Sri Lanka Institute of Information Technology

R24-002

FERNANDO R.D.S.A.

IT21011948

Information System Engineering

Table of Contents

Table of Figures

Project Component and Current Status

Component: Risk-Adjusted Time Forecast

Progress

- The system mainly focuses on giving time prediction for a project and measure the severity of risks identified in a certain project.
- A Time estimation model was trained after working on more than 6 Machine learning models to predict the time taken for a project which spent a very high effort to develop.

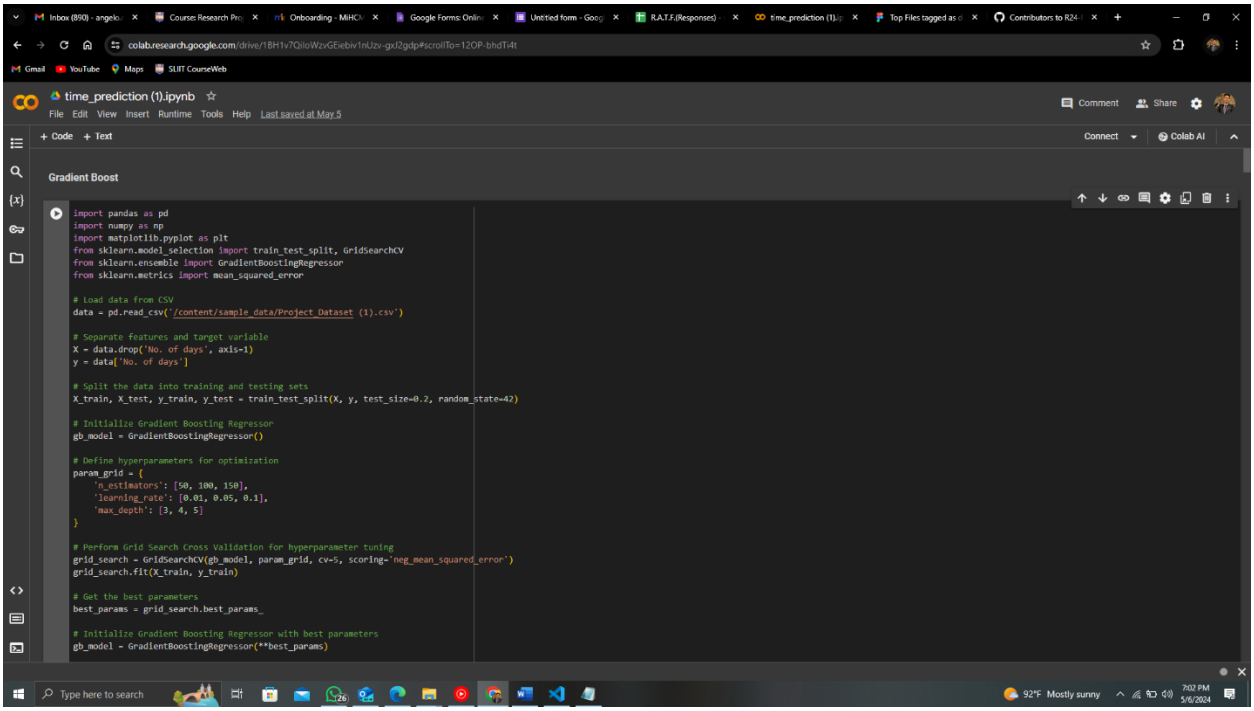


Figure 1 Machine Learning modeling.

