



Annexure -06

STUDENTS EXTERNAL ATTACHMENT INTERNSHIP LOG-BOOK

DURATION: 16 WEEKS

STUDENT'S PARTICULARS

Name of Student:

[illegible]

Enrolment No.:

2	1	0	3	0	5	1	0	5	3	0	2
---	---	---	---	---	---	---	---	---	---	---	---

Contact No.:

9	4	2	7	3	0	1	0	0	3
---	---	---	---	---	---	---	---	---	---

Email ID

Pdipen135@gmail.com

Name of Company:

[illegible]

Address of Company.:

[illegible]

Name of Industry Mentor

[illegible]

Contact No.:

7	4	0	5	4	7	4	8	8	7
---	---	---	---	---	---	---	---	---	---

Email ID

Fbvohra95@gmail.com

Name of Faculty Mentor

[illegible]

Contact No.:

9	0	0	9	5	2	6	2	7	0
---	---	---	---	---	---	---	---	---	---

Email ID

vinod.patidar28579@paruluniversity.ac.in

Training Duration: From:23/12/2024 To: 23/4/2025

WEEKLY PROGRESS CHART (FIRST WEEK) (WEEK ENDING: 27-12-24)

Day	Task Done
Monday	Met the employees and got to know the team leads and their respective tasks. There wasn't much work for me on the first day. I familiarized myself with the projects they were working on and discussed my strengths with the team. They assessed which team or project I might be suitable for. I was assigned a task to implement API CRUD operations for a CRM.
Tuesday	Worked on the CRM CRUD task. Added API calls to /crud/ for creating a user, deleting a user, updating user data, and reading a list of users with all their details.
Wednesday	Holiday
Thursday	Hosted the frontend and backend separately on Azure using Azure Function App and Azure Static Web App. Configured the API calls with CORS settings. Faced issues with the backend not allowing PATCH requests, which caused problems with update calls.
Friday	Integrated the user list data into a react-data-table-component. Added functionality for searching and sorting the data by name, role, age, and email. Updated the UI for the user data table using the selected color palette.

WEEKLY PROGRESS CHART (SECOND WEEK) (WEEK ENDING: 03-01-25)

Day	Task Done
Monday	Hosted the application with the new UI and populated it with 4,400 dummy user records generated through API calls using a JavaScript script executed with Node.js. Tested the sorting and searching functionalities.
Tuesday	Presented the project to the team lead. Received feedback to add backend validation for the data and a frontend UI to display errors and success messages. Started working on these tasks.
Wednesday	Holiday
Thursday	Implemented backend validation for the following: - Age must be a number. - Name must be a string. - Role must be a string. - Email must match a specific format using regular expressions.
Friday	Enhanced the frontend UI using <code>react-toastify</code> to display success and error messages. Added error handling for scenarios such as duplicate email, invalid age (non-numeric), invalid name (non-string), and incorrectly formatted email.

WEEKLY PROGRESS CHART (THIRD WEEK) (WEEK ENDING: 10-01-25)

Day	Task Done
Monday	Deployed the application again using GitHub YML files to Azure. Tested the react-toastify notifications and backend validation. Demonstrated the project to the team lead and asked for feedback.
Tuesday	Created a backend for user login functionality and tested it by hosting on Azure. Presented it to the team lead for feedback.
Wednesday	Tested login functionality without a password and with credentials for users. Created separate roles for admin and user access.
Thursday	Created a data flow diagram for authentication and showed it to team lead.
Friday	Added variables isFormFilled, isFormValidated, and Token to validate user accounts. Implemented functionality to ensure only admins can validate and allow users to update their accounts. Tested on Azure and demonstrated to the team lead.

WEEKLY PROGRESS CHART (FOURTH WEEK) (WEEK ENDING: 17-01-25)

Day	Task Done
Monday	Developed a routing system where routes are accessible based on active variables (isFormFilled, isFormValidated, etc.). Tested by hosting on Azure and presenting to the team lead.
Tuesday	Holiday
Wednesday	Holiday
Thursday	Added password-protected login functionality using NodeMailer for OTP verification. Separated routes and improved security measures. Hosted and tested on Azure. Showed the results to the team lead.
Friday	Integrated NodeMailer functionality to send OTP for secure login. Tested various use cases, including expired OTPs and incorrect emails, by hosting on Azure and demonstrating to the team lead.

