2024 CSC493 – Capstone Weekly Reports[1]

Your name: Zeke Sklar

Select Report Date: Nov 11, 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 production-readied and deployed the printer system. I cleaned and documented the codebase (removed the redundant admin status panel, fixed encoding issues, added clear docstrings, clarified URLs, and tightened views), and added a practical feature to the inventory scanner that captures a 'To' destination on OUT scans with lightweight audit logging. I hardened the configuration for real hosting by moving to env-driven settings (DEBUG/SECRET_KEY), parsing ALLOWED_HOSTS and CSRF_TRUSTED_ORIGINS, adding proxy-aware HTTPS settings, and documenting required environment variables. I set up deployment on DigitalOcean App Platform, resolved DisallowedHost and CSRF origin errors, wired build/run/post-deploy commands, created and pushed to a new public GitHub repo (including a one-time seed DB), and added optional Postgres support to enable persistent data going forward. I also began getting building managers to start using the system for testing and feedback.

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

  The main roadblock is data persistence and hosting constraints. The DigitalOcean App Platform's filesystem is ephemeral, so SQLite resets on redeployment. I need to provision a Managed PostgreSQL instance, set the DATABASE_URL environment variable, and migrate existing data to make the cloud deployment durable. A second challenge is network reachability for SNMP polling: the cloud app likely cannot reach on-campus printers over UDP/161 due to firewalling. Therefore, I either need to run the service on a campus machine (the wired workstation plan) or add a VPN/agent path for device status. Finally, deployment hardening items remain: keeping ALLOWED_HOSTS and CSRF_TRUSTED_ORIGINS exactly aligned with the live domain, creating the initial superuser in the cloud database, and rotating/removing any secrets committed for seeding (e.g., .env/SMTP app password), so onboarding building managers can continue testing without interruptions.

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

  I'd like feedback on the pilot-testing plan with building managers (what success metrics we want to track, which forms/labels need the most polish, and whether the new "To" destination in the scanner should feed a simple report for distribution tracking). I'd also like to align on production posture topics that aren't blockers but benefit from consensus: email summary cadence/content, manager dashboard metrics (e.g., alert counts, low-inventory views), and a lightweight logging/monitoring approach so we can quickly spot slow requests or SNMP reliability issues during the pilot.

- **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 will make the deployment durable and smooth for pilot users. Specifically, I will provision a Managed PostgreSQL instance, set DATABASE_URL in the platform, and migrate data from the current SQLite using a dumpdata/loaddata cycle; then remove any seeded secrets and commit-only DB files from the repo. I will also finalize admin access by creating a cloud superuser, tighten env settings to the exact domain for ALLOWED_HOSTS and CSRF_TRUSTED_ORIGINS, and add a simple home redirect so the root URL is not a 404. For device status, I will test SNMP reachability from a wired campus workstation and, if needed, run the polling component there while the DO app serves the portal/admin, so building managers can continue testing without network blockers. Finally, I will capture initial feedback from managers (forms clarity, scanner "To" usage) and prepare a short report of issues and quick wins for the next meeting.

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
11/04 3:00pm	11/04 6:00pm	3	Code cleanup and documentation: removed duplicate admin status panel, added clear docstrings, clarified URLs, tightened views, and set up safer env-driven settings for DEBUG/SECRET_KEY.
11/05 9:30am	11/05 12:00pm	2.5	Admin polish and encoding fixes: consolidated live status into template override, improved inventory picker/admin help, fixed mojibake across JS/templates, and verified UI behavior.
11/05 2:00pm	11/05 3:30pm	1.5	Scanner enhancement: added optional "To" destination for OUT scans, updated UI and results table, and added lightweight audit logging to data/inventory_scans.log.
11/06 10:00am	11/06 1:00pm	3.0	DigitalOcean deployment: configured build/run/post-deploy commands, mapped env vars (ALLOWED_HOSTS, CSRF_TRUSTED_ORIGINS, SECRET_KEY), added proxy-aware HTTPS settings, resolved DisallowedHost and CSRF errors, and tested admin login.
11/10 10:00am	11/10 12:00pm	2.0	PostgreSQL support: added dj-database-url and psycopg, wired DATABASE_URL override with secure defaults, documented migration path (dumpdata/loaddata), and prepared next steps.
11/11 8:30am	11/11 9:30am	1.0	Pilot prep: drafted manager testing instructions and training, reviewed scanner change, and compiled weekly report.
	WEEKLY TOTAL	14.5	

- **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: 112 hours 0 minutes (112.0 hours). I am on track: this week I completed production readiness work, deployed to DigitalOcean, added the scanner destination feature, and began manager onboarding. To hit the remaining goals, I will provision Managed PostgreSQL, set DATABASE_URL, migrate from SQLite, lock down ALLOWED_HOSTS and CSRF_TRUSTED_ORIGINS to the exact domain, create the cloud superuser, remove any committed secrets or seeded DB files, add a simple home redirect, and finalize the SNMP plan (poll from a wired campus workstation or via VPN/agent). If the Postgres or

networking setup takes longer than expected, I will continue the pilot from the campus workstation host and schedule the cloud cutover immediately after to keep testing on track.

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