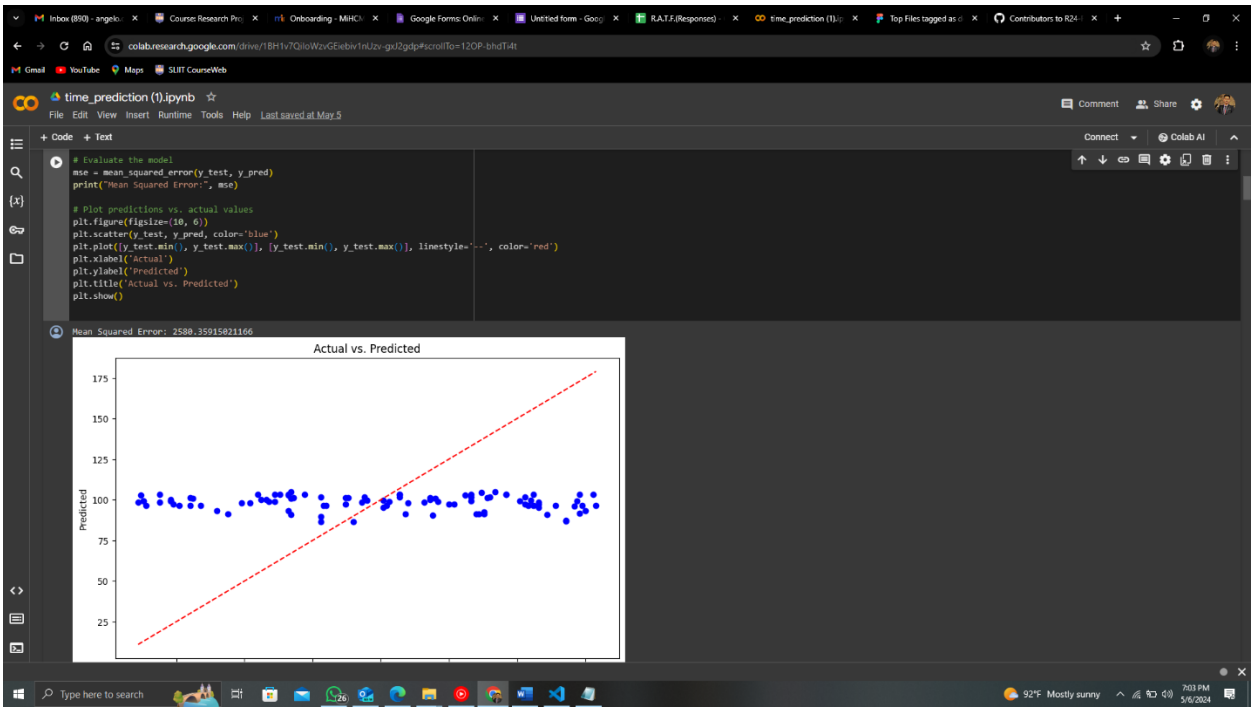


Figure 2- Actual vs predicted graph

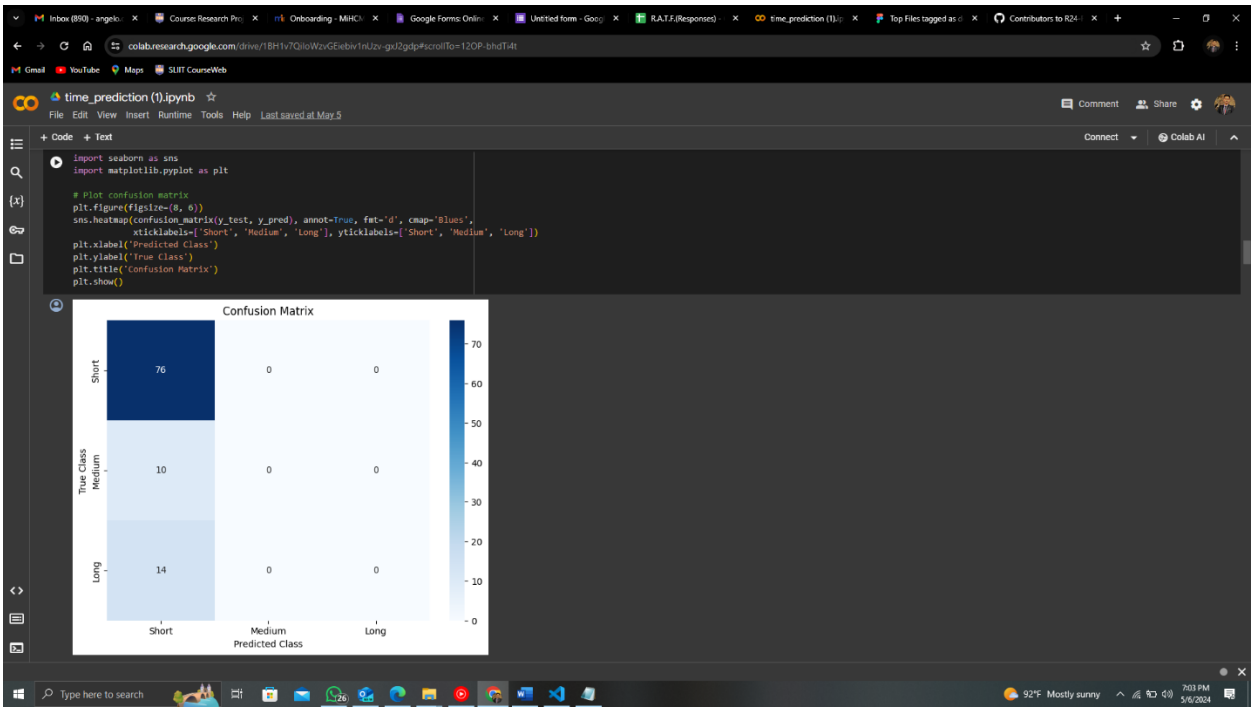


Figure 3-Confusion Matrix

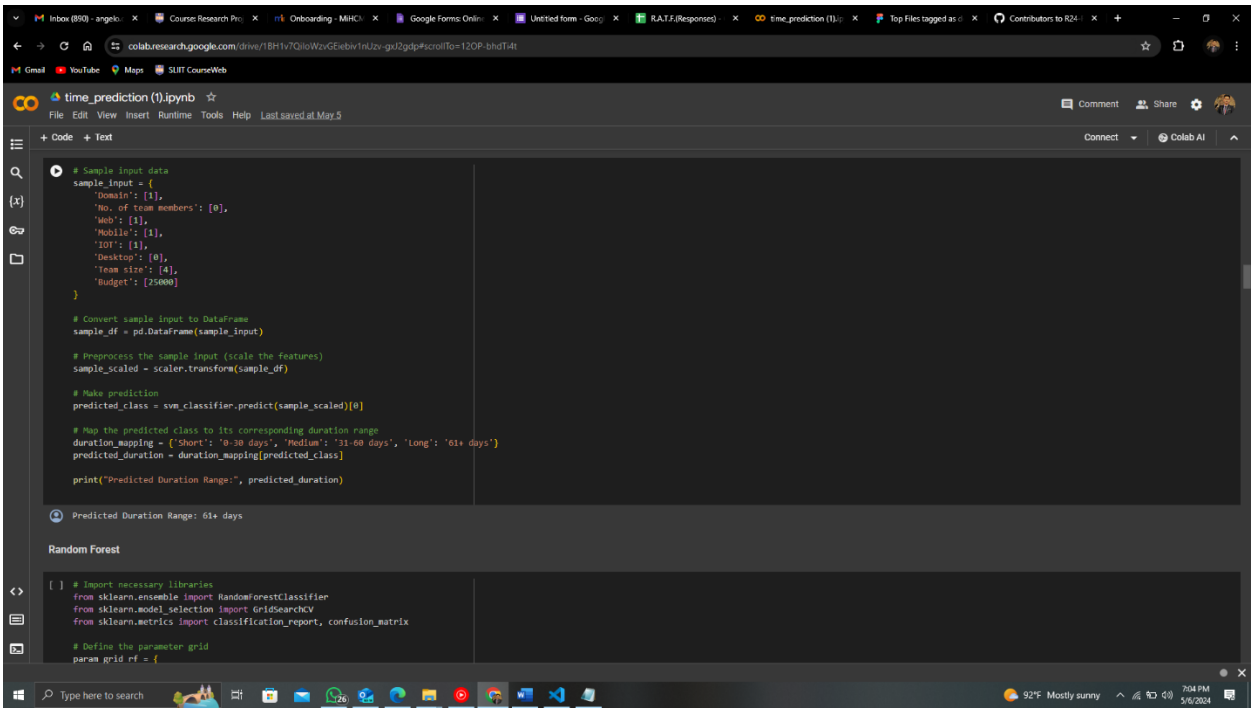


Figure 4- ML Model

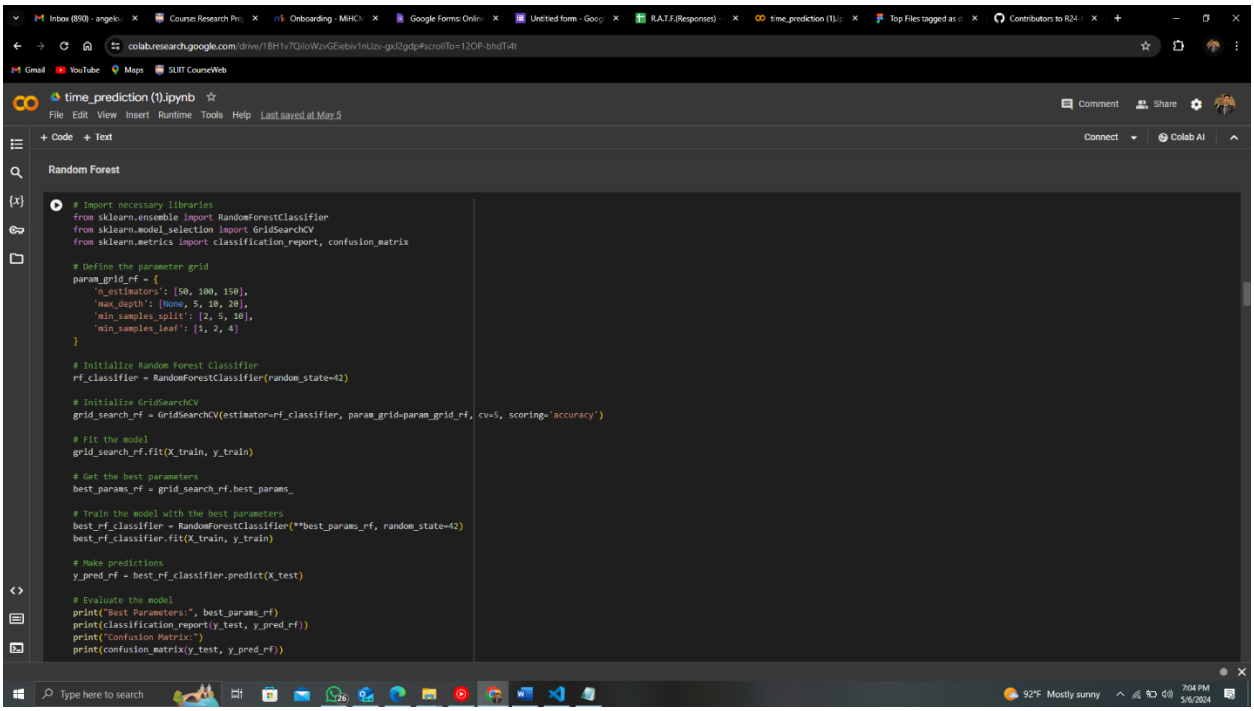


Figure 5-Random Forest model

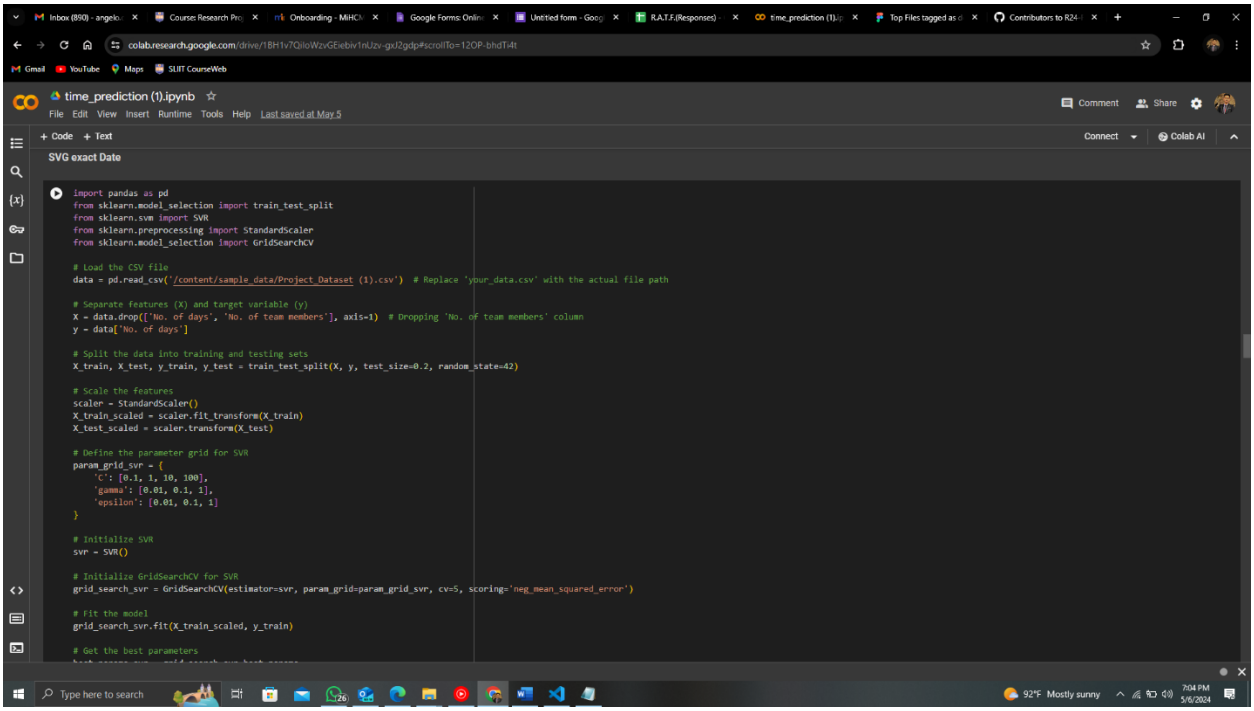


Figure 6-SVG exact date model

