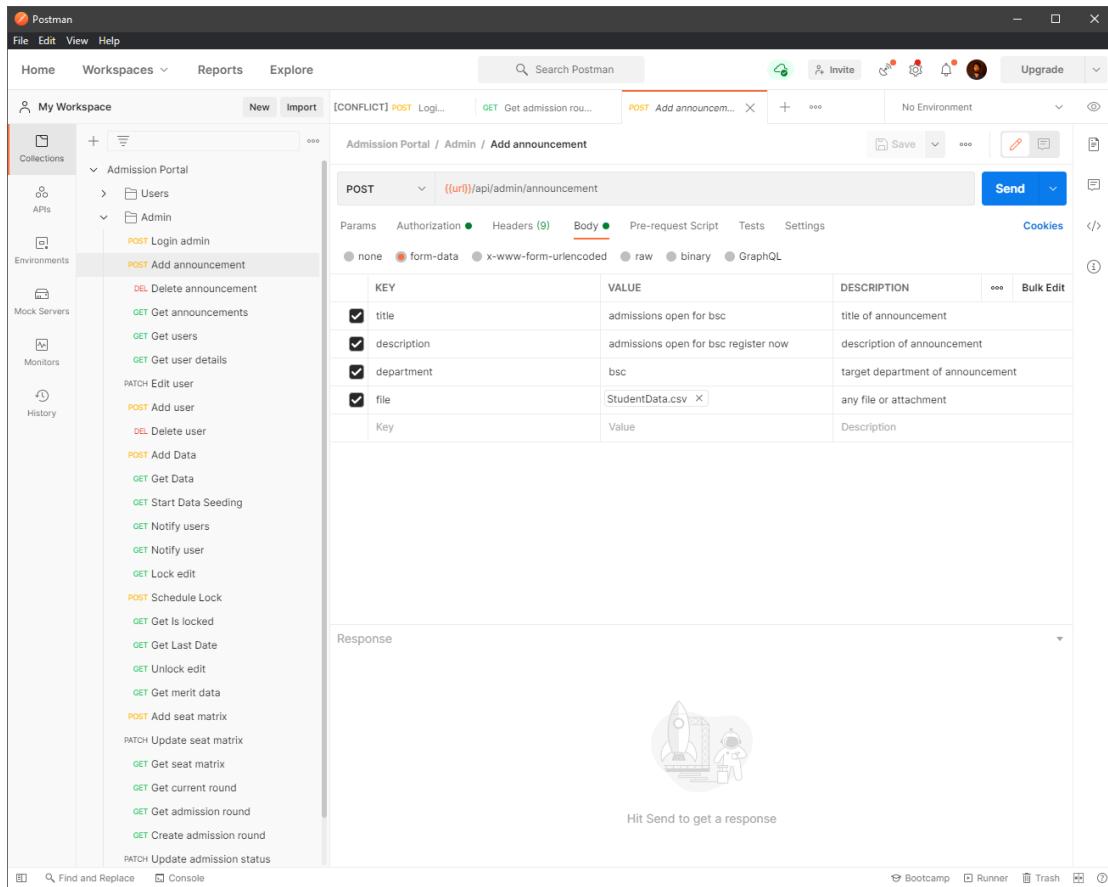


Fig: Postman (application) on local machine. To test and document APIs.

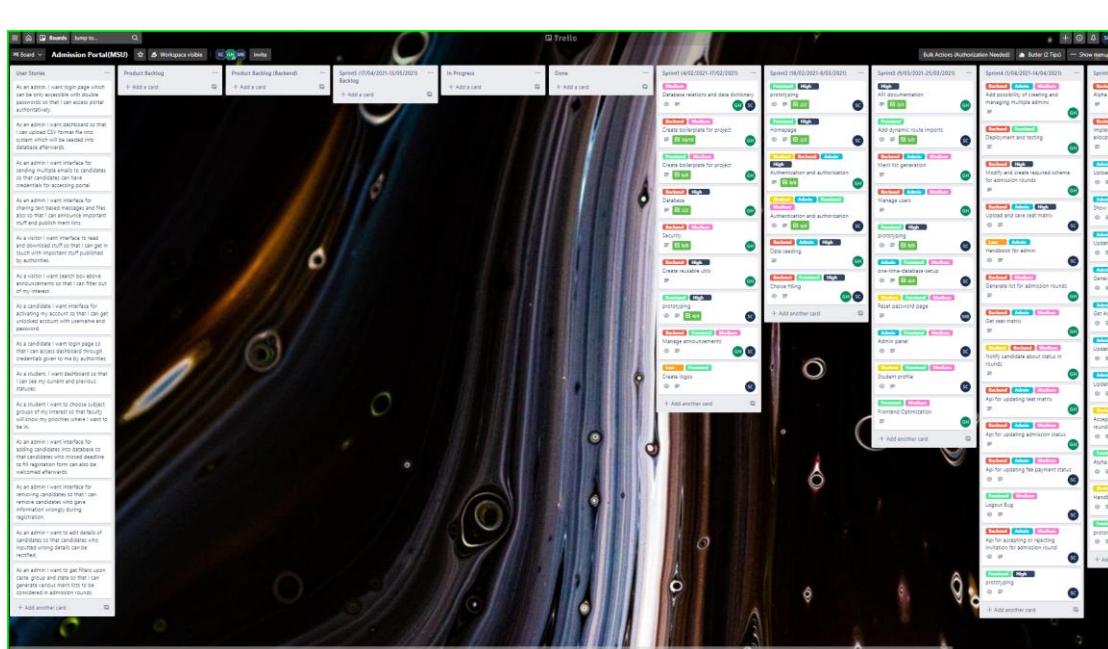


Fig: Trello to manage tasks (agile-scrum), we can put and move cards on this board.

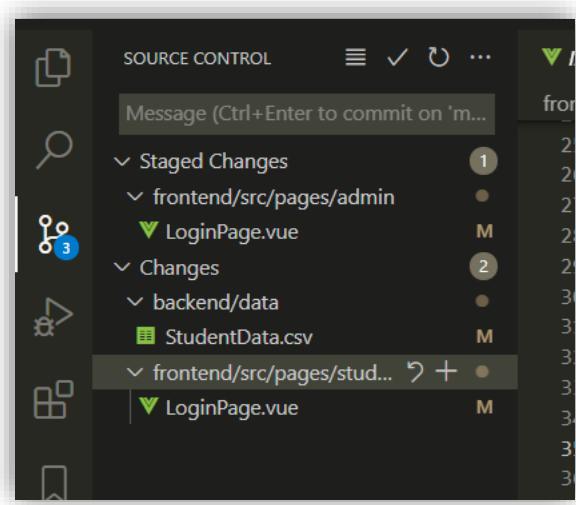


Fig: in-vs-code git for version control and Github for source code sharing.

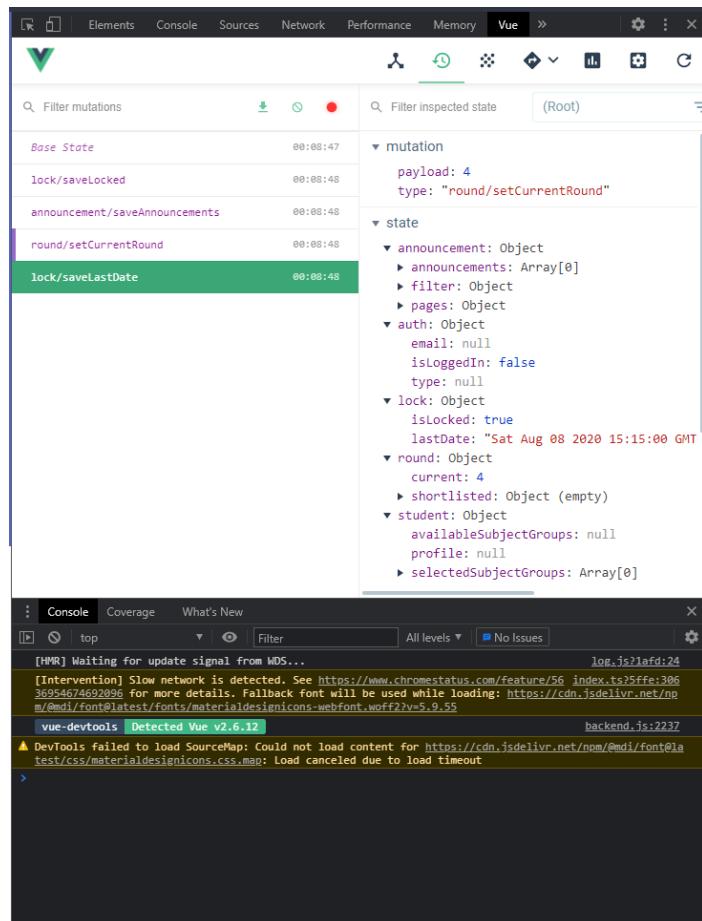
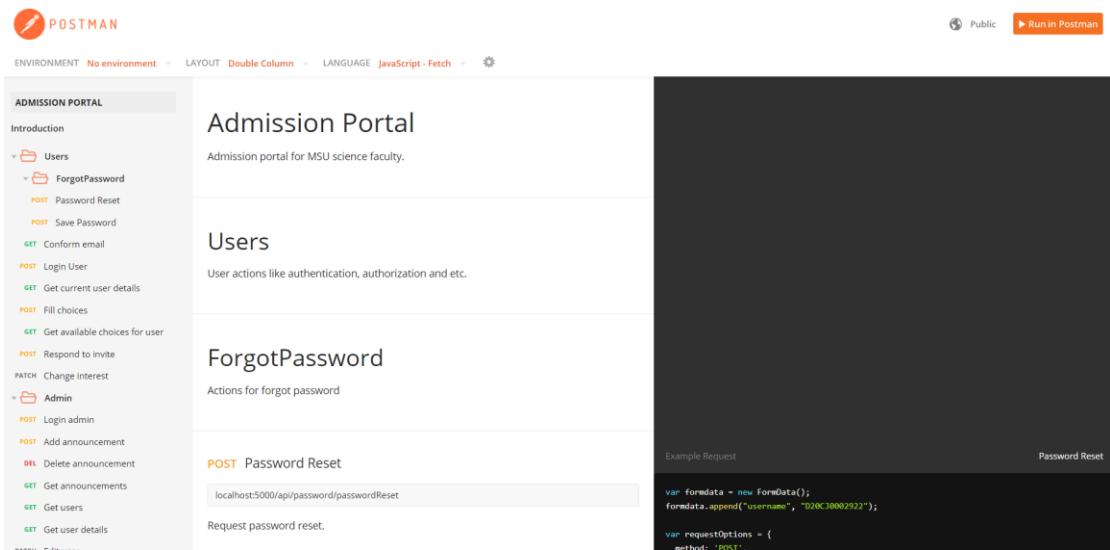


Fig: vue-chrome-dev-tools for debugging



```
var formdata = new FormData();
formdata.append("username", "020C30007922");

var requestOptions = {
  method: 'POST',
```

Fig: postman documentation.

You can see API's documentation here: [Admission Portal \(getpostman.com\)](https://documenter.getpostman.com/view/11722625/TWDTNzj7#60e7f9dd-e7af-42a2-928f-83d18fc68676) [ <https://documenter.getpostman.com/view/11722625/TWDTNzj7#60e7f9dd-e7af-42a2-928f-83d18fc68676> ]

---

## Admission Portal (Website):

**Announcements**

Filter by Title  Search

<b>dummy 2</b> <small>For: maths</small> this is also dummy <a href="#">READ MORE</a>	<b>Difficult test</b> <small>making dummy announcements are stressful job.</small> <a href="#">READ MORE</a>
---	--

**Mamata writes urges modi**  
For: Politics Mamata writes to PM Modi, urges him .. Read more at: [http://timesofindia.indiatimes.com/articleshow/82632317.cms?utm\\_source=contentofinterest&utm\\_medium=text...](http://timesofindia.indiatimes.com/articleshow/82632317.cms?utm_source=contentofinterest&utm_medium=text...)  
[READ MORE](#)

**India's Covid crisis**  
For: Biology India's Covid crisis pushes up the cost of living Read more at: [http://timesofindia.indiatimes.com/articleshow/82630004.cms?utm\\_source=contentofinterest&utm\\_medium=text...](http://timesofindia.indiatimes.com/articleshow/82630004.cms?utm_source=contentofinterest&utm_medium=text...)  
[READ MORE](#)

**Handbook**

For Candidates...

We understand that many students face difficulties while working with web-portals.  
That is why we are providing you a handbook, which is easy to understand.

[Open Handbook](#)

**Contact**

DEAN	
Prof. Haribhai R. Kataria	
FACULTY OF SCIENCE	
(O) +91 265 279 5329	
(R) 94277 88347	

[Office Bearers](#)

Developed by Students of Computer Applications | © 2021 Follow us :

Fig: Homepage of the admission portal, from which candidate or visitor can view announcements and handbook.

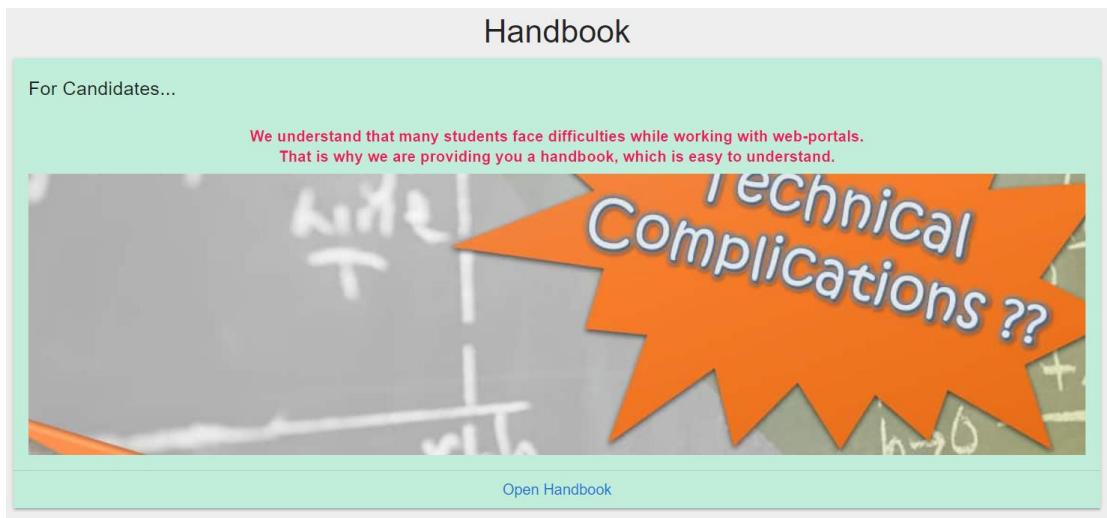


Fig: in homepage, visitors & participants can find 'handbook for candidates' to understand system flow from candidates' point of view. (It could not be copied in full page screenshot `due to CSS animation)

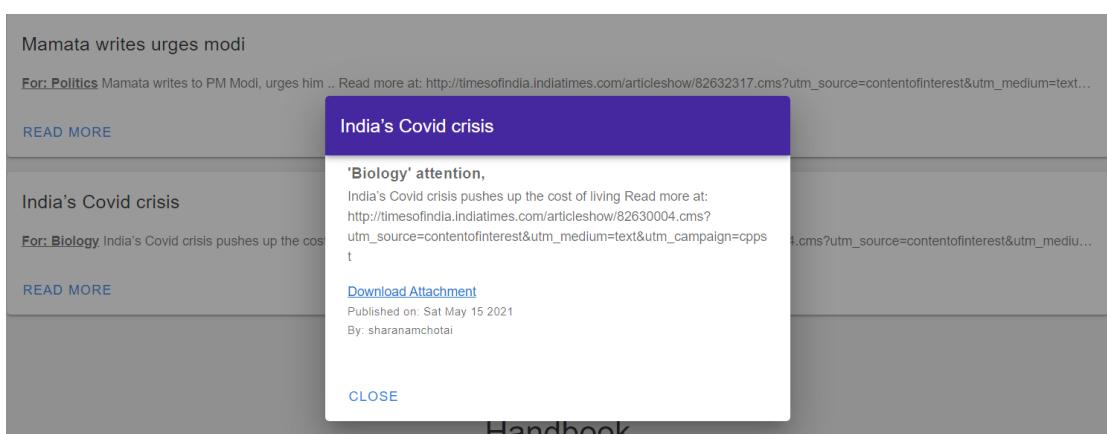


Fig: visitors and candidates reading announcement (dialog box) from homepage



Fig: user is searching for specific text in title, if multiple announcements will be found, user can navigate through pages. There is also a clear-button, which can remove search filter on one-click.

# Administrator's Place

E-mail

User ID is required

Password

>Password is required

Advance Password

Advance Password is required

**SUBMIT**

**GO TO STUDENT'S PAGE**

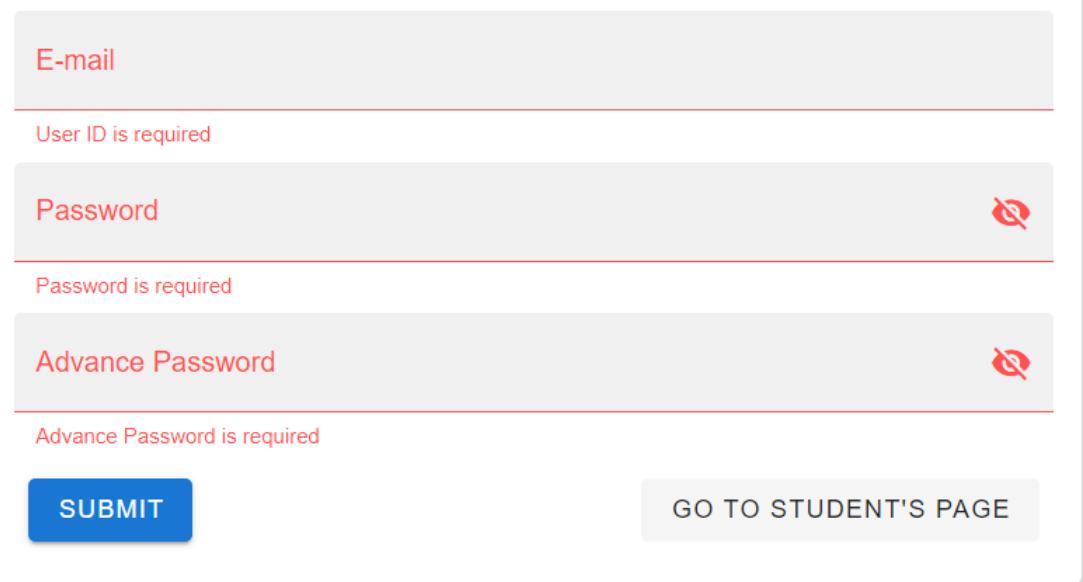
A screenshot of an admin login form. It has four input fields: 'E-mail', 'Password', 'Advance Password', and 'Advance Password'. Each field has a red error message below it: 'User ID is required', 'Password is required', 'Advance Password is required', and 'Advance Password is required' respectively. There are two buttons at the bottom: a blue 'SUBMIT' button and a white 'GO TO STUDENT'S PAGE' button.

Fig: admin's login page, where admin will need to provide username, password and advance password which is kind of 2 step verification.

All the messages in red font are errors.

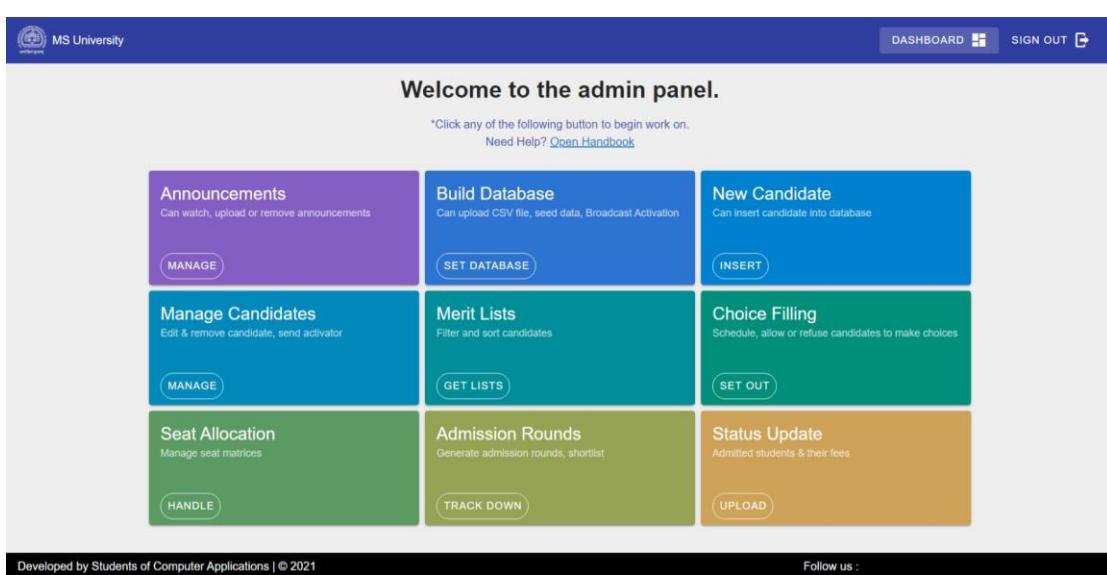
A screenshot of the admin panel dashboard. At the top, there is a header with the MS University logo, a dashboard link, and a sign out link. The main title is 'Welcome to the admin panel.' Below the title, there is a sub-instruction: '\*Click any of the following button to begin work on.' and a 'Need Help? Open Handbook' link. The dashboard is divided into a 3x3 grid of cards. The cards are: 'Announcements' (purple), 'Build Database' (blue), 'New Candidate' (blue), 'Manage Candidates' (blue), 'Merit Lists' (teal), 'Choice Filling' (teal), 'Seat Allocation' (green), 'Admission Rounds' (green), and 'Status Update' (orange). Each card has a description and a primary action button: 'MANAGE', 'SET DATABASE', 'INSERT', 'MANAGE', 'GET LISTS', 'SET OUT', 'HANDLE', 'TRACK DOWN', and 'UPLOAD'.

