SOFTWARE ENGINEERING PROJECT 15CSE313

FACULTY DASHBOARD

TEAM : iris

TEAM MEMBERS

Gayathri	CB.EN.U4CSE17420
Prathyusha	CB.EN.U4CSE17424
Neeraj P Reddy	CB.EN.U4CSE17445
Saiharsha B	CB.EN.U4CSE17451
Sanjay Tharagesh R S	CB.EN.U4CSE17453
Srishilesh P S	CB.EN.U4CSE17458

INDEX

ABSTRACT

In higher education institutions, managing a lot of tasks manually can be a hectic work for both faculties and students. There can be a lot of mismanagement and disorder in the information stored and retrieved. Moreover, it is a time-consuming process for faculties to monitor and integrate all the information to reduce data redundancy and generate reports from it. In some cases, it is also difficult for faculties to reach out to the Human Resource department to check their current attendance and leave status. Thus, we developed an integrated platform called Faculty Dashboard, for faculties to find tremendous amounts of information about their courses, attendance status of students and faculties, daily class schedules and other numerous functionalities that support faculties to complete the essential tasks right from the web portal. This portal will act as a one-stop source for all of the faculty's needs.

Introduction

In higher education institutions, people managing academic programs may require to monitor and check the progress of data regarding program offerings, course offerings, class attendance, etc.. Implementing a faculty dashboard could improve efficiency in data retrieval and optimize data storage. The dashboard implementation utilized database structure, faculty data was organized and stored in an AWS database and managed with SQL queries to retrieve data for the dashboard. The prototype data dashboard shows different types of functionalities and tables, even though there were some moderate limitations on the overall design due to time and implementation complexities. Future work could focus on streamlining the data preparation and automating the dashboard visualizations for even better efficiency in the faculty dashboard management system.

Need and Motivation

With the large number of courses offered every year and over 10+ academic programs, managing course details, class and faculty attendance, schedules of faculties and respective classes are quite labor-intensive. As in traditional universities, faculty members work with the department or individual to monitor and generate various student/faculty reports. Additionally, at universities, faculty serve as program leads and are responsible for managing the scheduling, staffing, and assessment of the monthly courses in their academic program. Assisting and mentoring students in a cohort is also a responsibility of the program lead. Thus a lot of work has to be integrated to achieve an efficient and optimized method to store and retrieve the information, we came up with an idea to develop a faculty dashboard which will offer specific functionalities to faculties to monitor their required tasks quickly. The proposed functionalities are limited to very basic requirements.

Tools Used

❏ Developing
1. Front-End　　　　　　: Nuxt.Js, VueJS
2. Back-End　　　　　　: NodeJS, Express
3. Database　　　　　　: AWS RDS, AWS S3, FireStore
4. Hosting Backend　　: AWS EC2
5. SCM　　　　　　　　: GIT

❏ Testing
1. Unit Testing　　　　: Mocha, Chai, Istanbul, NYC
2. Static Code analysis　: SonarQube
3. UI testing　　　　　: Selenium Webdriver (Javascript)

❏ Continuous Integration and Deployment
1. Tool　　　　　　　　: Jenkins
2. Webapp hosting　　: Netlify

Product Backlog

1. User Authentication (Login & Log out) and Registration:
   a. As a faculty, I must log in to my dashboard - [4]
   b. As a faculty, I should be able to register with the required details. - [5]
   c. As a faculty, I should be able to recover a password - [4]
   d. As a faculty, I should be able to recover my username - [4]
   e. Notifications bar to display important messages - [4]
2. Time table management
   a. As faculty, I must be abe to view my timetable and my class timetable [6]
   b. As a faculty, I must be able to download and save the timetable [4]
   c. As a faculty, I must be able to create a To-do list [4]
   d. faculty must be able to sync the timetable to google calendar [5]
   e. As a faculty, I must be able to view today's schedule [3]
3. Faculty and student details:
   a. As a faculty, I must be able to view my attendance and of students.
   b. As a faculty, I must download the attendance report of the student and I.
   c. As faculty, I can view marks scored by students in courses taught by me
   d. As a faculty, I must be able to view the FA warning list of my courses
   e. As a faculty, I must update the student's attendance daily
4. Courses handled by a faculty:
   a. As a faculty, I must be able to view the courses handled by me
   b. As a faculty, I must view course wise report of students
   c. Uploading resources
   d. Manage grades and marks sheets classwise
5. View student's profile:
   a. Student personal details
   b. Parent & Guardian details
   c. Course Wise Academic history
   d. Achievements and other documents
   e. As faculty, I must send a personal email to parents regarding his grades.
6. Update faculty profile and other settings:
   a. Change username
   b. Change password
   c. Help/Contact us
   d. Update personal details
   e. Customize Appearance
7. Continuous Integration and Deployment:
   a. Setting up the pipeline
   b. CI for cloning the git repository
   c. CI for Unit testing
   d. CI for Static code analysis
   e. CI for Selenium User Acceptance testing
   f. CD by Git push to netlify

JIRA REPORT

Summary	Status	Priority	Assignee	Sprint	Custom field (Story Points)
To test using SonarQube	Done	Medium	CB.EN.U4CSE 17453	EFD Sprint 1	
To test using SonarQube	Done	Medium	CB.EN.U4CSE 17458	EFD Sprint 1	
To send a message to parents if the student falls under FA warning list and has 3 or more indisciplinary actions filed against him/her	Done	Medium	CB.EN.U4CSE 17458	EFD Sprint 1	
Creation of a database for updating attendance	Done	Medium	CB.EN.U4CSE 17458	EFD Sprint 1	
To perform Unit testing	Done	Medium	CB.EN.U4CSE 17453	EFD Sprint 1	
To test the User Interface	Done	Medium	CB.EN.U4CSE 17420	EFD Sprint 1	
To perform Unit testing	Done	Medium	CB.EN.U4CSE 17458	EFD Sprint 1	
To select faculty/class wise report from the database	Done	Medium	CB.EN.U4CSE 17458	EFD Sprint 1	

To test the User Interface	Done	Medium	CB.EN.U4CSE 17424	EFD Sprint 1	
To perform Unit testing for validating the registered user details	Done	Medium	CB.EN.U4CSE 17445	EFD Sprint 1	
To test the User Interface for Registration page	Done	Medium	CB.EN.U4CSE 17424	EFD Sprint 1	
To perform Unit testing for validating the user details	Done	Medium	CB.EN.U4CSE 17453	EFD Sprint 1	
To test the User Interface for login page	Done	Medium	CB.EN.U4CSE 17420	EFD Sprint 1	
Update the new password to the database	Done	Medium	CB.EN.U4CSE 17458	EFD Sprint 1	
Validate the input fields	Done	Medium	CB.EN.U4CSE 17451	EFD Sprint 1	
Send email verification to the user	Done	Medium	CB.EN.U4CSE 17458	EFD Sprint 1	
password reset link webpage	Done	Medium	CB.EN.U4CSE 17453	EFD Sprint 1	
Display Forgot password page with procedures to reset password	Done	Medium	CB.EN.U4CSE 17451	EFD Sprint 1	

User name reset link webpage	Done	Medium	CB.EN.U4CSE 17453	EFD Sprint 1	
Update the new username to the database	Done	Medium	CB.EN.U4CSE 17445	EFD Sprint 1	
Send email verification to the user	Done	Medium	CB.EN.U4CSE 17458	EFD Sprint 1	
Validate the input fields	Done	Medium	CB.EN.U4CSE 17451	EFD Sprint 1	
Display Forgot Username page with procedures to reset id	Done	Medium	CB.EN.U4CSE 17451	EFD Sprint 1	
Show registration successful page.	Done	Medium	CB.EN.U4CSE 17445	EFD Sprint 1	
Validation of input field details in frontend.	Done	Medium	CB.EN.U4CSE 17451	EFD Sprint 1	
Email or OTP verification to verify the mobile number and email id.	Done	Medium	CB.EN.U4CSE 17458	EFD Sprint 1	
Add the registration details of the user to the database.	Done	Medium	CB.EN.U4CSE 17445	EFD Sprint 1	
Show download confirmation page	Done	Medium	CB.EN.U4CSE 17445	EFD Sprint 1	

Handle the download button	Done	Medium	CB.EN.U4CSE17458	EFD Sprint 1	
Creation of PDF/JPG format	Done	Medium	CB.EN.U4CSE17453	EFD Sprint 1	
Show FA List	Done	Medium	CB.EN.U4CSE17451	EFD Sprint 1	
Generate FA List	Done	Medium	CB.EN.U4CSE17453	EFD Sprint 1	
Create MySQL database with appropriate access permissions to store registration details	Done	Medium	CB.EN.U4CSE17453	EFD Sprint 1	
Review marks and export as PDF	Done	Medium	CB.EN.U4CSE17458	EFD Sprint 1	
Design an UI for the registration page.	Done	Medium	CB.EN.U4CSE17451	EFD Sprint 1	
Visualize student marks and class mean	To Do	Medium	CB.EN.U4CSE17451	EFD Sprint 1	
Update student marks	Done	Medium	CB.EN.U4CSE17458	EFD Sprint 1	
Add a remember me button to save username and password for a month	Done	Medium	CB.EN.U4CSE17453	EFD Sprint 1	
Display individual student marks	Done	Medium	CB.EN.U4CSE17451	EFD Sprint 1	

