2024 CSC493 – Capstone Weekly Reports[1]

Your name: Zeke Sklar

Select Report Date: Nov 4, 2025

Part 1: Weekly Progress Report

- **Accomplishments**: What did you accomplish since the last class meeting? *(up to 3 points)*

  Since the last class meeting, I reorganized the project for clarity and deployment readiness by moving local data to data/ (database now data/db.sqlite3), relocating utilities to scripts/, adding a comprehensive .gitignore, and licensing under MIT. I stabilized and polished the admin experience: fixed the missing "recent actions" include, removed the duplicate app list on the dashboard, added one-click inline delete, set dark mode as the default, and aligned the Issue Report page with the site's dark theme. I improved performance and reliability by rendering the Manager Dashboard from cached SNMP statuses (no synchronous polling on first load), sending the daily issue summary email asynchronously after the DB commit (no request blocking), making Whitenoise optional to prevent dev crashes, and fixing an admin change-form TypeError by appending fieldsets safely. I also added a required "Drop-off location" to all supply order flows and included it in ticket details, and constrained Inventory "Shelf Row" to a single A–Z letter with a data migration, admin/form validation, and a small JS input mask. For deployment, I added a pinned requirements.txt, detailed Windows (Waitress) setup in the README, and NSSM service scripts to install/remove the app as a Windows service. Throughout, I opened and linked GitHub issues for bugs and performance regressions, documenting root causes and associated fixes.

- **Challenges**: What are your current roadblocks? *(up to 3 points)*

  Current roadblocks center on performance, deployment, and permissions. SNMP remains a latency risk when devices are offline or a force refresh is requested; even after caching improvements, per-device timeouts (and the pysnmp-lextudio deprecation warning) make a proper background job/queue desirable before I can fully de-risk the manager views. On deployment, I still need production hardening items (finalize .env with real SMTP, ALLOWED_HOSTS, SECRET_KEY; collectstatic run; confirm firewall egress for email; and optionally move from SQLite to PostgreSQL for multi-user writes), plus a quick end-to-end test of the NSSM service on the target server with TLS/reverse proxy. Finally, GitHub permissions remain limited because this is a fork—issues are tracked on your repo, but I don't have triage on the upstream, which slows coordination and documentation of fixes across both repos.

- **Desired Discussion Points**: Do you have any desired discussion points that are not related to roadblocks? *(up to 2 points)*

  For production readiness, do we require migrating to PostgreSQL and adding a background worker (Celery/cron) for SNMP status refresh and daily emails, or is SQLite with on-demand polling sufficient for our expected usage?

- **Future Goal(s)**: What do you plan to accomplish before our next class meeting? These plans should be related to roadblocks or discussion points. If you plan to change direction, explain why. *(up to 2 points)*

[1] Detailed Weekly Report requirements can be found here: 2024 URCPP Capstone - Using Agile and Reporting Out

Created: 2024-09-22; Revised: 2024-09-26

Before next class, I'll finish production hardening and baseline performance: deploy to the target Windows server with NSSM (finalize .env, ALLOWED_HOSTS, SECRET_KEY, SMTP; run migrate and collectstatic), verify end-to-end (admin login, emails, manager views), and add lightweight timing logs to identify slow endpoints. I'll also set up scheduled tasks (daily send_issue_summary and an optional SNMP pre-warm command) and decide the database plan (document staying on SQLite or begin a Postgres migration plan) based on expected load.

Part 2: Time Reporting

Make sure that as you fill out the first prompt, you include in enough detail in the summary. For example, "debugging" is vague, but "debugged function X to make sure that when user does action Y, it is called and returns the value Z" is better.

- **Time Spent**: Briefly explain how much time, *outside of class*, spent on your project. If you worked on multiple components, each should get a detailed summary. Make sure to add up all the hours and minutes correctly. *Add as many rows as you need to the table below. Please do not include hours in class as part of this section. (up to 4 points)*

START	FINISH	HOURS	DETAILED SUMMARY
10/28 2:00pm	10/28 4:30pm	2.5	Reorganized repo (moved DB to data/, scripts to scripts/), fixed settings and script paths, added .gitignore and MIT license, updated README; ran checks/migrations to validate moves.
10/29 10:00am	10/29 12:00pm	2	Admin polish: added fallback for recent actions include, removed duplicate app list on dashboard, added one-click inline delete, set admin default to dark mode; opened GitHub issues and verified fixes.
10/30 1:30pm	10/30 3:30pm	2	Performance/robustness: manager dashboard uses cached SNMP status (no sync polling on initial render); made Whitenoise optional to prevent dev crash; filed related tracking issues.
10/31 3:00pm	10/31 4:30pm	1.5	Added required “Drop-off location” to supply orders (all flows) and included in ticket details; constrained Inventory “Shelf Row” to single A–Z (migration + admin/JS validation); added Windows deployment requirements and NSSM service scripts.
	WEEKLY TOTAL	8	

- **Total (Cumulative) Project Time Spent**: After the number of hours and minutes, make sure to briefly explain whether you are on track and if not, what you may need to do in order to achieve what you set out to accomplish. *(up to 2 points)*

Total time to date: 97 hours 30 minutes (97.5 hours). I’m on track: core reorg, admin polish, performance fixes, and deployment scripts are in place. Remaining to hit goals: finalize production hardening and deploy on the Windows server (set .env/ALLOWED_HOSTS/SECRET_KEY, migrate + collectstatic, verify SMTP, start NSSM service) and run end-to-end tests. If server access slips, I’ll use an interim host to keep testing on schedule; if SNMP latency crops up, I’ll add a simple scheduled pre-warm job to maintain responsiveness.

Rubric:

The following rubric will be used, but they might change as needed.

Accomplishments (3 points)

1 point for a general description of progress, 2 points for specifics on progress, 3 points for specifics AND referring to previous targets and explaining how current accomplishments build on previous ones.

Challenges (3 points)

1 point for mentioning there are roadblocks, 2 points for specifics, 3 points for specifics AND what was done already to try to overcome them.

Desired discussion points (2 points)

1 point for at least one relevant discussion point as a general question, 2 points for relevant discussion points with specifics

Future Goals (2 points)

1 point for concrete future targets (i.e. "working more on the project" is a zero, but "working on getting component X to interface with component Y" suffices), 2 points for tying in the targets with what was hopefully discussed in the meeting.

Time Spent (4 points)

1 point for including general statements of how much time was spent ("4 hours on coding"), 2 points for splitting time into specific parts ("1.5 hours on research on component X, 1 hour coding, 2.5 hours debugging"), 3 points for specific parts and details on the pieces ("1.5 hours researching Turtle interface for drawing concentric circles given inputs from the user, 1 hour coding function X that used that interface, 2.5 hours testing function X by giving it multiple values and fixing errors for values A, B, C, and D"). 1 Point for totalling the hours correctly.

What happens if your time on a task is interrupted and you don't have a concrete (or discrete) end time? In this case put the start time in, and the word "interrupted" for the end time and include the task total time. Rounding to the nearest 15 minutes is acceptable. *(This makes adding up times easier, especially when you use decimal hours, i.e. 3.75 rather than 3 hours 45 minutes.)*

Total (Cumulative) Project Time (2 points)

1 point for summing the values correctly, 2 points for the total time AND reflection on progress (you are confident to fit the target and if not, what course corrections you anticipate needing to make)

Resources

Here's a link to this Weekly Report Template – Make a copy and use it:

2024 CSC493 Weekly Report v2 TEMPLATE