# 3.4 Student Project Log

This document provides a week-by-week summary of my work on the final year project from January to April 2025. The log highlights key tasks, technical implementation, and reflective insights to demonstrate engagement, planning, and development progression. Although I uploaded most of the code to GitLab in batches instead of regular commits, this was due to forgetfulness and not a lack of progress. I recognize the importance of frequent commits and have ensured all code is now pushed and tracked. This log details what I accomplished during each week to support the submitted work.

## January 2025

### Week 1 (Jan 1–7)

* • Reviewed project goals and planned architecture.
* • Selected Flask, React, and Firebase as core technologies.
* • Outlined key features such as role-based access and inventory tracking.

### Week 2 (Jan 8–14)

* • Set up Flask backend and Firebase configuration.
* • Began implementing Firebase Authentication integration.
* • Mapped users to roles using UID-based logic.

### Week 3 (Jan 15–21)

* • Connected backend with Firebase Auth securely using headers.
* • Created company and users subcollections in Firestore.
* • Tested helper functions for UID mapping to full names and company.

### Week 4 (Jan 22–28)

* • Set up React project with basic routing and pages.
* • Implemented authentication UI and Google sign-in.
* • Created SignUp/Login layout with routing logic.

## February 2025

### Week 1 (Jan 29 – Feb 4)

* • Linked frontend and backend using Axios.
* • Added role-based dashboard navigation and redirects.
* • Restricted access to dashboards for approved users only.

### Week 2 (Feb 5–11)

* • Created CSV upload endpoint with Firestore logic.
* • Implemented validation and duplicate checks during upload.

### Week 3 (Feb 12–18)

* • Integrated CSV upload UI with feedback and error reporting.
* • Ensured new or existing items are correctly inserted or updated.

### Week 4 (Feb 19–25)

* • Implemented notification system for admins/managers.
* • Created Tasks UI and styled urgency with colors.
* • Connected frontend to task creation backend.

## March 2025

### Week 1 (Feb 26 – Mar 3)

* • Developed inventory upsert endpoint with quantity and price logic.
* • Integrated top-selling and low-stock analytics backend.

### Week 2 (Mar 4–10)

* • Built Reports & Analytics frontend with filters and report types.
* • Added charts for trends using Chart.js.

### Week 3 (Mar 11–17)

* • Implemented export features: PDF (with cover/chart), CSV, Excel.
* • Styled PDF tables and added low-stock highlights.

### Week 4 (Mar 18–24)

* • Created dual-query report endpoint using timestamps.
* • Converted Firestore timestamps to ISO strings for frontend.
* • Fixed frontend default date range and formatting issues.

## April 2025

### Week 1 (Mar 31 – Apr 6)

* • Polished frontend UI for responsiveness and accessibility.
* • Improved date handling in reports view using ISO strings.

### Week 2 (Apr 7–13)

* • Finalized Firebase re-auth flow for sensitive admin actions.
* • Added promote/demote/remove user endpoints with auth headers.

### Week 3 (Apr 14–20)

* • Cleaned backend with `require\_firebase\_auth` decorator.
* • Refactored to use Flask's `g` for consistent UID access.

### Week 4 (Apr 21–present)

* • Added frontend screenshots with captions to dissertation.
* • Began writing Results & Discussion section.
* • Resolved Firestore server timestamp issues for real-time reports.