Send attendance report to email ID	Done	Medium	CB.EN.U4CSE 17458	EFD Sprint 1	
Sign out button after logged in	Done	Medium	CB.EN.U4CSE 17445	EFD Sprint 1	
Show the download confirmation page	To Do	Medium	CB.EN.U4CSE 17445	EFD Sprint 1	
Forgot username button and functionality	Done	Medium	CB.EN.U4CSE 17458	EFD Sprint 1	
Forgot password button and functionality	Done	Medium	CB.EN.U4CSE 17458	EFD Sprint 1	
Handle the download button	Done	Medium	CB.EN.U4CSE 17453	EFD Sprint 1	
Creation of PDF/JPG format	Done	Medium	CB.EN.U4CSE 17453	EFD Sprint 1	
Display the selected report	Done	Medium	CB.EN.U4CSE 17451	EFD Sprint 1	
Validation of login details in frontend and backend.	Done	Medium	CB.EN.U4CSE 17445	EFD Sprint 1	
Select Faculty/Student Report	To Do	Medium	CB.EN.U4CSE 17445	EFD Sprint 1	
Visualize and display total attendance	Done	Medium	CB.EN.U4CSE 17451	EFD Sprint 1	

Calculate the number of hours for student to take leave	Done	Medium	CB.EN.U4CSE17453	EFD Sprint 1	
Calculate the number of hours for faculty to take leave	Done	Medium	CB.EN.U4CSE17453	EFD Sprint 1	
Visualize attendance class-wise	To Do	Medium	CB.EN.U4CSE17451	EFD Sprint 1	
Create an UI to display the Login page	Done	Medium	CB.EN.U4CSE17451	EFD Sprint 1	
Design a front-end to show summary	Done	Medium	CB.EN.U4CSE17451	EFD Sprint 1	
As a faculty, I should be able to recover my password (Forgot password)	Done	Medium		EFD Sprint 1	3
As a faculty, I should be able to recover my username (Forgot username)	Done	Medium		EFD Sprint 1	3
As a faculty, I must be able to view FA warning list of my courses	Done	Medium		EFD Sprint 1	5
As a faculty, I must view marks scored by students in each course handled by me	To Do	Medium		EFD Sprint 1	5

As a faculty, I must download attendance report of faculty and student	To Do	Medium		EFD Sprint 1	4
As a faculty, I must be able to view my attendance summary and of students	To Do	High		EFD Sprint 1	5
As a faculty, I should be able to register by providing the required details.	Done	High		EFD Sprint 1	5

As a faculty, I should be able to Log in to my account.	Done	Highest		EFD Sprint 1	6
As a faculty, I must be able to view my attendance summary and of students	To Do	High		EFD Sprint 1	5
Add/Remove List	Done	Medium	CB.EN.U4CSE17453	EFD Sprint 2	
Creation and updating of to-do databases	Done	Medium	CB.EN.U4CSE17453	EFD Sprint 2	
To perform Unit testing	Done	Medium	CB.EN.U4CSE17453	EFD Sprint 2	
Saving the message as a draft	In Progress	Medium	CB.EN.U4CSE17451	EFD Sprint 2	

Compose and send email	In Progress	Medium	CB.EN.U4CSE 17453	EFD Sprint 2	
Retrieve the details from the database	Done	Medium	CB.EN.U4CSE 17453	EFD Sprint 2	
To display page	Done	Medium	CB.EN.U4CSE 17451	EFD Sprint 2	
CI/CD	Done	High	CB.EN.U4CSE 17458	EFD Sprint 2	5
Creating deployment build	Done	High	CB.EN.U4CSE 17458	EFD Sprint 2	1
Creating pipeline for Unit testing	Done	High	CB.EN.U4CSE 17458	EFD Sprint 2	1
Creating pipeline for Static Code analysis	Done	High	CB.EN.U4CSE 17458	EFD Sprint 2	1
Creating pipeline for UI testing	In Progress	High	CB.EN.U4CSE 17445	EFD Sprint 2	1
Setting up the pipeline	Done	High	CB.EN.U4CSE 17458	EFD Sprint 2	1
To perform CI/CD	Done	High	CB.EN.U4CSE 17458	EFD Sprint 2	5
To perform Unit testing	Done	Medium	CB.EN.U4CSE 17453	EFD Sprint 2	
Show download confirmation page	Done	Medium	CB.EN.U4CSE 17451	EFD Sprint 2	

Export data	Done	Medium	CB.EN.U4CSE 17453	EFD Sprint 2	
Updation of database	Done	Medium	CB.EN.U4CSE 17451	EFD Sprint 2	
To edit the details	Done	Medium	CB.EN.U4CSE 17453	EFD Sprint 2	
Retrieve from the databases	Done	Medium	CB.EN.U4CSE 17451	EFD Sprint 2	
To display page	Done	Medium	CB.EN.U4CSE 17453	EFD Sprint 2	
Creation of s3 bucket to store resources	Done	Medium	CB.EN.U4CSE 17453	EFD Sprint 2	
Display uploaded resources	In Progress	Medium	CB.EN.U4CSE 17451	EFD Sprint 2	
Handling uploading files and uploading in cloud	In Progress	Medium	CB.EN.U4CSE 17451	EFD Sprint 2	
Upload button	In Progress	Medium	CB.EN.U4CSE 17453	EFD Sprint 2	
Select courses / classes	In Progress	Medium	CB.EN.U4CSE 17453	EFD Sprint 2	
Create UI to upload resources	In Progress	Medium	CB.EN.U4CSE 17451	EFD Sprint 2	

Show download confirmation page	Done	Medium	CB.EN.U4CSE 17451	EFD Sprint 2	
Download the course details	In Progress	Medium	CB.EN.U4CSE 17453	EFD Sprint 2	
Button to download the course details	Done	Medium	CB.EN.U4CSE 17451	EFD Sprint 2	
Retrieve course details from the database	To Do	Medium	CB.EN.U4CSE 17453	EFD Sprint 2	
Button to select a course	To Do	Medium	CB.EN.U4CSE 17451	EFD Sprint 2	
Display the list Courses	To Do	Medium	CB.EN.U4CSE 17453	EFD Sprint 2	
Display To-Do Lists	To Do	Medium	CB.EN.U4CSE 17451	EFD Sprint 2	
Send it as notification/reminders	To Do	Medium	CB.EN.U4CSE 17451	EFD Sprint 2	

UI elements to go to the download section	Done	Medium	CB.EN.U4CSE 17453	EFD Sprint 2	
Handle exceptions if timetable not found	Done	Medium	CB.EN.U4CSE 17451	EFD Sprint 2	
To retrieve the timetable from database	Done	Medium	CB.EN.U4CSE 17453	EFD Sprint 2	
Design UI to display the timetable	Done	Medium	CB.EN.U4CSE 17451	EFD Sprint 2	

Creation of a database of time tables	Done	Medium	CB.EN.U4CSE 17453	EFD Sprint 2	
Select the timetable	Done	Medium	CB.EN.U4CSE 17453	EFD Sprint 2	
To retrieve the timetable from database	Done	Medium	CB.EN.U4CSE 17451	EFD Sprint 2	
Create a PDF/JPEG format of the timetable	Done	Medium	CB.EN.U4CSE 17453	EFD Sprint 2	
To download the timetable	Done	Medium	CB.EN.U4CSE 17451	EFD Sprint 2	
UI to show download successful	Done	Medium	CB.EN.U4CSE 17453	EFD Sprint 2	
To test the User Interface for displaying Time table	Done	Medium	CB.EN.U4CSE 17420	EFD Sprint 2	
To test the User Interface for displaying Time table	Done	Medium	CB.EN.U4CSE 17424	EFD Sprint 2	
To test the User Interface	Done	Medium	CB.EN.U4CSE 17420	EFD Sprint 2	
As a faculty, I must keep track of students achievements and other documents	To Do	Medium			4

As a faculty, I must manage students personal details	To Do	Med ium			4
As a faculty, I must be able to view my timetable and my class's timetable	To Do	Med ium		EFD Sprint 2	4
As a faculty, I must be able to view the academic history of the student	To Do	Med ium			4
As a faculty, I must be able to contact administrator for help	To Do	Med ium			
As a faculty, I must be able to send a personal email to the student regarding his/her performance	To Do	Med ium		EFD Sprint 2	4
As a faculty, I must be able to view the details of Parents/Guardian	To Do	Med ium		EFD Sprint 2	4
As a faculty, I must be able to view the courses handled by me	To Do	Med ium		EFD Sprint 2	5
As a faculty, I must be able to view resources	To Do	Med ium		EFD Sprint 2	4

As a faculty, I must be able to download and save the timetable	To Do	Med ium		EFD Sprint 2	4
As a faculty, I must be able to create a To-do list	To Do	Med ium		EFD Sprint 2	4
As a faculty, I must be able to change password	To Do	Med ium			
As a faculty, I must download attendance report of faculty and student	To Do	Med ium		EFD Sprint 1	4
As a faculty, I must view marks scored by students in each course handled by me	To Do	Med ium		EFD Sprint 1	5
Visualize attendance class-wise	To Do	Med ium	CB.EN.U4CSE 17451	EFD Sprint 1	
Show the download confirmation page	To Do	Med ium	CB.EN.U4CSE 17445	EFD Sprint 1	
Select Faculty/Student Report	To Do	Med ium	CB.EN.U4CSE 17445	EFD Sprint 1	
As a faculty, I must view course wise report of students	To Do	Med ium			
As a faculty, I must be able to customize the	To Do	Med ium			

dashboard appearance					
As a faculty, I must be able to update personal details	To Do	Med ium			
As a faculty, I must be able to change username	To Do	Med ium			
As a faculty, I must be able to view today's schedule	To Do	Med ium			3
faculty must be able to sync the timetable to google calendar	To Do	Med ium			5
Faculty should be able to view, track and monitor his/her timetables.	To Do	Med ium			
Attendance summary, report, marks, etc.	To Do	Med ium			
Upload resources, manage grades, mark sheets, etc	To Do	Med ium			
Visualize student marks and class mean	To Do	Med ium	CB.EN.U4CSE 17451	EFD Sprint 1	
To test the User Interface	To Do	Med ium			
To test the user interface	To Do	Med ium			