Fig: admin panel (dashboard). Whenever admin will login, they will get to this page.

Department	Subject	Date	By	Attachment	View	Delete
maths	dummy 2 this is also dummy	Sat May 15 2021	sharanamchotai		<a href="#">OPEN</a>	<a href="#">DELETE</a>
	Difficult test making dummy announcements are stressful job.	Sat May 15 2021	sharanamchotai		<a href="#">OPEN</a>	<a href="#">DELETE</a>
Politics	Mamata writes urges modi Mamata writes to PM Modi, urges him ... Read more at: <a href="http://timesofindia.indiatimes.com/articleshow/82632317.cms?utm_source=contentofinterest&amp;utm_medium=text&amp;utm_campaign=cppst">http://timesofindia.indiatimes.com/articleshow/82632317.cms?utm_source=contentofinterest&amp;utm_medium=text&amp;utm_campaign=cppst</a>	Sat May 15 2021	sharanamchotai		<a href="#">OPEN</a>	<a href="#">DELETE</a>
Biology	India's Covid crisis India's Covid crisis pushes up the cost of living Read more at: <a href="http://timesofindia.indiatimes.com/articleshow/82630004.cms?utm_source=contentofinterest&amp;utm_medium=text&amp;utm_campaign=cppst">http://timesofindia.indiatimes.com/articleshow/82630004.cms?utm_source=contentofinterest&amp;utm_medium=text&amp;utm_campaign=cppst</a>	Sat May 15 2021	sharanamchotai	<a href="#">Download</a>	<a href="#">OPEN</a>	<a href="#">DELETE</a>

Fig: admin can manage (create, read, search, and delete) announcements.

**India's Covid crisis**

**For (Department): 'Biology',**  
India's Covid crisis pushes up the cost of living Read more at:  
[http://timesofindia.indiatimes.com/articleshow/82630004.cms?utm\\_source=contentofinterest&utm\\_medium=text&utm\\_campaign=cppst](http://timesofindia.indiatimes.com/articleshow/82630004.cms?utm_source=contentofinterest&utm_medium=text&utm_campaign=cppst)  
<http://localhost:5000/609ec72900a1a439a0bedab5Untitled.jpg>

Attachment: [Download](#)

Published on: Sat May 15 2021  
By: sharanamchotai

[DELETE](#)

Fig: admin reading announcement from inside of admin panel.

**Delete Announcement**

Are you sure you want to permanently delete this announcement?

[DELETE](#) [CANCEL](#)

Fig: admin deleting incorrectly uploaded announcement.

New Announcement X

---

Title Title is required.

Department

Description

Attach PDF File.

SAVE

Fig: admin creating new announcement.

---

UPLOAD CSV      VIEW CSV      SEED DATA      BROADCAST



Students' data  
Attach normalized CSV file containing student data.

UPLOAD

BACK TO PANEL

Fig: admin uploading student's data (CSV file), which was given by University Digital System. But before uploading file here, admin needs to normalize that received file manually as written in the admin's handbook.

---

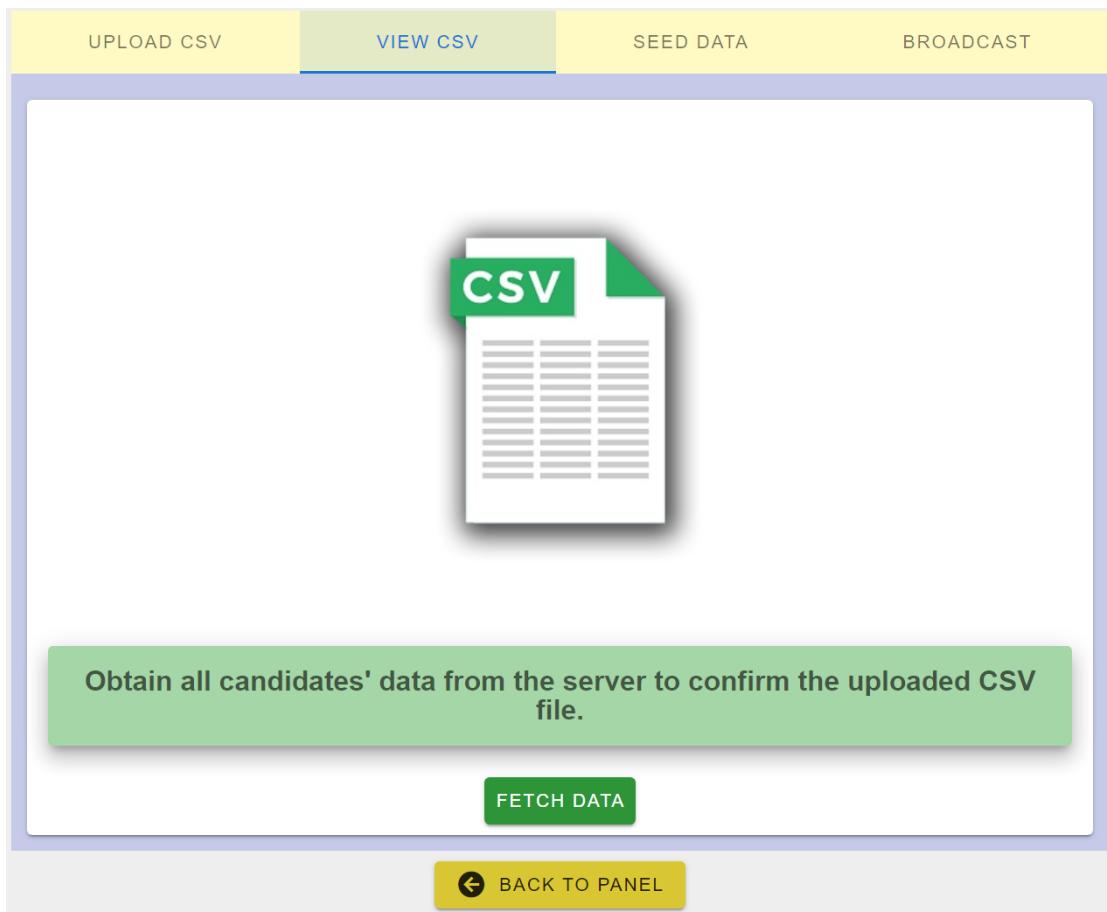


Fig: admin can confirm that uploaded CSV file is in correct form, by fetching it.

ApplicationNo	FirstName	LastName	MiddleName	MotherName	NameOnMarkSheet	Gender	DOB	Username	Citizenship	CorrespondenceAddress	PermanentAddress
1	KEYA	SHAH	JAYESHKUMAR SHAH	DEEPA SHAH	SHAH KEYA JAYESHKUMAR	Female	02-09-2002	D25_2000	India	Green Park Soc. Akota Behind Akota Stadium, Vadodara, 390020*	Green Park Soc. Akota Behind Akota Stadium, Vadodara, 390020*
2	NAISARGI	DUDHAT	VINODBHAI	JYOTIBALA	DUDHAT NAISARGI VINODBHAI	Female	02-08-2003	25_2000	India	MAIN ROAD, GANDHARI WADI, JUNAGADH, JOSHIPURA, JUNAGADH, 362002*	MAIN ROAD, GANDHARI WADI, JUNAGADH, JOSHIPURA, JUNAGADH, 362002*

FETCH DATA

Fig: After verifying that file is uploaded correctly and data is in correct form.... (in next fig.)

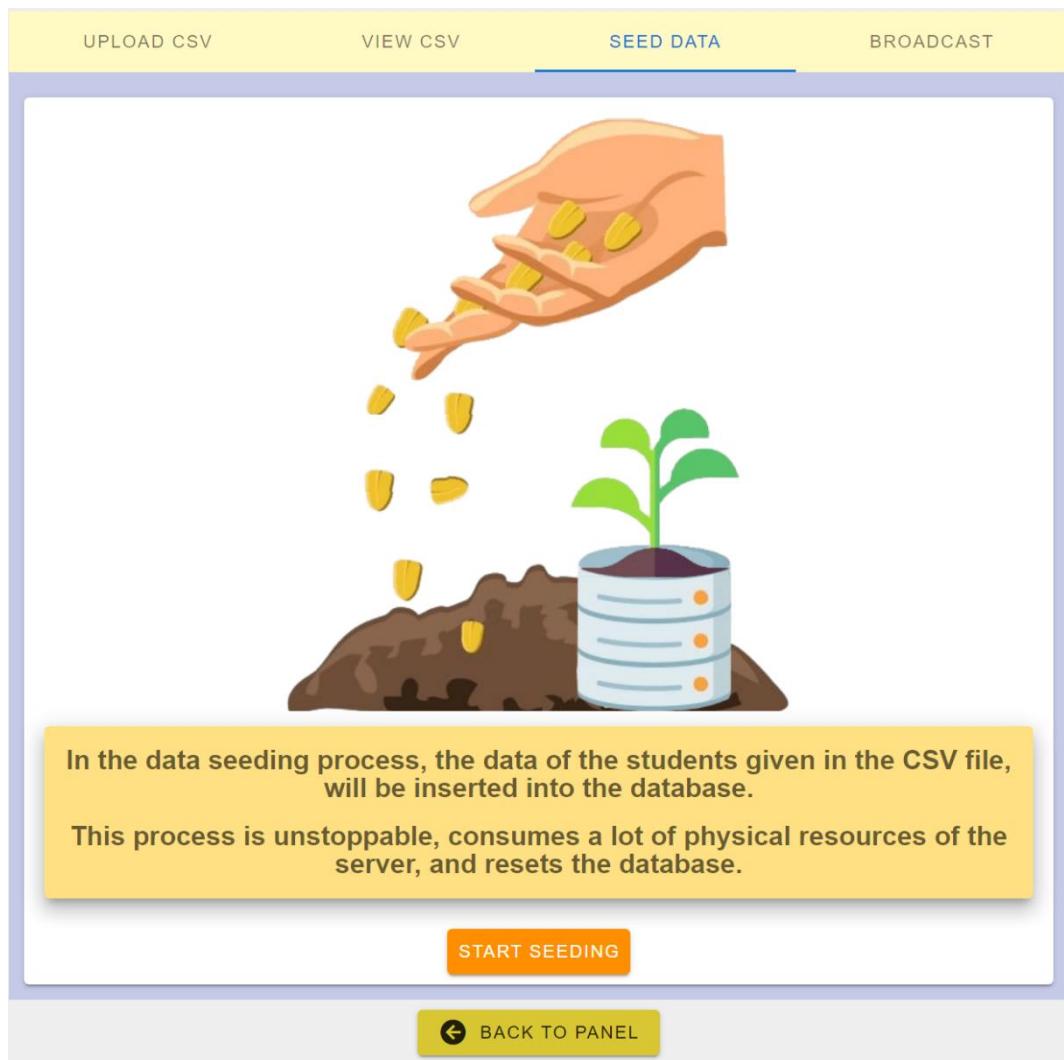


Fig: admin can seed data. Which means, data from that CSV file will be written into database. This job will be done by/on web server only.

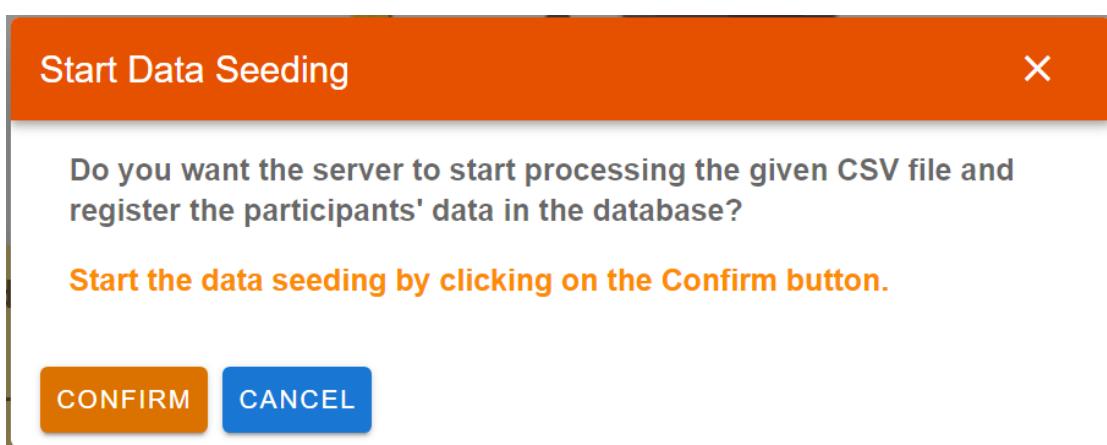


Fig: confirmation from admin to start data seeding into database. Whenever data seeding process will be ended, admin will be notified through SMS and email.

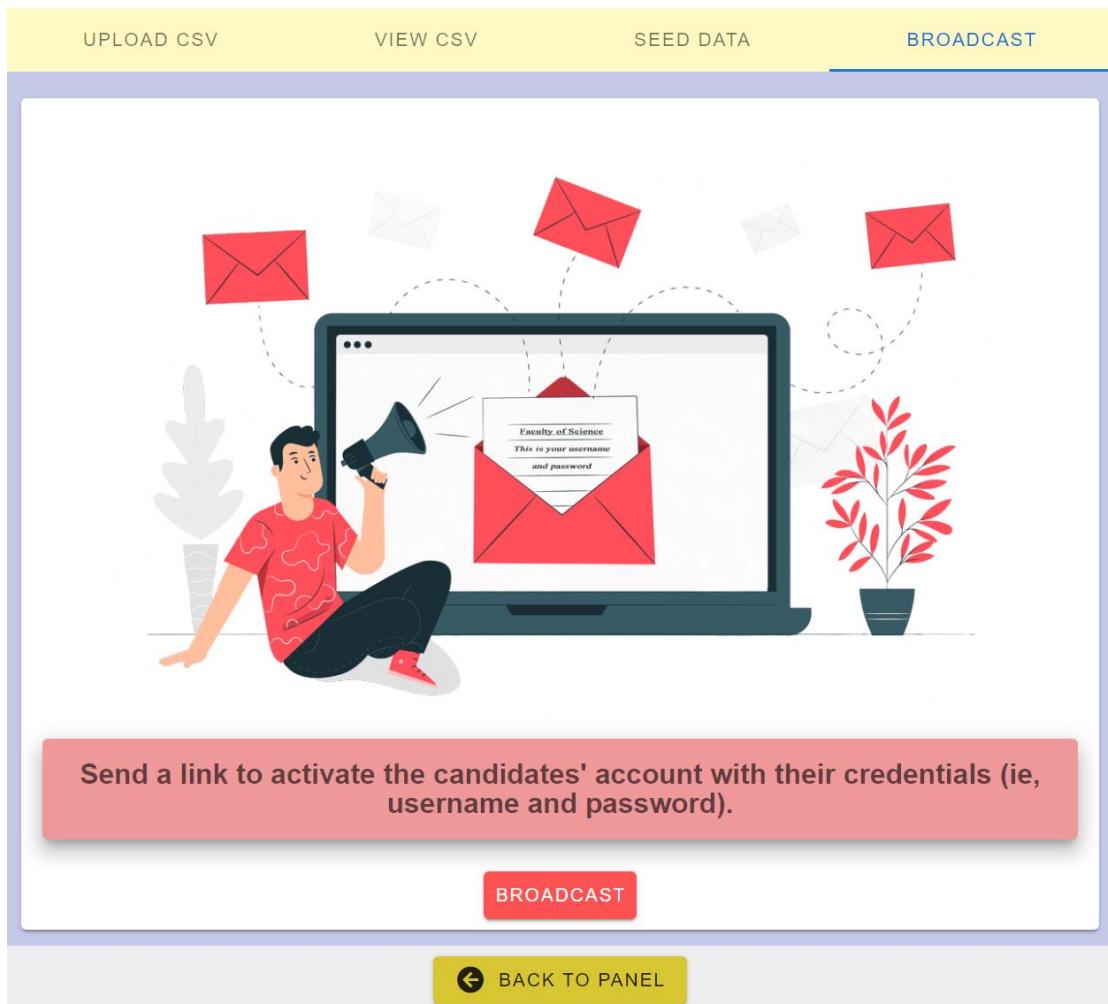


Fig: After seeding data successfully, admin can start sending credentials to candidates via emails and SMSs by clicking on this one button.

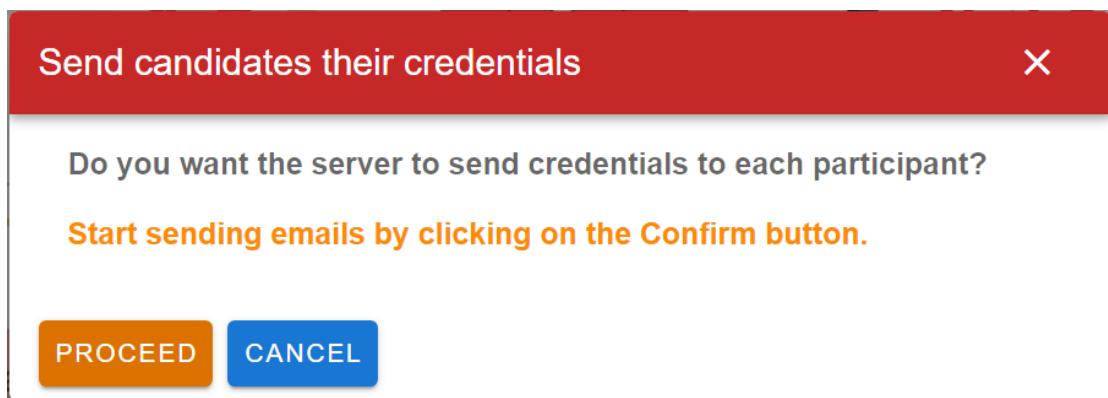


Fig: confirmation message for broadcasting email and SMS.

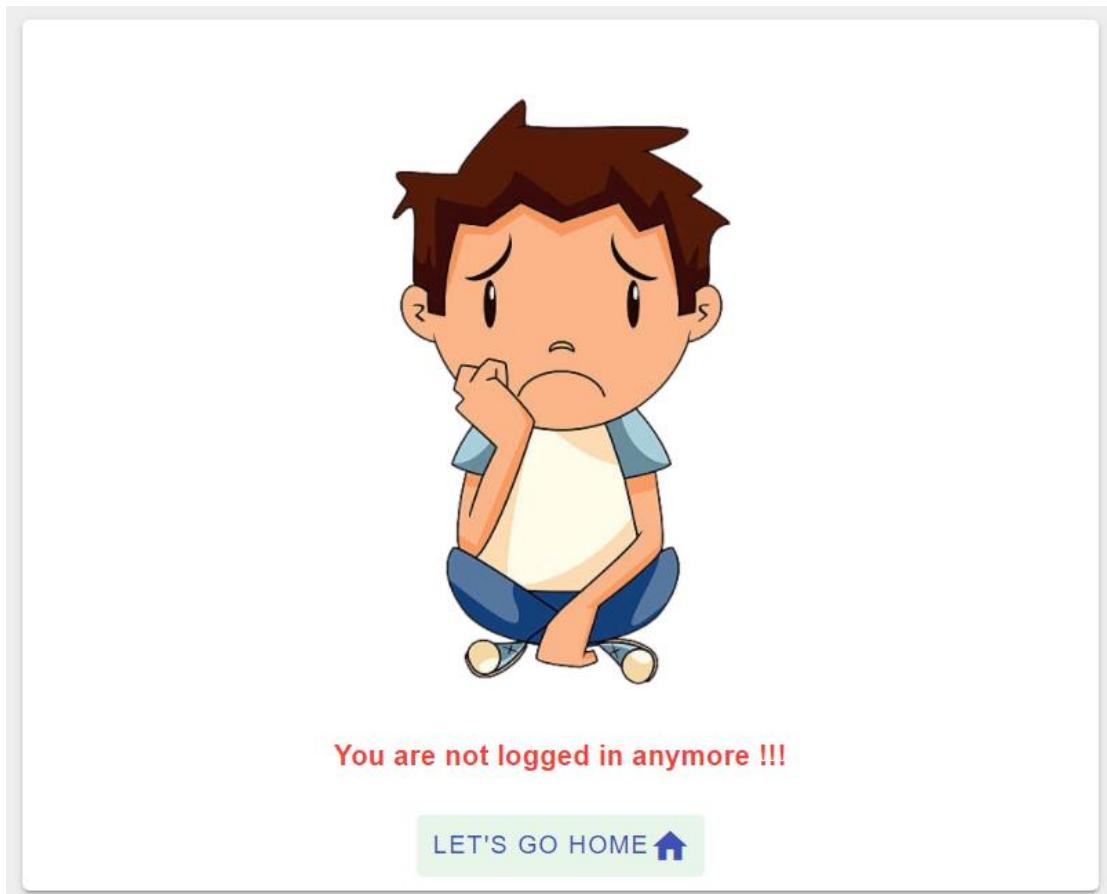


Fig: logout page, whenever admin or student logs out of their accounts, they will see this page. (here, if candidate or admin forgot to logout in previous session and comes back after an hour or later, they will be redirected to this page.)

