

Final Project Part II (30%)

Out: Monday, Oct. 17th, 2022

See project deadlines in Section 3, p.2

Included Handouts

1. Project overview
2. Deliverables
3. Deadlines
4. Project description
5. System example (screenshots)
6. Grading sheet

1. Project Overview

The term project, phase II, is a continuation of the interface your team started designing in the earlier phase I of the project. In this assignment you will be working on two more design evolutions. First, your team will create a **horizontal medium-fidelity prototype**, showing the features of the system. Second, you will create a **moderately robust system** that implements **vertical functionality**.

The project's main purpose is to give you hands-on experience applying some of the design concepts you have learnt in class, and to provide you with experience in developing a moderately robust interface. As part of phase II, you will implement your project prototype using C#, and perform a heuristic evaluation. The course weekly tutorials, as well as pre-recorded supplemental videos, will provide you with important insight on both C# and heuristic evaluation that would support your work on the project. **The project tutorials, videos and MVP group surveys are all mandatory and will continue throughout part II of your final project, till the end of the term.**

2. Deliverables

Deliverables are incremental. First, you will demonstrate your screen snapshots of a horizontal prototype (presentation will take place during your tutorials, graded by your TA/Project Advisors) and the re-design rationale. Second, you will deliver the final project, which includes the full working system, presented in person by your group to your TA/project advisor, a heuristic evaluation, a short design critique and a video figure of the final system. Your team mark will be composed based on the following scheme:

Horizontal prototype: 20%
Full working system: 80%

3. Project Deadlines

Nov. 14th Monday 10 am	Horizontal Prototype Redesign Deadline writeup (D2L submission) , including redesign rational (i.e. changes from the first prototype) + screen snapshots. Horizontal prototype Presentations Freeze Slides must be submitted to your TAs
Nov. 14th- 19th	Horizontal prototype Presentations (20% of mark) Horizontal Prototype/Screen snapshots demonstrated by the group to your TA during the tutorial.
Dec. 6th, 10am Final Project Submission Deadline	MAIN DEADLINE Dec. 6 th , at 10am via D2L: Complete Portfolio, including redesign rational, implementation freeze ¹ , latest screen snapshots, heuristic evaluation of final system, video figure and final discussion. Please submit all the project components electronically: all source code and required data files as well as a README file.
Dec. 6th-9th	Project demonstrations (80% of mark) scheduled by your TA/Project Advisor. Demos will be run strictly from the submitted media. All team group members must participate.

4. Project Description

Overview. In this project, you will gain further hands-on experience applying concepts learnt in class, as well as experience designing and developing medium and high fidelity prototypes. You will also learn how to program using a graphical user interface toolkit, and how to do a heuristic evaluation. Your design will ideally continue the interface you prototyped in Assignment 1 (or, possibly follow a new design).

¹. This means that your team must stop coding at this point - this is why you need to submit a full copy of your code.

A note on organization. You must hand in the *entire* portfolio on the Main Deadline, including your team Assignment 1 work, as this will show us how your work evolved and progressed.

What you do

1. Implement a horizontal prototype, plus re-design Rationale (20% of project mark)

- *Redesign* your interface. To do this, you should *review* your Assignment 1 prototypes and walkthrough results. You should also apply the design knowledge you are gaining in class to your design. You may want to develop a few more paper prototypes here and do further walkthroughs to check your ideas out. This part is up to you.
- **or**
*In case your assignment 1 design direction was unsuccessful, you can try to come up with a novel system design that stretches your creative talents. **Make sure your TA is aware of this change, and that they supports and approves it.***
- *Implement* your design as a medium fidelity horizontal prototype. Using C#, implement your primary screen(s). Most of this will involve widget selection and placement, although you may have to do some more sophisticated coding if your interface has esoteric components.
- *Deliverable: **Your redesign writeup via D2L.*** It should contain
 - a two page redesign rationale that describes your main reasons behind the changes made
 - illustrations of your screens, which you can generate by a screen snapshot tool of your choice. Note that this component is not included in the two-page writeup count.
- *Presentations, during your tutorial:* You will present and discuss these prototype snapshots. You **must** freeze your presentation and submit your slides electronically by the Horizontal Prototype Redesign Deadline.

2. Implement a Vertical prototype, and do a heuristic evaluation (80% of project mark)

- *Redesign* your interface. To do this, you should *evaluate* your interface, selecting from the evaluation techniques you now know e.g., walkthrough, heuristic evaluation, and from applying the interface design techniques described in the lectures and tutorials.