Sending the message as an email to the administrator	To Do	Med ium			
To test the user interface	To Do	Med ium			
To perform unit testing	To Do	Med ium			
To perform Unit Testing	To Do	Med ium			
To perform Unit Testing	To Do	Med ium			
To test the User Interface	To Do	Med ium			
To perform Unit Testing	To Do	Med ium			
To display grades	To Do	Med ium			
To perform Unit testing	To Do	Med ium			
To test the User Interface	To Do	Med ium			
To display Class-wise report	To Do	Med ium			
To perform Unit testing	To Do	Med ium			

To test the User Interface	To Do	Med ium			
To perform Unit testing	To Do	Med ium			
To test the User Interface	To Do	Med ium			
Display downloading or downloaded grades	To Do	Med ium			
Handle download by selecting the correct format	To Do	Med ium			

To perform Unit testing	To Do	Med ium			
To test the User Interface	To Do	Med ium			
Button to download the report	To Do	Med ium			
To retrieve the timetable from database	To Do	Med ium			
Add, remove or modify contents in database	To Do	Med ium			
Button to show/add grades of the student	To Do	Med ium			
Display course progress	To Do	Med ium			

Visualization of grades with class-wise comparison	To Do	Med ium			
Grading displayed class-wise	To Do	Med ium			
Display lesson plan class-wise	To Do	Med ium			
Visualization for class-wise reports	To Do	Med ium			
To perform Unit testing	To Do	Med ium			
To test the User Interface for the notification panel	To Do	Med ium			
Get current date and time to retrieve the details	To Do	Med ium			
Retrieve the schedule from database	To Do	Med ium			
UI to show today's classes and respective timings	To Do	Med ium			
Webpage to guide the user to use Google calendar	To Do	Med ium			
Show confirmation/error page	To Do	Med ium			

Implement connecting timetable to google calendar	To Do	Med ium			
Get confirmation from the user using the checkbox	To Do	Med ium			
Select the timetable	To Do	Med ium			
Show downloading/downloade d confirmation page	To Do	Med ium			
Button and code to download the details	To Do	Med ium			
Change user photo	To Do	Med ium			
Change display size	To Do	Med ium			
Backend code to change from light theme to the dark theme of web app	To Do	Med ium			
Display the forms	To Do	Med ium			
Update database	To Do	Med ium			
Backend to send the new details to database	To Do	Med ium			
Design form to enter new details	To Do	Med ium			

Validate using email	To Do	Med ium			
Validate using OTP	To Do	Med ium			
Display personal details	To Do	Med ium			
Add it to contact logs in the database	To Do	Med ium			
Input validation	To Do	Med ium			
Design contact form	To Do	Med ium			
Contact page	To Do	Med ium			
Creation of a contact-us database	To Do	Med ium			
Send confirmation email	To Do	Med ium			
Send confirmation OTP	To Do	Med ium			
Update the new password in the database	To Do	Med ium			
Validate the new password	To Do	Med ium			
Validate the current password	To Do	Med ium			

Change password - frontend	To Do	Med ium			
Update database with new username	To Do	Med ium			
Change username after confirmation email is clicked	To Do	Med ium			
Send confirmation email	To Do	Med ium			
Send confirmation OTP	To Do	Med ium			
Validate current username	To Do	Med ium			
Design icon	To Do	Med ium			
Wrap the dropdown menu	To Do	Med ium			
Handle the dropdown button	To Do	Med ium			
Display announcements	To Do	Med ium			
Change username - UI	To Do	Med ium			
Dropdown button design	To Do	Med ium			

Design notification page	To Do	Med ium			
Button to show and handle the achievements report	To Do	Med ium			
To add student achievement to database	To Do	Med ium			
To create display page	To Do	Med ium			
Updation of database	To Do	Med ium			
To edit the details	To Do	Med ium			
Retrieve from the databases	To Do	Med ium			
Button to download the report	To Do	Med ium			
Downloading/downloade d/error messages to user	To Do	Med ium			
Save the Image in cloud storage bucket	To Do	Med ium			
Creating and update databases for OCR	To Do	Med ium			
Confirm button	To Do	Med ium			

Updation of database	To Do	Med ium			
To edit the details	To Do	Med ium			
Retrieve from the databases	To Do	Med ium			
Display Course Wise Academic history	To Do	Med ium			
Display a loading page	To Do	Med ium			
Button to upload an image	To Do	Med ium			
OCR Algorithm	To Do	Med ium			
Show download confirmation page	To Do	Med ium			
Export the data	To Do	Med ium			
Update the database	To Do	Med ium			
Edit the details	To Do	Med ium			
Retrieve details from the database	To Do	Med ium			
Display the Student details page	To Do	Med ium			

To manage the details of the students	To Do	Med ium			
To update the faculty details	To Do	Med ium			
As a faculty, I must manage grades and mark sheet class wise	To Do	Med ium			
Faculty should be able to register and login into the account.	To Do	Med ium			
A notification bar to view recent announcements and other important details.	To Do	Low est			3
As a faculty, I must update the students attendance daily	To Do	Low est			5

SPRINT 1 & 2 STANDUP NOTES

AMRITA VISHWA VIDYAPEETHAM Academic Year 2019-20

Semester: VI Sem   B.Tech 15CSE313 SOFTWARE ENGINEERING
STANDUP NOTES 1 on 31/12/2019

Section: E   Project Title: Faculty Dashboard

Name	Question	Answer	Sign
Member 1 Saiharsha B	What did you do yesterday?	Decided on the tech stack (Web)	
	What will you do today?	Read docs and learn VueJS	
	Are there any impediments in your way?	VueJS' complicated folder structuring	
Member 2 Sanjay Tharagesh RS	What did you do yesterday?	Zoom meeting with developers	
	What will you do today?	Deciding the framework, Cloud services, and functionalities	
	Are there any impediments in your way?	None.	
Member 3 Srishilesh P S	What did you do yesterday?	Worked on deciding best tech stack	
	What will you do today?	Decided on frameworks and cloud services	
	Are there any impediments in your way?	None	
Member 4 Neeraj Kumar Reddy P	What did you do yesterday?	Explored about jasmine framework for testing	
	What will you do today?	Learning more about test phases in jasmine	
	Are there any impediments in your way?	None	
Member 5 Ilam Prathyusha	What did you do yesterday?	Explored about Selenium framework for testing	
	What will you do today?	Learn more about web drivers	
	Are there any impediments in your way?	None	
Member 6 Gayathri E	What did you do yesterday?	Explored about Selenium framework for testing	
	What will you do today?	Learn more about web drivers	
	Are there any impediments in your way?	None	

AMRITA VISHWA VIDYAPEETHAM Academic Year 2019-20

Semester: VI Sem   B.Tech 15CSE313 SOFTWARE ENGINEERING
STANDUP NOTES 2 on 7/01/2020

Section: E   Project Title: Faculty Dashboard

Name	Question	Answer	Sign
Member 1 Saiharsha B	What did you do yesterday?	Integrated NuxtJS to the project and refactored code	
	What will you do today?	Start working on the front-end	
	Are there any impediments in your way?	None	
Member 2 Sanjay Tharagesh RS	What did you do yesterday?	Checked final requirements and understood the workflow.	
	What will you do today?	Meeting with developers for final discussion.	
	Are there any impediments in your way?	None	
Member 3 Srishilesh P S	What did you do yesterday?	Finalized the cloud services.	
	What will you do today?	Zoom meeting with developers for final discussion on requirements and functionalities.	
	Are there any impediments in your way?	None	
Member 4 Neeraj Kumar Reddy P	What did you do yesterday?	Started learning JavaScript	
	What will you do today?	Moving on to JavaScript frameworks	
	Are there any impediments in your way?	None	
Member 5 Ilam Prathyusha	What did you do yesterday?	Learning about web drivers	
	What will you do today?	Learn a little bit of JavaScript	
	Are there any impediments in your way?	None	
Member 6 Gayathri E	What did you do yesterday?	Tried installing Selenium	
	What will you do today?	Learn a little bit of JavaScript	
	Are there any impediments in your way?	None	

Semester: VI Sem   B.Tech 15CSE313 SOFTWARE ENGINEERING
STANDUP NOTES 3 on 21/01/2020