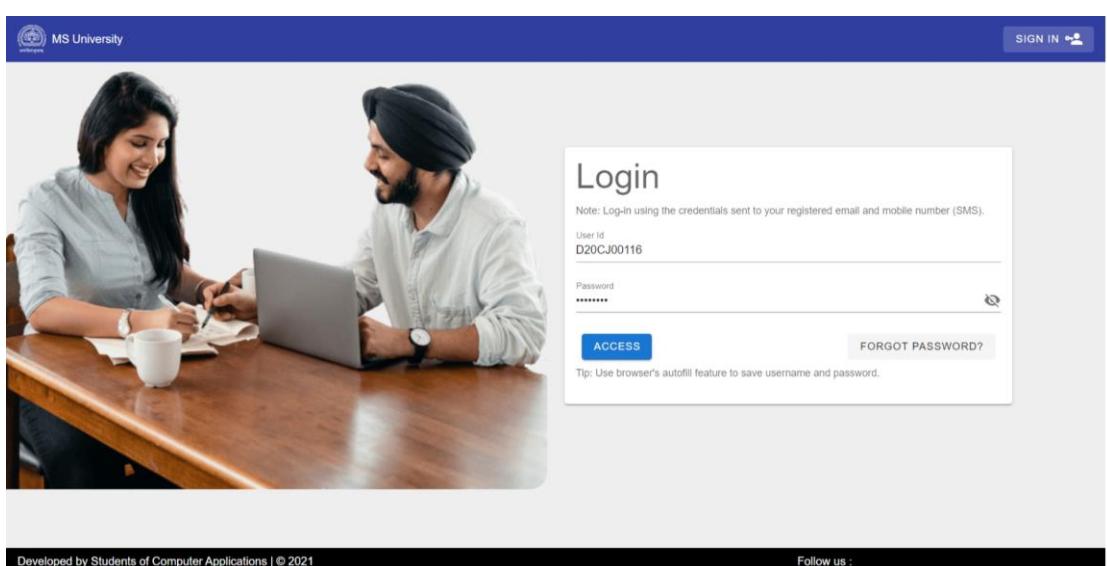


Fig: login page for candidates. (in desktop or with large display devices)

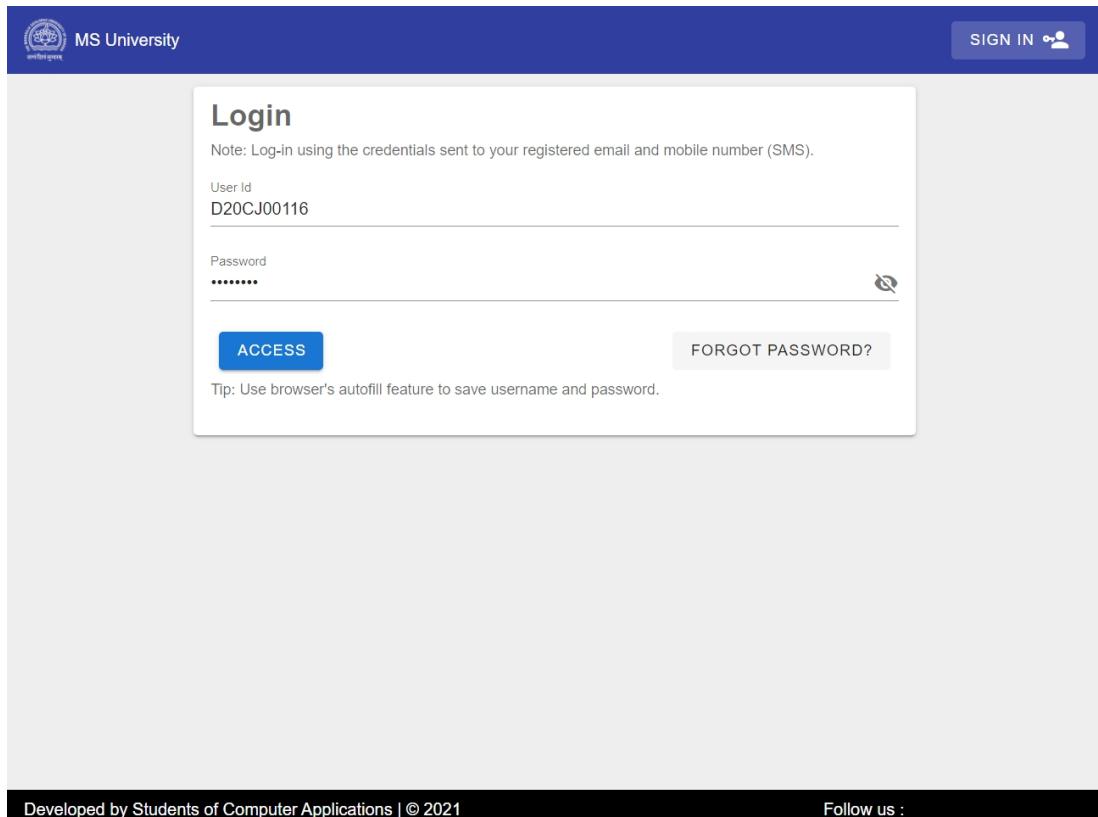


fig: login page for candidates. (tablet or smaller devices)

This login page will not be in use before admin broadcasts user-credentials to candidates.

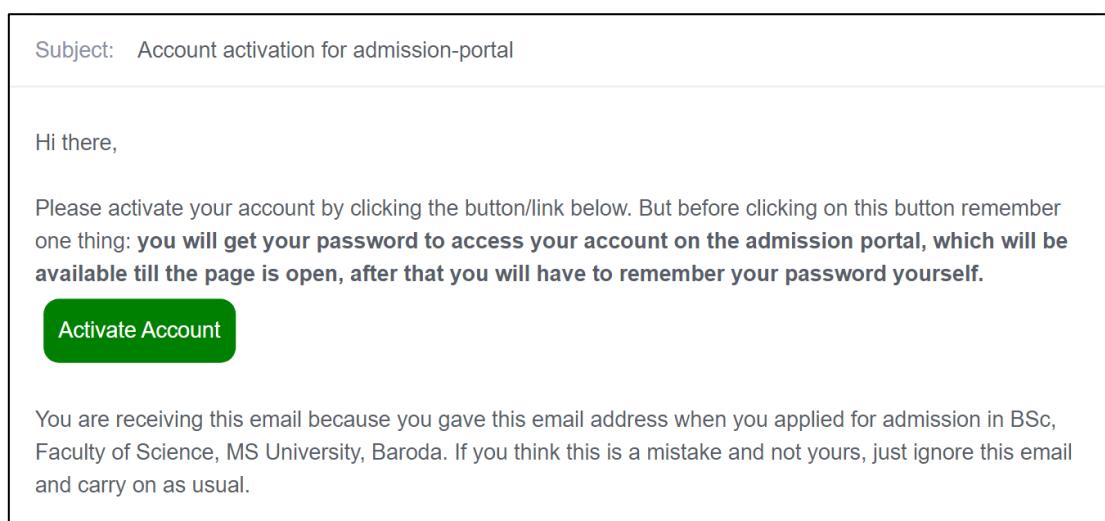


Fig: Email received by the candidate to activate his account. By clicking on that given button, the user will be redirected to page where he/she can see his/her username and system generated password.

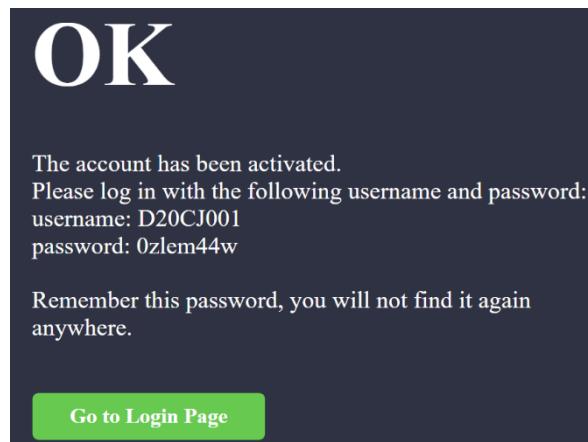


Fig: Candidates are looking at credentials to access the admission portal.

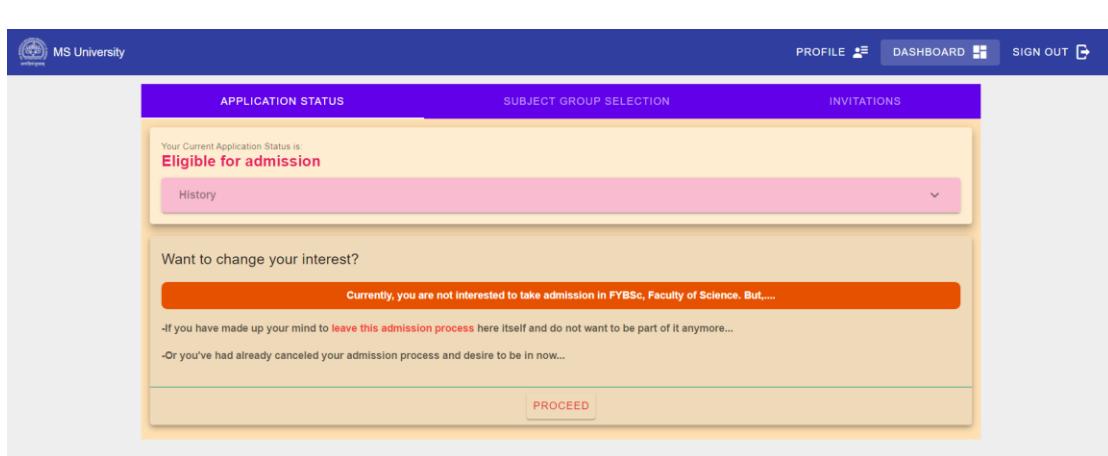


Fig: dashboard for student/candidates, from where they can view their current and previous application status. And can cancel their admission.

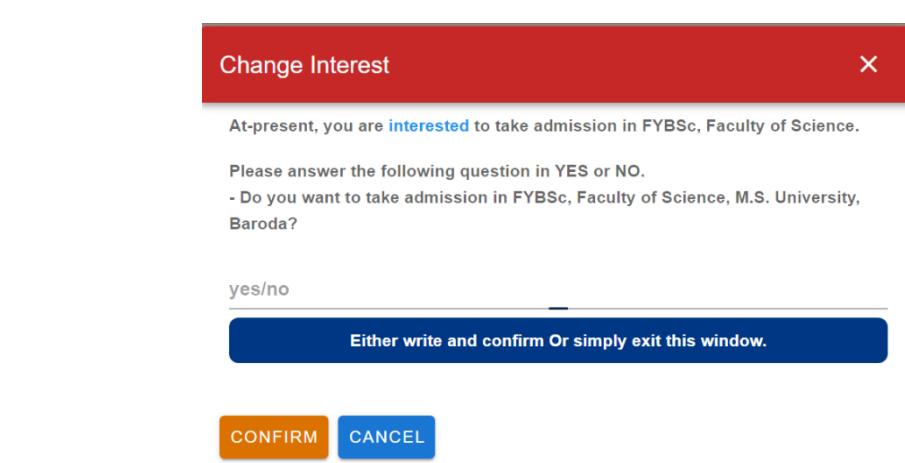


Fig: change interest. Which means, if candidate wants to leave admission process in amid of admission rounds, they have to write 'no' in the given text box. And if it was done by mistake or by someone else, they can get back to their position by writing 'yes' in textbox. And will be able to save their post in upcoming admission rounds.

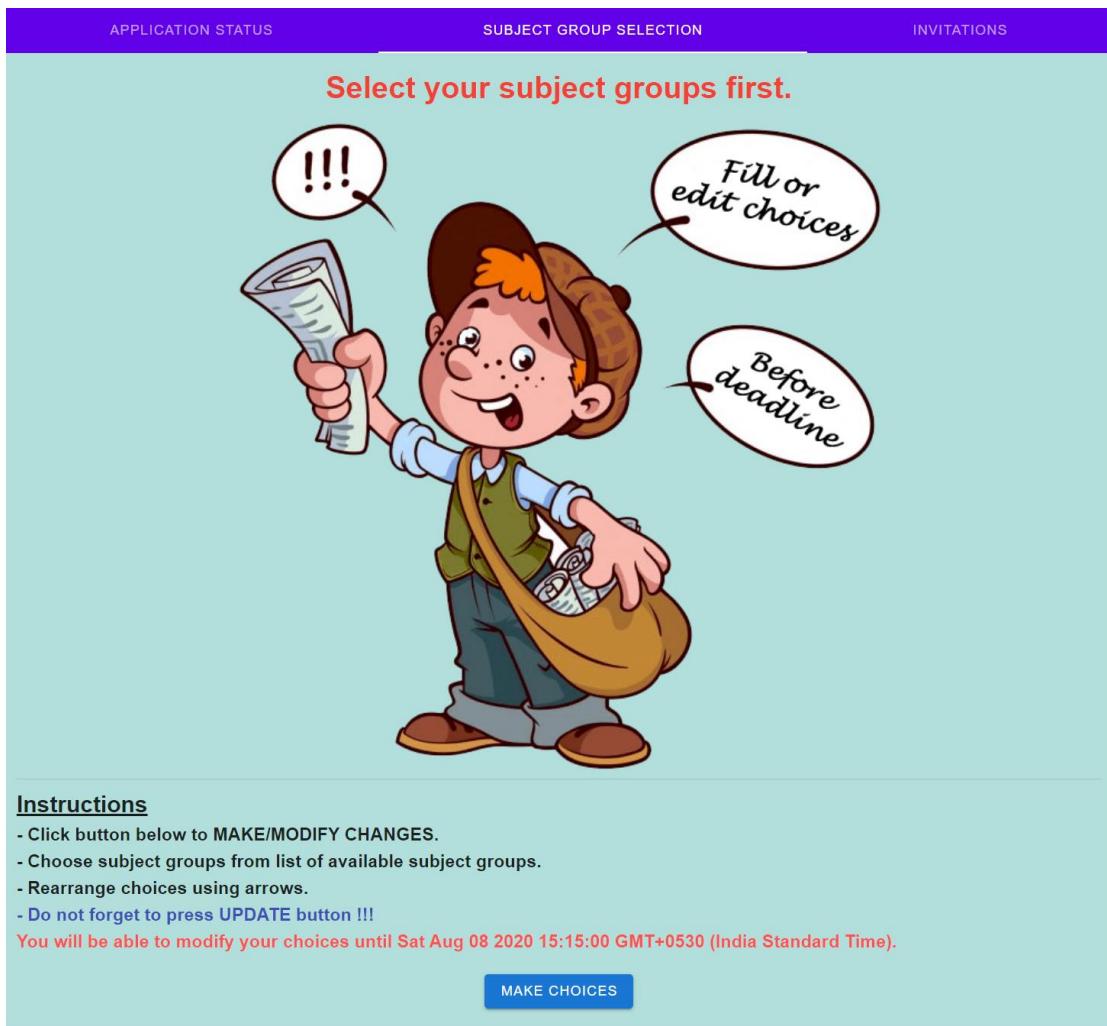


Fig: after activating their accounts, candidate needs to fill their choices here. (This is the case when choices of candidates are being taken or accepted by administrator and candidate has not yet selected any subject group.)

Fig: candidate selecting his/her favorite subject groups from list of available subject groups.

These available subject groups are based on candidate's HSC's subject group (either A or B or Both).

\*\* It is mandatory to select a minimum of 5 groups.

\* Change priorities using up and down arrows.

Subject Groups	Move Up	Move Down
<b>1. Geology Chemistry Physics</b>		
<b>2. Geography Chemistry Physics</b>		
<b>3. Geography Geology Physics</b>		
<b>4. Geography Botany Zoology</b>		

Chosen 4 out of 16 subject-groups.

I am fully aware that clicking on proceed/update button will set above subject groups as 'My Preference' along with priorities given by me.

UPDATE

Fig: Candidate must select more than or equal to five subject groups.

**13. Chemistry Geography Geology**

Chosen 14 out of 16 subject-groups.

I am fully aware that clicking on proceed/update button will set above subject groups as 'My Preference' along with priorities given by me.

UPDATE

Fig: Candidates will have to confirm their action of choice filling by ticking up that checkbox.

Using up and down arrows, candidate can change order or priority of choices.

APPLICATION STATUS SUBJECT GROUP SELECTION INVITATIONS

You will be able to modify your choices until Sat Aug 08 2020 15:15:00 GMT+0530 (India Standard Time).

Subject groups you have selected are as follows:

1. Botany Zoology Chemistry
2. Geography Geology Physics
3. Botany Zoology Geology
4. Botany Zoology Geography
5. Geology Botany Zoology
6. Geography Chemistry Physics
7. Geography Botany Zoology
8. Zoology Chemistry Physics
9. Zoology Botany Geography
10. Zoology Botany Chemistry
11. Botany Chemistry Physics

Wonderful,  
I still have  
a chance  
to make a  
change.

[MODIFY CHOICES](#)

Fig: candidate can modify their choices until admin has not locked the choice filling.

APPLICATION STATUS SUBJECT GROUP SELECTION INVITATIONS

You can no longer make or modify choices.

Subject groups you have selected are as follows:

1. Botany Zoology Chemistry
2. Geography Geology Physics
3. Botany Zoology Geology
4. Botany Zoology Geography
5. Geology Botany Zoology
6. Geography Chemistry Physics
7. Geography Botany Zoology
8. Zoology Chemistry Physics
9. Zoology Botany Geography
10. Zoology Botany Chemistry
11. Botany Chemistry Physics

Did I just  
secure these  
subject  
groups  
????

Fig: if admin locks down choice filling, candidates cannot make or modify their choices afterwards. Whatever candidate had selected remains there.

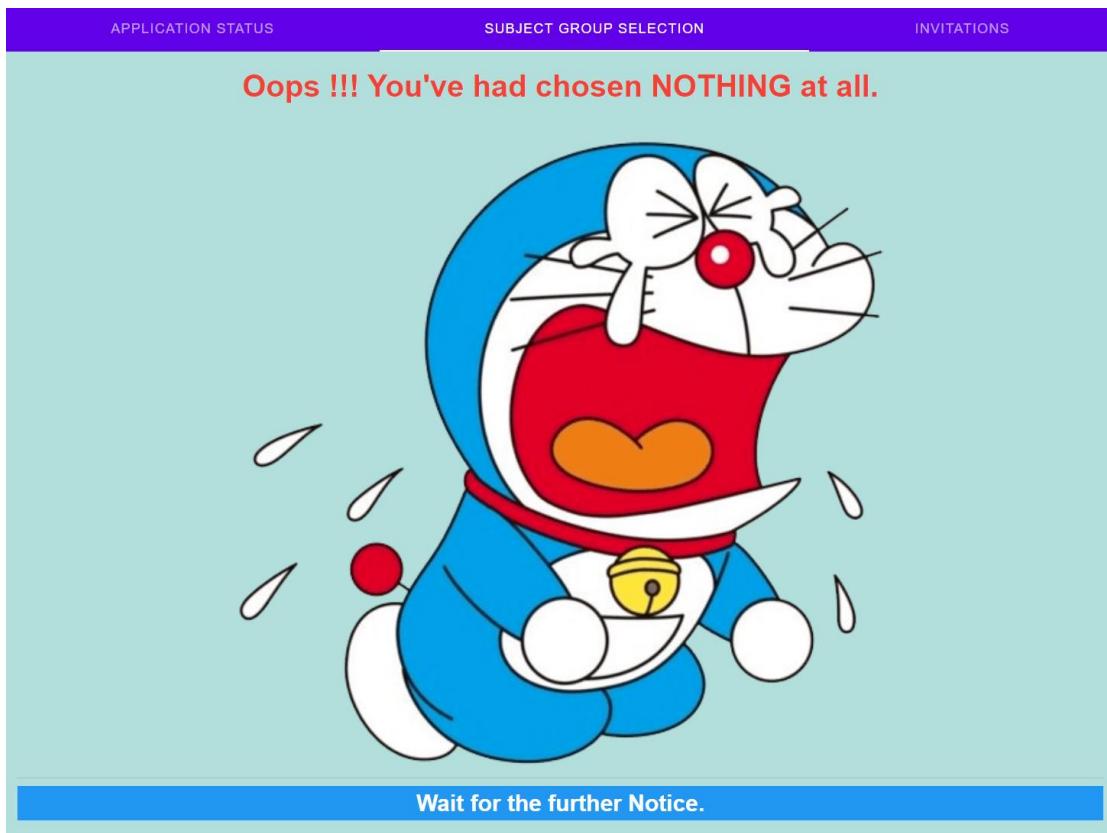


Fig: if candidate failed to choose subject groups, and administrator stops taking responses from candidates, this will be shown to candidate. Candidates will be helpless if they are getting this, and have to wait for further notice, if in case administrator starts taking choices from candidates again.

Fig: profile page of candidate, where he/she can see their personal data.

STUDENT	CONTACT	QUALIFICATION	PHYSICAL	CIVILIZED
<b>Candidate</b> E-mail ID: <a href="mailto:sharanamchotai@gmail.com">sharanamchotai@gmail.com</a> Phone Number: 8780361061				
<b>Candidate's Father</b> E-mail ID: - Phone Number: 7567470311				
<b>Candidate's Mother</b> E-mail ID: - Phone Number: 7567470311				
<b>Address</b> Present: 402, MY NEST I APARTMENT, SARDARPARA MAIN ROAD, GANDHARI WADI, JUNAGADH, JOSHIPURA, JUNAGADH, 362002 Permanent: 402, MY NEST I APARTMENT, SARDARPARA MAIN ROAD, GANDHARI WADI, JUNAGADH, JOSHIPURA, JUNAGADH, 362002				

STUDENT	CONTACT	QUALIFICATION	PHYSICAL	CIVILIZED
<b>HSC Examination</b> Examination Board: GUJARAT SECONDARY AND HIGHER SECONDARY EDUCATION BOARD, GANDHINAGAR State: Gujarat Country: India Year of Passing: 2020 Total attempt(s): 1 Seat Number: b137784 Completed HSC with B group.				
<b>Theory</b> Marks: 250 out of 300 . Percentage: 83.33				
<b>Practical</b> Marks in theory: 149 out of 150 Percentage: 99.33				

STUDENT	CONTACT	QUALIFICATION	PHYSICAL	CIVILIZED
Gender: Female				
<b>Sports</b> <ul style="list-style-type: none"> <li>Not a sports-person.</li> </ul>				

STUDENT	CONTACT	QUALIFICATION	PHYSICAL	CIVILIZED
Nationality: India Caste: General				
<b>Special Category</b> <ul style="list-style-type: none"> <li>Not Applicable.</li> </ul>				

Fig: other sections of Profile Page of Candidate.

