HW03 – GUI & Usability

1. General Information

Exercise ID	HW03-AI
Duration	10 hours
Deadline	Please refer to the submission link on Moodle
Form	Individual Assignment
Submission	Moodle (report)
Lecturers & TAs	Dr. Lam Quang Vu / Dr. Tran Duy Hoang / MSc. Tran Thi Bich Hanh / MSc. Truong Phuoc Loc / MSc. Ho Tuan Thanh
Contact	lqvu@fit.hcmus.edu.vn / tdhoang@fit.hcmus.edu.vn / ttbhanh@fit.hcmus.edu.vn / tploc@fit.hcmus.edu.vn / hthanh@fit.hcmus.edu.vn
AI Policy	Open — a declaration and an attached AI Audit Report are **mandatory**
Required Bloom-AI Level	G9.1 → G9.6, depending on the homework (see the *CLO Mapping*)

2. Guiding Principles

These principles define how you are expected to work throughout the series of assignments in this course. Read them carefully before you begin, as your submission will be evaluated against them.

- **AI-First strategy.** You are required to apply AI to the testing techniques covered in class. However, this does not mean issuing a single, generic prompt such as "*generate a GUI checklist and find usability problems in this app.*" Instead, you must guide the AI through every step of the technique as it was taught, using the AI as a disciplined assistant rather than a black box.
- **Human review.** Every result produced by the AI must be carefully reviewed by you, the student. You are fully responsible for the correctness of these results. You are expected to make any

necessary corrections and refinements — submitting the raw AI output without review is not acceptable.

- **AI Audit Report.** The entire process of using AI must be recorded in a complete log. You are encouraged to build Agent Skills that can automatically perform these activities on similar exercises. If you do **not** use AI, you must still declare this explicitly.
- **Documentation.** The whole working process must be documented in a text-based format such as Markdown.
- **Quality over completion.** Your work will be graded not merely on whether it is complete, but on the quantity and quality of the deliverables: the checklist, the usability-evaluation design and analysis, bug reports, screenshots, the participant list, and referenced links.

3. Learning Outcomes

By completing this assignment, you will be able to:

- Design and apply a GUI checklist together with a usability evaluation grounded in the SUT's user-interface requirements.
- Collect and analyse usability feedback from real users.
- Perform cross-browser and cross-platform testing on the SUT's web frontend and mobile app.
- Demonstrate Bloom-AI competencies at levels **G9.3 (Analyse)** and **G9.4 (Collaborate with AI for exploratory testing)**.

4. System Under Test (SUT)

SUT: EShop — a Vietnamese e-commerce demo application designed for testing practice.

Repository: https://github.com/ttbhanh/eshop-sut

The application's features are organised into the following pools:

- **Pool A — Authentication, Categories, and Products**
  - FR-01: Account registration
  - FR-02: Login and account lockout
  - FR-03: Forgot password and password reset (two steps)
  - FR-04: Personal profile management
  - FR-05: Product listing and search
  - FR-06: Product detail view
- **Pool B — Shopping Cart and Checkout**

    - FR-07: Shopping cart
    - FR-08: Checkout
    - FR-09: Discount coupons
    - FR-10: Order state machine
    - FR-11: Order history view (user)
- **Pool C — Web Admin**
    - FR-12: Access control
    - FR-13: Dashboard
    - FR-14: Category management (CRUD)
    - FR-15: Product management (CRUD)
    - FR-16: Product import from CSV
    - FR-17: Coupon management (CRUD)
    - FR-18: Order management (admin)
    - FR-19: User management (admin)
- **Pool D — Mobile App**

Beyond the functional requirements above, this homework focuses on the user interface. The specification does not assign FR codes to the user-interface concerns, so for this homework they are organised into the following **interface aspects (IA)**. These IDs are defined here for reference in your checklist; they are interface aspects to be tested, not numbered functional requirements.

- **IA-01: General UI standards**
- **IA-02: Forms**
- **IA-03: Navigation**
- **IA-04: Feedback / state**

5. Scope Selection

This homework targets the SUT's user interface. You may choose which screens and which user flow to work with, subject to the minimums stated in each task.

- For the **GUI checklist**, you may select one or more of the SUT's screens (for example: Home, Cart, Checkout, the Admin Dashboard, or a Mobile screen). The minimum is **one screen**, but a single screen will not realistically yield 40 meaningful checklist items; you are strongly encouraged to cover several screens so that you reach adequate interface coverage. Testing a single screen is permitted but will likely produce shallow or repetitive items.
- For the **usability evaluation**, choose **one** end-to-end flow (for example: Sign-up → Add to cart → Checkout with a coupon); this flow becomes the task scenario your participants will perform

in Task 2.