Section: E   Project Title: Faculty Dashboard

Name	Question	Answer	Sign
Member 1 Saiharsha B	What did you do yesterday?	Completed the UI for Login Epic	
	What will you do today?	Will start working on UI for Faculty Details	
	Are there any impediments in your way?	Having difficulties in integrating the auth module for login	
Member 2 Sanjay Tharagesh RS	What did you do yesterday?	Completed the backend for login and authentication	
	What will you do today?	Setting up Jasmine for unit testing	
	Are there any impediments in your way?	Setting up cloud services for hosting the API	
Member 3 Srishilesh P S	What did you do yesterday?	Completed backend for faculty and student details	
	What will you do today?	SonarQube static code analysis	
	Are there any impediments in your way?	Had JVM version mismatches during SonarQube installation	
Member 4 Neeraj Kumar Reddy P	What did you do yesterday?	Set up and learned the basics of Selenium	
	What will you do today?	Since I've switched to the developer team, I'm learning VueJS	
	Are there any impediments in your way?	Since I'm new to JavaScript frameworks, I'm taking some time to understand it	
Member 5 Ilam Prathyusha	What did you do yesterday?	Set up a Selenium framework for testing and learned to use it	
	What will you do today?	Find test cases for the registration module	
	Are there any impediments in your way?	Facing difficulty while implementing automated cross-browser testing	
Member 6 Gayathri E	What did you do yesterday?	Set up Selenium framework and worked on the backend for login	
	What will you do today?	Find test cases for the login module	
	Are there any impediments in your way?	Facing difficulty while performing testing using chrome	

AMRITA VISHWA VIDYAPEETHAM Academic Year 2019-20

Semester: VI Sem   B.Tech 15CSE313 SOFTWARE ENGINEERING
STANDUP NOTES 4 on 24/01/2020

Section: E   Project Title: Faculty Dashboard

Name	Question	Answer	Sign
Member 1 Saiharsha B	What did you do yesterday?	Completed some tasks for faculty details	
	What will you do today?	Will fix bugs with auth module	
	Are there any impediments in your way?	Having difficulties in using JSON tokens	
Member 2 Sanjay Tharagesh RS	What did you do yesterday?	Completed unit testing for login and authentication	
	What will you do today?	Start working on the faculty attendance portal - setting up databases and queries to update and modify.	
	Are there any impediments in your way?	Difficulties in managing SQL pool connection sessions.	
Member 3 Srishilesh P S	What did you do yesterday?	Completed unit testing for Faculty attendance portal	
	What will you do today?	Integrate Backend with frontend	
	Are there any impediments in your way?	Having difficulties in sending API response to frontend	
Member 4 Neeraj Kumar Reddy P	What did you do yesterday?	Designed front-end for student marks	
	What will you do today?	Attendance summary story	
	Are there any impediments in your way?	Difficulty in understanding the Vue Lifecycle	
Member 5 Ilam Prathyusha	What did you do yesterday?	Found various test cases for the register page.	
	What will you do today?	Manual testing for the test cases found.	
	Are there any impediments in your way?	None	
Member 6 Gayathri E	What did you do yesterday?	Found various test cases for the login page.	
	What will you do today?	Manual testing for the test cases found.	
	Are there any impediments in your way?	None	

AMRITA VISHWA VIDYAPEETHAM Academic Year 2019-20

Semester: VI Sem   B.Tech 15CSE313 SOFTWARE ENGINEERING
STANDUP NOTES 5 on 28/01/2020

Section: E   Project Title: Faculty Dashboard

Name	Question	Answer	Sign
Member 1 Saiharsha B	What did you do yesterday?	Have started work on auth module	
	What will you do today?	Run backend in the local system and test request and responses	
	Are there any impediments in your way?	None	
Member 2 Sanjay Tharagesh RS	What did you do yesterday?	Set up faculty attendance database	
	What will you do today?	Start working on email verification through OTP	
	Are there any impediments in your way?	Azure credits running low	
Member 3 Srishilesh P S	What did you do yesterday?	Completed Unit testing	
	What will you do today?	Changing Azure DBS to Azure MySql	
	Are there any impediments in your way?	Losing Azure credits faster, so changing database	
Member 4 Neeraj Kumar Reddy P	What did you do yesterday?	Started work on attendance summary	
	What will you do today?	Help in integrating front-end and backend	
	Are there any impediments in your way?	None	
Member 5 Ilam Prathyusha	What did you do yesterday?	Basic testing using selenium IDE	
	What will you do today?	tried implementing manual testing for register page	
	Are there any impediments in your way?	None	
Member 6 Gayathri E	What did you do yesterday?	Basic testing using selenium webdriver	
	What will you do today?	tried implementing automated testing for google search	
	Are there any impediments in your way?	None	

AMRITA VISHWA VIDYAPEETHAM Academic Year 2019-20

Semester: VI Sem   B.Tech 15CSE313 SOFTWARE ENGINEERING
STANDUP NOTES 6 on 31/01/2020

Section: E   Project Title: Faculty Dashboard

Name	Question	Answer	Sign
Member 1 Saiharsha B	What did you do yesterday?	Completed work on auth module	
	What will you do today?	Work on Attendance report - frontend	
	Are there any impediments in your way?	Issues regarding NuxtJS Configuration	
Member 2 Sanjay Tharagesh RS	What did you do yesterday?	Completed working on email verification through OTP and use-case diagram for forgot-password.	
	What will you do today?	Start working on forgot password link backend	
	Are there any impediments in your way?	Installing Java11	
Member 3 Srishilesh P S	What did you do yesterday?	Completed Unit Testing	
	What will you do today?	Changing database from Azure to AWS	
	Are there any impediments in your way?	Completion of front end for integration	
Member 4 Neeraj Kumar Reddy P	What did you do yesterday?	Completed work on faculty story	
	What will you do today?	Going to start work on student story	
	Are there any impediments in your way?	None	
Member 5 Ilam Prathyusha	What did you do yesterday?	Learnt how to automated testing is done	
	What will you do today?	Started working on automated testing for the registration page.	
	Are there any impediments in your way?	Prompting for re-installation of driver again during execution	
Member 6 Gayathri E	What did you do yesterday?	Learnt how to automated testing is done	
	What will you do today?	Started working on automated testing for the login page	
	Are there any impediments in your way?	Running multiple test cases simultaneously	

AMRITA VISHWA VIDYAPEETHAM Academic Year 2019-20

Semester: VI Sem   B.Tech 15CSE313 SOFTWARE ENGINEERING
STANDUP NOTES 7 on 04/02/2020

Section: E   Project Title: Faculty Dashboard

Name	Question	Answer	Sign
Member 1 Saiharsha B	What did you do yesterday?	Work almost done on attendance report	
	What will you do today?	Start student details - front end	
	Are there any impediments in your way?	Working with promises in front-end	
Member 2 Sanjay Tharagesh RS	What did you do yesterday?	Completed forgot password reset function and use case diagram for the same.	
	What will you do today?	Set up SonarQube for static code analysis	
	Are there any impediments in your way?	Working with async functions	
Member 3 Srishilesh P S	What did you do yesterday?	Changed Azure to AWS	
	What will you do today?	Integration with front end	
	Are there any impediments in your way?	Front end completion for epic 2	
Member 4 Neeraj Kumar Reddy P	What did you do yesterday?	Finish work on student story	
	What will you do today?	Helping with integration	
	Are there any impediments in your way?	None	
Member 5 Ilam Prathyusha	What did you do yesterday?	Automated a few basic test cases for the register page.	
	What will you do today?	Exploring some more test cases possible for the register page.	
	Are there any impediments in your way?	None	
Member 6 Gayathri E	What did you do yesterday?	Automated a few basic test cases for the register page.	
	What will you do today?	Exploring some more test cases possible for the login page.	
	Are there any impediments in your way?	None	

AMRITA VISHWA VIDYAPEETHAM Academic Year 2019-20

Semester: VI Sem   B.Tech 15CSE313 SOFTWARE ENGINEERING
STANDUP NOTES 8 on 07/02/2020

Section: E   Project Title: Faculty Dashboard

Name	Question	Answer	Sign
Member 1 Saiharsha B	What did you do yesterday?	Finish Notification module in the front-end	
	What will you do today?	Work on Class Details route	
	Are there any impediments in your way?	Figuring out the response payload from the back-end	
Member 2 Sanjay Tharagesh RS	What did you do yesterday?	Finished setting up SOnarQube	
	What will you do today?	Start working on static code analysis and unit testing	
	Are there any impediments in your way?	SonarQube and Java version clash	
Member 3 Srishilesh P S	What did you do yesterday?	Integration with front end	
	What will you do today?	Developer documentation for Use case diagram	
	Are there any impediments in your way?	None	
Member 4 Neeraj Kumar Reddy P	What did you do yesterday?	Helped with UI integration	
	What will you do today?	Work on UI enhancement	
	Are there any impediments in your way?	None	
Member 5 Ilam Prathyusha	What did you do yesterday?	Checking for valid details being entered in the register page.	
	What will you do today?	Learn how to check if a prompt occurs when validation is not met	
	Are there any impediments in your way?	Accessing the buttons in the webpage	
Member 6 Gayathri E	What did you do yesterday?	Checking for valid details being entered in the login page.	
	What will you do today?	Learn how to check if a prompt occurs when invalid email is given	
	Are there any impediments in your way?	Accessing the buttons in the webpage	

AMRITA VISHWA VIDYAPEETHAM Academic Year 2019-20

Semester: VI Sem   B.Tech 15CSE313 SOFTWARE ENGINEERING
STANDUP NOTES 9 on 11/02/2020