MS University
DASHBOARD
SIGN OUT

**New Candidate**

**Personal**

Information

Form No	Username
---------	----------

Candidate

First Name	Middle Name	Last Name	Phone
Email			

Name as per HSC Marksheets

Gender	Nationality	DOB
Male	India	dd-mm-yyyy

Father

Name	Phone Number	Email
------	--------------	-------

Mother

Name	Phone Number	Email
------	--------------	-------

**Address**

Present address is permanent.

Present Address

Permanent Address

**Qualification**

Admission done.  Fee-payment done.

Details	City	State
Examination Board	VADODARA	Gujarat
Seat Number	Year Of Passing	Attempts
		1
	Group	
	A	

PCM Theory	Total Marks - PCM Theory	0% in PCM Theory
Obtained Marks - PCM Theory	Total Marks - PCM Theory	0

PCM Practical	Total Marks - PCM Practical	0% in PCM Practical
Obtained Marks - PCM Practical	Total Marks - PCM Practical	0

**Sports**

Candidate is sports-person.

**Civilized**

Categories

Caste

General

EWS  
 Handicapped  
 Defence Person  
 Kashmiri Migrant  
 Ward of MSU Employee

• Click on below button to update data.

**SAVE**

**BACK TO PANEL**

Fig: admin can add new candidate directly into database. If someone cannot get through in data seeding process.

Ward of MSU Employee

- first name is required
- middle name is required
- last name is required
- email is required
- phone number is required
- username is required

**UPDATE**

Fig: Displaying errors while adding new candidate into system. (validations)

MS University

DASHBOARD SIGN OUT

### Manage Candidates

Search by Form No

Username	Form No	name	Activated	Email	Phone	View	Send mail	Modify	Remove
007	11525	sharanam maheshbhai chotai	Yes	sharanamchotai@gmail.com	1234567890	<b>VIEW</b>	<b>ACTIVATE</b>	<b>EDIT</b>	<b>DELETE</b>
D20CJ0002922	14062	KEYA JAYESHKUMAR SHAH SHAH	Yes	bawama130@threep.com	6353195124	<b>VIEW</b>	<b>ACTIVATE</b>	<b>EDIT</b>	<b>DELETE</b>
D20CJ0011687	14080	NAISARGI VINODBHAI DUDHAT	Yes	sharanamchotai@gmail.com	8780361061	<b>VIEW</b>	<b>ACTIVATE</b>	<b>EDIT</b>	<b>DELETE</b>

**1**

**BACK TO PANEL**

Developed by Students of Computer Applications | © 2021

Follow us :

Fig: admin can manage candidates. Which means, admin can view details of individual-candidate. From here, if candidate missed to receive activation-email, admin can send email again. If candidate needs to correct their personal details and he/she approaches admin, then admin can edit candidate's details. And admin can even remove candidate from system/database.

Here, admin has facility to search participants through their unique form no.

MS University

DASHBOARD

SIGN OUT

View Candidate

Personal

Information

Form No: 11525

Username: 007

Verified Candidate

Candidate

First Name: sharanam

Middle Name: mathashish

Last Name: chotali

Phone: 1234567890

Email: sharanamchotali@gmail.com

Name as per HSC Marksheets: 1234567890

Father

Name: 1234567890

Phone Number: 1234567890

Email: s@i.com

Mother

Name: 1234567890

Phone Number: 1234567890

Email: e@i.com

Application Status

- Eligible for admission
- Not admitted in Admission Round 14 group A
- Not admitted in Admission Round 14 group B
- Not admitted in Admission Round 15 group A
- Not admitted in Admission Round 15 group B
- Not admitted in Admission Round 16 group A
- Not admitted in Admission Round 16 group B
- Called for round 17 group A
- Called for round 17 group B
- Accepted invitation for round 17
- Called for round 1 group A
- Called for round 1 group B
- Called for round 2 group B
- Called for round 2 group A
- Accepted invitation for round 2
- Called for round 3 group A
- Called for round 3 group B
- Called for round 4 group B
- Called for round 4 group A

Filled Choices

- 4
- 2
- 3
- 1
- 5

Address

Present Address: tere ghar ke baaju me Permanent Address: tere ghar ke baaju me

Qualification

Admission done.

Fee-payment done.

Details

Examination Board: GSWEWB

State: Gujarat

City: VADODARA

Country: India

Seat Number: b-69 />

Year Of Passing: 2020

Attempts: 1

Group: AB

PCM Theory

Obtained Marks: 10

Total Marks: 15

69.66666666666666% in PCM Theory

PCM Practical

Obtained Marks: 10

Total Marks: 15

66.66666666666666% in PCM Practical

PCB Theory

Obtained Marks: 10

Total Marks: 15

69.66666666666666% in PCB Theory

PCB Practical

Obtained Marks: 10

Total Marks: 15

66.66666666666666% in PCB Practical

Sports

Candidate is sports-person.

State Level

Participated in state level.

Won medal in state level.

National Level

Participated in national level.

Won medal in national level.

International Level

Participated in international level.

Won medal in international level.

Civilized

Categories

Caste: Scheduled Castes

EWS

Handicapped

Defence Person

Kashmiri Migrant

Ward of MSU Employee

GO BACK

Downloaded by Shalabh of Computer Applications © 2024

GATE 2024

Fig: admin is viewing individual candidate's record. Here, everything looks like form-view for adding and updating candidate's details, but it is in read only format.

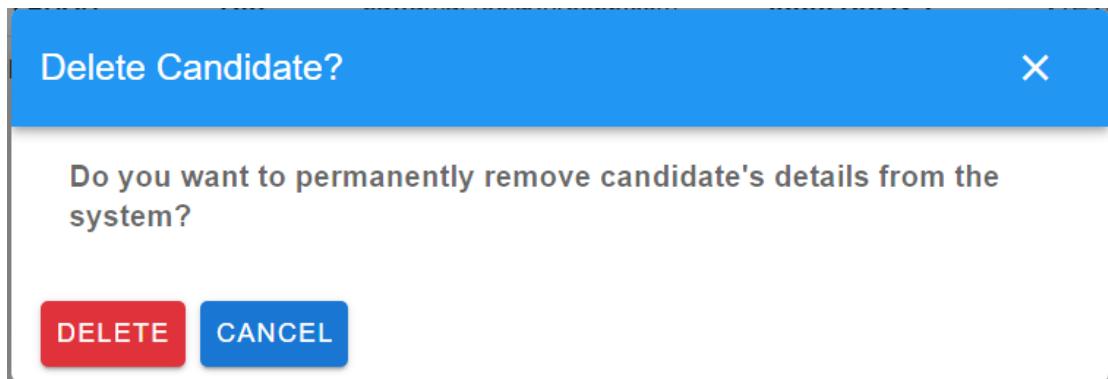


Fig: if in the case, administrator wants to remove a candidate, then this dialog box confirms the action.

---

Select any of the following filter to fetch list.								
Caste	Group	State						
All	A	Gujarat						
Merit No Form No Name Marks Total Percentage Caste Gender								
1	14062	KEYA JAYESHKUMAR SHAH SHAH	227	300	75.67	GEN	Female	
2	11525	sharanam maheshbhai chotai	10	15	69.67	SC	Male	

[PRINT](#)  
[BACK TO PANEL](#)

Fig: generate merit list; here, admin can select various parameters from list above and print them.

---

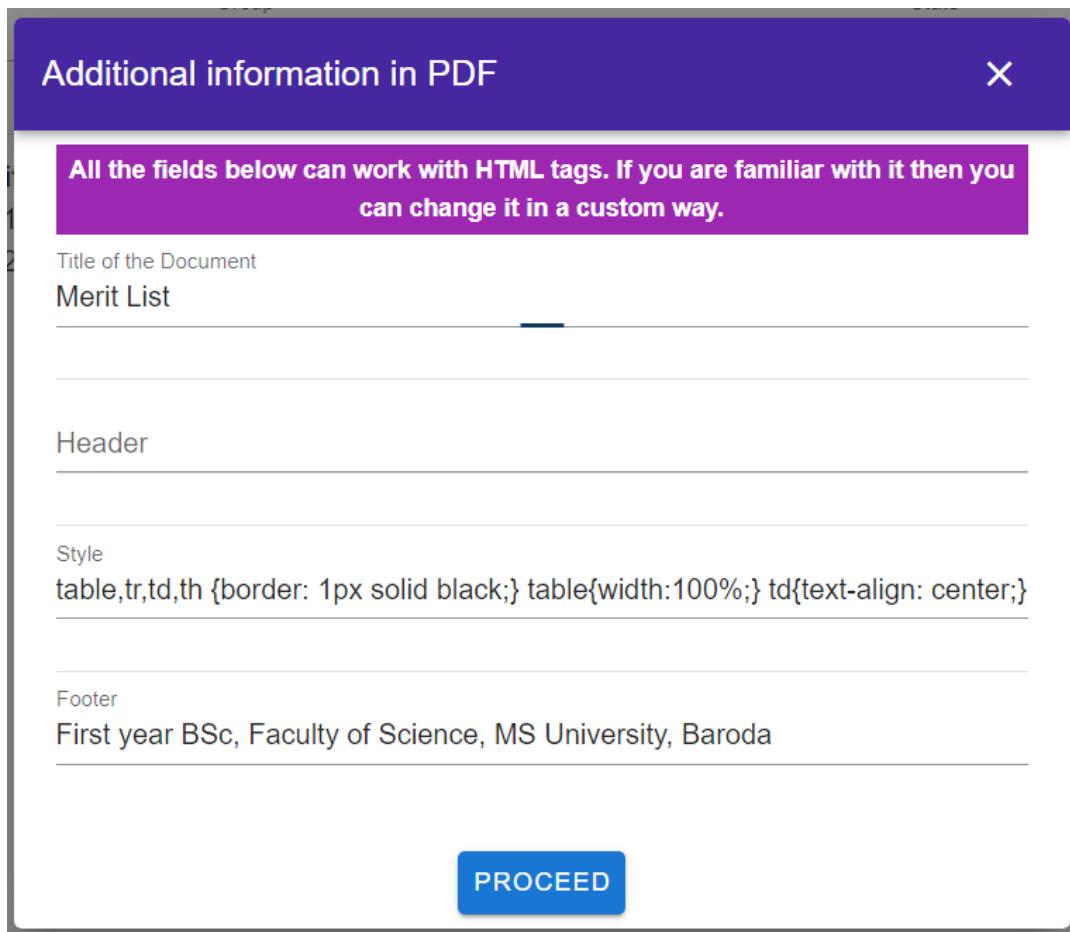


Fig.: exporting merit list into pdf dialog box. It is very flexible, with it, admin can add additional information into PDF. Refer handbook if you are interested to know more.

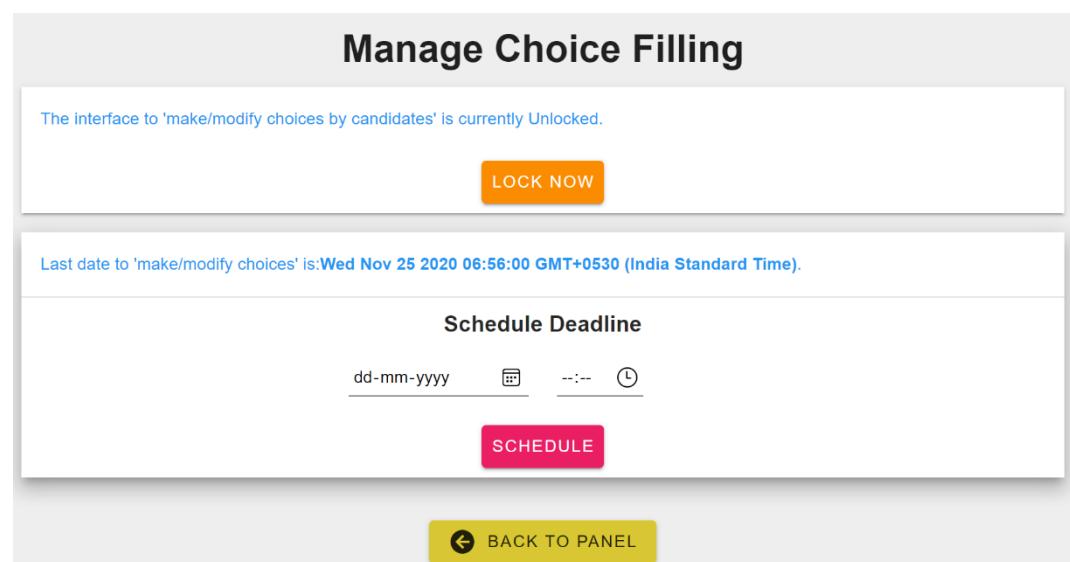


Fig: manage choice filling page. Here, admin can lock or unlock (accept or deny) choices from candidates immediately, by clicking orange button. If admin wants to make deadline for it, he can choose date and time for that, which can be seen from candidate's dashboard, too.

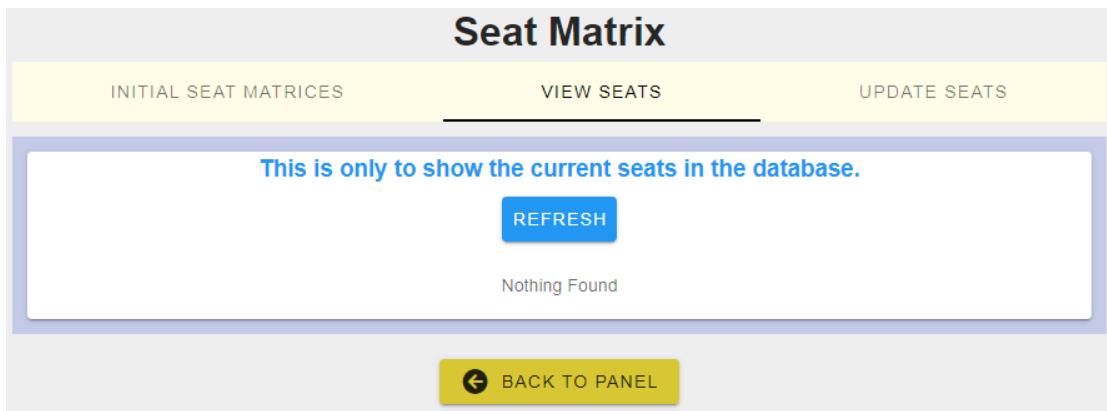


Fig: admin tries to view seat matrix, before uploading them.

Seat Matrix											
INITIAL SEAT MATRICES				VIEW SEATS				UPDATE SEATS			
This is only to show the current seats in the database.											
REFRESH											
Nothing Found											
BACK TO PANEL											
Gujarat											
SrNo	Subject Groups	General  A	General  B	SEBC  A	SEBC  B	SC  A	SC  B	ST  A	ST  B	EWS  A	EWS  B
1	Botany Zoology Chemistry	Total 9, Current 9	Total 54, Current 54	Total 2, Current 2	Total 4, Current 4	Total 5, Current 5	Total 2, Current 2	Total 6, Current 6	Total 4, Current 4	Total 0, Current 0	Total 0, Current 0
2	Botany Zoology Geography	Total 41, Current 41	Total 52, Current 52	Total 2, Current 2	Total 4, Current 4	Total 5, Current 5	Total 2, Current 2	Total 6, Current 6	Total 4, Current 4	Total 0, Current 0	Total 0, Current 0
3	Botany Zoology Geology	Total 35, Current 35	Total 28, Current 28	Total 2, Current 2	Total 4, Current 4	Total 5, Current 5	Total 2, Current 2	Total 6, Current 6	Total 4, Current 4	Total 0, Current 0	Total 0, Current 0
4	Botany Chemistry Physics	Total 27, Current 27	Total 67, Current 67	Total 2, Current 2	Total 4, Current 4	Total 5, Current 5	Total 2, Current 2	Total 6, Current 6	Total 4, Current 4	Total 0, Current 0	Total 0, Current 0
5	Zoology Botany Chemistry	Total 5, Current 5	Total 71, Current 71	Total 2, Current 2	Total 4, Current 4	Total 5, Current 5	Total 2, Current 2	Total 6, Current 6	Total 4, Current 4	Total 0, Current 0	Total 0, Current 0
6	Zoology Botany Geography	Total 62, Current 62	Total 57, Current 57	Total 78, Current 78	Total 63, Current 63	Total 13, Current 13	Total 17, Current 17	Total 60, Current 60	Total 63, Current 63	Total 0, Current 0	Total 0, Current 0
7	Zoology Botany Geology	Total 60, Current 60	Total 55, Current 55	Total 90, Current 90	Total 77, Current 77	Total 77, Current 77	Total 6, Current 6	Total 4, Current 4	Total 77, Current 77	Total 0, Current 0	Total 0, Current 0
8	Zoology Chemistry Physics	Total 89, Current 89	Total 1, Current 1	Total 57, Current 57	Total 40, Current 40	Total 71, Current 71	Total 95, Current 95	Total 41, Current 41	Total 40, Current 40	Total 0, Current 0	Total 0, Current 0
9	Chemistry Botany Zoology	Total 88, Current 88	Total 96, Current 96	Total 92, Current 92	Total 83, Current 83	Total 38, Current 38	Total 60, Current 60	Total 52, Current 52	Total 80, Current 83	Total 0, Current 0	Total 0, Current 0
10	Chemistry Geography Geology	Total 10, Current 10	Total 66, Current 66	Total 63, Current 63	Total 92, Current 92	Total 83, Current 83	Total 45, Current 45	Total 83, Current 83	Total 92, Current 92	Total 0, Current 0	Total 0, Current 0
11	Geography Botany Zoology	Total 20, Current 20	Total 15, Current 15	Total 18, Current 18	Total 66, Current 66	Total 34, Current 34	Total 53, Current 53	Total 82, Current 82	Total 66, Current 66	Total 0, Current 0	Total 0, Current 0
12	Geology Botany Zoology	Total 45, Current 43	Total 96, Current 96	Total 40, Current 40	Total 99, Current 99	Total 60, Current 60	Total 81, Current 81	Total 26, Current 26	Total 98, Current 99	Total 0, Current 0	Total 0, Current 0
13	Chemistry Mathematics Physics	Total 84, Current 84	Total 99, Current 99	Total 83, Current 83	Total 6, Current 6	Total 24, Current 24	Total 100, Current 100	Total 38, Current 38	Total 6, Current 6	Total 1, Current 1	Total 1, Current 1
14	Chemistry Mathematics Statistics	Total 61, Current 61	Total 72, Current 72	Total 92, Current 92	Total 95, Current 95	Total 33, Current 33	Total 58, Current 58	Total 35, Current 35	Total 95, Current 95	Total 2, Current 2	Total 2, Current 2
15	Mathematics Chemistry Physics	Total 100, Current 100	Total 90, Current 90	Total 66, Current 66	Total 60, Current 60	Total 73, Current 73	Total 52, Current 52	Total 89, Current 89	Total 60, Current 60	Total 3, Current 3	Total 3, Current 3
16	Mathematics Chemistry Statistics	Total 3, Current 3	Total 37, Current 37	Total 99, Current 99	Total 45, Current 45	Total 22, Current 22	Total 7, Current 7	Total 88, Current 88	Total 45, Current 45	Total 3, Current 3	Total 3, Current 3
17	Mathematics Physics Statistics	Total 51, Current 51	Total 54, Current 54	Total 6, Current 6	Total 53, Current 53	Total 16, Current 16	Total 30, Current 30	Total 84, Current 84	Total 53, Current 53	Total 6, Current 6	Total 6, Current 6
18	Physics Mathematics Chemistry	Total 63, Current 63	Total 27, Current 27	Total 95, Current 95	Total 81, Current 81	Total 20, Current 20	Total 63, Current 63	Total 81, Current 81	Total 81, Current 81	Total 4, Current 4	Total 4, Current 4
19	Physics Mathematics Statistics	Total 76, Current 76	Total 93, Current 93	Total 60, Current 60	Total 100, Current 100	Total 38, Current 38	Total 12, Current 12	Total 46, Current 46	Total 100, Current 100	Total 5, Current 5	Total 5, Current 5
20	Statistics Mathematics Chemistry	Total 90, Current 90	Total 43, Current 43	Total 45, Current 45	Total 58, Current 58	Total 90, Current 90	Total 60, Current 60	Total 16, Current 16	Total 58, Current 58	Total 1, Current 1	Total 1, Current 1
21	Statistics Mathematics Physics	Total 32, Current 32	Total 34, Current 34	Total 53, Current 53	Total 52, Current 52	Total 57, Current 57	Total 26, Current 26	Total 40, Current 40	Total 52, Current 52	Total 3, Current 3	Total 3, Current 3
22	Statistics Mathematics Geology	Total 76, Current 76	Total 3, Current 3	Total 81, Current 81	Total 7, Current 7	Total 17, Current 17	Total 79, Current 79	Total 22, Current 22	Total 7, Current 7	Total 4, Current 4	Total 4, Current 4
23	Geography Geology Mathematics	Total 83, Current 83	Total 81, Current 81	Total 100, Current 100	Total 30, Current 30	Total 53, Current 53	Total 83, Current 83	Total 92, Current 92	Total 30, Current 30	Total 2, Current 2	Total 2, Current 2
24	Geography Geology Statistics	Total 60, Current 60	Total 43, Current 43	Total 58, Current 58	Total 63, Current 63	Total 27, Current 27	Total 23, Current 23	Total 8, Current 8	Total 63, Current 63	Total 2, Current 2	Total 2, Current 2
25	Geography Geology Physics	Total 26, Current 26	Total 29, Current 29	Total 52, Current 52	Total 12, Current 12	Total 20, Current 20	Total 66, Current 66	Total 75, Current 75	Total 12, Current 12	Total 1, Current 1	Total 1, Current 1

Fig: admin viewing uploaded seat matrix after uploading correctly.