WEEKLY PROGRESS CHART (FIFTH WEEK) (WEEK ENDING: 27-01-25)

Day	Task Done
Monday	Enhanced login security by separating admin and user routes and ensuring restricted access to sensitive functionalities. Tested by hosting on Azure.
Tuesday	Optimized and refactored the backend code for the login and validation system. Added tests for edge cases. Hosted on Azure and demonstrated improvements to the team lead.
Wednesday	Debugged and resolved minor issues in the OTP-based login functionality. Improved error handling for failed API calls and incorrect user inputs.
Thursday	Added frontend UI feedback for invalid OTPs and incorrect email formats using react-toastify. Tested integration thoroughly on Azure.
Friday	Improved admin routes to allow bulk validation of user accounts. Tested by simulating high-traffic scenarios on Azure.

WEEKLY PROGRESS CHART (SIXTH WEEK) (WEEK ENDING: 31-01-25)

Day	Task Done
Monday	Fixed issues with the user update functionality. Verified changes and tested compatibility with backend validation logic.
Tuesday	Implemented additional backend security checks for OTP validity duration and enhanced error logging.
Wednesday	Improved UI responsiveness for mobile devices, especially for login and validation pages. Tested cross-browser compatibility.
Thursday	Streamlined the deployment process by automating Azure deployments through GitHub Actions. Tested the pipeline end-to-end.
Friday	Fixed edge cases in the routing logic to handle scenarios where variables like <code>isFormFilled</code> and <code>isFormValidated</code> are undefined or null.

WEEKLY PROGRESS CHART (SEVENTH WEEK) (WEEK ENDING: 07-02-25)

Day	Task Done
Monday	Conducted a final round of testing and documentation for the login, validation, and routing functionalities. Prepared for a project review.
Tuesday	Created three types of users: Admin, Customer, and Employee. Implemented role-based access control to ensure users could only access specific functionalities based on their roles.
Wednesday	Developed a task creation system where a Customer can create a task request. Integrated the feature with the backend API and ensured data validation.
Thursday	Implemented task verification functionality. An Admin can review and verify a task created by a Customer before assigning it further.
Friday	Added the ability for an Admin to assign a Customer to an Employee after verification. Implemented role-based API calls to restrict actions based on user type.

WEEKLY PROGRESS CHART (EIGHT WEEK) (WEEK ENDING: 14-02-25)

Day	Task Done
Monday	Restricted Customers and Employees to view only their own profiles. Introduced an access control mechanism ensuring they cannot access or modify others' profiles.
Tuesday	Developed a chat functionality for assigned tasks. Customers and Employees can communicate only with their assigned individual to discuss tasks.
Wednesday	Implemented real-time chat functionality for assigned tasks. Messages are stored in the backend, and users receive real-time updates.
Thursday	Created an Admin Panel UI for task assignment and chat monitoring. Developed a backend system to enable admins to assign and track tasks efficiently.
Friday	Tested the full system, including user roles, task creation, verification, assignment, and chat. Fixed bugs related to user access restrictions and task assignment.

WEEKLY PROGRESS CHART (NINE WEEK) (WEEK ENDING: 16-02-25)

Day	Task Done
Monday	Created Chat functionality using Socket.io in sync with both frontend and backend.
Tuesday	Completed Integration of Google login and Facebook login using google Oauth2.0 and Facebook developer.
Wednesday	Worked on Payment options with Stripe.
Thursday	Configured stripe in frontend and backend as well as tested some test credit card purchase successfully.
Friday	Tested the full system, including payment, login, verification, assignment, and chat. Fixed bugs related to user access restrictions and socket disconnect.