Section: E   Project Title: Faculty Dashboard

Name	Question	Answer	Sign
Member 1 Saiharsha B	What did you do yesterday?	Almost completed front-end work for Faculty-Class epics	
	What will you do today?	Add Material Design Icons to front-end to improve UX	
	Are there any impediments in your way?	None	
Member 2 Sanjay Tharagesh RS	What did you do yesterday?	Completed SonarQube analysis and created test cases for unit testing for student mark checking.	
	What will you do today?	Will complete unit-testing	
	Are there any impediments in your way?	None	
Member 3 Srishilesh P S	What did you do yesterday?	Completed use case diagram	
	What will you do today?	To complete the Data Flow Diagram	
	Are there any impediments in your way?	None	
Member 4 Neeraj Kumar Reddy P	What did you do yesterday?	Worked on UI enhancement	
	What will you do today?	Add some CSS styling	
	Are there any impediments in your way?	None	
Member 5 Ilam Prathyusha	What did you do yesterday?	When all details entered correctly, check for successful registration.	
	What will you do today?	Automating some of the negative and boundary test cases.	
	Are there any impediments in your way?	None	
Member 6 Gayathri E	What did you do yesterday?	Check for successful login with correct credentials	
	What will you do today?	Automating some of the negative and boundary test cases.	
	Are there any impediments in your way?	None	

Semester: VI Sem   B.Tech 15CSE313 SOFTWARE ENGINEERING
STANDUP NOTES 10 on 14/02/2020

Section: E   Project Title: Faculty Dashboard

Name	Question	Answer	Sign
Member 1 Saiharsha B	What did you do yesterday?	Added Icons and Styling	
	What will you do today?	Complete work on front-end	
	Are there any impediments in your way?	None	
Member 2 Sanjay Tharagesh RS	What did you do yesterday?	Completed unit testing and working middleware for direct file downloads bypassing the cloud storage.	
	What will you do today?	Set up the middleware and complete the DFD diagram.	
	Are there any impediments in your way?	POST request error in unit testing.	
Member 3 Srishilesh P S	What did you do yesterday?	Completed Data Flow Diagram	
	What will you do today?	Integration with front end	
	Are there any impediments in your way?	Integration with Front end	
Member 4 Neeraj Kumar Reddy P	What did you do yesterday?	Worked on CSS Styling	
	What will you do today?	Help with middleware	
	Are there any impediments in your way?	None	
Member 5 Ilam Prathyusha	What did you do yesterday?	Tried a few test cases regarding the register page.	
	What will you do today?	Test the user interface to view attendance report	
	Are there any impediments in your way?	Multiple test cases simultaneous execution	
Member 6 Gayathri E	What did you do yesterday?	Tried a few test cases regarding the login page.	
	What will you do today?	Test the user interface to download attendance report	
	Are there any impediments in your way?	Prompting for re-installation of driver again during execution	

AMRITA VISHWA VIDYAPEETHAM Academic Year 2019-20

Semester: VI Sem   B.Tech 15CSE313 SOFTWARE ENGINEERING
STANDUP NOTES 11 on 03/03/2020

Section: E   Project Title: Faculty Dashboard

Name	Question	Answer	Sign
Member 1 Saiharsha B	What did you do yesterday?	Planned for sprint 2	
	What will you do today?	Testing out different CSS frameworks	
	Are there any impediments in your way?	None	
Member 2 Sanjay Tharagesh RS	What did you do yesterday?	Plan sprint 2 outline	
	What will you do today?	Start working on backend	
	Are there any impediments in your way?	To learn code coverage	
Member 3 Srishilesh P S	What did you do yesterday?	Plan the CI/CD pipeline structure	
	What will you do today?	Installation of Jenkins	
	Are there any impediments in your way?	Learn about DevOps	
Member 4 Neeraj Kumar Reddy P	What did you do yesterday?	Working on the installation of Jenkins	
	What will you do today?	Learning the basics of DevOps	
	Are there any impediments in your way?	Installation of Jenkins	
Member 5 Ilam Prathyusha	What did you do yesterday?	Improved automated test cases for the registration page.	
	What will you do today?	Planning test cases regarding the Attendance Report.	
	Are there any impediments in your way?	None.	
Member 6 Gayathri E	What did you do yesterday?	Improved automated test cases for the login page.	
	What will you do today?	Planning test cases regarding the to-do list.	
	Are there any impediments in your way?	None.	

AMRITA VISHWA VIDYAPEETHAM Academic Year 2019-20

Semester: VI Sem   B.Tech 15CSE313 SOFTWARE ENGINEERING
STANDUP NOTES 12 on 06/03/2020

Section: E   Project Title: Faculty Dashboard

Name	Question	Answer	Sign
Member 1 Saiharsha B	What did you do yesterday?	Experimented with Bulma CSS for styling	
	What will you do today?	Will add some Sass stylesheets	
	Are there any impediments in your way?	None	
Member 2 Sanjay Tharagesh RS	What did you do yesterday?	Completed a few modules in backend	
	What will you do today?	Unit test the modules	
	Are there any impediments in your way?	Installing code coverage packages	
Member 3 Srishilesh P S	What did you do yesterday?	Finished Jenkins and other plugins installation	
	What will you do today?	Created pipeline and wrote pipeline script to clone the git repository and install dependencies	
	Are there any impediments in your way?	No	
Member 4 Neeraj Kumar Reddy P	What did you do yesterday?	Browsing for required plugins	
	What will you do today?	Working on understanding Basic pipeline script components	
	Are there any impediments in your way?	Pipeline functionalities	
Member 5 Ilam Prathyusha	What did you do yesterday?	Got few test cases on retrieving Attendance Report	
	What will you do today?	Automating the planned test cases on the Attendance report	
	Are there any impediments in your way?	None	
Member 6 Gayathri E	What did you do yesterday?	Got a few test cases on working with the to-do list	
	What will you do today?	Automating the planned test cases on the to-do list	
	Are there any impediments in your way?	None	

AMRITA VISHWA VIDYAPEETHAM Academic Year 2019-20

Semester: VI Sem   B.Tech 15CSE313 SOFTWARE ENGINEERING
STANDUP NOTES 13 on 10/03/2020

Section: E   Project Title: Faculty Dashboard

Name	Question	Answer	Sign
Member 1 Saiharsha B	What did you do yesterday?	Added some sass stylesheets	
	What will you do today?	Fix errors with SPA rendering when hosted	
	Are there any impediments in your way?	SPAs do not render properly on hosting services	
Member 2 Sanjay Tharagesh RS	What did you do yesterday?	Completed unit testing and code coverage	
	What will you do today?	Discuss with DevOps to migrate code coverage to sonarqube	
	Are there any impediments in your way?	Making code coverage > 80 %	
Member 3 Srishilesh P S	What did you do yesterday?	Wrote pipeline script for Unit testing	
	What will you do today?	Run unit testing and generate coverage reports	
	Are there any impediments in your way?	Faced difficulty in generating the coverage reports	
Member 4 Neeraj Kumar Reddy P	What did you do yesterday?	Undergoing CI/CD process with the testing phase	
	What will you do today?	Writing pipeline scripts	
	Are there any impediments in your way?	Report failure	
Member 5 Ilam Prathyusha	What did you do yesterday?	Automated test cases on Attendance Report	
	What will you do today?	Automating the test cases for page navigation	
	Are there any impediments in your way?	None	
Member 6 Gayathri E	What did you do yesterday?	Automated test cases on the to-do list	
	What will you do today?	Automating the test cases for page scroll	
	Are there any impediments in your way?	None	

AMRITA VISHWA VIDYAPEETHAM Academic Year 2019-20

Semester: VI Sem   B.Tech 15CSE313 SOFTWARE ENGINEERING
STANDUP NOTES 14 on 13/03/2020

Section: E   Project Title: Faculty Dashboard

Name	Question	Answer	Sign
Member 1 Saiharsha B	What did you do yesterday?	Fixed SPA render issues	
	What will you do today?	Add some finishing touches to the app	
	Are there any impediments in your way?	None	
Member 2 Sanjay Tharagesh RS	What did you do yesterday?	Implemented and integrated other backend modules.	
	What will you do today?	Host server in AWS EC2	
	Are there any impediments in your way?		
Member 3 Srishilesh P S	What did you do yesterday?	Created a freestyle project for running sonarqube	
	What will you do today?	Integrating selenium and doing Continuous Deployment in the Github repository	
	Are there any impediments in your way?	Running selenium tests using chromedriver	
Member 4 Neeraj Kumar Reddy P	What did you do yesterday?	Learning how to connect Jenkins with Sonarcube	
	What will you do today?	Integrating testing connection with Jenkins	
	Are there any impediments in your way?	Deployment	
Member 5 Ilam Prathyusha	What did you do yesterday?	Automated basic test cases like buttons checking	
	What will you do today?	Improving the quality of automated test cases	
	Are there any impediments in your way?	None	
Member 6 Gayathri E	What did you do yesterday?	Automated basic test cases like scrolling	
	What will you do today?	Improving the quality of automated test cases	
	Are there any impediments in your way?	None	

Retrospective Notes

START

1. Identify proper test cases before implementation.
2. Run the server code in AWS/Azure virtual compute machine.
3. Normalize the database tables to reduce data redundancy.

STOP

1. Keep manual testing minimal.
2. Avoid last minute changes in the codebases (UI+ Backend)

CONTINUE

1. Keep doing frequent checkouts to GitHub Repository.
2. Weekly Zoom meetings among the developers.
3. Checking test cases even in Selenium IDE along with automating for accuracy.
4. Keep using the UI framework to make the task of the designer easy.
5. Do static code analysis to fix some critical security errors.
6. Writing modularized code.