This process will reset the seat matrix already stored in the database.

Please upload appropriate seat matrices as shown in the example via CSV files; A small mistake can cause a lot of trouble.

Vadodara  
Attach CSV file containing matrix for VADODARA.

Example,

Subject Group	General A	General B	SC A	SC B	ST A	ST B	SEBC A	SEBC B	EWS A	EWS B
Botany Zoology Chemistry	9	54	5	2	6	4	2	4	0	0
####	#	#	#	#	#	#	#	#	#	#

Gujarat  
Attach CSV file containing matrix for GUJARAT.

Example,

Subject Group	General A	General B	SC A	SC B	ST A	ST B	SEBC A	SEBC B	EWS A	EWS B
Botany Zoology Chemistry	9	54	5	2	6	4	2	4	0	0
####	#	#	#	#	#	#	#	#	#	#

India  
Attach CSV file containing matrix for INDIA.

Example,

Subject Group	General A	General B	SC A	SC B	ST A	ST B	SEBC A	SEBC B	EWS A	EWS B
Botany Zoology Chemistry	9	54	5	2	6	4	2	4	0	0
####	#	#	#	#	#	#	#	#	#	#

Sub-Categories  
Attach CSV file containing seat matrix for Sub categories.

Principal Subject	VADODARA DEFENCEWARDS	VADODARA DIFFERENTLYABLED	VADODARA KASHMIR	GUJARAT
Botany				
####				
Total	7	34	7	2

Fig: admin uploading seat matrix files as per written in the handbook. Admin can also confirm columns' name in CSV files with examples given here.

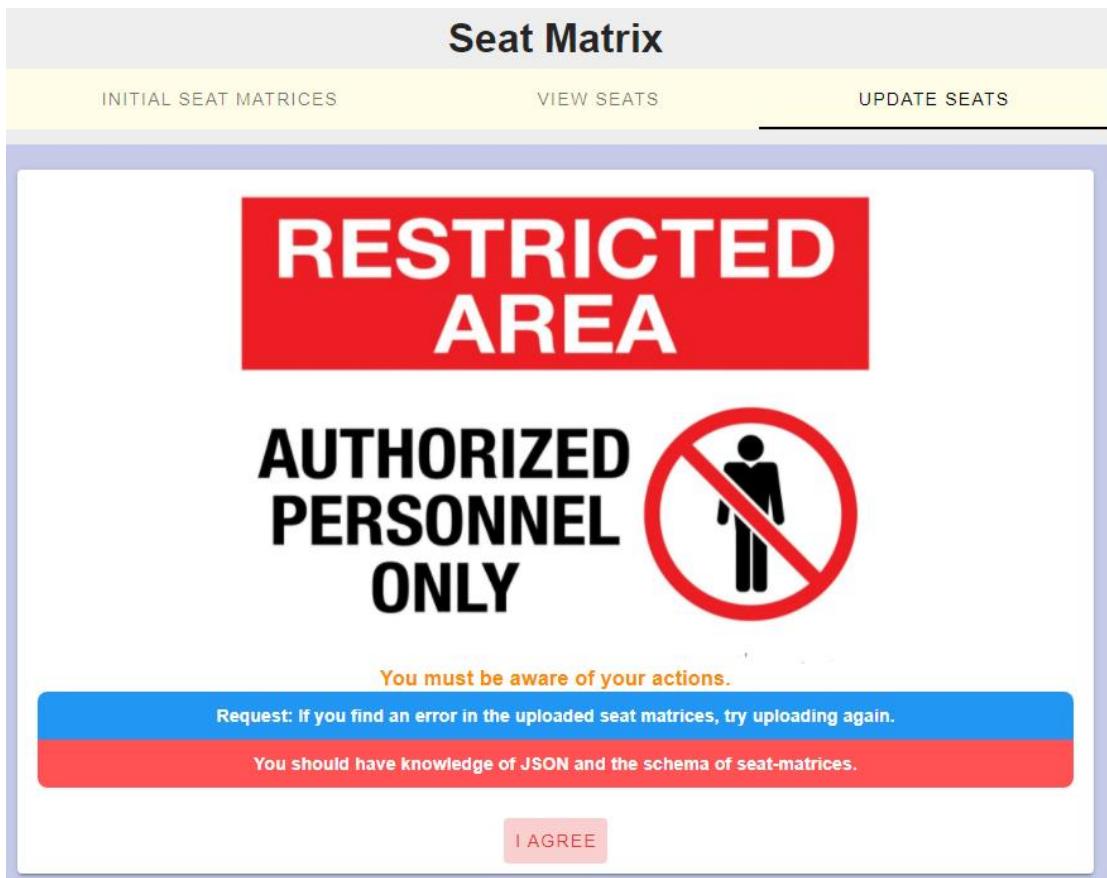
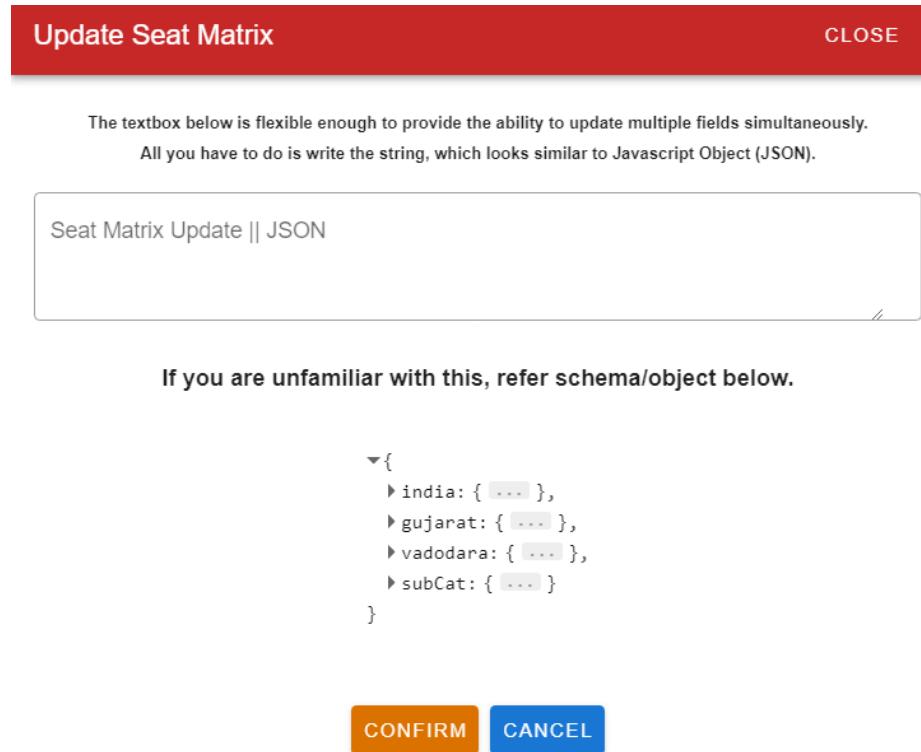


Fig: updating seat matrix. It is not easy to update them directly within database, as they are dynamic JavaScript object for that reason admin needs to write JSON for that.



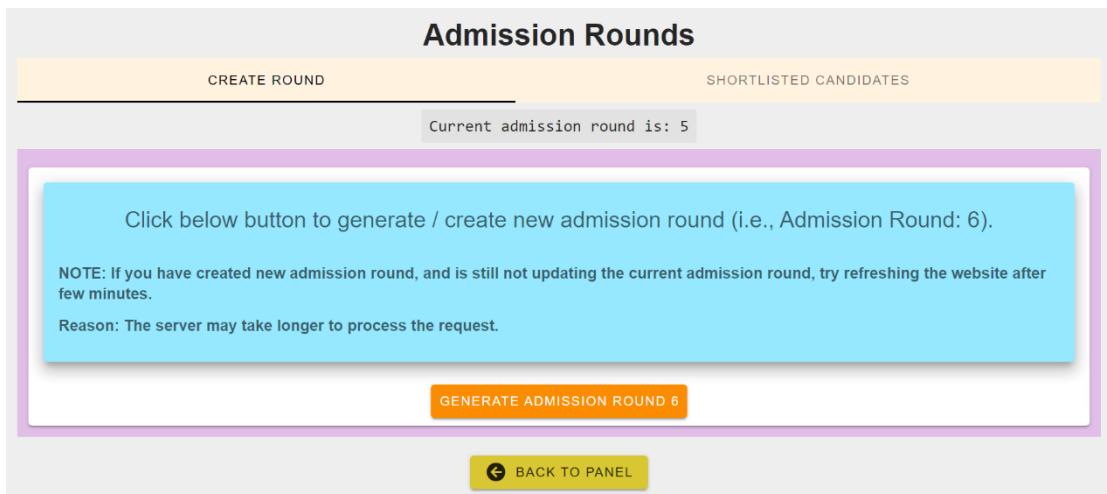


Fig: admin generating new admission round.

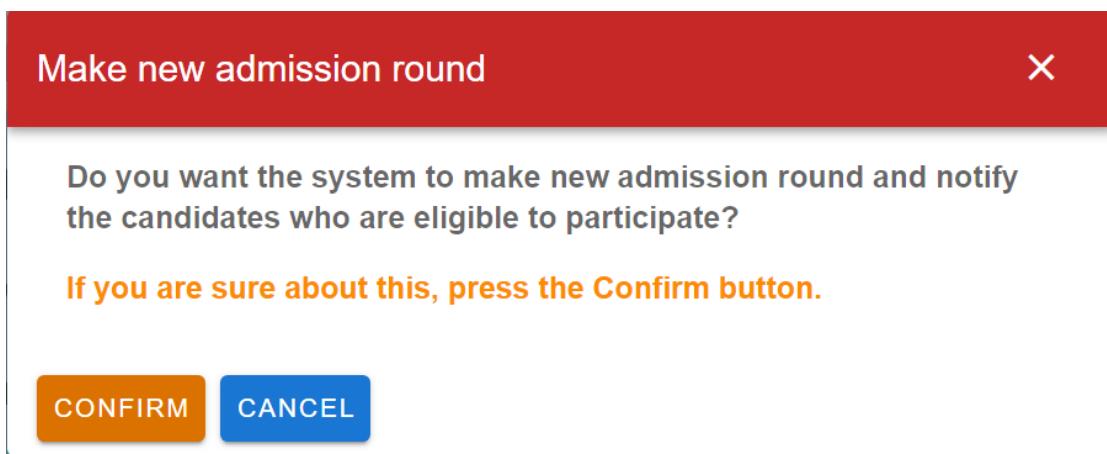


Fig. Dialog box is confirming administrator's request to generate new admission round.

In this process, candidates with sufficient merit rank will be notified that they are being called to appear in the current or newly created admission round.

(The administrator can see the list of those students in the list of shortlisted candidates.)

Admission Rounds																																																												
CREATE ROUND				SHORTLISTED CANDIDATES																																																								
Current admission round is: 1																																																												
You are currently viewing shortlisted candidates for admission round 1																																																												
<span style="border: 1px solid #ccc; border-radius: 50%; padding: 2px 5px; margin-right: 5px;">&lt;</span> <span style="border: 1px solid #0070C0; border-radius: 50%; padding: 2px 5px; color: white; background-color: #0070C0; text-decoration: none;">1</span> <span style="border: 1px solid #ccc; border-radius: 50%; padding: 2px 5px; margin-left: 5px;">&gt;</span>																																																												
<span style="background-color: #0070C0; color: white; padding: 2px 10px; border-radius: 5px; text-decoration: none;">EXPORT AS CSV</span>																																																												
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Sr No</th> <th>FormNo</th> <th>Name</th> <th>AllottedCategory</th> <th>DefaultCategory</th> <th>Location</th> <th>Group</th> <th>Handicapped</th> <th>DefenceWard</th> <th>KashmiriMigrant</th> <th>SubjectGroup</th> <th>E</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>14062</td> <td>SHAH KEYA JAYESHKUMAR</td> <td>general</td> <td>General</td> <td>vadodara</td> <td>A</td> <td>false</td> <td>false</td> <td>false</td> <td>chemistry mathematics statistics</td> <td>f</td> </tr> <tr> <td>2</td> <td>11525</td> <td>1234567890</td> <td>general</td> <td>Scheduled Castes</td> <td>vadodara</td> <td>A</td> <td>false</td> <td>false</td> <td>false</td> <td>botany chemistry physics</td> <td>t</td> </tr> <tr> <td>3</td> <td>11525</td> <td>1234567890</td> <td>general</td> <td>Scheduled Castes</td> <td>vadodara</td> <td>B</td> <td>false</td> <td>false</td> <td>false</td> <td>botany chemistry physics</td> <td>t</td> </tr> </tbody> </table>													Sr No	FormNo	Name	AllottedCategory	DefaultCategory	Location	Group	Handicapped	DefenceWard	KashmiriMigrant	SubjectGroup	E	1	14062	SHAH KEYA JAYESHKUMAR	general	General	vadodara	A	false	false	false	chemistry mathematics statistics	f	2	11525	1234567890	general	Scheduled Castes	vadodara	A	false	false	false	botany chemistry physics	t	3	11525	1234567890	general	Scheduled Castes	vadodara	B	false	false	false	botany chemistry physics	t
Sr No	FormNo	Name	AllottedCategory	DefaultCategory	Location	Group	Handicapped	DefenceWard	KashmiriMigrant	SubjectGroup	E																																																	
1	14062	SHAH KEYA JAYESHKUMAR	general	General	vadodara	A	false	false	false	chemistry mathematics statistics	f																																																	
2	11525	1234567890	general	Scheduled Castes	vadodara	A	false	false	false	botany chemistry physics	t																																																	
3	11525	1234567890	general	Scheduled Castes	vadodara	B	false	false	false	botany chemistry physics	t																																																	

Fig: Students shortlisted in the Admission Round (i.e., First). The administrator can export this list as a CSV file by clicking on that purple button.

NOTE: To make changes in "admission success" and "fees paid" columns, the administrator must download that CSV file. With those changes, the administrator must upload the same file to update status. (as shown below)

Update Status												
This is the example table (CSV) you got from the admission round module.												
<i>iMigrant</i>	<i>GotInKashmiriMigrant</i>	<i>DefenceWard</i>	<i>GotInDefenceWard</i>	<i>EWS</i>	<i>GotInEWS</i>	<i>ChoiceNumber</i>	<i>Betterment</i>	<i>AdmissionSuccess</i>	<i>FeePaid</i>			
SE	FALSE	FALSE	FALSE	FALSE	FALSE	1	FALSE	FALSE	FALSE			
Make changes into "AdmissionSuccess" and "FeePaid" columns and upload CSV file here.												
<div style="background-color: #f0f0f0; padding: 5px;"> <p>Update fees payment status</p> <p><input type="file"/> Fees Payment</p> <p>CSV file to update fees payment status.</p> <p style="text-align: center;"><b>UPLOAD</b></p> </div>												
<div style="background-color: #f0f0f0; padding: 5px;"> <p>Update admission status</p> <p><input type="file"/> Admission</p> <p>CSV file to update Admission Status.</p> <p style="text-align: center;"><b>UPLOAD</b></p> </div>												

Fig.: The administrator is updating the admission success status and fee payment status of the candidates after confirming from the digital university system.

# Forgot Password

You are going to trigger the system to send email containing link to reset password by submitting your user id.

User Id

PROCEED



Fig: forgot password feature for candidates. It sends link to registered email address.

Subject: Reset password

**Hi, ~~KEYA JAYESHKUMAR SHAH SHAH~~ with username ~~D20CJ900022~~**

Looks like a request was made to reset the password for Admission Portal of MSU. No problem! You can reset your password now using the lovely button below. But remember, This password-reset request is only valid for the next 1 hour, after which it will become inactive.

**Reset your password**

If you didn't want to reset your password, you can safely ignore this e-mail and carry on as usual.  
Thanks,  
Team ASP.

If you are having trouble with the above button, then copy and paste the URL below in your web browser.  
<http://localhost:5000/api/password/resetPassword/60884aaee0a764240cc50a42/eyJhbGciOiJIUzI1NlslnR5c0XAXVRCdATcEMFei5kNdmwePyAtBGyg>

Fig: email to reset password. (candidate's email box)

# Reset Password

Create new password to access the Admission Portal.

Password

Confirm Password

Submit

Fig: save password. Constraints: Password must be 8 characters long and should contain at least one Digit, Capital letter and Special character

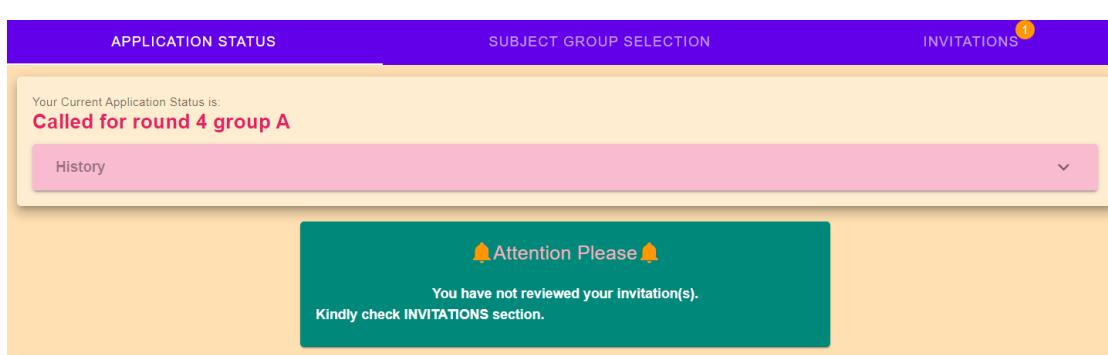


Fig: notifying candidate to review invitation on the dashboard. (using marquee effect and color toggling animation, moreover, badge on the notification section)

Email and SMS will be sent to each shortlisted candidate for same.

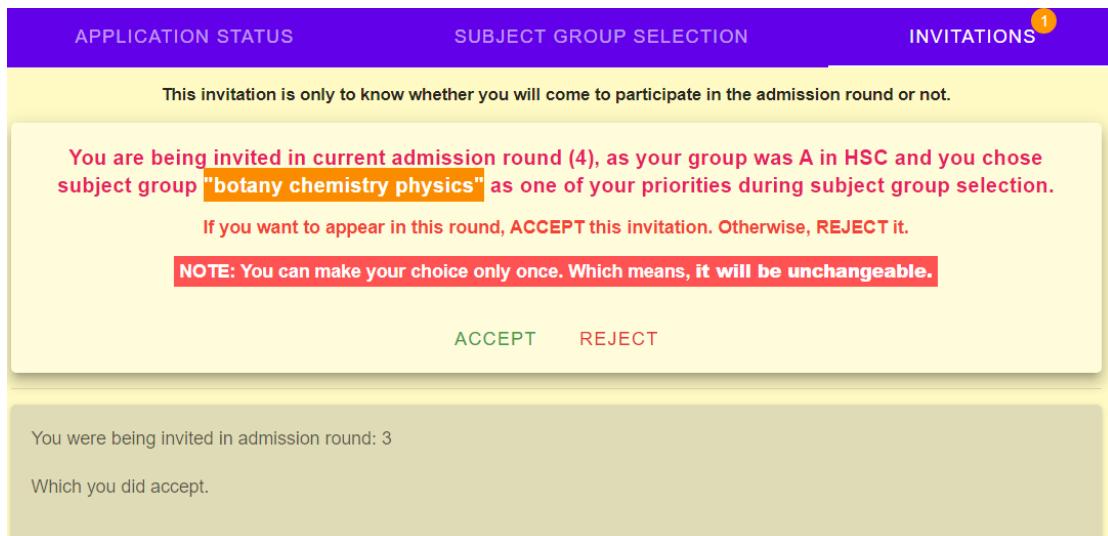


Fig: UI for candidate to accept or reject the invitation.