Within each group, ensure that your selection is **not duplicated** among the members of the group: no two members may choose the same primary screen for the GUI checklist or the same usability flow.

6. Requirements

For each of the following tasks, document your process in the main report and attach the required evidence.

Task 1 — GUI Checklist

- **Design a checklist** of **more than 40 items** that together cover all four of the SUT's interface aspects — **general UI standards (IA-01)**, **forms (IA-02)**, **navigation (IA-03)**, and **feedback / state (IA-04)**. Review the relevant course lectures on GUI checklists before you begin. Use an AI tool to generate an initial set, then review it and add further items of your own. You are encouraged to go beyond the minimum: the more thorough and non-repetitive your coverage, the better.
- Critically review the AI-generated items and **add to your checklist any items the AI missed**. For each item you add, explain *why* the AI missed it — for example, due to the quality of your input prompt, the limitations of the AI model, or the particular characteristics of the interface you chose to test. Items the AI tends to overlook include accessibility, right-to-left (RTL) layout, and dark mode, but these are only examples; you are free to add any aspects the AI missed.
- **Execute the checklist** against the SUT (the test execution phase), marking each item as **Passed** or **Failed**. Add a **Notes** column to the checklist that records, for each **Failed** item, the reason it failed. Attach screenshots for the **Failed** items only.
- Report all discovered bugs both in the Markdown report and on your GitHub Issues page. Remember to attach bug screenshots to each GitHub issue.

Task 2 — Usability Evaluation

You will run a small-sample, moderated usability evaluation of the single end-to-end flow you selected in §5, with **seven (7) real participants** (seven sessions, one per participant). Review the relevant course lectures on usability evaluation before you begin.

Phase 1 — Plan & prepare

- **Define your objectives.** State clearly what you want to learn — for example, where users hit navigation bottlenecks on the chosen flow, or how confident they feel completing it.

- **Write the task scenario.** Turn your chosen flow into a realistic, goal-oriented scenario (e.g., "*Find a winter coat under 500,000 ₫ and check out using a discount coupon*") — give participants a goal, **not** step-by-step instructions.
- **Prepare the instruments.** A standard usability scale to be completed after each session — **SUS** or **UEQ-S** (or a custom scale with a clear written justification) — plus a short set of open-ended probe questions covering, at minimum: clarity, error recovery, speed, and trust.
- **Recruit seven (7) real participants** who match your target user profile, with verifiable contact details (Zalo, email, or phone, with the middle four digits masked). Participants **must be people outside this class** — students currently enrolled in HW03 are **not** eligible. Non-IT / non-tester participants are preferred for authentic usability feedback, though this is not strictly required.
- **Run a pilot session** with one person to catch an unclear scenario, broken flow, or timing problems; refine before the real sessions.

Phase 2 — Conduct the sessions (one per participant)

- **Set the stage.** Tell the participant you are testing the *product*, not them, and ask them to **think aloud** while they work.
- **Observe neutrally.** Do not give leading hints or explain the interface; step in only if the participant is completely stuck.
- **Capture evidence.** Record the screen (and audio, with consent) and take **structured notes** on friction points, errors, hesitations, and verbalised frustration.
- **Close the session.** Have the participant complete the **SUS / UEQ-S** scale, then ask your probe questions to dig into the difficulties you observed.

Phase 3 — Analyse & report

- **Score** the SUS / UEQ-S results across the seven participants.
- **Synthesise** your notes: group similar pain points together, and separate isolated bugs from systemic design issues.
- **Prioritise by severity** (for example, blockers that prevent task completion versus minor visual complaints).
- **Report** genuine bugs both in the Markdown report and on your GitHub Issues page, with a screenshot attached to each issue.
- The TA may randomly call **two (2)** participants to verify them. Impersonation results in **0 points for Task 2**.

Task 3 — Cross-Browser / Cross-Platform

- Perform Task 1 across **at least three (3) platforms**.

- Use a **BrowserStack** or **LambdaTest** trial; these are strongly preferred. If your trial has expired, you may substitute another cloud testing tool (for example, Sauce Labs or CrossBrowserTesting) or use real physical devices, provided your screenshots clearly show the browser / OS / device name alongside the SUT's localhost URL. You are responsible for obtaining your own trial access.
- Cover the web frontend on **Chrome, Firefox, and Safari** (or Android Chrome).
- You may also test the **SUT's mobile app via Expo Go on a real phone**. Expo Go counts as a valid platform and may replace one of the three required browser platforms (for example, in place of Safari); it is not bonus-only — including it satisfies one of the three required platforms.
- Each screenshot must overlay your username in the form **your student email**.