Implementation

The project, Faculty Dashboard is developed keeping the core functionalities of the portal easy and elegant to use by users. The web app is built upon the latest state-of-the-art technologies with scalable cloud services with comprehensive UI testing, unit testing, and code coverage. The project is continuously integrated with Jenkins and deployed on netlify platform. The server code is modularized with appropriate functions and integrated with the frontend. Modularized code ensures effortless debugging and making the development process fast.

Sample Code

Importing packages

```
var express = require('express');
var app = express();
var mysql = require('mysql');
var bodyParser = require('body-parser');
var json2xls = require('json2xls');
```

AWS RDS credentials

```
var credentials = {
  host: "iris-aws-db.c4hq5iosxryf.us-east-1.rds.amazonaws.com",
  port: "3306",
  user: "admin",
  password: "",
  database: 'iris_db',
  ssl: true //ssl=true (TYPO)
};
```

Middlewares

```
var conn = mysql.createConnection(credentials);
app.use(json2xls.middleware);
app.use(bodyParser.urlencoded({
  extended: true
}));
```

POST requests

```
app.post("/login", function (req, res) {
  login.login(req, res, conn);
})
app.post('/register', function (req, res) {
  register.send_email_otp(req, res, conn);
  register.verifyotp(req, res, conn);
})
app.post('/forgotpassword', function (req, res) {
  forgotpassword.send_email_otp(req, res, conn);
});
app.post('/download_fa_list', function (req, res) {
  fa_report.download_fa_list(req, res, conn);
})
```

Functions

```
function download_class_attendance_report(req, res, conn) {
    let email = req.body.email;
    let date = req.body.date; // mm/dd/yyyy
    let course_code = req.body.ccode;
    let dept = req.body.dept;
    let sec = req.body.section;
    let sem = req.body.semester;
    let batch = req.body.batch;
    conn.connect(function(err) {
        // if(err) throw err
        //(0-absent, 1-present, 2-OD)
        conn.query(`select s_roll, department, section, semester, batch,
c_date, c_period, attd_status, course_code from student_attendance where
f_email=? and c_date=? and course_code=? and department=? and section=?
and semester=? and batch=?;`, [email, date, course_code, dept, sec, sem,
batch], function(error, rows, fields) {
            // if(err) throw err
            console.log('student report data retrieved');
            let date = new Date(); // add date&tijme to file name
            res.xls('student_report-'+date+'.xlsx', rows);
        })
    });
}
function calculate_free_student_leave(req, res, conn) {
    let email = req.body.email; //faculty email id
    let course_code = req.body.ccode;
    let dept = req.body.dept;
```

```javascript
    let sec = req.body.section;
    let sem = req.body.semester;
    let batch = req.body.batch;

    const min_working_days = 20;
    const attendance_threshold = 0.5; //50 percent

    conn.connect(function (err) {
    conn.query(`select s_roll, ?-count(*) as cnt as percent from
student_attendance where f_email=? and course_code=? and department=? and
section=? and semester=? and batch=?  and (attd_status='1' or
attd_status='2') group by s_roll;`, [min_working_days, email, date,
course_code, dept, sec, sem, batch, attendance_threshold], function
(error, rows, fields) {
        // if(err) throw err
        console.log('data retrieved');
        res.send(rows);
    })
    });
}
```

SOFTWARE ENGINEERING TOOLS

1. Unit Testing

1.1. Tools used

- Mocha (JavaScript test runner and framework)
- Chai (Test-driven development TDD for NodeJS)
- Istanbul (JavaScript test framework to generate a code coverage report)
- NYC (command-line interface for Istanbul)
- NodeJS and Express for core code modules.

1.2. Setup details

- Installation of required packages

```
sudo npm i -g mocha istanbul nyc --save-dev
npm i mocha chai chai-http express body-parser --save-dev
```

- Configure package.json with appropriate commands

```
"scripts": {
 "start":"node server.js",
 "test":"mocha ./test/test.js --timeout 10000 --exit",
 "coverage":"nyc --reporter=lcov --reporter=text-lcov npm
test"
},
```

```
"nyc": {
  "register": "babel-register",
  "include": [
    "**/*.js",
    "./*.js"
  ],
  "exclude": [
    "**/node_modules/**",
    "**/test/**",
    "**/coverage/**"
  ],
  "all": true
},
```

- Run the test case - `npm test`
- Run test cases with istanbul code coverage - `npm run coverage`
- View the report - `nyc report`

1.3.   **Unit test cases statistics**

The unit test code is implemented for 8 important modules in the backend. 2 to 3 test cases are developed to check the boundary, pass, failure results. The unit test is executed successfully. The failures and errors that were reported are fixed with appropriate logical functionalities.

```
PS D:\SE-project\testing\unitTestingOverflow> npm test

> unittestingoverflow@1.0.0 test D:\SE-project\testing\unitTestingOverflow
> mocha ./test/test.js --timeout 10000 --exit

Server running on port 3000:
 http://localhost:3000


  Testing Faculty Backend Website
{}
    √ should read the todolist.txt file successfully (80ms)
then post
    √ should add a new todo item to the todo list successfully (151ms)
then post my 1
    √ correct marks
then post my 2
    √ incorrect marks
then post my 3
    √ username pass mismatch (2225ms)
then post my 4
    √ username pass failure (544ms)
then post my 5
    √ username pass mismatch (537ms)
    √ download faculty report - success
    √ download faculty report - invalid
    √ download class report - success
    √ download class report - invalid
    √ generate FA report - success
then post my 8
    √ retrieve student leave available


  13 passing (4s)
```

1.4. Unit test case code

```javascript
const app = require('../server'), chai = require('chai'),
chaiHttp = require('chai-http'),
    expect = chai.expect ;
chai.use(chaiHttp);
describe('Testing Faculty Backend Website', function () {
    after(() => {});
    var url = 'http://localhost:3000';
    var requester = chai.request.agent(url);
    it('correct marks', function(done) {
        var input = {
            'P1': 45,
            'P2': 23 }
        requester
            .post('/update_marks')
            .send(input)
```

```javascript
                .then(function(res) {
                    console.log('then post my 1');
                    expect(res.body.message).to.equal('success');
                    done();

                })
        })
    it('incorrect marks', function(done) {
        var input = {
            'P1': -45,
            'P2': 23 }
        requester
            .post('/update_marks')
            .send(input)
            .then(function(res) {
            console.log('then post my 2');
            expect(res.body.message).to.equal('Invalid marks');
                done();
            })
        })
    it('username pass mismatch', function(done) {
        var input = {
            'email': 'abcxyz@gmail.com',
            'pswrd': 'qwe123',
            'rem': '0' }
        requester
            .post('/login')
            .send(input)
            .then(function(res) {
                console.log('then post my 3');
                expect(res.body.message).to.equal('user name
and password mismatch');
                done();
            })});
    it('username pass failure', function(done) {
        var input = {
            'email': 'srishilesh@gmail.com',
            'pswrd': 'qwe123',
            'rem': '0' }
        requester
            .post('/login')
            .send(input)
            .then(function(res) {
```
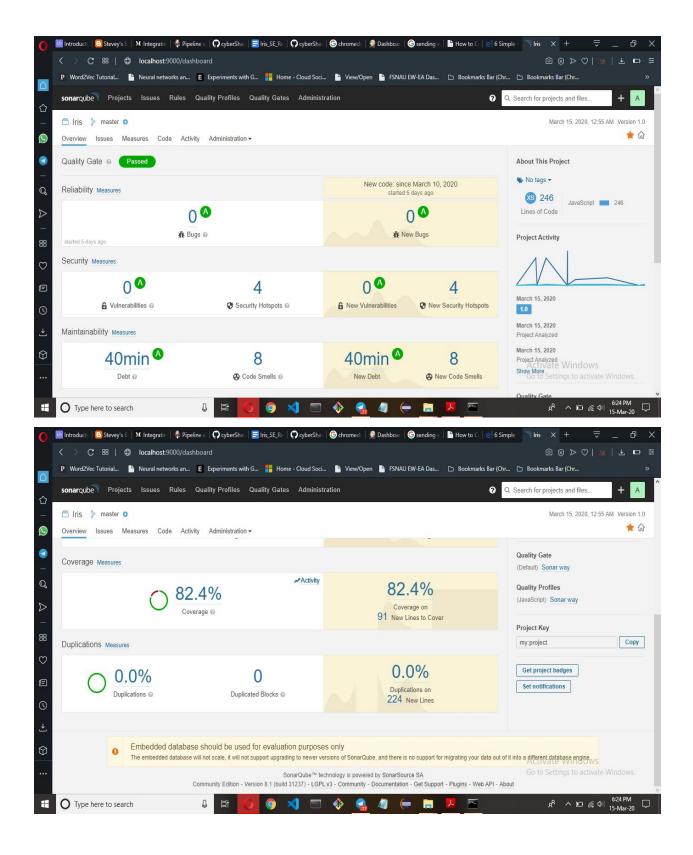
```javascript
                console.log('then post my 4');
                expect(res.body.message).to.equal('failure');
                done();
            })});
    it('username pass mismatch', function(done) {
        var input = {
            'email': 'srishilesh@gmail.com',
            'pswrd': 'pass',
            'rem': '0'
        }
        requester
            .post('/login')
            .send(input)
            .then(function(res) {
                console.log('then post my 5');
                expect(res.body.message).to.equal('success');
                done();
            })});
    it("download faculty report - success", () => {
        var input = {"email":"srishilesh@gmail.com"}
        requester
            .post('/download_faculty_report')
            .send(input)
            .then(function(res) {
                console.log('then post my 6');
                expect(res.body.message).to.equal('success');
                done();
            })});
    it("download faculty report - invalid", () => {
        var input = {"email":"srishilesh@gmail.com"}
        requester
            .post('/download_faculty_report')
            .send(input)
            .then(function(res) {
            console.log('then post my 7');
            expect(res.body.message).to.equal('Invalid entry');
                done();
            })});
    it("download class report - success", () => {
        var input = {
            "email":"srishilesh@gmail.com",
            "date":"01/12/2020",
```

```javascript
                "ccode":"15cse381",
                "dept":"cse",
                "section":"e",
                "semester":"6",
                "batch":"2017"
            }
        requester
            .post('/download_student_report')
            .send(input)
            .then(function(res) {
                console.log('then post my 7');
                expect(res.body.message).to.equal('success');
                done();
            })});
    it("download class report - invalid", () => {
        var input = {
            "email":"hello",
            "date":"01/07/2020",
            "ccode":"15cse311",
            "dept":"cse",
            "section":"e",
            "semester":"6",
            "batch":"2017"
        }
        requester
            .post('/download_student_report')
            .send(input)
            .then(function(res) {
            console.log('then post my 8');
            expect(res.body.message).to.equal('Invalid entry');
                done();
            })});
    it("generate FA report - success", () => {
        var input = {
            'email': 'srrishilesh@gmail.com',
            'ccode': '15cse381',
            'dept': 'cse',
            'section': 'e',
            'semester': '6',
            'batch': '2017',
        }
        requester
```

```
                .post('/generate_fa_list')
                .send(input)
                .then(function(res) {
    console.log('then post my 9');
    expect(res.body.message).to.equal('FA report generated');
    done();
                })});
    it('retrieve student leave available', () => {
        var input = {
            'email': 'srrishilesh@gmail.com',
            'ccode': '15cse381',
            'dept': 'cse',
            'section': 'e',
            'semester': '6',
            'batch': '2017',
        }
        requester
            .post('/calculate_free_student_leave')
            .send(input)
            .then(function(res) {
                console.log('then post my 10');
                expect(res.body.message).to.equal('success');
                done();
            }) });
});
```