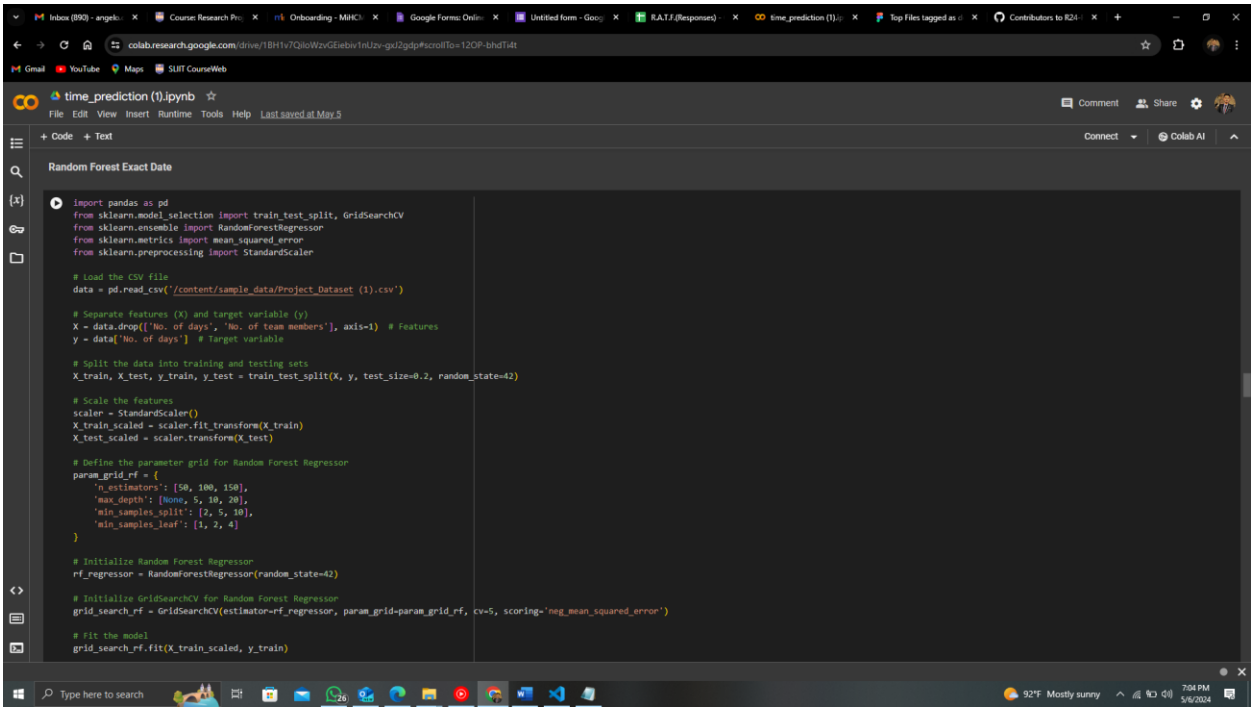


Figure 7-Random Forest exact date

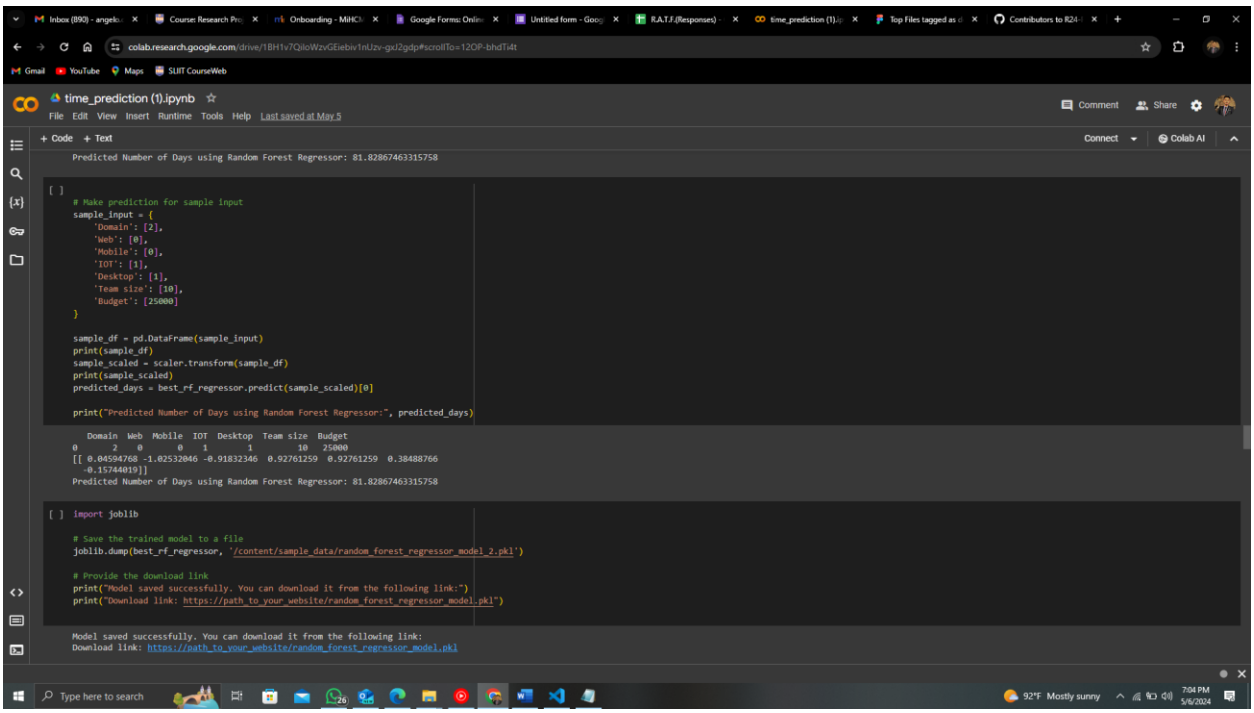


Figure 8- RF implementation

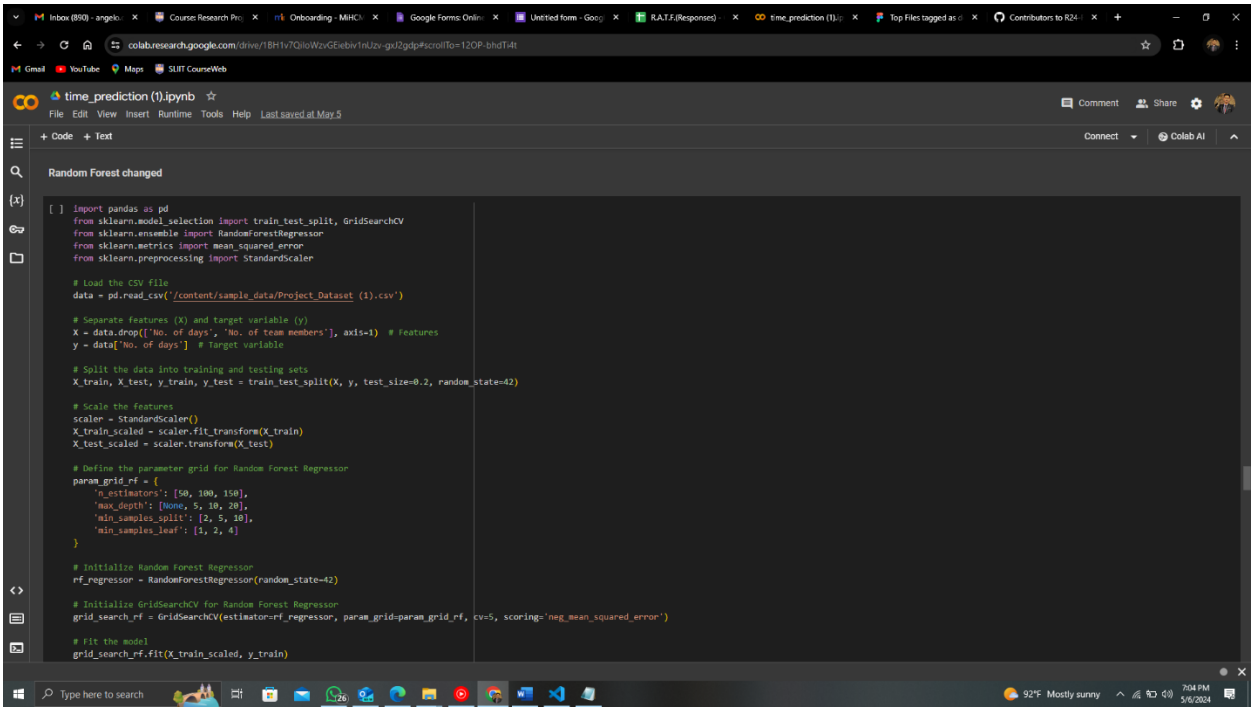
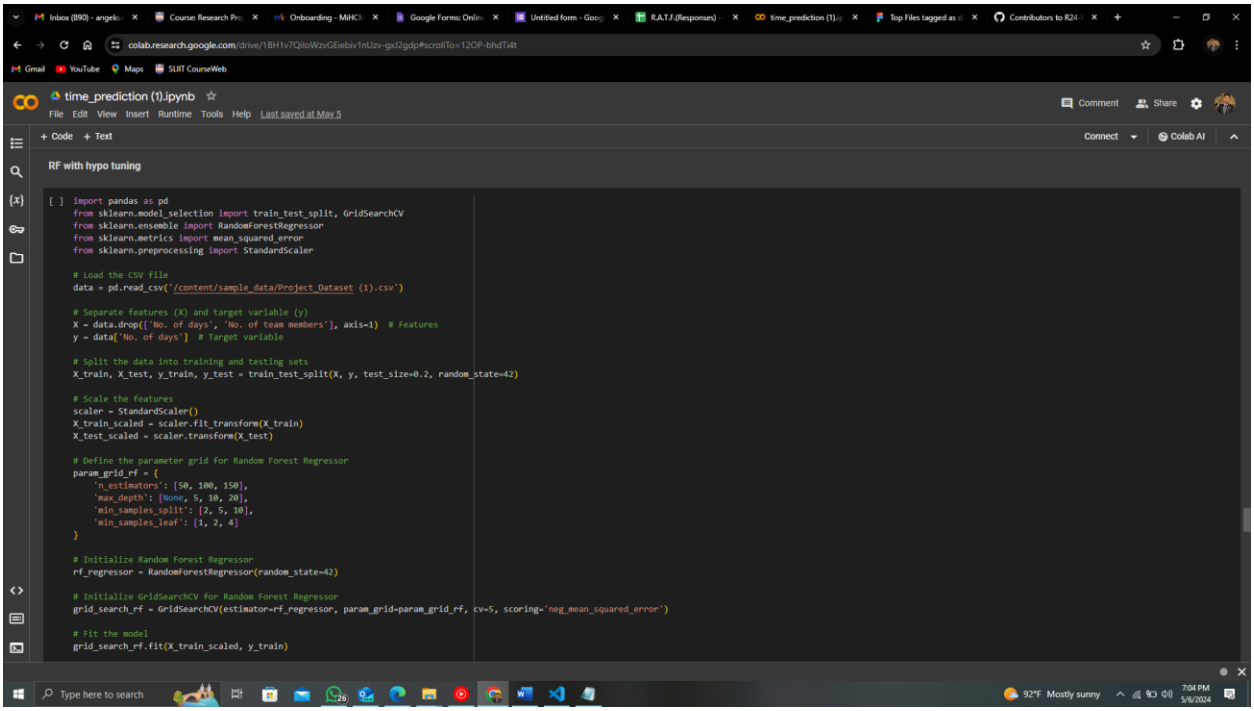


Figure 9-Random Forest changed



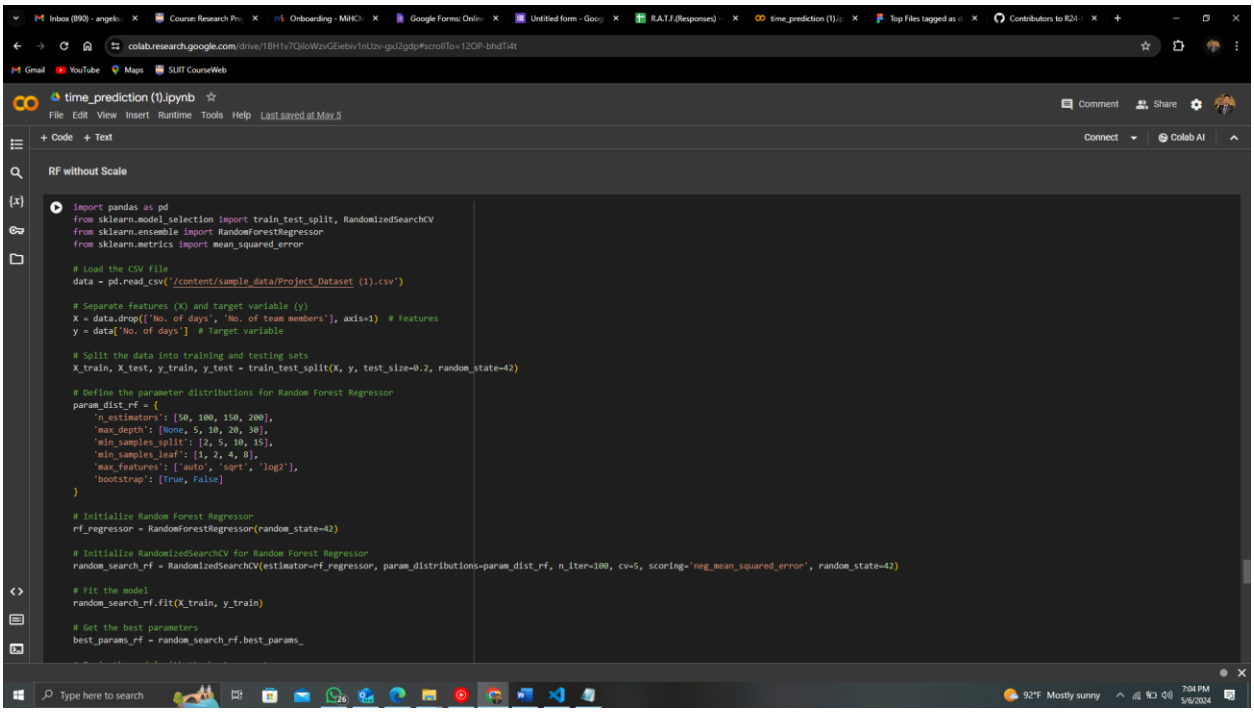
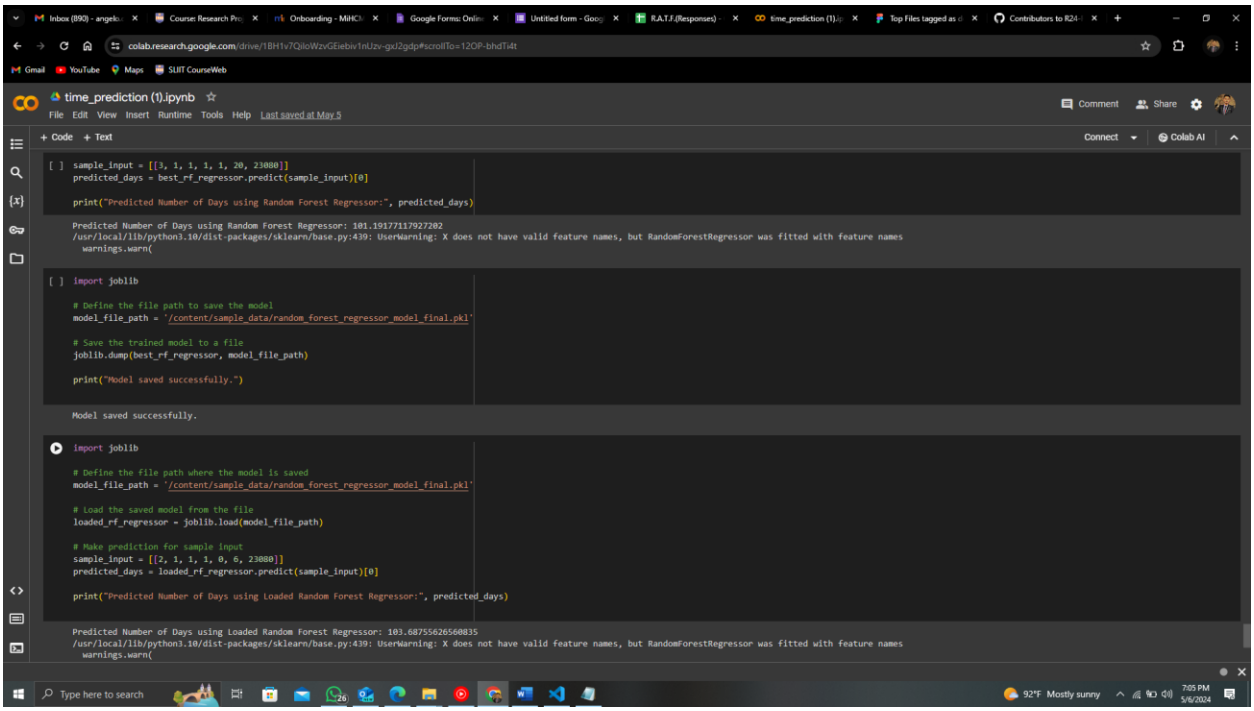