7. Agent Skill

- You are encouraged to build Agent Skills that apply the GUI-checklist and usability-evaluation activities, so that they can be reused on additional screens and flows in future testing tasks.
- Submit the skills together with demonstration videos (YouTube links) that show, end to end, how you used the skills on a complete screen or flow.

8. Allowed Tools and Bloom-AI Level

You may use the following tools, and you must declare them in your AI Audit Report:

- Any AI tool of your choice (e.g., ChatGPT, Claude, Gemini, Copilot, Cursor).
- A BrowserStack or LambdaTest trial.

The required Bloom-AI level for this homework is **G9.3 (Analyse)** and **G9.4 (Collaborate)**.

9. AI Audit Report (Mandatory Appendix)

Attach the AI Audit Report as an appendix. Use the content of the given AI Templates if needed.

- If you did not use AI, declare: "*I do not use any AI help in this exercise.*"
- If you did use AI, declare: "*I use AI tools for the following tasks,*" and include the following information for each interaction:
  - Name of the AI tool
  - Date and time
  - Your prompt
  - The AI output

To simplify this process, you are encouraged to create a skill or rule that extracts the information above automatically after an AI session.

10. AI Critique (200–300 words, Mandatory)

Write a paragraph of 200–300 words critiquing the AI. Address the following questions: Where did the AI get something wrong, biased, or incomplete? Why did it fail to catch the issue? What principle have you learned about collaborating with AI during this assignment?

Use the content of the given AI Templates if needed.

11. Anti-AI-Cheat Constraints

This homework relies on genuine human participants and real cross-platform runs. The following must not be AI-generated or fabricated, and the TAs verify them during grading:

- The list of 7 participants (name plus Zalo / phone, middle four digits masked). The TA may randomly call two of them.
- The cross-platform screenshots, which must show your student ID and full name in the form.

12. Git Commit Log

- Create a new Git commit for each step of the testing procedure (for example: checklist design, checklist execution, bug logging, each usability session, and the analysis).
- Provide the Git commit log in a text-based file format.

13. Oral Defense

A randomly selected **30% of students** may be invited to a 5–7-minute oral defense during the week following the deadline, to explain how they completed this homework.

14. Submission Regulations

- **Filename format:** `<StudentID>_HW03_AI_GUIUsability_<SelfAssessedGrade>.zip`
  - *SelfAssessedGrade:* a 3-digit number in the range [000, 100].
  - *Example:* `25127001_HW03_AI_GUIUsability_090.zip`
- **Required contents of the `.zip` :**

    - Main report (Markdown + PDF), including the GUI checklist report and the usability evaluation report.
    - Bug report, with screenshots of the bugs on the GitHub Issues page.
    - AI Critique and AI Audit Report (Markdown + PDF).
    - Git commit log (text file).
    - The Excel checklist (more than 40 items) and the test summary.
    - The usability-session evidence (task scenario, observation notes, SUS / UEQ-S responses, severity-ranked findings, and screen recordings where available) together with the table of 7 participants.
    - The cross-browser / cross-platform screenshots.
    - A `README.md` containing the self-assessment table (below) and a test summary report: number of screens / flows tested; number of checklist items designed, executed, passed, and failed; number of bugs; number of participants; and demo videos.
    - Any other supporting materials.
- Submit to Moodle. For the deadline, refer to the submission link.

15. Assessment Template

No.	Criteria	Grade	Self-Assessed Grade
1	Task 1 — GUI Checklist (design + execution + bug report)	30	
2	Task 2 — Usability Evaluation (task scenario + 7 sessions + analysis)	40	
3	Task 3 — Cross-Browser / Cross-Platform (≥ 3 platforms)	20	
4	Agent Skills	10	
	Total	**100**	

16. References

- ISTQB Foundation Level Syllabus (latest edition).
- Hardman, P. (2025). *A Post-AI Learning Taxonomy.*
- Fuster Rabella, M. (2025). *OECD Education Working Paper No. 338.*

- Anthropic (2025). *Building Reliable AI Test Agents* — engineering blog.
- DeepEval & Promptfoo documentation — LLM testing frameworks.

17. Other Regulations

- Late submission is **not permitted**.
- Missing any required document results in **0 points**.
- Copying between students — **including prompts** — results in a **grade of 0 for both parties**.