2.   Static Code Analysis

Actions were taken to fix issued reported by SonarQube

1) Fixed severe security bugs.
2) Improved unit test coverage to cover at least 80% line in the server code.
3) Removed code duplicates.

3. UI Testing

3.1. **Tools used**   - Selenium Webdriver, Chrome/gecko driver

3.2. **Setup Details -**

- Set up a new npm project.
- Next, install a framework so as to allow us to work with Selenium from the inside node.

```
D:\GitHub>cd SeleniumJS

D:\GitHub\SeleniumJS>npm install selenium-webdriver
npm WARN seleniumexample@1.0.0 No repository field.

+ selenium-webdriver@4.0.0-alpha.7
updated 1 package and audited 173 packages in 5.17s

1 package is looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

- Installed Selenium Webdriver and the latest GeckoDriver and ChromeDriver drivers inside the project folder.
- And finally added the driver's location to the System PATH variable.
- Then, run the test case - `node test_name.js`

```
D:\GitHub\SeleniumJS\scroll_test>node scroll_test.js

DevTools listening on ws://127.0.0.1:50405/devtools/browser/92ca655b-77bb-48eb-8ae4-318b398006e4
Test passed!
Quitting driver...

D:\GitHub\SeleniumJS\scroll_test>cd ../login_tests

D:\GitHub\SeleniumJS\login_tests>node login_test_all.js

DevTools listening on ws://127.0.0.1:51010/devtools/browser/393f1937-29b5-461e-811d-2242564c2d5c

DevTools listening on ws://127.0.0.1:51046/devtools/browser/f0ef44c2-2525-45f3-a6b6-8288cdb1cf68

DevTools listening on ws://127.0.0.1:51071/devtools/browser/9a4f1b03-70fc-4636-b096-cc710d01f9ec
Test passed
Test failed
Test failed

D:\GitHub\SeleniumJS\login_tests>
```

3.3. **Test Cases Statistics for register and login page**

Test ID	Description	Expected Results	Actual Results	Automated?

1.1	**Verify if a user can log in with a valid username and password**	Successful login	pass	yes
1.2	**Verify if a user can log in with an invalid username and password**	Error displayed during login attempt	pass	yes
1.3	**Verify if a user can login with no password entered.**	Prompt for password. "Please enter Password".	fail	yes
1.4	**Verify if the data in password field is either visible as asterisk or bullet signs**	To display the password in asterisk when typed in that particular column	pass	**no**
1.5	**Verify if the 'Enter' key of the keyboard is working correctly on the login page**	Successful login	pass	**yes**
1.6	**Invalid email format in username field login page.**	Prompt for correct format of email.	fail	**no**
2.1	After opening the home page, once register button is clicked in the side menu, should be directed to register	Register page to be displayed.	pass	**no**

	page rather than any other page.			
2.2	Not entering enough characters in the password box as per the validation rules, register shouldn't be successful.	Register not successful	pass	**yes**
2.3	**Not entering the correct format in the Email box where the registration shouldn't be successful.**	Register not successful	pass	**yes**
2.4	**Entering all the details as per the mentioned validation rules.**	Registration successful	pass	**yes**
2.5	**Not mentioning the department**	Register not successful	pass	**no**
2.6	Not mentioning the gender	Register not successful	pass	**no**
2.7	**After redirecting to the Register page, once Login page is clicked, should be redirected to Login Page.**	Login Page must be displayed	pass	**no**

3.1	Forgot password: Actions regarding the 'Forgot password' option	To navigate into the forgot password page to reset the password for a particular username by sending him a mail	pass	**no**
3.2	Invalid email format	Error displaying invalid email format	pass	**no**
3.3	**Submit email that is not registered and check the result (it may be an error, or a general success message - this depends if how security is implemented)**	Error displaying that mail id entered is not registered	fail	**no**

3.4. **Test Case Code**

3.4.1 For Register Page

```
const webdriver = require('selenium-webdriver'),
    By = webdriver.By,
    until = webdriver.until;
async function reg_test1(){
let driver =await new webdriver.Builder()
    .forBrowser('firefox')
    .build();

await driver.get('https://iris-se.netlify.com/register')

    await
driver.findElement(By.id("input-51")).sendKeys("Prathyusha")
    await driver.findElement(By.id("input-54")).sendKeys("Ilam")
```

```javascript
    const button =
driver.findElement(By.xpath("/html/body/div/div/div/div[2]/main
/div/div/div/div/div[2]/form/button[1]"))
  button.click()

await driver.getPageSource().then(function(content)
{
  if(content.indexOf('Welcome ') !==-1 ) {
   console.log('Test passed');
   /*console.log('Target page');
   d.get('https://iris-se.netlify.com') */
  } else {
      console.log('Test failed');
      return false;
  }
});
}

async function reg_test2(){
    let driver =await new webdriver.Builder()
        .forBrowser('firefox')
        .build();

    await driver.get('https://iris-se.netlify.com/register')

      await
driver.findElement(By.id("input-51")).sendKeys("Prathyusha")
      await
driver.findElement(By.id("input-54")).sendKeys("Ilam")
    const button =
driver.findElement(By.xpath("/html/body/div/div/div/div[2]/main
/div/div/div/div/div[2]/form/button[1]"))
    button.click()

    await driver.getPageSource().then(function(content)
    {
      if(content.indexOf('Must be at least 8 characters') !==-1
) {
       console.log('Test passed');
      } else {
```

```
                console.log('Test failed');
                return false;
        }
    });
    }


    async function reg_test3(){
        let driver =await new webdriver.Builder()
            .forBrowser('firefox')
            .build();
        await
driver.get('https://iris-se.netlify.com/register')
            await
driver.findElement(By.id("input-58")).sendKeys("prathyu@gmail.c
om")
            await
driver.findElement(By.id("input-61")).sendKeys("prathyu0398")
        await(await
driver.findElement(By.id('registerButton'))).click()



        await driver.getPageSource().then(function(content)
        {
          if(content.indexOf('Welcome ') !==-1 ) {
           console.log('Test passed');
          } else {
                console.log('Test failed');
                return false;
        }
    });
        }

reg_test1();
reg_test2();
reg_test3();
```

3.4.2 For page Scroll

```
const webdriver = require('selenium-webdriver'),
    By = webdriver.By,
    until = webdriver.until;
async function scroll(){
```

```
    let driver =await new webdriver.Builder()
    .forBrowser('firefox')
    .build();


    await driver.get('https://iris-se.netlify.com/register')
    await driver.executeScript("window.scrollBy(0, 1000)")


    if (driver.scroll===false){
        console.log("Test failêd!");
    }
    else{
        console.log("Test passed!");
    }
}
scroll();
```

3.4.3 For login Page