Figure 10-RF without scale


time_prediction (1).ipynb
File Edit View Insert Runtime Tools Help Last saved at May 5
Comment Share
+ Code + Text
Connect Colab AI
sample_input = [[3, 1, 1, 1, 1, 20, 23080]]
predicted_days = best_rf_regressor.predict(sample_input)[0]
print("Predicted Number of Days using Random Forest Regressor:", predicted_days)
Predicted Number of Days using Random Forest Regressor: 101.19177117927202
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:439: UserWarning: X does not have valid feature names, but RandomForestRegressor was fitted with feature names
warnings.warn(
import joblib
# Define the file path to save the model
model_file_path = '/content/sample_data/random_forest_regressor_model_final.pkl'
# Save the trained model to a file
joblib.dump(best_rf_regressor, model_file_path)
print("Model saved successfully.")
Model saved successfully.
import joblib
# Define the file path where the model is saved
model_file_path = '/content/sample_data/random_forest_regressor_model_final.pkl'
# Load the saved model from the file
loaded_rf_regressor = joblib.load(model_file_path)
# Make prediction for sample input
sample_input = [[2, 1, 1, 1, 0, 6, 23080]]
predicted_days = loaded_rf_regressor.predict(sample_input)[0]
print("Predicted Number of Days using Loaded Random Forest Regressor:", predicted_days)
Predicted Number of Days using Loaded Random Forest Regressor: 103.68755626560835
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:439: UserWarning: X does not have valid feature names, but RandomForestRegressor was fitted with feature names
warnings.warn(
Type here to search
92°F Mostly sunny
7:05 PM
5/6/2024

Team Communication

The team chose Microsoft Teams as their primary communication channel, forming a dedicated Team with all four group members. We also used Zoom to communicate with supervisors, provide updates, and receive comments on the project's progress. Regular team conversations were arranged to discuss, share knowledge, and plan.

The crew also used WhatsApp as an additional tool to stay in constant communication with their supervisors. This enabled timely updates and cooperation between the supervisor and co-supervisor, ensuring that everyone was informed and on the same page throughout the project.

Teams Channel

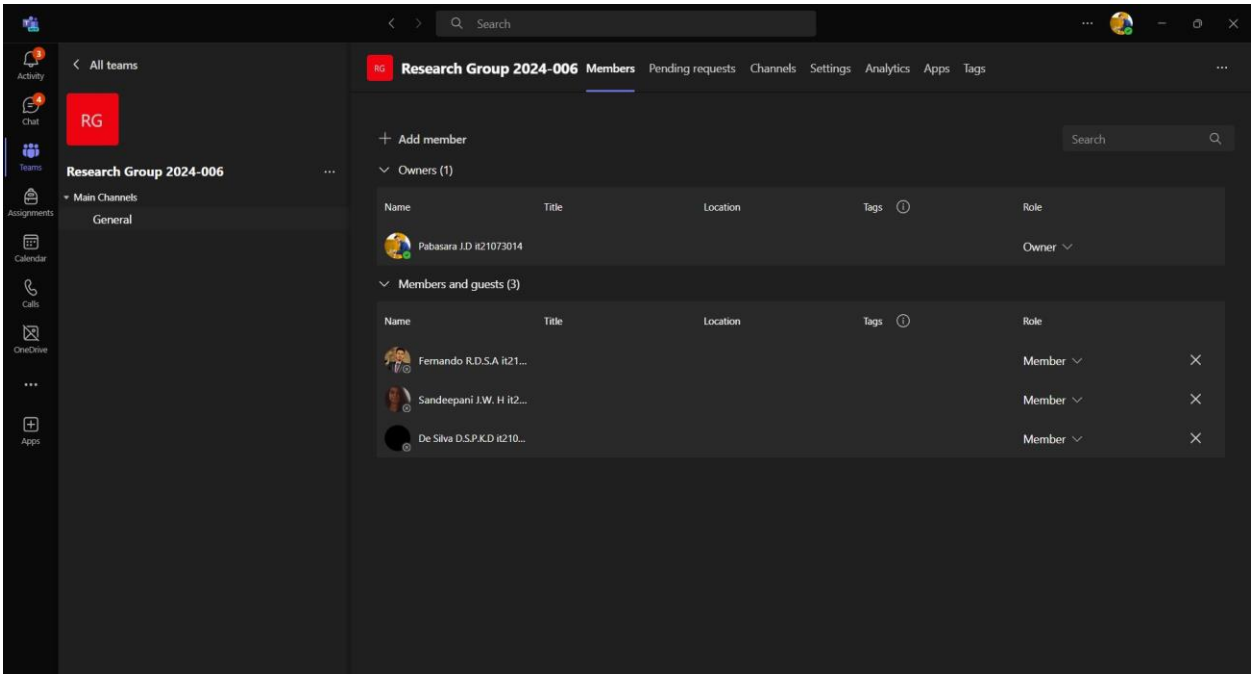


Figure 11 - Teams Channel Creation

Teams Calls with the Research Team

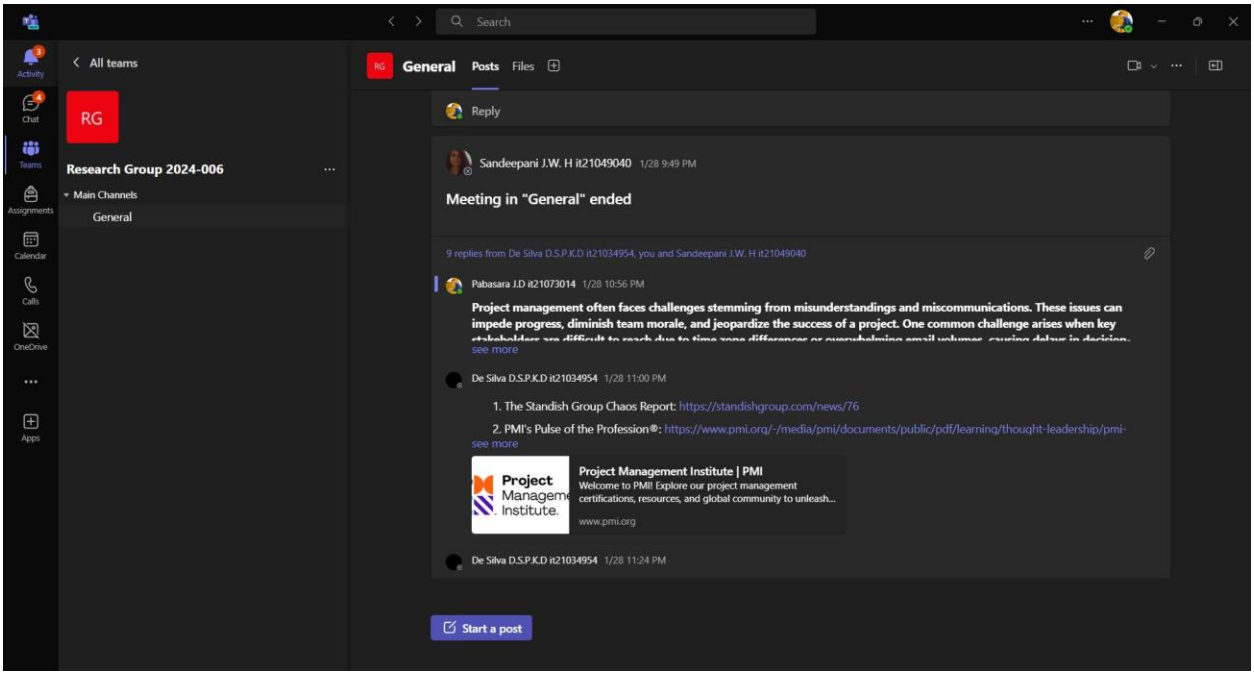


Figure 12 - Overview of Team Calls and Communication

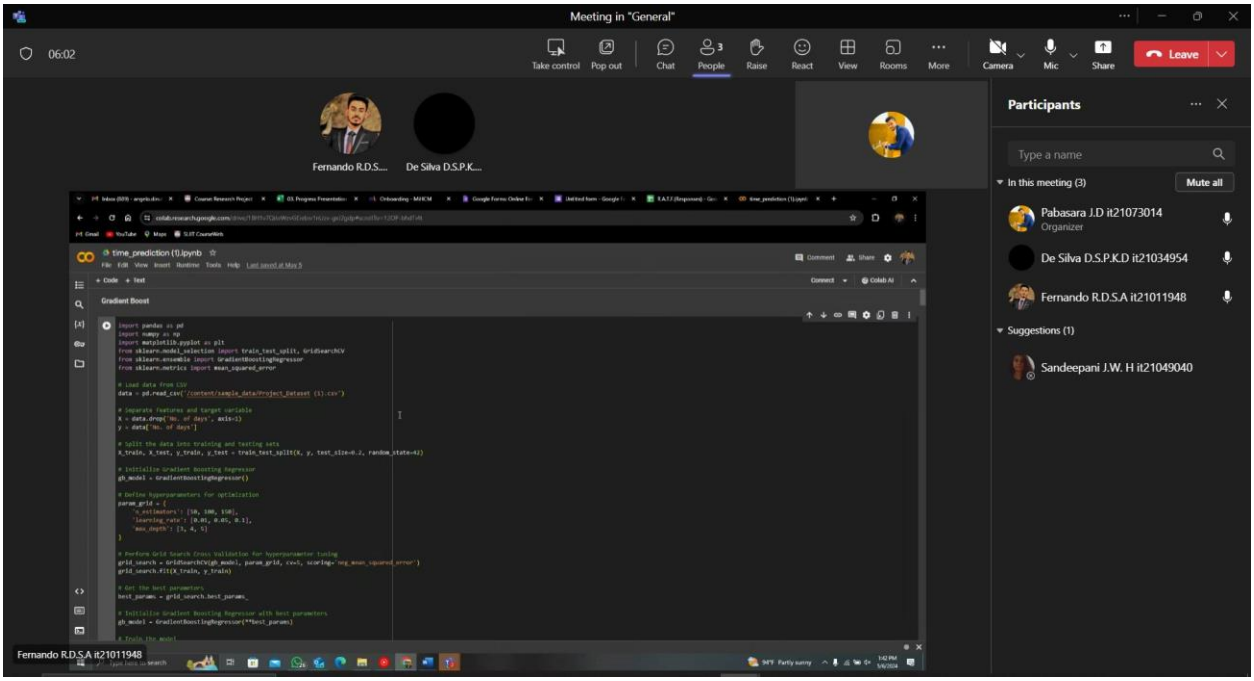


Figure 13 - Team Member Teams Calls (Example 01)