WEEKLY PROGRESS CHART (TENTH WEEK) (WEEK ENDING: 23-02-25)

Day	Task Done
Monday	Improved the chat functionality using Socket.io in sync with both frontend and backend.
Tuesday	Made sure that all logic functionalities work with in sync of globalContext and logout function.
Wednesday	Tested the Payment functionalities with the stripe test acocunts.
Thursday	Made sure every payment try either completed or pending is noted in backend in sync with client.
Friday	Tested the full system, including payment, login, verification, assignment, and chat. Fixed bugs Payment through stripe.

WEEKLY PROGRESS CHART (ELEVENTH WEEK)

(WEEK ENDING: 02-03-25)

Day	Task Done
Monday	Started Working on UI after completing the testing the required functionalities from client.
Tuesday	Improved the UI with motion and styling using Material ui sx and tradition css styling.
Wednesday	Worked some more on the UI with motion and styling using Material ui sx and tradition css styling.
Thursday	Worked some more on the UI with motion and styling using Material ui sx and tradition css styling.
Friday	Completed the UI work with motion and styling using Material ui sx and tradition css styling. Got the design approved from the team lead

WEEKLY PROGRESS CHART (TWELVE WEEK) (WEEK ENDING: 09-03-25)

Day	Task Done
Monday	Created Video Call functionality using peerjs in sync with both frontend and backend.
Tuesday	Made sure to keep the video call a global component passed through globalcontext so call are received as long as user is online.
Wednesday	Created UI for floating video call component which floats around which you do other works on CRM in call.
Thursday	Work on the Ui of video call and movable window of call and being able to change the sized of the floating video call panel.
Friday	Holiday

WEEKLY PROGRESS CHART (THIRTEENTH WEEK) (WEEK ENDING: 16-03-25)

Day	Task Done
Monday	Work more on the Ui of video call and movable window of call and being able to change the sized of the floating video call panel and got it approved.
Tuesday	Integrated screen sharing functionality in the video call component using PeerJS. Ensured audio and screen sharing work seamlessly for all assigned CRM users via Socket.io.
Wednesday	Tested and debugged video call features, ensuring smooth switching between camera, screen share, and mute/unmute options.
Thursday	Finalized UI improvements for the video call component with better transitions and styling.
Friday	Conducted final testing and received team lead approval for video call and screen share features.

WEEKLY PROGRESS CHART (FOURTEENTH WEEK) (WEEK ENDING: 23-03-25)

Day	Task Done
Monday	Set up the hosting process, deployed the CRM frontend and backend using Azure and GitHub Actions.
Tuesday	Wrote automated deployment scripts to update dummy user data on every deployment.
Wednesday	Conducted end-to-end testing of hosted CRM, checking video, screen share, and audio calls.
Thursday	Optimized API calls and socket events to ensure real-time updates work smoothly without performance issues.
Friday	Final review with the team, bug fixes, and documentation for video call and screen share functionality.

WEEKLY PROGRESS CHART (FIFTEENTH WEEK)

(WEEK ENDING: 30-03-25)

Day	Task Done
Monday	Started working on RFM (Recency, Frequency, Monetary) model implementation for user payment tracking.
Tuesday	Developed backend logic to fetch and calculate RFM scores based on user transaction history.
Wednesday	Created frontend visualization for RFM scores using charts and analytics UI.
Thursday	Integrated real-time updates for RFM scores when payments are made or updated.
Friday	Completed and tested the RFM model, ensuring it syncs properly with the user's payment data.

WEEKLY PROGRESS CHART (SIXTEENTH WEEK)

(WEEK ENDING: 06-04-25)

Day	Task Done
Monday	Implemented Multer and GridFS to store and manage user profile pictures in MongoDB.
Tuesday	Created API endpoints for uploading, updating, and retrieving user profile pictures.
Wednesday	Built UI for profile picture management with real-time updates using Socket.io.
Thursday	Integrated profile picture updates across all CRM-related features where user images appear. Making sure they are compressed accordingly.
Friday	Completed final testing and approved the user profile management system with image handling.

Student's Signature: _____

Date: _____

Comments by Industry Mentor:

Name: _____

Signature of Industry Mentor _____

Date: _____