```
const webdriver = require('selenium-webdriver'),
    By = webdriver.By,
    until = webdriver.until;
async function reg(){
    let driver=await new webdriver.Builder()
    .forBrowser('firefox')
    .build()
    await driver.get('https://iris-beta.netlify.com/')
    await(await
driver.findElement(By.xpath('/html/body/div/div/div/div/div/div
/div/div/p/a'))).click()


    await

driver.findElement(By.name('email')).sendKeys("prathyu0398@gmai
l.com")
    await

driver.findElement(By.name('password')).sendKeys("prathyu0398")
    await(await
driver.findElement(By.xpath('/html/body/div/div/div/div/div/div
/div/form/div[3]/button'))).click()
}
reg();
```

4. Continuous Integration

4.1. Tools used

- Jenkins, Git

4.2. Setup details and workflow details

- Install and setup Jenkins with JDK and .war file
- Install the default and other necessary plugins needed
- Create a new pipeline with SCM as GitHub repository
- Write pipeline scripts for cloning the git repository and installing all the dependencies specified

```
pipeline {
  agent any
  stages {
    stage('Cloning Git') {
      steps {
        git 'https://github.com/srishilesh/iris.git'
      }
    }

    stage('Install dependencies') {
      steps {
        bat 'npm install'
      }
    }
    stage('Unit testing'){
        steps{
            dir('./unitTesting/unitTestingOverflow') {
                bat 'npm install'
                bat 'npm run coverage'
            }
        }
    }
  }
}
```

- Write a script for running the Unit test cases and generate the Coverage report
- Create a freestyle project for running SonarQube, and connect with original pipeline
- Specify the sonarqube properties with the hosting URL in Jenkins

```
sonar.login = admin
sonar.password = admin
sonar.projectKey=my:project
```

```
sonar.projectName = Iris
sonar.projectVersion=1.0


sonar.exclusions = assets/**


sonar.javascript.lcov.reportPaths =
unitTesting/unitTestingOverflow/coverage/lcov.info
```

- On successful build and coverage report generation, the post-build action is to perform User Acceptance test using Selenium web drivers and chrome drivers
- On build completion, create a webhook trigger from hosted Jenkins server to the Github webhook to commit the necessary changes and push them into the repository
- On pushing the code, Continuous Deployment takes place

4.3. **Analysis**
- The above Continuous Integration proceeds to the next build only on successful builds
- The test cases are validated by Unit testing stage and create a Coverage report
- With a minimum threshold of 75%, it passes to the next build in SonarQube for Static code analysis

5. Any additional software engineering practices followed

5.1. **Scalable Websites**
- Usage of AWS EC2 instances, we have hosted our backend codes, which allows scalability and takes cares of performance issues

5.2. **Normalized and Scalable Database**
- Usage of AWS RDS instances, we have hosted using MySQL database with normalized schemas

5.3. **Modularized code**
- All the backend and frontend codes are modularized for right reuse of codes avoiding duplications

5.4. **API**
- We have created separated APIs for each functionality, which ensures good code quality with effective usage of function calls

5.5. **Coverage Static code analysis**
- Developing the codes only on static code analysis with overall coverage results above 75%, to ensure less bugs, code smells and less vulnerabilities

5.6. **Daily Online Stand up meetings**
- Conducted Developer standup meetings through Zoom calls

5.7. Jenkins successful build
- ■ Email notifications after successfully build

CONCLUSION

This project aimed to improve the efficiency of faculty's tasks through an elegant and simplistic web portal. Integrating the results of separate functionality into a new product and providing a visual dashboard user interface improves the efficiency of data retrieval, analysis and decision making for faculties. The dashboard view displays updated dynamic information, eliminating the need to manually complete the tasks. The dashboard also allows faculties to track their attendance and check the number of different leaves available.

Future work could involve building a single dashboard that supports multiple functionalities like integrated attendance management with automated OD updates. Another potential area of future work is to include a feature for faculties to upload study resources and subject notes. Future studies could research dashboards focusing on different metrics, such as student retention, graduation rates, and the average time to graduation by leveraging data analytics and machine learning.

Registration form

Rgister New User <<uses>> Update User Table

Website

User 1

<<uses>>

Confirmed Data store <<extends>>

Return Message

Incorrect Details

OR

Database

<<extends>>

Login to Application <<uses>> Authenticate User details

Website

User 2

Return message

Incorrect details

OR <<extends>>

Return message

Access granted <<uses>>

Application

Data Flow Model: DFD Diagram

```
                              ┌─────────────────┐
                              │  Registration   │
                              └─────────────────┘
          ┌───────────────────────┼───────────────────┬──────────────────┐
          ▼                       ▼                   ▼                  ▼
  ┌──────────────┐        ┌──────────────┐    ┌──────────────┐   ┌──────────────┐
  │   Collect    │        │  Validate    │    │   Update     │   │    Show      │
  │   Faculty    │        │    the       │    │    the       │   │ Registration │
  │   details    │        │   details    │    │  database    │   │   status     │
  └──────────────┘        └──────────────┘    └──────────────┘   └──────────────┘
          │ Using Form             │ POST in backend   │ Verification status │ Insertion status
          ▼                        ▼                   ▼                  ▼
  ┌──────────────┐        ┌──────────────┐    ┌──────────────┐   ┌──────────────┐
  │    Data      │        │    Back      │    │ Verify OTP   │   │    Send      │
  │  collection  │        │    end       │    │ confirmation │   │ registration │
  │   using      │        │  validation  │    │              │   │ confirmation │
  │    form      │        └──────────────┘    └──────────────┘   │    status    │
  └──────────────┘                │ Verify details    │ Insert   └──────────────┘
          │ Input validation       ▼                   ▼
          ▼                ┌──────────────┐    ┌──────────────┐
  ┌──────────────┐        │    Send      │    │ Add details  │
  │    Front     │        │    Email     │    │ to original  │
  │     End      │        │ verification │    │  database    │
  │  Validation  │        └──────────────┘    └──────────────┘
  └──────────────┘                │ Add email
          │ POST to backend        ▼
          ▼                ┌──────────────┐
  ┌──────────────┐        │  Temporary   │
  │   Submit     │        │    DB for    │
  │   details    │        │    Email     │
  └──────────────┘        │ verification │
                          └──────────────┘       ┌──────────────┐
                   ┌──────────────┐               │   Confirm    │
                   │  Retrieve    │──────────────▶│     OTP      │
                   │    email     │               │  validation  │
                   │   address    │               └──────────────┘
                   │    from      │
                   │ Temporary    │──┐
                   │     DB       │  │  ┌──────────────┐   ┌──────────────┐
                   └──────────────┘  └─▶│  Generate    │──▶│    Send      │
                                        │     OTP      │   │    Email     │
                                        └──────────────┘   └──────────────┘
```

Forgot Password

- Select Forgot Password
- «include» Input username
- «include» Send OTP to e-mail id
- «extend» Username validation
- «include» Update OTP to DB
- Enter e-mail id and OTP to reset password
- «include» Check OTP from DB
- «include» Verify OTP and Reset password
- OTP correct / OTP incorrect
- «include» Update Password to DB

Faculty

iris-backend

Faculty

email_id, new_password,
email_id

OTP

0

Forgot
Password

email_id, acc_date, acc_time, OTP

OTP

email_verification

success/failure

email_id,
new_password

emp_login

SE DataFlowModel2-DFDDiagram2

email_id, OTP

correct/incorrect

| 4 |
| Update new password |

success/failure → emp_login

email_id, new_password →

email_verification

new_password

| 3 |
| Verify OTP |

email_id

true/false

INSERT(email_id, acc_date, acc_time, OTP)

success/failure

OTP, new_password

Faculty

email_id →

| 1 |
| Validate Email ID |

email_id, num_of_digits →

| 2 |
| Generate and send OTP |

OTP

2.2
Return Time

acc_time

2.1
Generate OTP using mathematical algorithm

email_id, number_of_digits

email_id, OTP, acc_date, acc_time

2.4
Update OTP to DB

email_id, OTP

2.5
Email the OTP to user

INSERT(email_id, acc_date, acc_time, OTP)

success/failure

success/failure

forgot_pass.py

acc_date

2.3
Return Date

email_verification

2.6
Spawn the child process (Python program)

success/failure

to_addr, OTP

2.7
Send Email

OTP

Forgot Password

Select Forgot Password

«include»

Input username

«include»

«extend»

Username validation

Send OTP to e-mail id

«include»

Update OTP to DB

Faculty

iris-backend

Enter e-mail id and OTP to reset password

«include»

«include»

Check OTP from DB

Verify OTP and Reset password

«include»

OTP correct / OTP incorrect

Update Password to DB

Faculty

email_id, new_password,
email_id

OTP

0

Forgot
Password

email_id, acc_date, acc_time, OTP

OTP

email_verification

success/failure

email_id,
new_password

emp_login

email_id, OTP

correct/incorrect

4

Update new password

success/failure

emp_login

email_id, new_password

email_verification

new_password

3

Verify OTP

email_id

true/false

INSERT(email_id, acc_date, acc_time, OTP)

success/failure

OTP, new_password

1

Validate Email ID

email_id, num_of_digits

2

Generate and send OTP

Faculty

email_id

OTP

2.2
Return Time

acc_time

2.1
Generate OTP using mathematical algorithm

email_id, number_of_digits

email_id, OTP, acc_date, acc_time

acc_date

2.3
Return Date

2.4
Update OTP to DB

INSERT(email_id, acc_date, acc_time, OTP)

success/failure

email_verification

2.5
Email the OTP to user

email_id, OTP

success/failure

forgot_pass.py

2.6
Spawn the child process (Python program)

success/failure

to_addr, OTP

2.7
Send Email

OTP