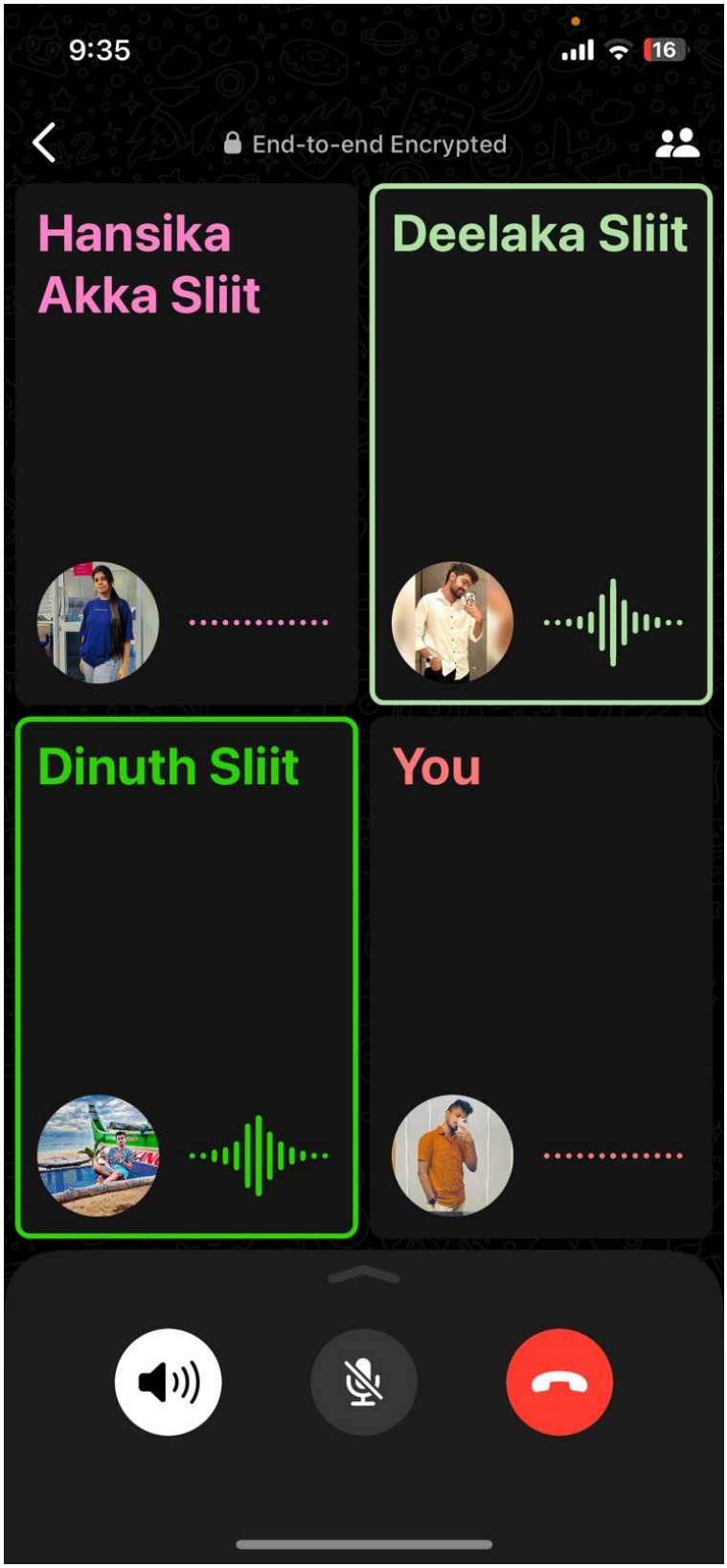


Figure 14 - Team Member WhatsApp Calls (Example 02)

Online Calls with Supervisors

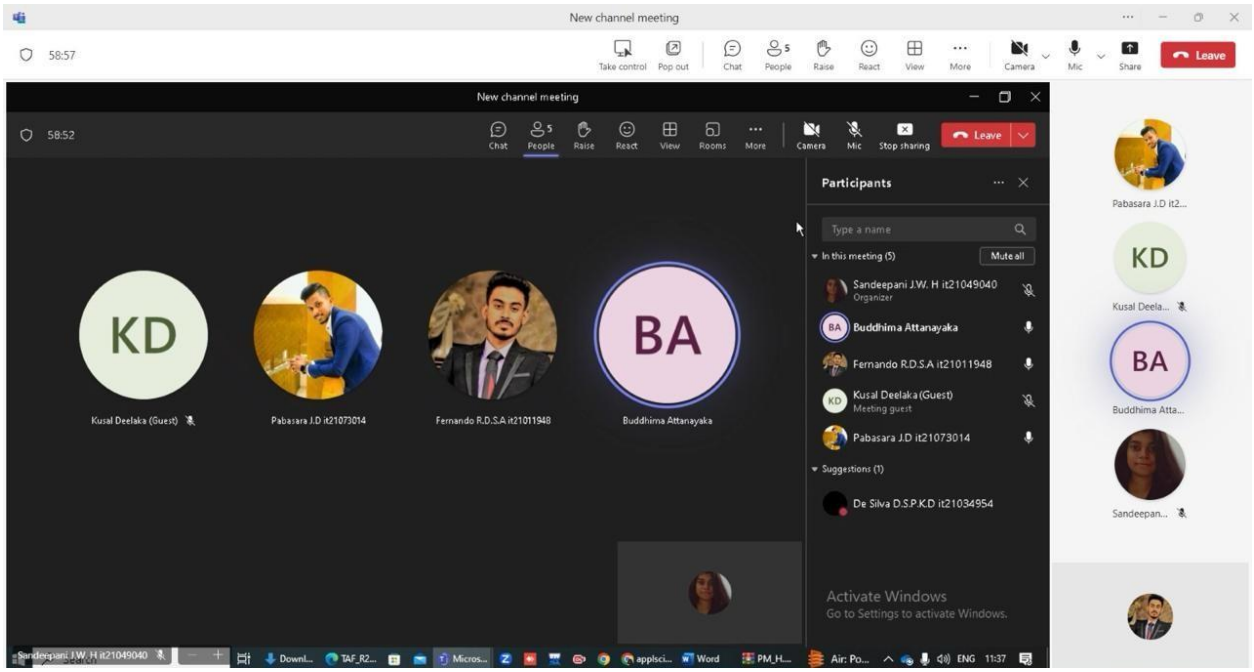


Figure 15 - Teams Meeting with the Co-Supervisor (Example 01)

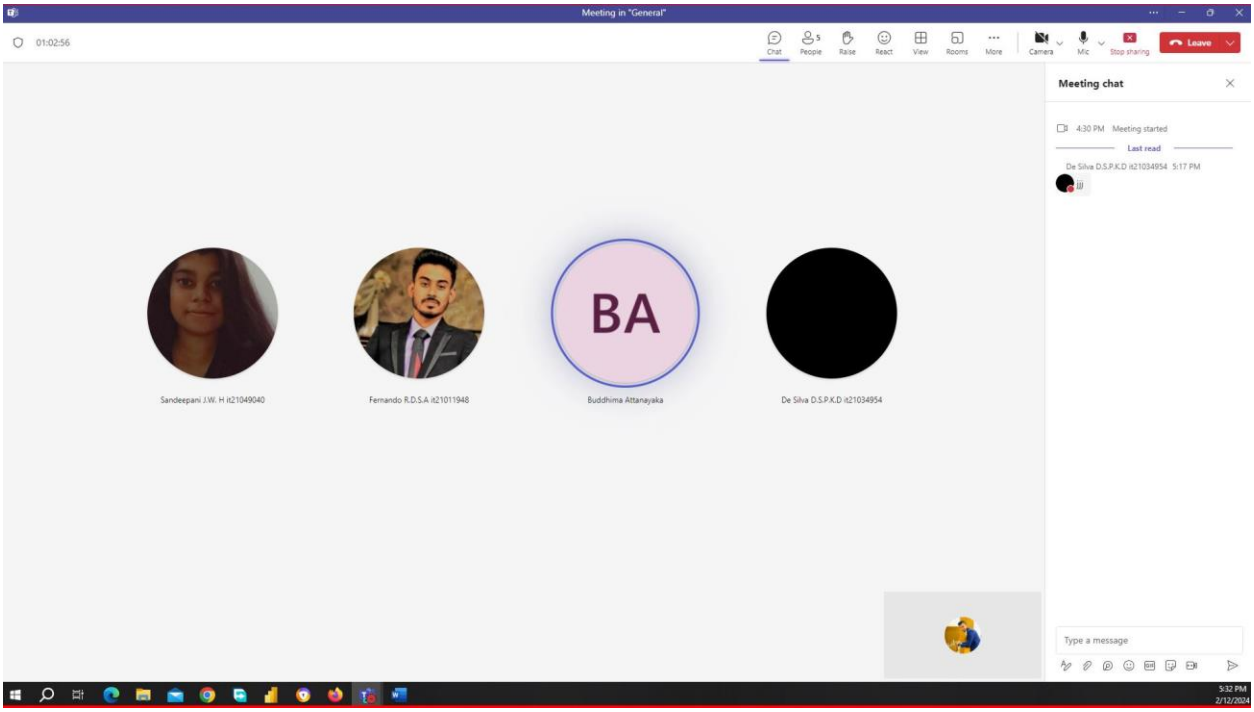


Figure 16 - Teams Meeting with the Co-supervisor (Example 02)