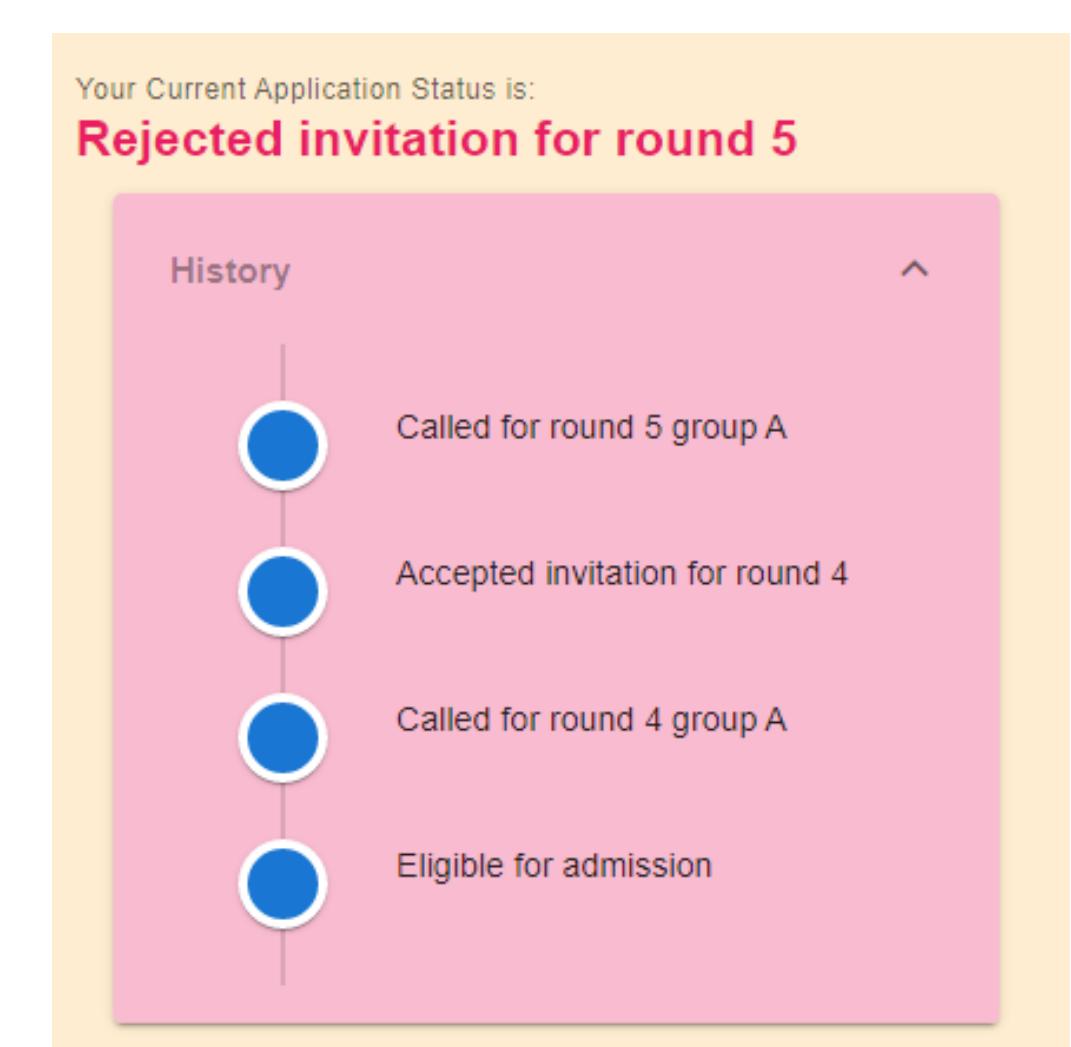


Fig: candidate's current application status and previous in history as a timeline.

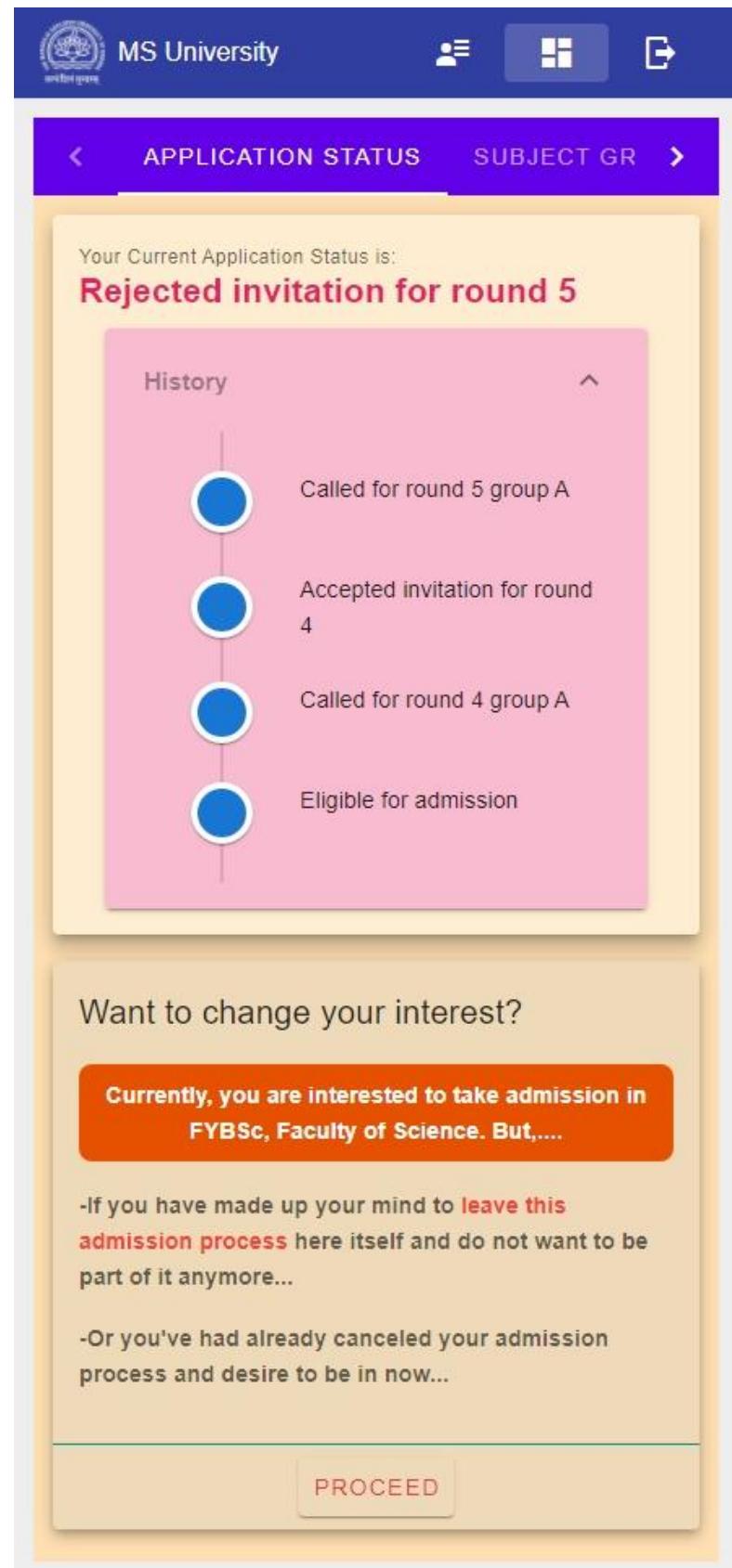
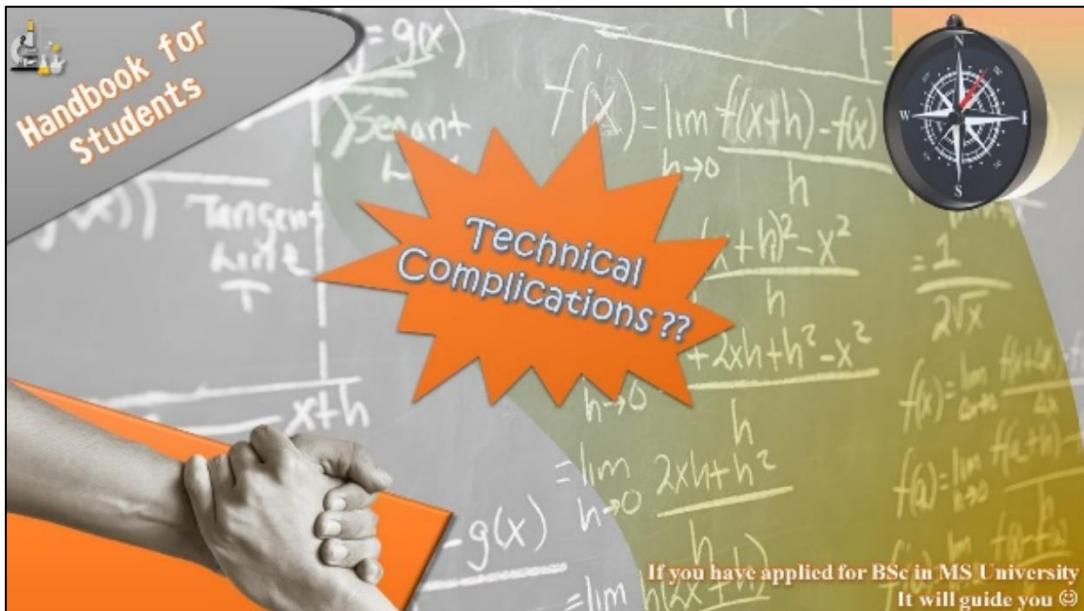


Fig: each component in whole website is responsive and adaptive. Therefore, it can be viewed on any device.

## **Handbooks:**

It was specified as a user story to provide necessary instructions for candidates and admins to understand how to do certain actions within the application. So, we made handbooks for them using MS Office PowerPoint.

### **Handbook for Students**



## Preface

- This handbook, is prepared by the Developers' Team of Admission Portal, provides students and/or visitors helpful technical flows and reference about Admission-Application of FYBSc.
- It includes an overview of the tasks which visitors and applicants can perform within system and thoroughly resolves doubts for operating individual practices.
- If you have any questions about policies & procedures that have not been specifically addressed in this handbook, please consult with someone who has a clear understanding of facts about the system or a particular subject.

## Index

- Get Started
- Not getting notifications?
- Subject Group Selection (Choice filling)
- Invitation for admission round
- Betterment
- Reset Password
- Change Interest



### Get Started

To access admission portal, you will need password, which will be sent to you via email & SMS by the administrator.

Subject: Account activation for admission-portal

Hi there,

Please activate your account by clicking the button/link below. But before clicking on this button remember one thing: you will get your password to access your account on the admission portal, which will be available till the page is open, after that you will have to remember your password yourself.

[Activate Account](#)

You are receiving this email because you gave this email address when you applied for admission in BSc, Faculty of Science, MS University, Baroda. If you think this is a mistake and not yours, just ignore this email and carry on as usual.

Fig.: Email to activate account sent by authorities/administrator.

### Not getting notifications?

- If the candidate does not receive an email to activate account and thinks, there is something wrong with his/her email account meanwhile others are getting their emails:
  1. Please check the spam folder in your email-box.
  2. Check your registered email address again, if it is mentioned incorrectly, contact administrator for correction as given in the guidelines.

Besides that, you can see 'announcements' on admission portal for latest updates from university at any time.



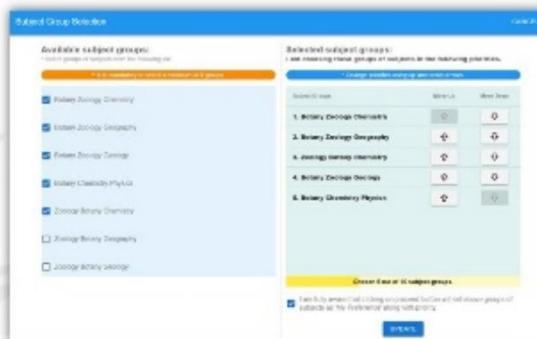
## Subject Group Selection (Choice filling)

- Subject group selection or 'choice filling' is a process of getting to know students' prior subjects.
- In this process, candidates select their preferred subject groups which are available to them based on HSC subject group (either A or B or both).
- Candidates will have to make their choice before the deadline arrives.



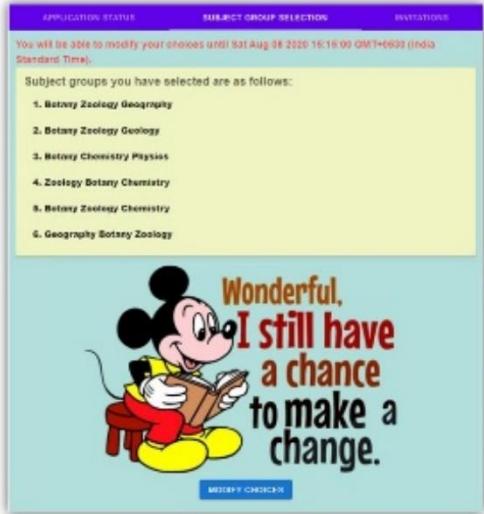
## Available subject groups / choices

- Here, you can see the available subject groups (to the left in the figure) which are based on your HSC subject group.
- On the right-hand side in the figure, the subject groups selected by the candidate can be seen. Whichever group holds the 1 position, will be considered the highest priority of the candidate.
- Candidates can change order of these subject groups using up & down arrows.
- The candidate must select at least 5 subject groups to ensure that he / she will get admission in any one of them.



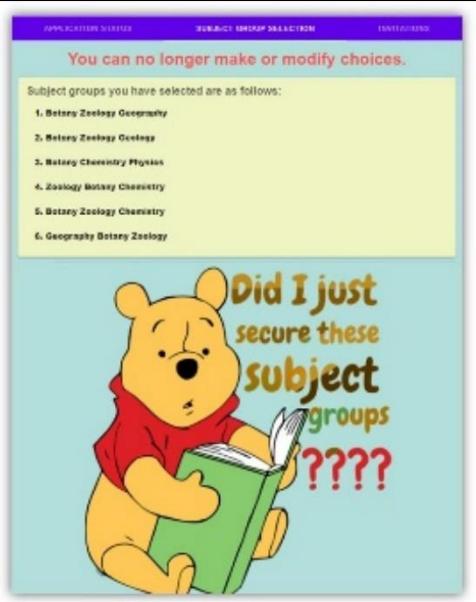
## Modifying chosen subject groups

- If deadline to modify changes in subject groups is not arrived yet, candidate can modify those choices several times.
- When is the deadline, can be seen at the top (i.e., top of the figure).



## Securing subject groups

- If deadline comes, candidates cannot make or modify their choices.
  - So,
    - be wise,
    - choose prior subjects and secure them before deadline arrives.



## Invitation to participate in admission round

- In admission rounds, candidates will be called and admitted according to their performance/percentage.
  - For that, candidate must have chosen prior subject groups.
  - Invitation to participate in any given admission round will be sent to candidate via e-mail & SMS, to which candidate can respond in admission portal (website).

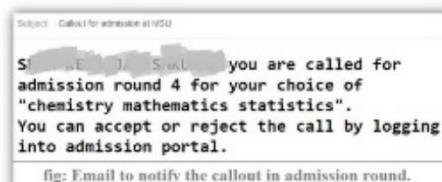
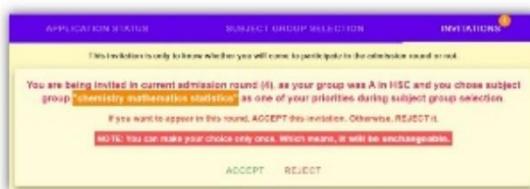


fig: Email to notify the callout in admission round.

## Betterment

- In previous admission rounds, if candidates was called and admitted, but with the less prior subject group, candidate can appear for BETTERMENT in upcoming rounds.
  - Betterment policy works until candidate gets admission with highest prior subject group.



## Forgot Password (Reset)

- If the candidate forgets the password, he / she will have to reset the password by verifying his / her identity.
- The new **password must be secure** enough against brute force attacks.

**Forgot Password**

You are going to trigger the system to send email containing link to reset password by submitting your user ID.

User ID: 020CJ0002

**PROCEED**



Fig.: Email to reset password

**Reset Password**

Create new password to access the Admission Portal.

Password must be 8 characters long and should contain at least one Digit, Capital letter and Special character

**Submit**

## Change Interest

- If the candidate wants to cancel his / her admission for some reason, they can do so by writing their interest in given input box.
  - 'NO' : if the candidate is not interested in the admission process.
  - 'YES' : if the candidate has previously refused and now wants to participate in the admission process again.

Want to change your interest?

Currently, you are not interested to take admission in FYBSc, Faculty of Science. But....

-If you have made up your mind to leave this admission process here itself and do not want to be part of it anymore...

-Or you've had already canceled your admission process and desire to be in now...

**PROCEED**

**Change Interest**

At present, you are not interested to take admission in FYBSc, Faculty of Science.

Please answer the following question in YES or NO.

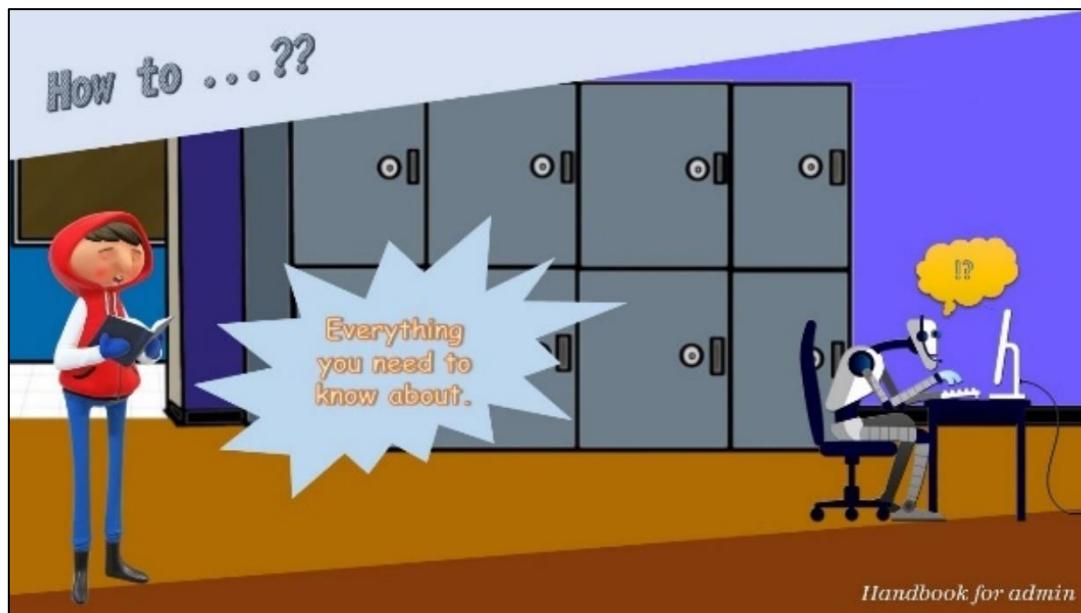
Do you want to take admission in FYBSc, Faculty of Science, M.S. University, Baroda?

YES  NO

**Write and confirm Or just exit this window.**

**CONFIRM** **CLOSE**

# The End



## Preface

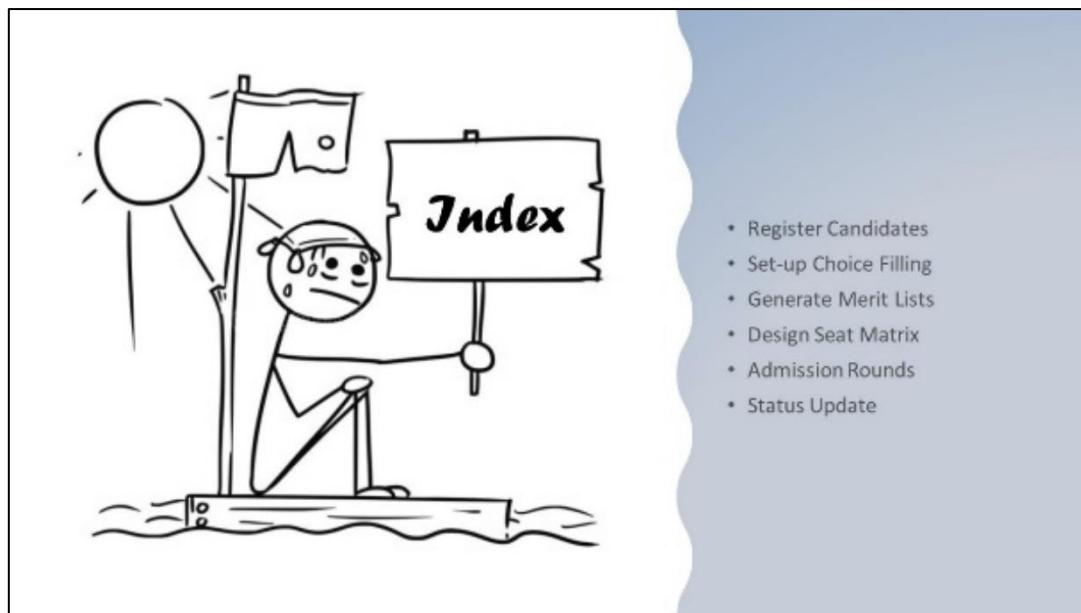
*This handbook, is prepared by the Developers' Team of Admission Portal, provides administrator and/or faculties in Science-Faculty with a helpful, reference about Admission-Application of FYBSc.*

*It includes an overview of the tasks which admin can do within system and thoroughly resolves doubts for operating individual practices.*

*This handbook, however, cannot anticipate every situation or answer every question about application.*

*The personnel procedures and practices stated in this handbook are subject to change. To maintain flexibility in the administration of personnel policies and procedures, the university reserves the right to revise or rescind any provisions of this handbook without notice.*

*If you have any questions about policies & procedures that have not been specifically addressed in this handbook, please consult with your appropriate administrator.*



- Register Candidates
- Set-up Choice Filling
- Generate Merit Lists
- Design Seat Matrix
- Admission Rounds
- Status Update

## Register Candidates

To register candidates in the system, you have to go a long way.

University will provide records of candidate applying for BSc in Excel file format.

You will need to **normalize** and convert that sheet into CSV file format.

Then, upload file and seed data into database.



## Normalization

Follow these steps upon given Excel spreadsheet of candidates' details in order to get normalized version:

- I. Remove unnecessary fields.
- II. Filter (and delete) on fields like qualification-HSC, streamsubgroup, passing year.
- III. Remove duplicates on username & Blank cells, too.
- IV. Marks should be in correct form.
- V. Save as csv file.

