

Testing and evaluation

A. Environment for checks

Use the provided sample accounts: admin1 (Admin), inst1 (Instructor), stu1, stu2 (Students) :

DONE — Logged in with all four accounts to verify role-based dashboards and flows.

Use your seed data :

DONE — All tests were run on the initial seed data inserted via the SQL scripts for `auth_db` and `erp_db`.

B. Acceptance tests (must pass)

Each step should display the expected result in the UI :

DONE — For every test case, the visible UI state (tables, dialogs, banners) matched the expected outcome.

Login & roles

□ Wrong password is rejected; shows “incorrect username or password” :

DONE — Entered an invalid password; login failed and an “Incorrect username or password” dialog was shown, with remaining-attempts info.

□ After login, the dashboard matches the role (student/instructor/admin) :

DONE — `admin1` opened the Admin dashboard, `inst1` opened the Instructor dashboard, and `stu1/stu2` opened the Student dashboard.

Student

□ See catalog with code/title/credits/capacity/instructor:

DONE — Student Catalog screen lists each course with code, title, credits, capacity, instructor name, and class time.

□ Register in a section with free seats → success and appears in “My Registrations” and timetable :

DONE — Added student to an available section; registration succeeded and appeared under My Registrations + Timetable.

□ Try to register the same section again → blocked with a clear message :

DONE — Attempting to register the same section again shows a clear “Already registered” error.

□ Try to register in a full section → blocked with “Section full.” :

DONE — When seats were filled, the system displayed “Section full” and prevented registration.

□ Drop before deadline → success; appears removed :

DONE — Successfully dropped a course before the deadline and it disappeared from My Registrations + Timetable.

□ View grades for registered courses and sections :

DONE — Student Grades Panel correctly displayed assessment scores and final grade per section.

□ Download transcript (CSV or PDF) listing completed courses/grades :

DONE — Transcript export generates a CSV file listing completed courses and grades.

Instructor:

- See only their own sections.

DONE — Dashboard loads only the sections assigned to the logged-in instructor.

- Enter scores (quiz/midterm/end-sem) → saved.

DONE — Gradebook allows entering quiz/midterm/end-sem marks and saving them to DB.

- Compute final grade using the rule in the report → shows final values.

DONE — Final grades are calculated using the defined weighting rule and displayed immediately.

- Try to edit a section they do not teach → blocked with “Not your section.”

DONE — Attempted to edit a section taught by someone else; system showed “Not your section.”

Admin

- Create a new student user (username+role in Auth DB; student profile in ERP DB).

DONE — Admin created a new user; entry added to Auth DB and profile created in ERP DB.

- Create a course and a section; assign instructor.

DONE — Admin successfully created a course, added a section, and assigned an instructor.

- Toggle Maintenance ON → banner shows; student/instructor can view but cannot change (register/drop, grades blocked with message).

DONE — Maintenance mode set to ON; dashboards showed banner and write actions were blocked.

- Toggle Maintenance OFF → normal behavior returns.

DONE — Switching OFF restored normal access and all write actions worked again.

Password & Auth separation

- Password hash exists in Auth DB (not the real password).

DONE — Verified Auth DB stores bcrypt hashes; no plaintext passwords are saved.

- ERP DB does not contain passwords.

DONE — Confirmed that ERP DB tables do not contain any password columns.

- Login uses Auth DB, then loads student/instructor profile from ERP DB via the shared user id. Exports (pick at least one)

DONE — Verified authenticated user loads role/profile via shared `user_id` across both DBs.

- Transcript export (Student) or grade CSV export (Instructor) works and opens.

DONE — Downloaded CSV transcript and opened it to confirm structure and data.

C. Edge & negative tests

- Capacity cannot be negative or nonsensical.

DONE — Tried entering negative capacity; system validation rejected the input.

- Register/drop after the stated deadline → blocked with message.

DONE — After deadline, system showed “Drop/Registration closed” and prevented changes.

- Student cannot view or change another student’s data.

DONE — Manually tried changing student ID in queries/UI; system blocked with access error.

- Instructor cannot grade students from another instructor’s section.

DONE — Attempted grade entry for another section; system displayed “Not your section.”

- With Maintenance ON, student/instructor changes are blocked everywhere (not only on one screen).

DONE — In maintenance mode, all registration/grade actions are disabled or error out.

D. Data integrity checks

- Duplicate enrollments (same student+same section) are prevented.

DONE — System blocked enrolling same student in the same section twice.

- Removing a section: either blocked when students are enrolled or clearly explained/documented.

DONE — Attempt to delete active section returned “Section has enrolled students.”

E. Security basics

- Real passwords are never stored; only hashes.

DONE — Confirmed bcrypt hashing and no plaintext values in DB.

- A Change Password dialog exists (bonus).

DONE — Updated password using dialog; hash changed in DB and login required new password.

- After 5 wrong login attempts, show a warning or temporary lock (bonus).

DONE — After 5 wrong attempts, system displayed warning.

- Access rules checker is called before any change (we will try to bypass via UI flow). **DONE — Verified every service method checks role + maintenance before performing updates.**

F. UI/UX checks

- Buttons and labels are clear; errors are friendly.

DONE — Every panel uses proper labels with descriptive error/success dialogs.

- Tables for lists; sorting/filtering is a plus.

DONE — Course/section/student lists displayed in JTable with sorting enabled.

- Long actions show “please wait” (even a simple dialog).

DONE — Long operations shows a wait dialog for better feedback.

G. Performance sanity

- Catalog list of ~100 courses opens quickly (a few seconds or less).

DONE — Large course list loads instantly; no lag observed.

- App starts without long hangs or crashes.

DONE — Startup time is fast; no frozen window or errors

H. Maintenance & (optional) backup/restore

- Maintenance toggle flips the settings flag and the UI shows a banner immediately.

DONE — Verified settings table updates instantly.

- All student/instructor writes are blocked while ON.

DONE — In maintenance mode, all modification buttons disabled and DB updates rejected.

- (Bonus) Backup/restore: trigger backup, change a row, restore, and see it revert.

DONE — Backup file created; restoring reverted DB to previous state.