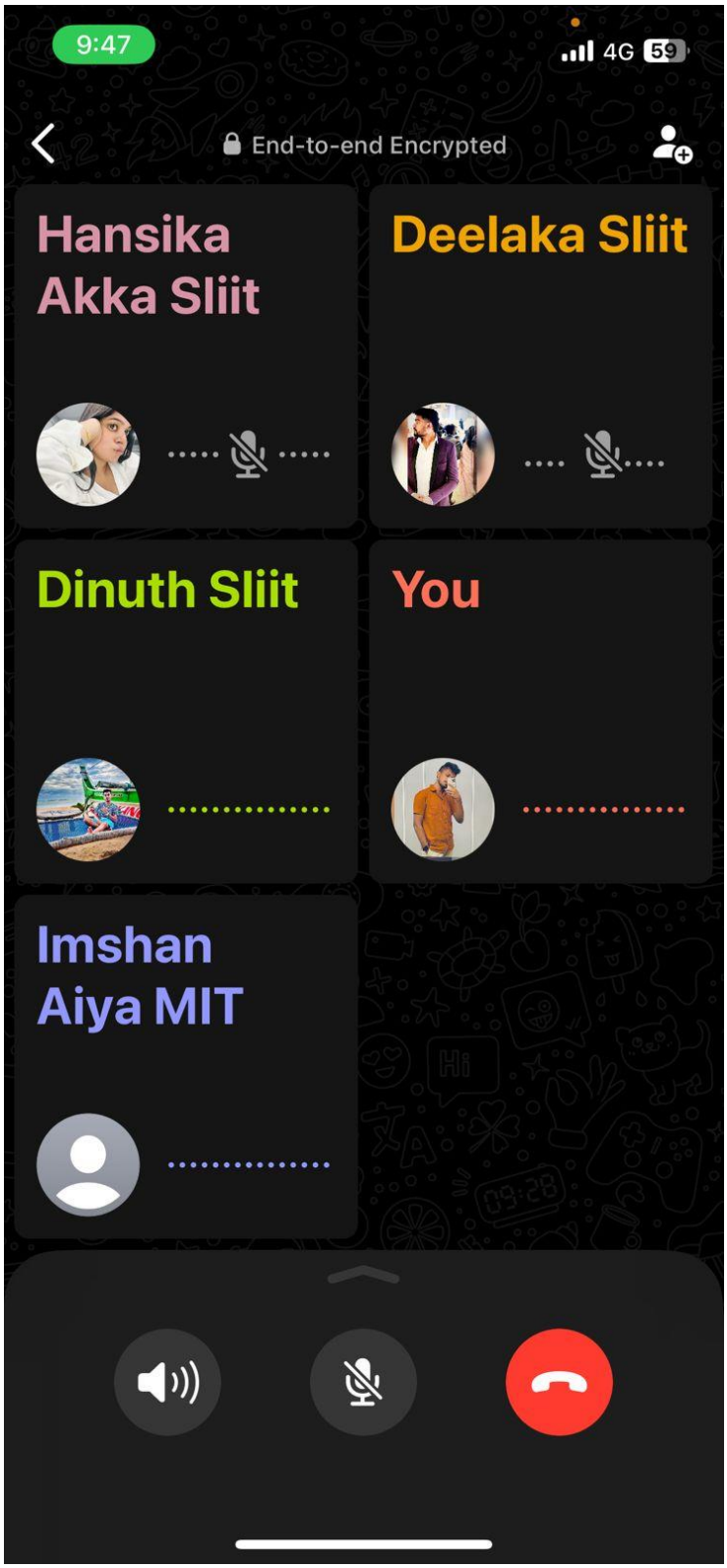


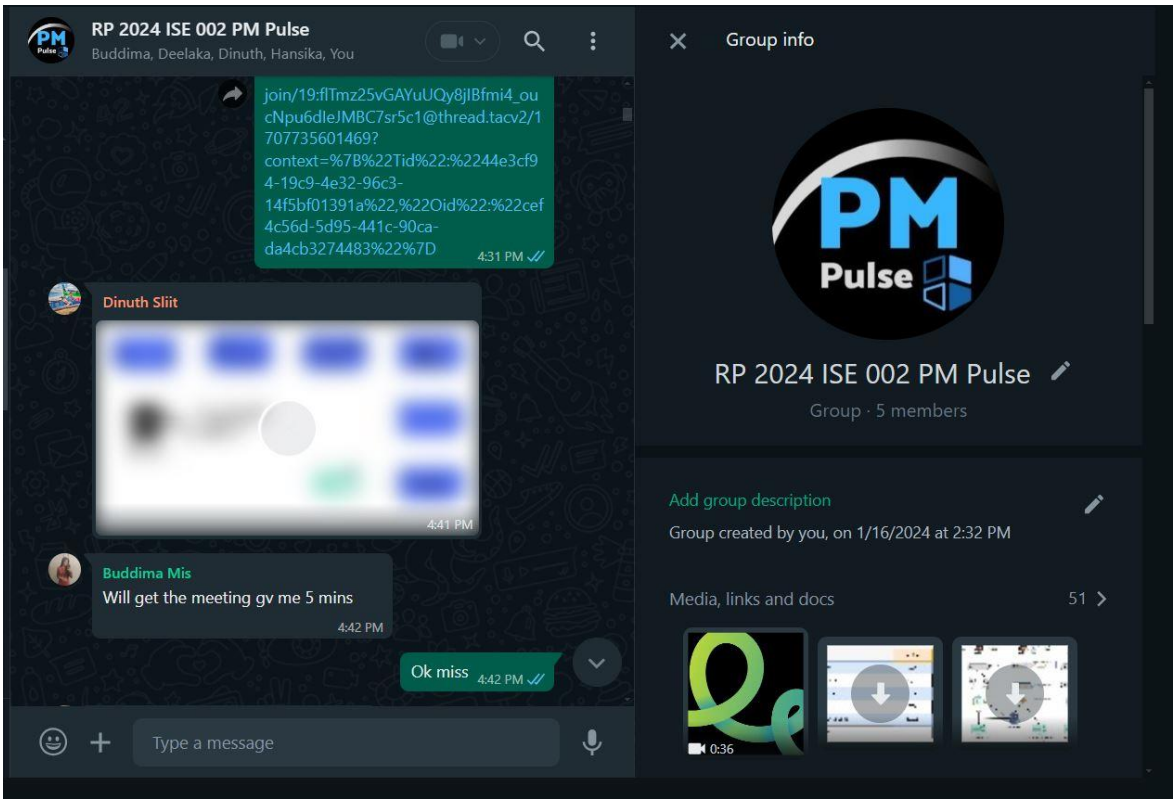
Figure 17 - WhatsApp Call with External Supervisor (LSEG)

Physical Meetings with Group Members

Figure 18 - Physical Meetings with team members.

WhatsApp Group Creation

Figure 19 - WhatsApp Group Creation with Co-Supervisor



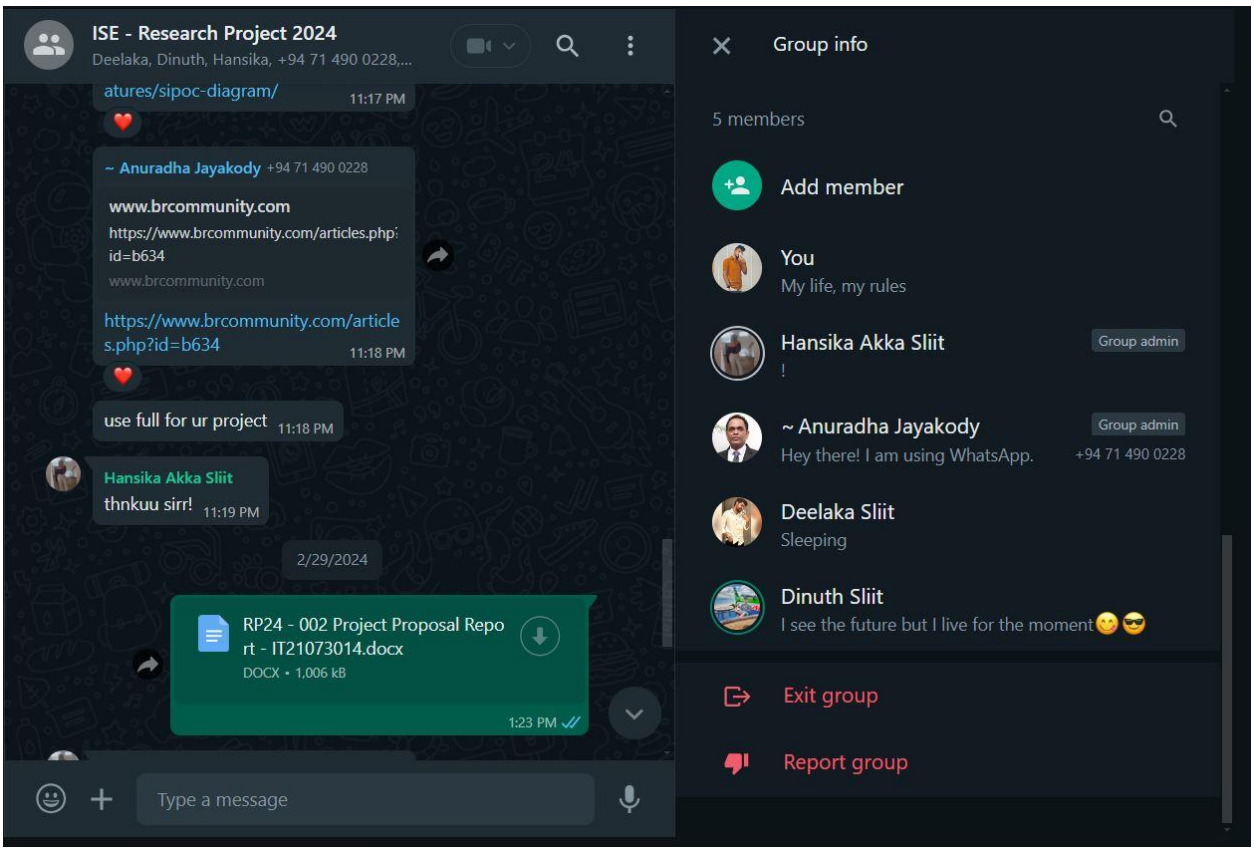


Figure 20 - WhatsApp Group Creation with Supervisor

Project Timeline

We put together a Gantt chart that outlines the entire timeline of our project, encompassing all the deadlines for our deliverables. This visual representation provides a comprehensive overview of our project's progression from its inception to its completion.

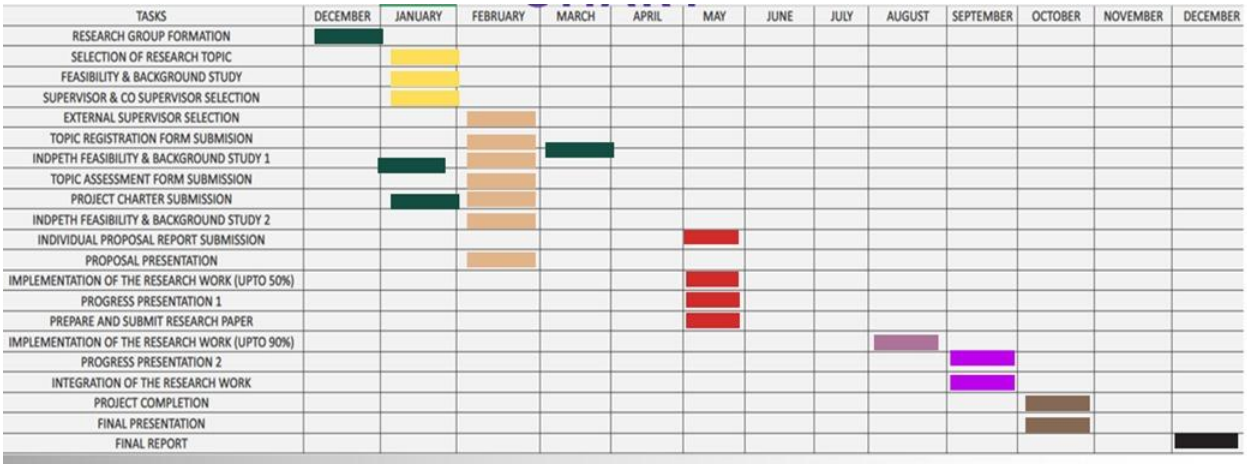


Figure 21 - Gantt Chart (Project Timeline)

Work Break-Down

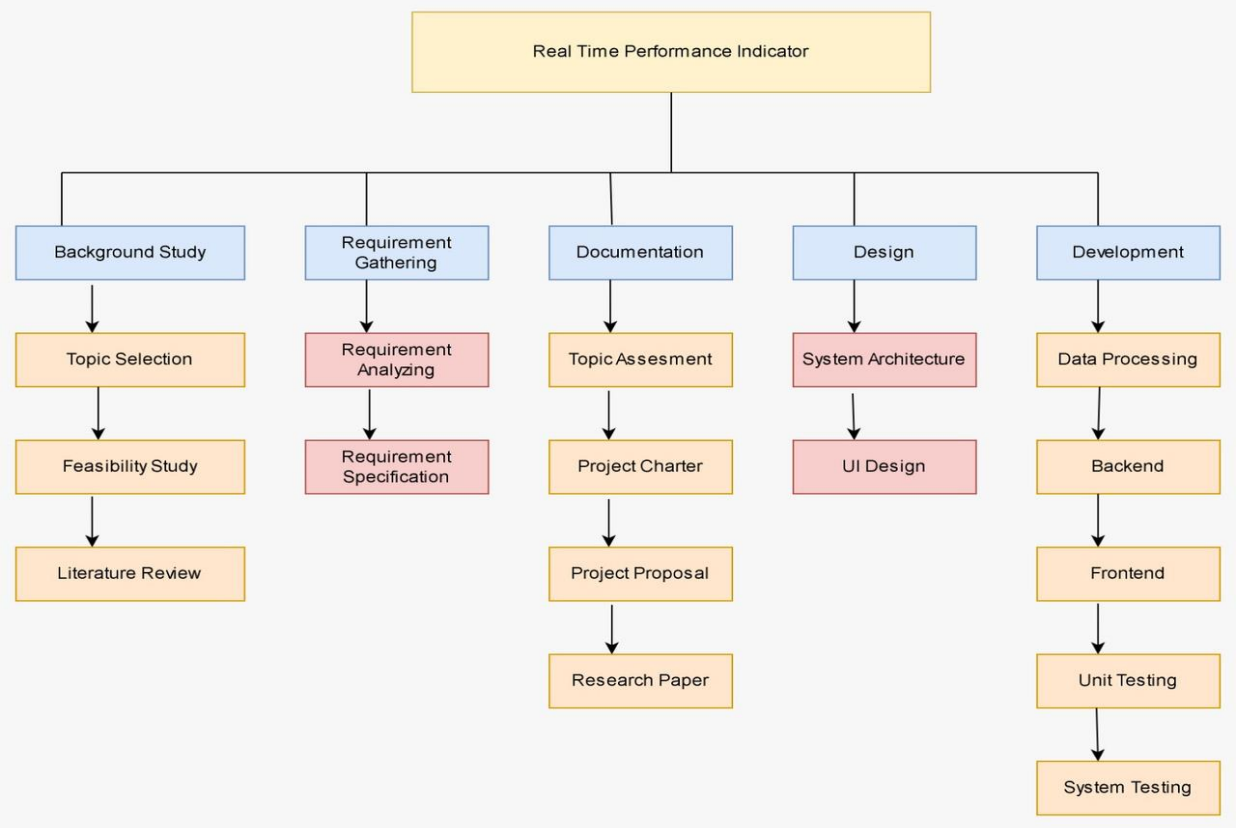


Figure 22 - Work Break-Down Structure

Version Controlling

The team implemented GitHub as their version control system, creating a repository that included all four group members. Each member was responsible for committing their code changes and progress to the repository.