Roll No.	Name	Gender	Stream	Subgroup	Qualification	Year	Score
1	John Doe	Male	Computer Science	IT	HSC	2018	85
2	Jane Smith	Female	Electrical Engineering	EE	HSC	2018	78
3	Mike Johnson	Male	Mechanical Engineering	ME	HSC	2018	82
4	Sarah Williams	Female	Electrical Engineering	EE	HSC	2018	75
5	David Lee	Male	Computer Science	IT	HSC	2018	88
6	Emily Davis	Female	Mechanical Engineering	ME	HSC	2018	72
7	James Wilson	Male	Electrical Engineering	EE	HSC	2018	80
8	Amy Green	Female	Computer Science	IT	HSC	2018	76
9	Benjamin Brown	Male	Mechanical Engineering	ME	HSC	2018	83
10	Olivia White	Female	Electrical Engineering	EE	HSC	2018	79
11	Matthew Black	Male	Computer Science	IT	HSC	2018	86
12	Charlotte Grey	Female	Mechanical Engineering	ME	HSC	2018	77
13	Alexander Green	Male	Electrical Engineering	EE	HSC	2018	81
14	Isabella Blue	Female	Computer Science	IT	HSC	2018	74
15	Matthew Black	Male	Mechanical Engineering	ME	HSC	2018	87
16	Charlotte Grey	Female	Electrical Engineering	EE	HSC	2018	80
17	Isabella Blue	Female	Computer Science	IT	HSC	2018	73
18	Matthew Black	Male	Mechanical Engineering	ME	HSC	2018	85
19	Charlotte Grey	Female	Electrical Engineering	EE	HSC	2018	76
20	Isabella Blue	Female	Computer Science	IT	HSC	2018	82

Fig.: Spreadsheet is, containing data of candidates who registered for FYBSc, given by the authorities.

### Fields to be in spreadsheet

Fig.: These fields must be in final version of spreadsheet

## Set-up Choice Filling

- By setting-up the choice filling, admin can decide that how long candidates can fill their subject groups' choices.
  - Admin can lock immediately to get candidates away from making or modifying choices, and also unlock it.
  - By scheduling deadline, admin can make system to do this job on specific date & time. (Web-server must not have downtime at that moment)

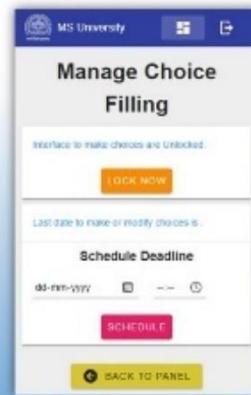


Fig.: User Interface to set-up choice filling

## Generate Merit Lists

To generate/get merit data, try changing the selected values for CASTE, GROUP, and STATE

This will provide columns as given in figure aside.

To export this list (HTML Table) into PDF, tap on PRINT button.

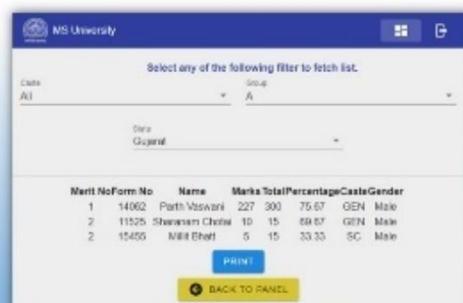


Fig. 3: User Interface to generate merit list

## Export Merit List – additional information

In the published merit list, it is necessary to give the merit table information. i.e., when was merit-list published, by whom, for whom.

- For that, the administrator can modify everything with the basic knowledge of HTML and CSS.
- i.e.,
  - Any text written in the title field can be printed on every page of PDF. By allowing 'Header & Footer' field in web-browser.
  - In the header field, the administrator can write long paragraphs that state the context.
  - Meanwhile, text written in footer field will be displayed after merit table.
  - The style field contains all the CSS that can be manipulated when exporting. (No changes are needed)

Add Details in PDF

All the fields below can work with html tags. If you are familiar with HTML then you can change it in the custom way.

Title of the Document  
Merit List Year: 1920-21 || Faculty of Science || Other header details

Header  
<p>Only candidates whose caste is SEBC are included in this merit list.</p>

Paragraph to be printed or short message

Style  
table,tr,td,th {border: 1px solid black; table{width:100%} td{border-align: center;}

Footer  
First year BSc, Faculty of Science, MS University, Baroda

**PROCEED** **CLOSE**

Fig.: Dialogbox for exporting merit-table with additional information.

## Print Merit List – Web Browser

- In every browser, most of these features will be same.
  - Here, administrator will decide about page size, margin and orientation (portrait or landscape).
  - Also, admin can decide whether they want to allow headers and footers, background graphics and some other functionalities.



Fig.: Printbox of web browser for exporting merit-table.

## Design Seat Matrix (CSVs)

Suppose, you are given an **Excel spreadsheet** that contains data or can also be called **seat matrices**. To insert these information into the system, the steps below need to be done.

- Remove unnecessary data
- Make header row sole & data appropriate
- Export in CSV File Format

Categories and Details wise Seat Matrix for Vellore/Coimbatore/Indira										
Subject Group	Total Seats	Karnataka		Tamil		Sri L.		Total		State
		Gr A	Gr B	Gr A	Gr B	Gr A	Gr B	Gr A	Gr B	
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6	13	8	5	21	9
Maths, English, Chemistry	18	5	14	19	2	6	13	8	5	13
Maths, English, Chemistry	5	14	19	7	6					

## 1. Remove unnecessary data

- a) In 'total seats' column, summation of cells are given which is useless here. Remove those columns.
  - b) Blank and/or descriptive lines in sheet are only meant for humans to understand and get a clear readable view. Remove them as well.

Fig.: Data inside the red rectangles has to be removed.

## 2. Make header row sole & data appropriate

- a) The row of header in a given Excel spreadsheet uses the unique features of MS Office Excel. E.g., merge and center.

  - Unmerge cells
  - Remove extra row of headers

b) Rename the titles (or fields) as given below.

✓ in a proper way, that is, take capital and small letters sensitively.

Fig.: An amendment in header row is needed

Get additional fields from different sheet

It may be possible that field like EWS is not given in the actual datasheet.

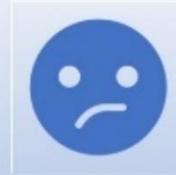


For that, the administrator must get the data from another seat-matrix and merge them into the files of the respective locations.

## Subject names must be concordant



Either they are subject groups or only subject (Principal subject), they must be spelt identically.



E.g., 'Maths' in one file and 'Mathematics' in another are **not acceptable**.

## Fields to be in spreadsheet

These field are for the files: **Vadodara, Gujarat, India**.

- a) Subject Group
- b) General|A
- c) General|B
- d) SC|A
- e) SC|B
- f) ST|A
- g) ST|B
- h) SEBC|A
- i) SEBC|B
- j) EWS|A
- k) EWS|A

	A	B	C	D	E	F	G	H	I	X
1	Subject Group	General A	General B	SC A	ST A	ST B	SEBC A	SEBC B	EWS A	EWS B
2	Biology	9	54	5	2	6	4	2	4	0
3	Zoology	43	52	5	8	6	4	2	4	0

Fig.: Final version of **Spreadsheet** is ready to be exported

**Note:** straight line or *vertical bar* in field-name is known as pipe which is used here as a delimiter.

## Merge different sub-categories' seat-matrices



These seat matrices can be given in separate spreadsheets.



The administrator must merge them all into a single file with the column-name specified below.

Differently-Add			
Principal Subject	VADODARA	GUJARAT	INDIA Total
Biology			1
Chemistry			2
Geography			1
History		1	2
Maths			2
Physics			2
Statistics			1
Total	2	3	10

B-GROUP Results			
Principal Subject	VADODARA	GUJARAT	INDIA Total
Biology			2
Chemistry			3
Geography			1
History		1	2
Maths			1
Physics			1
Zoology			1
Total	2	1	10

E-GROUP			
Principal Subject	VADODARA	GUJARAT	INDIA Total
Biology			3
Chemistry			1
Geography			1
History			6
Maths			6
Physics			6
Zoology			6
Total	42	9	51 43

Fig.: Data inside the **red rectangles** has to be removed.

## Fields to be in spreadsheet

These field are for the file: Sub-Category

- a) Principal Subject
- b) VADODARA|DEFENCEWARDS
- c) VADODARA|DIFFERENTLYABLED
- d) VADODARA|KASHMIR
- e) GUJARAT|DEFENCEWARDS
- f) GUJARAT|DIFFERENTLYABLED
- g) GUJARAT|KASHMIR
- h) INDIA|DEFENCEWARDS
- i) INDIA|DIFFERENTLYABLED
- j) INDIA|KASHMIR
- k) TOTAL|DEFENCEWARDS
- l) TOTAL|DIFFERENTLYABLED
- m) TOTAL|KASHMIR

Principal Subject	VADODARA DEFENCEWARDS	VADODARA DIFFERENTLYABLED	VADODARA KASHMIR	GUJARAT DEFENCEWARDS	GUJARAT DIFFERENTLYABLED	GUJARAT KASHMIR	INDIA DEFENCEWARDS	INDIA DIFFERENTLYABLED	INDIA KASHMIR	TOTAL DEFENCEWARDS	TOTAL DIFFERENTLYABLED	TOTAL KASHMIR
Maths							3	6	2	3	9	3
Chemistry							3	6	2	3	9	3
Biology							3	6	3	3	9	3
Physics							3	6	2	3	9	2
Geography							3	6	3	3	9	3
Maths							3	6	2	3	9	2
Physics							3	6	2	3	9	2
Geography							3	6	3	3	9	3
Total	7	36	3	2	9	2	3	3	3	3	9	3

Fig.: Final version of Spreadsheet is ready to be exported

Note: straight line or vertical /barin field-name is known as pipe which is used here as a delimiter.

## 3. Export in CSV File Format

You'll find different interfaces for exporting sheets to csv in different versions of MS Office Excel.

You'll have to save/export **4 csv files** for India,

Gujarat, Vadodara, Sub-Category respectively.

You are recommended to use 'File -> Save As' and select CSV as a type of file.

File name: Seat Matrix - India.csv  
Save as type: CSV (Comma delimited) (\*.csv)

Fig.: Dialog box for exporting file

Subject Group	General A	General B	SC A	SC B	ST A	ST B	SCBC A	SCBC B	EWS A	EWS B
Barotri Zoology Chemistry	9	54	5	2	6	6	2	4	0	0
Barotri Zoology Geography	41	52	5	2	6	4	2	4	0	0

File: Vadodara.csv

Subject Group	General A	General B	SC A	SC B	ST A	ST B	SCBC A	SCBC B	EWS A	EWS B
Barotri Zoology Chemistry	8	54	5	2	6	6	2	4	0	0
Barotri Zoology Geography	41	52	5	2	6	4	2	4	0	0

File: Gujarat.csv

Subject Group	General A	General B	SC A	SC B	ST A	ST B	SCBC A	SCBC B	EWS A	EWS B
Barotri Zoology Chemistry	9	54	5	2	6	6	2	4	0	0
Barotri Zoology Geography	41	52	5	2	6	4	2	4	0	0

File: India.csv

Prinicipal Subject	INDIA DEFENCEWARDS	INDIA DIFFERENTLYABLED	INDIA KASHMIR	TOTAL DEFENCEWARDS	TOTAL DIFFERENTLYABLED	TOTAL KASHMIR
Maths				3	6	2
Chemistry				3	6	2
Biology				3	6	3
Physics				3	6	2
Total	7	36	3	2	9	3

File: Sub-Categories.csv

## Admission Rounds

After installing seat-matrices correctly, administrator can start creating admission rounds.

In which, the administrator will get the list of students who can participate first based on their merit number.

The administrator should export this data as a CSV, which will be required to update the admission status and fees payment status.

### Admission Rounds

< SHORTLISTED CANDIDATES >

Current admission round is: 3

You are currently viewing shortlisted candidates for admission round 3

EXPORT AS CSV

Sr. No. Name

1. 11002 SHALINI KUMARI JAYESHBHUVAN

2. 11152 TANAYA BHAVNA

Fig.: User Interface to export admission round – students' list

Admission status & Fees payment status update

✓ In downloaded CSV file, administrator have to change admission and fees payment status respectively. (TRUE to FALSE, or vice versa).

✓ And upload that updated CSV file. (Fields/Columns will remain the same).

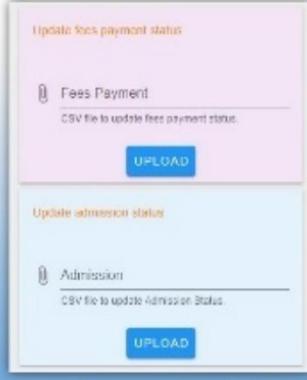


Fig.: User Interface to upload updated CSVs



## Handbook for Superuser

**Manual for Super-User**

**Introduction:**

The superuser is a special user account used for system administration. In systems which implement a role-based security model, any user with the role of superuser (or its synonyms) can carry out all actions of the superuser account. Superusers can make unrestricted, potentially adverse, system-wide changes.

**Administration-portal:**

In "Administration Portal" system, super-user can perform following tasks through Command Line Interface:

- View list of commands using 'help' option.

```
node admin [-get] [-help] [-change] [-delete] [-create] [-args]
```

These are the commands used in various situations:

- 1)Get all admin:

```
node admin -get
```

2)Get help:

```
node admin -help
```

3)Change credentials for an admin:

```
node admin -change EMAIL_PASSWORD ADVANCE_PASSWORD
```

4)Delete an admin:

```
node admin -delete EMAIL
```

5)Create an admin:

```
node admin -create EMAIL_PASSWORD ADVANCE_PASSWORD
```

Note: Enter the special characters followed by "-" in the EMAIL, PASSWORD and ADVANCE\_PASSWORD arguments.

6)Get all admin:

```
node admin [-get] [-help] [-change] [-delete] [-create] [-args]
```

These are the commands used in various situations:

- 1)Get all admin:

```
node admin -get
```

- 2)Get help:

```
node admin -help
```

- 3)Change credentials for an admin:

```
node admin -change EMAIL_PASSWORD ADVANCE_PASSWORD
```

4)Delete an admin:

```
node admin -delete EMAIL
```

5)Create an admin:

```
node admin -create EMAIL_PASSWORD ADVANCE_PASSWORD
```

Note: Enter the special characters followed by "-" in the EMAIL, PASSWORD and ADVANCE\_PASSWORD arguments.

6)Get list of admins using 'get' option:

```
node admin -get
```

id	email	date
60a390c426c24f4ed2c1877	partimvaasant@gmail.com	2021-05-18T09:20:04.591Z
60a390c426c24f4ed2c1877	partimvaasant@gmail.com	2021-05-18T09:20:10.632Z

Note: id is autogenerated. id by MongoDB. email is also known as username of admin. Table also contains timestamp when the administrator account was registered into the database.

7)Manage admin:

```
node admin [-get] [-help] [-change] [-delete] [-create] [-args]
```

8)Register new administrator:

- Use 'create' option and pass 3 arguments: EMAIL\_ID, PASSWORD, and ADV\_PASSWORD.

9)Change password of administrator's account:

- Use 'change' option and pass 3 arguments: EMAIL\_ID, PASSWORD, and ADV\_PASSWORD.

10)Delete administrator's account:

- Use 'delete' option and pass EMAIL\_ID as an argument.

## **Super User CLI:**

```
parth@parth-HP-Laptop-15-bs1xx:~/Desktop/AdmissionPortal/backend$ node admin --help
Usage: node admin [--get] [--help] [--change] [--delete] [--create] [<args>]

These are the commands used in various situations:

1)Get all admins:
   node admin --get

2)Get help:
   node admin --help

3)Change credentials for an admin:
   node admin --change EMAIL PASSWORD ADVANCE_PASSWORD

4)Delete an admin:
   node admin --delete EMAIL

5)Create an admin:
   node admin --create EMAIL PASSWORD ADVANCE_PASSWORD

Note: Enter the special characters followed by "\" in the EMAIL, PASSWORD and ADVANCE_PASSWORD arguments.
```

Fig: Super user's manual & CLI. Admin can see list of commands using option 'help'.

---

```
parth@parth-HP-Laptop-15-bs1xx:~/Desktop/AdmissionPortal/backend$ node admin --get
```

(index)	email	date
60a386c42b628416ad3c1677	'parthnvaswani@gmail.com'	2021-05-18T09:20:04.391Z
60a386d33bbd5a16c0852851	'parthnvaswani@gmail.co'	2021-05-18T09:20:19.532Z

Fig. List of admins. Super user can view the list of admins using option “get”.

---

```
parth@parth-HP-Laptop-15-bs1xx:~/Desktop/AdmissionPortal/backend$ node admin --change parthnvaswani@gmail.com Dingo\$1do Dingo\$1do
Password changed successfully.
parth@parth-HP-Laptop-15-bs1xx:~/Desktop/AdmissionPortal/backend$ node admin --create parthnvaswani@gmail.com Dingo\$1do Dingo\$1do
Admin created successfully.
parth@parth-HP-Laptop-15-bs1xx:~/Desktop/AdmissionPortal/backend$ node admin --delete parthnvaswani@gmail.com
Admin deleted successfully.
```

Fig. Tasks of super user. Super user can create, change, or delete admin using respective options.

## Restful APIs

This were documented in postman for ease of accessing on front-end by developers. For detailed docs: [Admission Portal \(getpostman.com\)](https://Admission%20Portal%20(getpostman.com))

HTTP Method	Route	Who can access?	Short Description
<b>post</b>	api/admin/login	everyone	Login as admin
<b>get</b>	api/admin/announcement	everyone	Get announcements
<b>post</b>	api/admin/announcement	Only Admin	Create an announcement
<b>delete</b>	api/admin/announcement	Only Admin	Delete an announcement
<b>post</b>	api/admin/data	Only Admin	Add csv file for seeding
<b>get</b>	api/admin/data	Only Admin	Get csv file for seeding
<b>get</b>	api/admin/seed	Only Admin	Start data seeding
<b>get</b>	api/admin/notify	Only Admin	Send mail to all users for activating account
<b>get</b>	api/admin/notify/:id	Only Admin	Send mail to all users for activating account
<b>get</b>	api/admin/lock	Only Admin	Lock all choices
<b>post</b>	api/admin/lock	Only Admin	Schedule Lock all choices
<b>get</b>	api/admin/unlock	Only Admin	Unlock all choices
<b>get</b>	api/admin/locked	everyone	Get status locked or unlocked
<b>get</b>	api/admin/lastDate	everyone	Get last date to fill choices
<b>get</b>	api/admin/merit	Only Admin	Get merit data
<b>get</b>	api/admin/users	Only Admin	Search for users
<b>get</b>	api/admin/user/:userId	Only Admin	Get user details
<b>patch</b>	api/admin/user/:userId	Only Admin	Change user details
<b>delete</b>	api/admin/user/:userId	Only Admin	Delete user
<b>post</b>	api/admin/user	Only Admin	Add new user details

<b>get</b>	api/admin/round	Only Admin	Get current admission round
<b>post</b>	api/admin/seat	Only Admin	Add csv file for seat matrix
<b>patch</b>	api/admin/seat	Only Admin	Update seat matrix
<b>get</b>	api/admin/seat	Only Admin	Get seat matrix
<b>get</b>	api/admin/admission Round	Only Admin	Create next admission round
<b>get</b>	api/admin/admission Round/:number	Only Admin	Get admission round data
<b>patch</b>	api/admin/admission Status	Only Admin	Update admission status and seat matrix
<b>patch</b>	api/admin/feesPaym entStatus	Only Admin	Update fees payment status and seat matrix
<b>post</b>	api/password/passwo rdReset	everyone	send email for reset password
<b>post</b>	api/password/savedR eset	everyone	save the new password
<b>get</b>	api/password/resetPa ssword/:id/:token	everyone	set new password
<b>get</b>	api/users/conform/:to ken	everyone	Confirm email
<b>post</b>	api/users/login	everyone	Login User / Returning JWT Token
<b>get</b>	api/users/details	Only Candidate	Return current user details
<b>get</b>	api/users/choices	Only Candidate	Get available choices for user
<b>post</b>	api/users/choices	Only Candidate	Fill choices for user
<b>post</b>	api/users/invite	Only Candidate	Accept or Reject Invitation for participation in admission round
<b>patch</b>	api/users/interest	Only Candidate	Change interest status

# Testing

*“Discovering the unexpected is more important than confirming the known”*

—George E. P. Box

## What is testing in agile methodology?

In agile, requirements evolve through collaboration between the customer and self-organizing teams (developers). And agile aligns development with customer needs.

There are few advantages of agile testing:

- It saves time because developers do not need to document everything as ‘less documentation’ is feature of agile and regular feedback from the end user can help to determine the issues well in advance.
- Developers and end users are testing everything therefore agile do not consists of testing phase in its lifecycle. It runs continuously alongside the development effort. Here testing team is not required, therefore cost reduces.

## What did we have then?

Instead of every kind of test, which is not being documented in agile methodology, we are putting user scenarios here, which can also be considered as validation, verification, or acceptance test.

View Announcements				
Note	User is currently viewing homepage			
	Scene1	Scene2	Scene3	Scene3
Type of user	Visitors	Visitors	Visitors	Visitors
Action	Clicks on read more button	Writes text into "filter by title"	changes page of announcement	Clicks on download button under announcement
Output	Individual announcement gets open	Announcement which contains same text somewhere in the title can be seen into cards-panel of announcements	Announcements for given number of pages loaded	file gets open in new tab, PDFs are readable in browser directly

Navbar					
	Scene1	Scene2	Scene3	Scene4	Scene5
Type of user	Visitor	Admin	Candidate	Candidate	Admin, Candidate
Action	Clicks on sign in button	Clicks on dashboard	Clicks on dashboard	Clicks on profile	Clicks on logout
Output	User was redirected to candidate's login page	User was redirected to admin-dashboard	User was redirected to /dashboard	User was redirected to /profile	User was redirected to /logout

Candidate Login			
Note	Scene1	Scene2	Scene3
Type of user	Visitor/Candidate	Visitor/Candidate	Visitor/Candidate
Action	Writes correct username password	enters wrong credentials	clicks on view password button
Output	User was redirected to candidate's dashboard	User can see errors below individual input box	User can see their password

Candidate Forgot Password			
Note	Scene1	Scene2	Scene3
Type of user	Visitor/Candidate	Candidate	Candidate
Action	Enters UserID properly	Clicks on button given in the mail	Enters weak password
Output	It asked to see email box of candidate	User was asked to enter password and confirm password.	It shows, how to make proper-valid password

Change Interest			
Note	Candidates is logged in		
	Scene1	Scene2	Scene3
Type of user	Candidate	Candidate	Candidate
Action	Proceeds to change interest	user enters yes and no respectively	User writes other than YES or NO
Output	dialog box with input box was visible	It shows Interest updated successfully.	it shows: Either YES or NO. Everything else is invalid.

