### **Academic Issue Tracking System (AITS)**

### **Progress Report**

**Course:** CSC 1202: Software Development  
**Lecturer:** Dr. Peter Khisa Wakholi  
**Team Name:** GROUP C  
**Submission Date:** April 3, 2025  
**Prepared by:** Group C

**INTRODUCTION**

This report covers what our team has accomplished and the problems we’ve faced building the Academic Issue Tracking System (AITS). We’ve got a working system where students can log issues, the Academic Registrar can view them, and users can log in based on their roles.

**Week 1 and 2: Getting Started**  
We Started by setting up a GitHub repository to track our work. We got the backend going with Django and PostgreSQL and set up the frontend using React. We also started figuring out what the system needs, like letting students report issues and the registrar handle them.

**Week 3: Planning the System**  
we wrote down what each user (students, registrar, and lecturers) needs to do and sketched out simple designs for the screens, like a student dashboard. I built the database structure (e.g., linking issues to users) and set up APIs so the frontend could talk to the backend. The team started building basic pages in React to match those designs.

**Week 4: Login and Roles**  
We made it so users can log in and only see what they’re allowed to based on their role (e.g., students can’t assign issues). I set up the login system in Django with secure tokens, and the team connected it to react so the right dashboard shows up after login—like a student page or a registrar page.

**Week 5: Student Features**  
we built a way for students to report issues, like missing marks. I created an API to save issue details (e.g., course code, issue type) in the database, and the team made a form in React that sends the info to my API. We also added a simple student dashboard to show their logged issues, though it’s still basic.

**Week 6: Registrar Features**  
we worked on letting the Academic Registrar see and manage issues. I made APIs to list issues, filter them (e.g., by status), and assign them to lecturers, plus a log to track changes. The team built a table in React to show the issues and buttons to assign them, so the registrar can do their job. It’s working, but the table needs some tweaking to look better.

#### Problems We Ran Into

We faced some common headaches that slowed us down:

* **Learning Curve:** Some of us were new to tools like Django or React, so setting things up took longer than expected. We had to watch tutorials and troubleshoot errors—like getting the database connected—which ate into our time.
* **Bugs:** We kept getting bugs on both the frontend and backend side that took a lot of trial and error to fix.
* **Time Crunch:** We underestimated how long the issue submission form and dashboards would take. Which left us with a basic version that’s not as polished as we wanted.

#### Completed Tasks

* Let’s users log in and see their issues (student or registrar).
* Allows students to submit issues and view their status.
* Let’s the registrar view issues.
* Stores everything in the database and tracks changes.

#### What’s next

* Allow the registrar to assign issues to lecturers.
* Add features for lecturers to see issues that have been assigned to them and add comments.
* Email verification.
* Start sending notifications (like emails) when issues status.
* Test everything and make it ready for the final hand-in.