To ensure transparency and collaboration, all code changes were incrementally committed and pushed to individual branches. These branches were later merged into the main branch during weekly meetings. This approach allowed for effective version control and streamlined collaboration among team members.

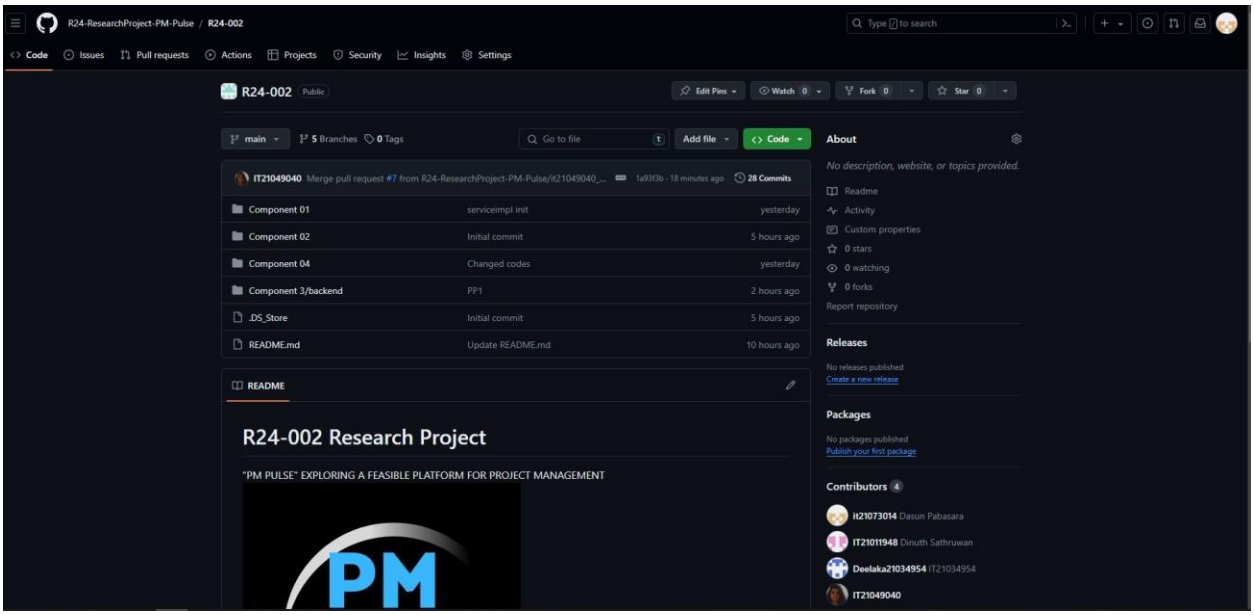


Figure 23 - GitHub Repository

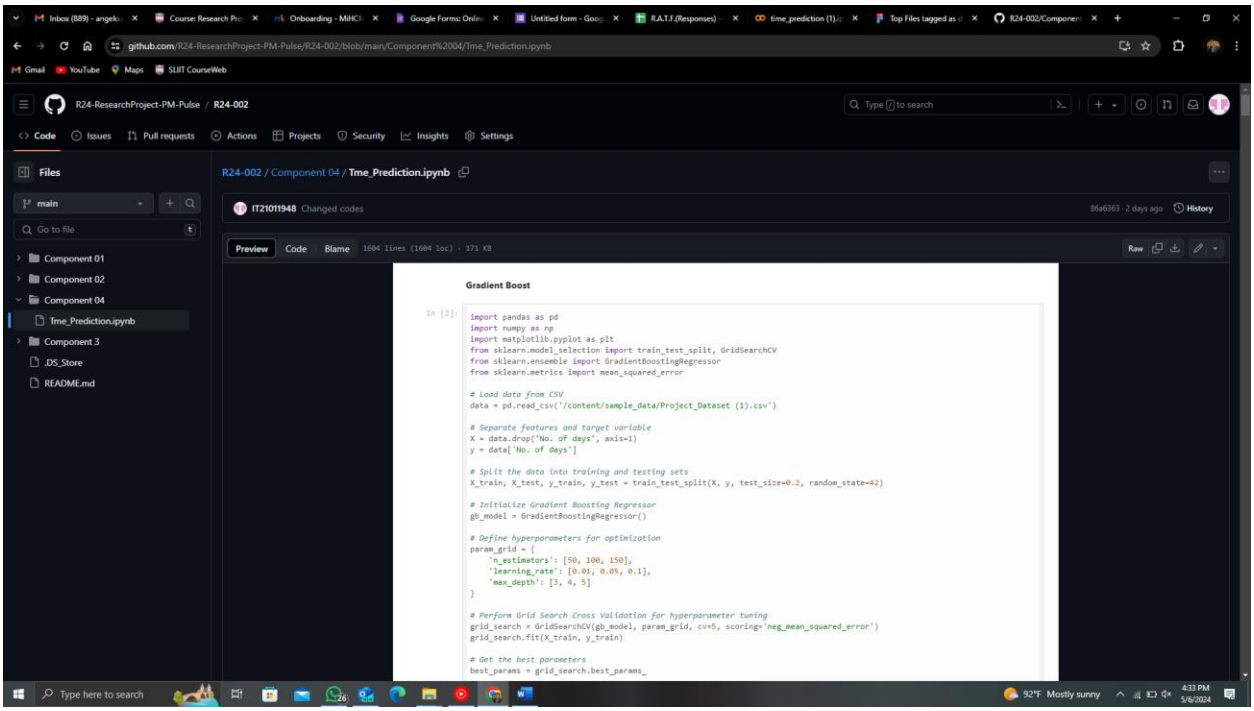
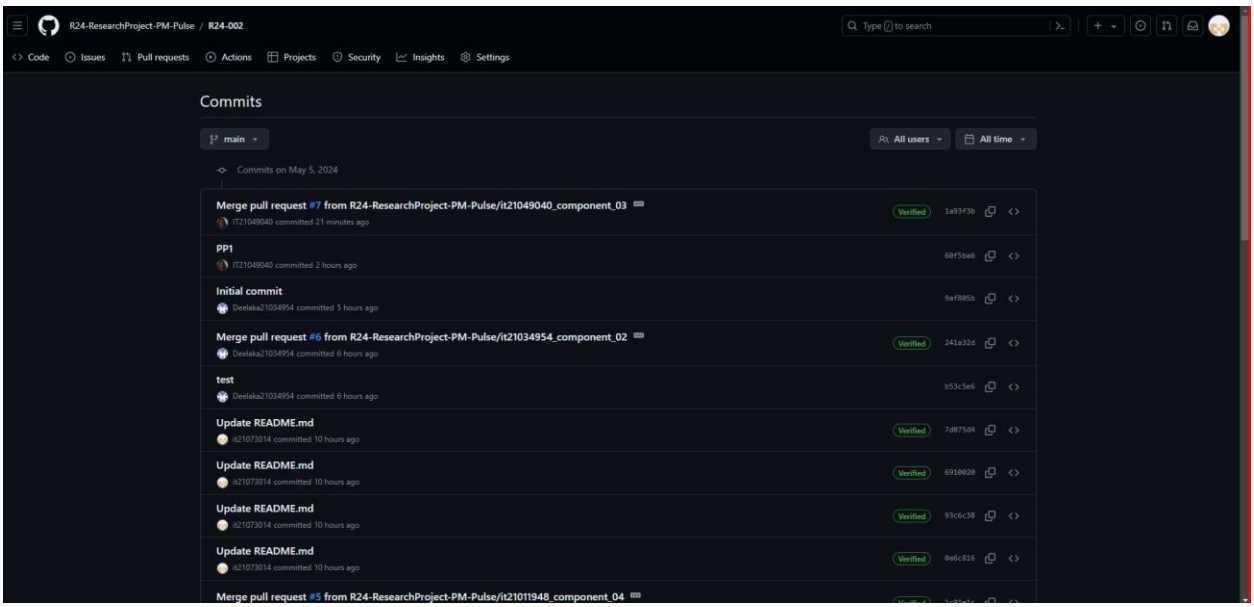
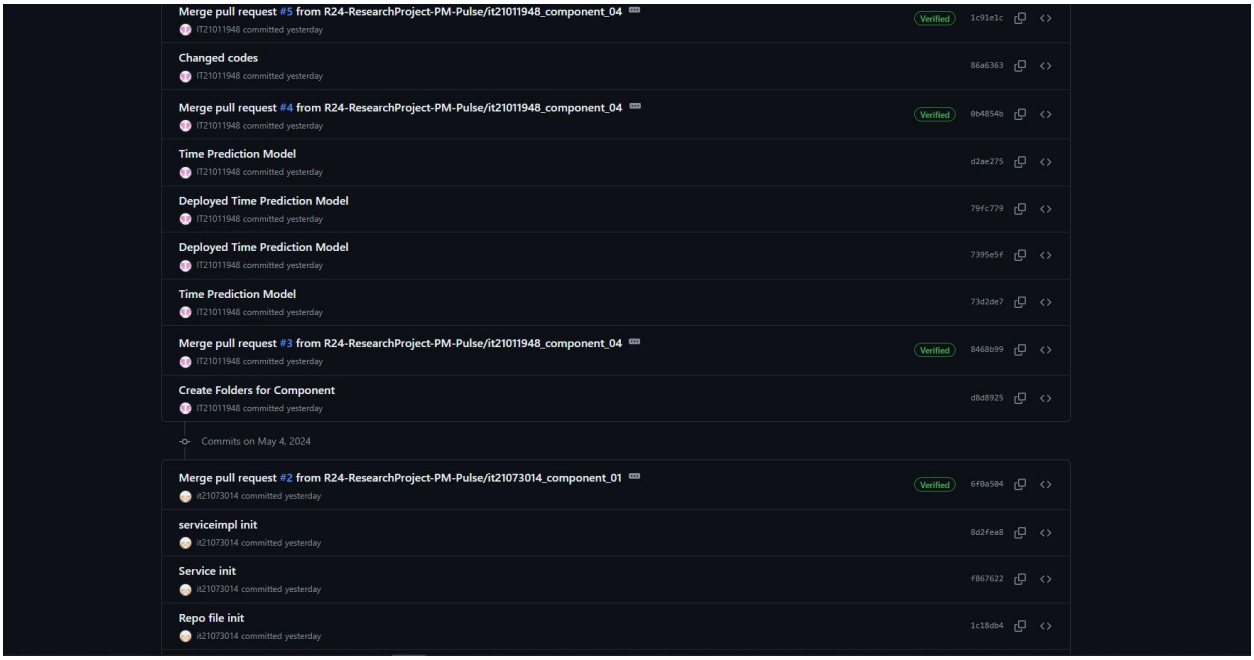


Figure 24 - GitHub Repository (Component 04 Branch)