Choice filling				
Note	Candidates is logged in and is in subject group selection section and choices are not locked			
	Scene1	Scene2	Scene3	Scene4
Type of user	Candidate	Candidate	Candidate	Candidate
Action	Clicks on make / modify button	Selects from available subject groups	untick from available subject groups	Clicks on up and down arrows respectively
Output	dialog box was opened	Subject Group was added in selected subject groups	given subject group from selected subject group was removed	Order of selected subject groups changes accordingly

Choice filling				
Note	Candidates is logged in and is in subject group selection section and choices are not locked			
	Scene5	Scene6	Scene7	Scene8
Type of user	Candidate	Candidate	Candidate	Candidate
Action	Selects less than 5 subjects	unticks condition below (checkbox)	selects more than 5 subjects and tick checkbox below	Clicks on update button
Output	Update button remains/becomes disabled	Update button remains/becomes disabled	update button is clickable	dialog box closes and updated list becomes visible

Invitation		
<b>Note</b>	Candidates is logged in and is in invitation section	
	<b>Scene1</b>	<b>Scene2</b>
<b>Type of user</b>	Candidate	Candidate
<b>Scenario</b>	User has pending invitations	User accepts or rejects invitation and confirms
<b>Output</b>	it shows badge on the invitation section's header	It updates invitation section, showing what candidate has done(chosen)

Admin Login			
<b>Note</b>	User is not login yet		
	<b>Scene1</b>	<b>Scene2</b>	<b>Scene3</b>
<b>Type of user</b>	Admin	Admin	Admin
<b>Action</b>	Writes Everything corrects (Email, password, advPassword)	enters wrong credentials (any of those three)	clicks on view password button
<b>Output</b>	User was redirected to Admin's dashboard	User can see errors below individual input box	User can see their password

Manage Announcements			
<b>Note</b>	Admin is logged in		
	<b>Scene1</b>	<b>Scene2</b>	<b>Scene3</b>
<b>Type of user</b>	Admin	Admin	Admin
<b>Action</b>	User clicks on read announcement	User clicks on delete button and confirms	Clicks on create announcement button
<b>Output</b>	Dialog box for individual announcement got open	Announcement was deleted and cards on page refetched	Dialog box for new announcement became visible

Make Announcements			
Note	Admin is logged in and dialog box for making new announcement is open		
	Scene1	Scene2	Scene3
Type of user	Admin	Admin	Admin
Scenario	User clicks on save button without providing anything in title, description	User provides more than 30 characters in title or less than 12 characters in description	Clicks on save after entering valid information
Output	Error messages were being displayed	Error messages were being displayed	Dialog box disappeared and announcement can be seen in list of announcements

Build Database				
Note	Admin is logged in			
	Scene1	Scene2	Scene3	Scene4
Type of user	Admin	Admin	Admin	Admin
Scenario	Admin attaches CSV file, and clicks on Upload button	Admin clicks on Fetch CSV button	Admin clicks on seed data button	Admin clicks on broadcast button
Output	Alert box for acknowledgment was shown.	Uploaded CSV file was displayed in table format	Alert box for acknowledgment was shown.	Alert box for acknowledgment was shown. And emails to each candidate was being sent.

New Candidates		
Note	Admin is logged in	
	Scene1	Scene2
Type of user	Admin	Admin
Scenario	Admin fills data of new candidate into form, but something was missed to mention and clicks on proceed button	Admin fills correct data
Output	Error messages from server was shown above the proceed button	Last inserted student details can be seen below form, and new blank form is ready to accept new candidate

Manage Candidates					
Note	Admin is logged in				
	Scene1	Scene2	Scene3	Scene4	Scene5
Type of user	Admin	Admin	Admin	Admin	Admin
Scenario	Admin searches for specific candidate by form no	Admin navigates through page numbers below	Admin clicks on view button to view individual candidate's details	Admin clicks on activate button and confirms	Admin clicks on delete candidate button and confirms
Output	Matching candidate with same form no. was appeared in the list	Data for specific page number was appeared properly	admin was redirected to specific userId's page and on that page Read only (form like) structure was displayed	Alert box showing that email was sent to candidate.	Candidate was removed from the list(database)

Generate Merit List				
Note	Admin is logged in			
	Scene1	Scene2	Scene3	Scene4
Type of user	Admin	Admin	Admin	Admin
Scenario	Admin selects or changes parameters through dropdown list (it was a case when candidates was available)	Admin selects or changes parameters through dropdown list (it was a case when candidates was not available)	Admin clicks on print button	Admin provides additional information and confirms
Output	List of those candidates was displayed in table format	Image showing that no candidates are there, try to look for other parameters.	dialog box for accepting additional information was displayed.	Print dialog box was shown.

Manage Choice Filling			
Note	Admin is logged in		
	Scene1	Scene2	Scene3
Type of user	Admin	Admin	Admin
Action	Clicks on lock now button	Clicks on unlock now button	Provides date and time and proceeds
Output	alert box showing choice filling was locked immediately	alert box showing choice filling was unlocked immediately	alert box for acknowledgment was shown

Manage Seat Matrix			
Note	Admin is logged in		
	Scene1	Scene2	Scene3
Type of user	Admin	Admin	Admin
Action	Uploads seat matrix files according to each field	Admin clicks on refresh button to see uploaded seat matrix files	Admin writes JSON and Proceed to update seat matrix data
Output	alert box for acknowledgment was shown and email to admin was sent for whatever happened with those files, either operation was successful or not.	Files was fetched from server and shown in the table format.	alert box for acknowledgment was shown, either it was successful or not.

Manage admission rounds			
Note	Admin is logged in		
	Scene1	Scene2	Scene3
Type of user	Admin	Admin	Admin
Action	Clicks on generate round button and confirms	Clicks on page number to see shortlisted students for individual round	Clicks on Export as CSV button
Output	alert box for acknowledgment was shown and current round number changes.	Data was displayed in table format	CSV file for given admission round was downloaded with its admission round number as file name

Manage admission rounds		
Note	Admin is logged in	
Type of user	Scene1	Scene2
Action	Attaches admission status.csv file and confirms	Attaches fees payment status.csv file and confirms
Output	alert box for acknowledgment was shown, whether file was uploaded correctly or not.	alert box for acknowledgment was shown, whether file was uploaded correctly or not.

Create Admin		
	Scene1	Scene2
Type of user	Super User	
Action	Wrote option --create and passed valid EMAIL ID, PASSWORD, and ADVANCE PASSWORD as an argument.	Wrote option --create and passed any invalid argument.
Output	Admin created and message displayed for same.	Error displayed for respective argument.

Change Admin		
	Scene1	Scene2
Type of user	Super User	
Action	Wrote option --change and passed Email Id, password, and advance password as an argument.	Wrote option --change and passed any invalid argument.
Output	Password changed and message displayed for same.	Error displayed for respective argument.

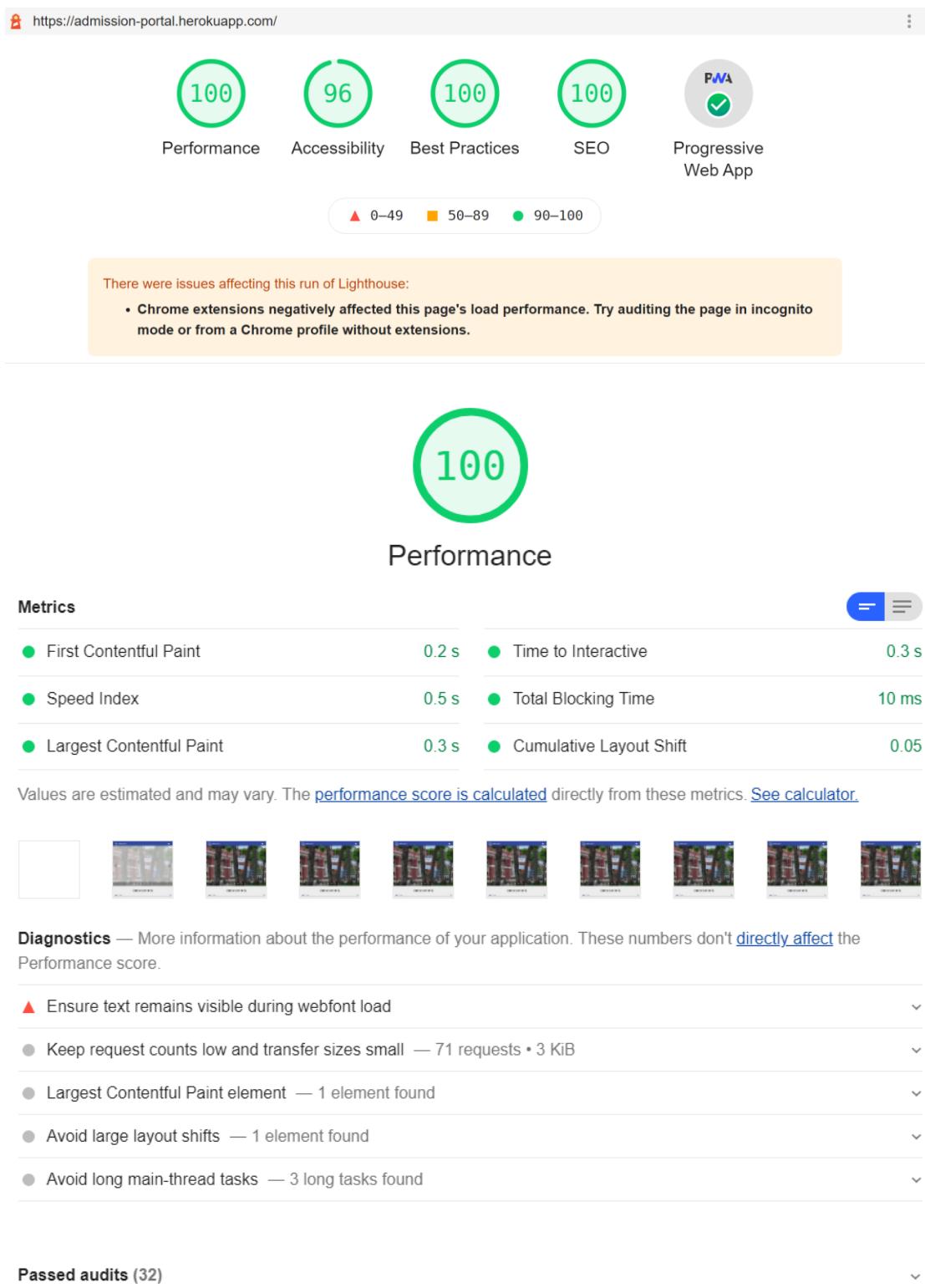
Delete Admin		
	Scene1	Scene2
Type of user	Super User	
Action	Wrote option --delete and passed Email Id as an argument.	Wrote option --delete and passed invalid email id.
Output	Admin deleted and message displayed for same.	Error displayed.

## Security and Other Features

- XSS (Cross Site Scripting)
  - XSS is a web security vulnerability that allows an attacker to compromise the interactions that users have with a vulnerable application. It allows an attacker to circumvent the same origin policy, which is designed to segregate different websites from each other. Cross-site scripting vulnerabilities normally allow an attacker to masquerade as a victim user, to carry out any actions that the user is able to perform, and to access user's data. If the victim user has privileged access within the application, then the attacker might be able to gain full control over all of the application's functionality and data.
  - We have done following things to ensure our application is secure against XSS.
  - We have used `xss-clean` as a middleware to sanitize user input coming from POST body, GET queries, and url params.
  - We have also done validation of data coming from user input before storing it to the database.
  - We have also used Helmet which ensures that no inline or 3rd party scripts or assets are loaded or executed on front end by setting some headers.
  - And at last, we are using Vue as our frontend framework which filters the data coming from server before rendering it.
- ReDoS (Regular expression Denial of Service)
  - ReDoS is a Denial-of-Service attack, that exploits the fact that most Regular Expression implementations may reach extreme situations that cause them to work very slowly (exponentially related to input size). An attacker can then cause a program using a Regular Expression to enter these extreme situations and then hang for a very long time.
  - We are escaping special characters which have any meaning in regex from user input before inserting it into Regular Expression.
- Denial of Service & Brute force
  - Brute force attacks use a technique of attempting to try every combination of passwords/keys to gain access to a particular system.
  - We are using a middleware named `express-rate-limit` that is used to limit repeated requests to public APIs and/or endpoints such as password reset, login etc. by only allowing certain number of requests from an unique IP address.
- NoSQL Injection
  - NoSQL injection vulnerabilities allow attackers to inject code into commands for databases that do not use SQL queries, such as MongoDB. Which allows an attacker to do things like bypass authentication, access restricted data etc.
  - We are using `Express Mongoose Sanitize` as a middleware to sanitize data provided by user to prevent MongoDB Operator Injection.

- **CSRF (Cross Site Request Forgery)**
  - Cross-Site Request Forgery (CSRF) is an attack that forces an end user to execute unwanted actions on a web application in which they are currently authenticated.
  - We are using JWT tokens stored in local-storage which can only be used on the same site to validate user. And we have already mitigated XSS so, CSRF is not possible.
  - We are also ensuring to warn user about any critical state change to their account like resetting password, deleting account etc. so that if they click any random link then also, they are safe.
- **Caching**
  - Caching is the process of storing copies of files in a cache, or temporary storage location, so that they can be accessed more quickly.
  - All the static assets are cached on initial load of website using service worker, so that user do not have to wait for website to load every-time they surf it. And it also works offline because of it.
- **Installable (PWA)**
  - A progressive web application is a type of application software delivered through the web, built using common web technologies including HTML, CSS, and JavaScript. It is intended to work on any platform that uses a standards-compliant browser, including both desktop and mobile devices.
  - Our application is installable on any platform, so that they can access it from home-screen without opening any browser and it also allows us to use some native device features.
- **Single Page Application (Vue)**
  - A single-page application (SPA) is a web application or website that interacts with the user by dynamically rewriting the current web page with new data from the web server, instead of the default method of a web browser loading entire new pages.
  - Our application is single page application, so user do not have to wait for the loading new page because only content of the page is changed, and data consumption is low.
- **Image Compression**
  - For modern browsers we are compressing and storing images in webp format for optimization.
  - And for older browsers we are storing them in a globally accepted format like JPEG.
- **HTTP Compression**
  - HTTP compression is a capability that can be built into web servers and web clients to improve transfer speed and bandwidth utilization.
  - We are using Brotli compression which is best for optimization on our server for modern browsers which support it and for older browsers we are using GZIP as fallback which is not as good Brotli but way better then serving uncompressed assets.

Fig: Report. -- tested and generated by Lighthouse 7.2.0





## Accessibility

These checks highlight opportunities to [improve the accessibility of your web app](#). Only a subset of accessibility issues can be automatically detected so manual testing is also encouraged.

**Contrast** — These are opportunities to improve the legibility of your content.

▲ Background and foreground colors do not have a sufficient contrast ratio.



**Names and labels** — These are opportunities to improve the semantics of the controls in your application. This may enhance the experience for users of assistive technology, like a screen reader.

▲ Links do not have a discernible name



**Additional items to manually check (10)** — These items address areas which an automated testing tool cannot cover. [Learn more in our guide on conducting an accessibility review](#).



**Passed audits (22)**



**Not applicable (20)**



## Best Practices

**Passed audits (17)**



**Not applicable (1)**





## SEO

These checks ensure that your page is optimized for search engine results ranking. There are additional factors Lighthouse does not check that may affect your search ranking. [Learn more](#).

**Additional items to manually check (1)** — Run these additional validators on your site to check additional SEO best practices. ▼

**Passed audits (11)** ▼

**Not applicable (3)** ▼



## Progressive Web App

These checks validate the aspects of a Progressive Web App. [Learn more](#).

### **Installable**

- Web app manifest and service worker meet the installability requirements ▼

### **PWA Optimized**

- Registers a service worker that controls page and `start_url` ▼
- Redirects HTTP traffic to HTTPS ▼
- Configured for a custom splash screen ▼
- Sets a theme color for the address bar. ▼
- Content is sized correctly for the viewport ▼
- Has a `<meta name="viewport">` tag with `width` or `initial-scale` ▼
- Provides a valid `apple-touch-icon` ▼
- Manifest has a maskable icon ▼

Runtime Settings

<b>URL</b>	<a href="https://admission-portal.herokuapp.com/">https://admission-portal.herokuapp.com/</a>
<b>Fetch Time</b>	May 13, 2021, 10:35 AM GMT+5:30
<b>Device</b>	Emulated Desktop
<b>Network throttling</b>	40 ms TCP RTT, 10,240 Kbps throughput (Simulated)
<b>CPU throttling</b>	1x slowdown (Simulated)
<b>Channel</b>	devtools
<b>User agent (host)</b>	Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/90.0.4430.93 Safari/537.36
<b>User agent (network)</b>	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/90.0.4420.0 Safari/537.36 Chrome-Lighthouse
<b>CPU/Memory Power</b>	900
<b>Axe version</b>	4.1.2

Generated by **Lighthouse** 7.2.0 | [File an issue](#)

---

## Coding Standard

We are using tools like prettier and ESLint to enforce *coding conventions and naming standards*, so that the code is maintainable, consistent, understandable, and flexible.

ESLint performs automated scans of JavaScript files for common syntax and style errors.

Prettier scans files for style issues and automatically reformats code to ensure consistent rules are being followed for indentation, spacing, semicolons, single quotes vs double quotes, etc.

We use them on project because:

- ✓ They keep everybody on the same page, following the same rules.
- ✓ They save time in code reviews, because we can safely ignore all style issues, and focus on things that matter, like the structure and semantics of our code.
- ✓ They catch errors. Prettier, not so much, but ESLint catches a lot of syntax errors and simple forms of type errors, such as undefined variables.
- ✓ Setting these things up is a one-time cost, but the time-saving benefits compound over time.

For files, naming conventions are used as well.

We are using functional programming to achieve following things:

- ✓ It reduces code redundancy.
- ✓ It improves modularity.
- ✓ It helps us to solve complex problems.
- ✓ It increases maintainability.

We are using express as a backend framework which is based on MVC (Model-view-controller) approach which helps us in achieving following things:

- ✓ Development of the application becomes fast.
- ✓ Easy for multiple developers to collaborate and work together.
- ✓ Easier to Update the application.
- ✓ Easier to Debug as we have multiple levels properly written in the application.

## Limitations and Future Enhancements

*“Perfection is a goal that will forever remain impossible for any human being to achieve. Therefore, the only achievable goal is to only strive to become the best that you can possibly be.”*

— Edmond Mbiaka

We tried our best to make the application as safe, fast, and well-optimized as possible, although some drawbacks cannot be ruled out.

i.e., our application cannot currently provide information to participants via short message service (SMS), as we have not yet subscribed to any services used in commercial areas.

Another thing is, all the images shown on the website and all the files uploaded by the user(administrator) are being kept on the primary web server, for which cloud storage of amazon or something else like that, can be used.

## References and Bibliography

Almost everything in this document is followed by knowledge and wisdom of our guide and teachers. Beyond this, we took care of notes from the following websites.

For learning,

Vue - <https://vuejs.org>

Vuetify - <https://vuetifyjs.com/en>

Node JS - <https://nodejs.org>

Mongo DB - <https://docs.mongodb.com/manual/tutorial>

Express - <https://expressjs.com>, and  
[https://developer.mozilla.org/enUS/docs/Learn/Server-side/Express\\_Nodejs](https://developer.mozilla.org/enUS/docs/Learn/Server-side/Express_Nodejs)

Mongoose - <https://mongoosejs.com/docs>

Nodemailer - <https://nodemailer.com/about/>

Vuex - <https://vuex.vuejs.org>

JSON Web token - <https://jwt.io/>

Axios - <https://www.npmjs.com/package/axios>

Local Storage:

[https://developer.mozilla.org/enUS/docs/Web/API/Web\\_Storage\\_API](https://developer.mozilla.org/enUS/docs/Web/API/Web_Storage_API)

Bcryptjs - <https://www.npmjs.com/package/bcryptjs>

Passport - <http://www.passportjs.org/docs>

Validator - <https://www.npmjs.com/package/validator>

Safe-Regex - <https://www.npmjs.com/package/safe-regex>

Nodemon - <https://nodemon.io>

For dependencies, being used in system:

NPM JS: npm | build amazing things ( <https://www.npmjs.com> )

Yarn: <https://yarnpkg.com>

## Conclusion

This project boosted our knowledge and skills required in a corporate world specifically. Also, it exposed us to many new technologies and opportunities like VueJS and its whole environment of libraries and allowed us to improve on our previous knowledge of node JS and mongo DB. We learnt teamwork, planning and human resource management.

It was very difficult to manage our team from a distant and keep everyone on the same page which slowed down our progress. There were a lot of changes in the requirements in the middle and end of our project that contributed towards a lot of work added in the last sprint but our choice of following agile proved to be right and we were able to complete the project in timely manner.

Moreover, we are the only batch, hopefully, who got affected by a pandemic and completed whole project without any physical interaction with team members, therefore it was a kind of challenge for every one of us, but we learnt to overcome these limitations and achieved our goal.