GitHub Commits





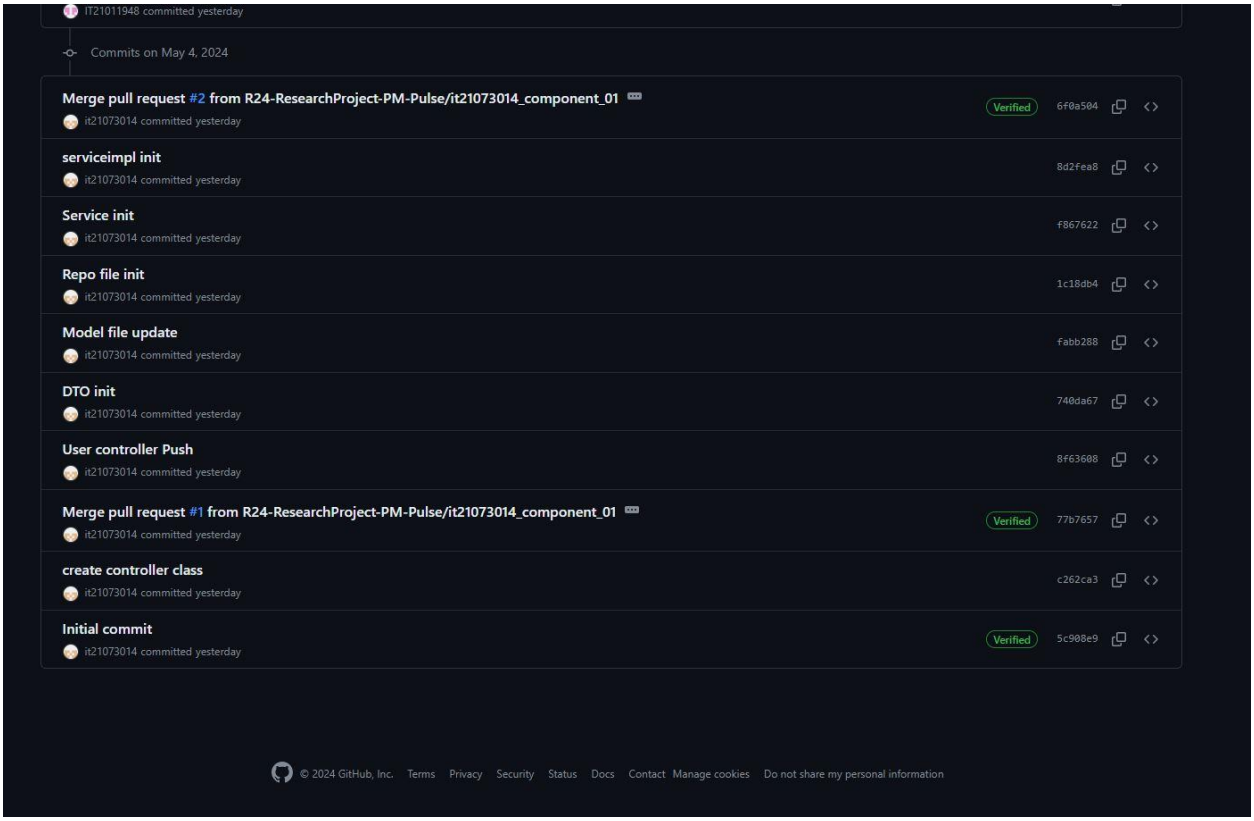


Figure 25 - GitHub Commits

GitHub Graph of Commits

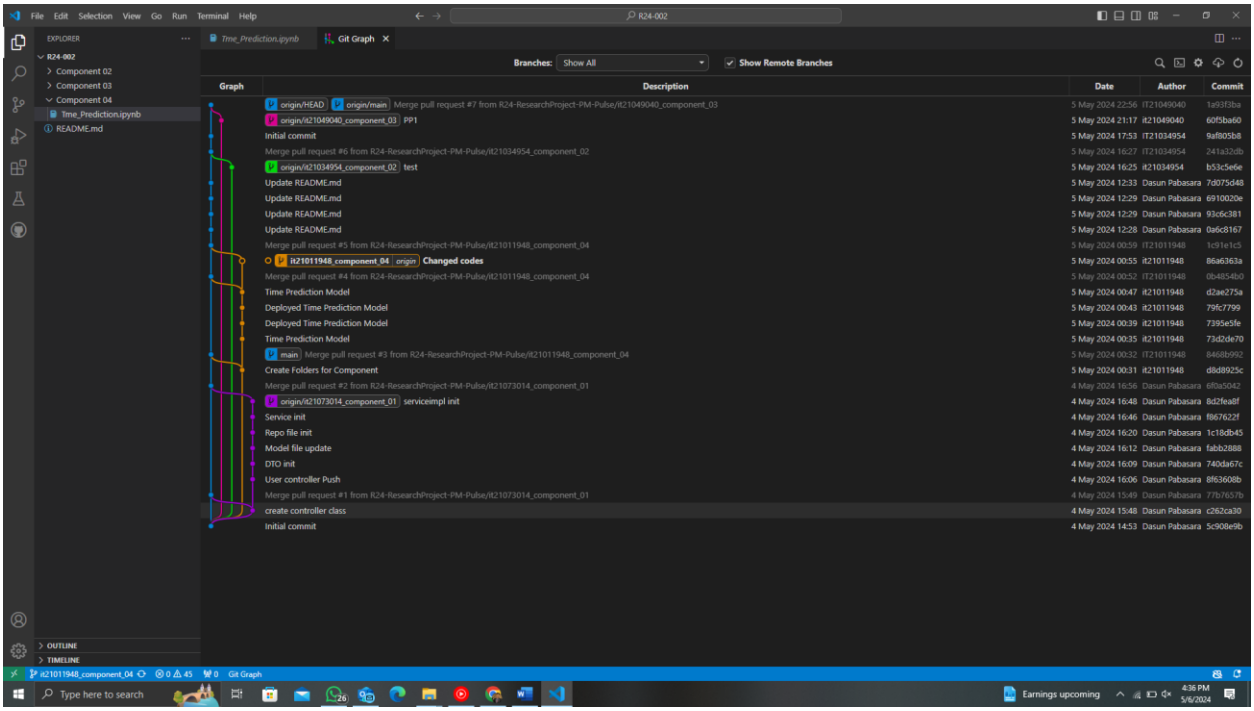
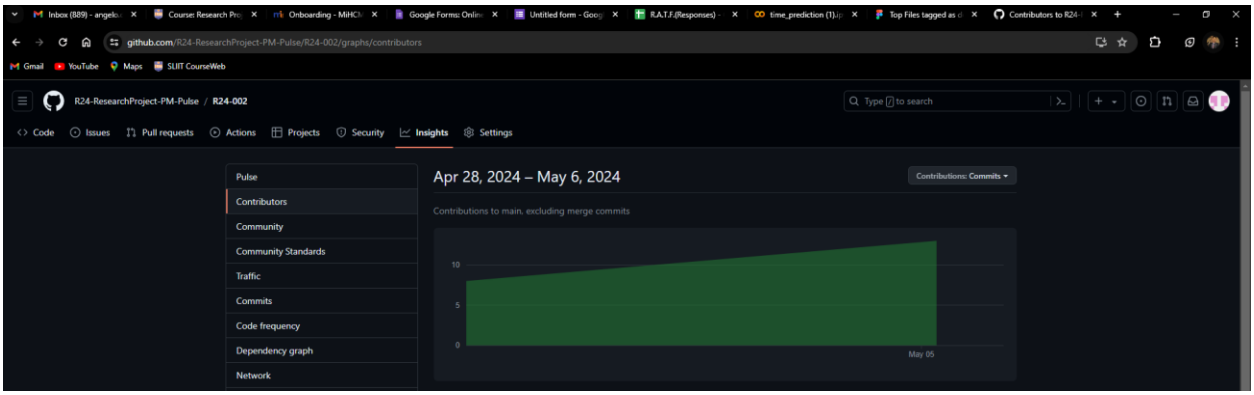
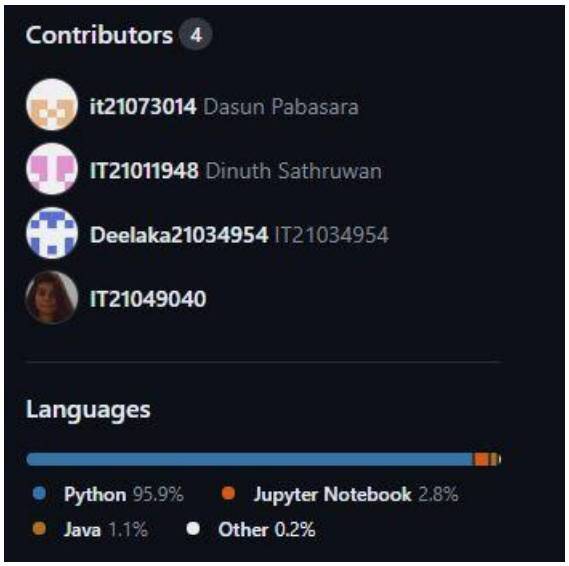


Figure 26 - Git Graph

GitHub Contributors





Contribution Charts

Figure 27 - GitHub Contributors

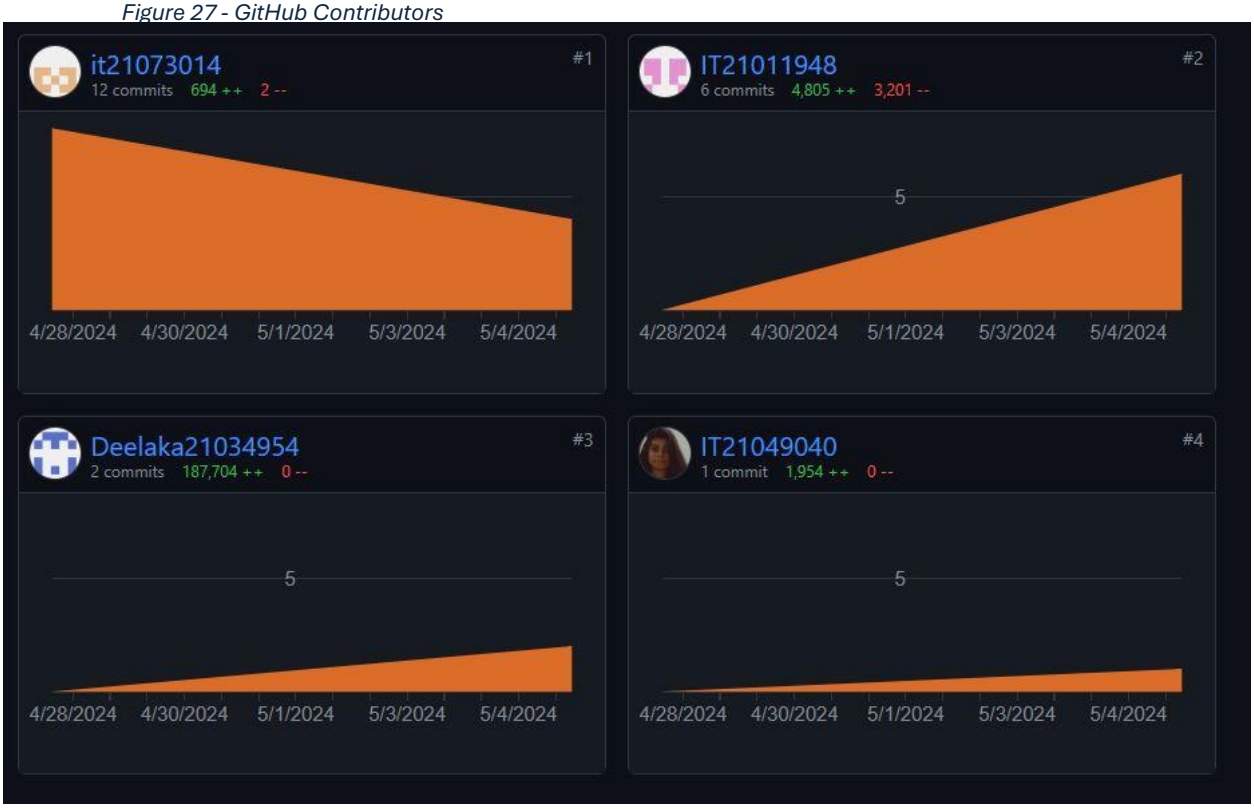


Figure 28 - Contributors Charts