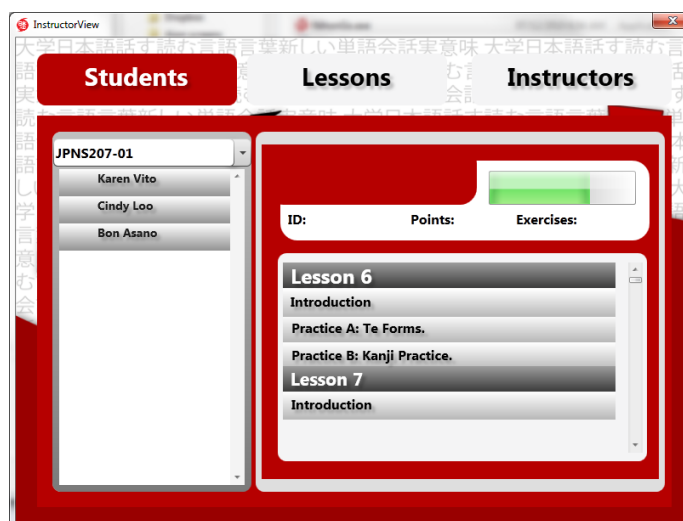
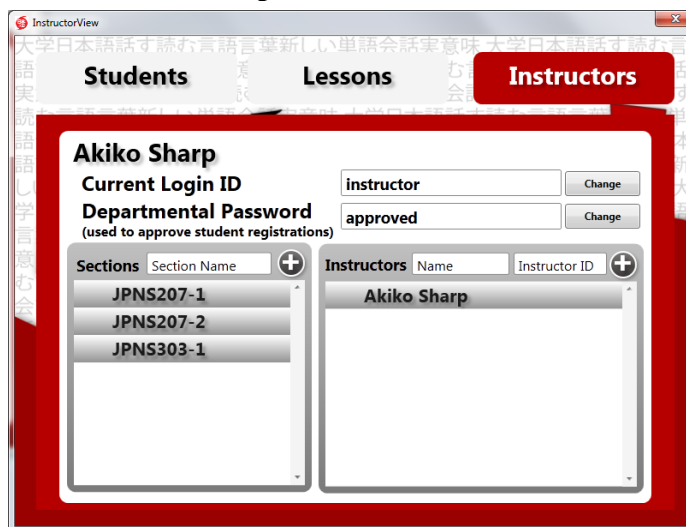
- **Implement a substantial part of the vertical functionality of your interface.** 'Substantial part' means that examples of the more interesting features (screens, error messages, handling of unexpected input, defaults, robustness, ...) should be functional and demonstrable. (You may program in 'stubs' for sub-tasks you are not implementing at this time, e.g., certain actions may return some kind of 'Under development' message).
- Perform a *heuristic evaluation* of your final interface.
- *Deliverables, in portfolio:* Your final portfolio should contain:
 - Illustrations of your final implementation, using new screen snapshots,
 - The results of the heuristic evaluation:
 - List the problems detected, categorized by heuristics. Include a severity rating of the problems noted
 - Summarize the main findings of your heuristic evaluation
 - A video figure of your final system (more information on this will be provided later)
 - Final design rationale and discussion (two - three pages) of the state of your design. Discuss the quality of your system design. What parts of the design work well and what still needs improvement? Do you really believe that the system would work well for your identified users and tasks? What are your design strengths? What are the weaknesses? Explain and justify.
- **You must electronically submit (via D2L):**
 - All source code and data files pertaining to the final version of your project. This must include a **README file** containing your full name and any instructions for using the system. What cases/functions were implemented? What data should be entered at which times? To ensure that we don't miss any of the best features of your system you should word your instructions as an exact walkthrough of what should be typed and what controls should be set to what values. Everything must run from the installation directory (**Hint:** use App. Path. This means that you should use relative rather than absolute paths).
- *Demonstration.* Your team will demonstrate your running system to the project advisors at the end of the term in person. A timetable will be posted by your TA/Project Advisor. All team members must participate.

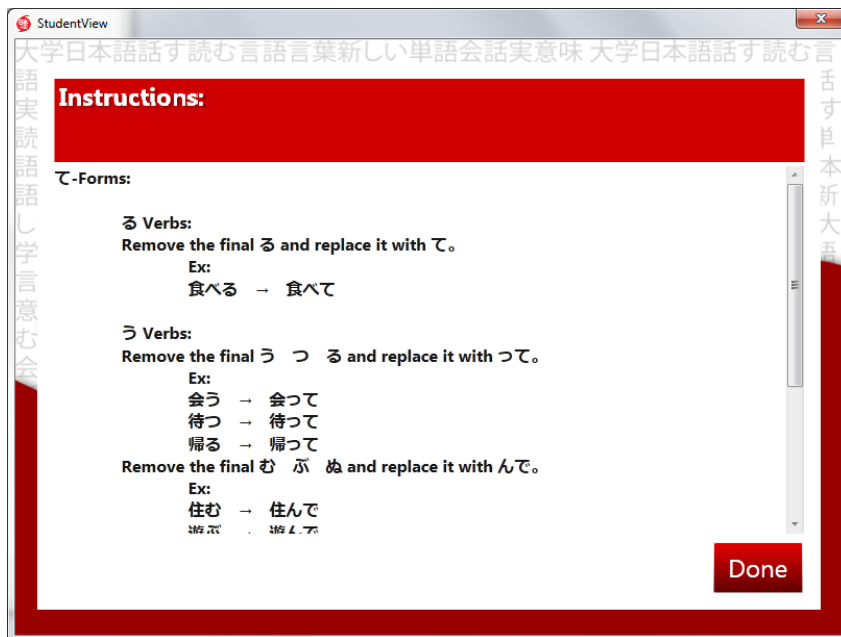
Grading. Grades are based on the quality, sophistication and creative elements of the evolving design and its implementation, on your prototype demonstration and the professional nature of the written and oral submissions. Remember that you are implementing both a horizontal and vertical high-fidelity prototype --- the balance between the two depends on your design. It should contain enough 'meat' to show what it would be like to interact with the real thing. **Grades are not based on the complexity of underlying application code that is not related with the user interface.**

You are emphatically cautioned against biting off more than you can chew! A modest carefully implemented project often scores much higher than an ambitious project that is not well done. Note, this project requires considerable effort. If you want a good mark in this assignment, ***Start Immediately!***

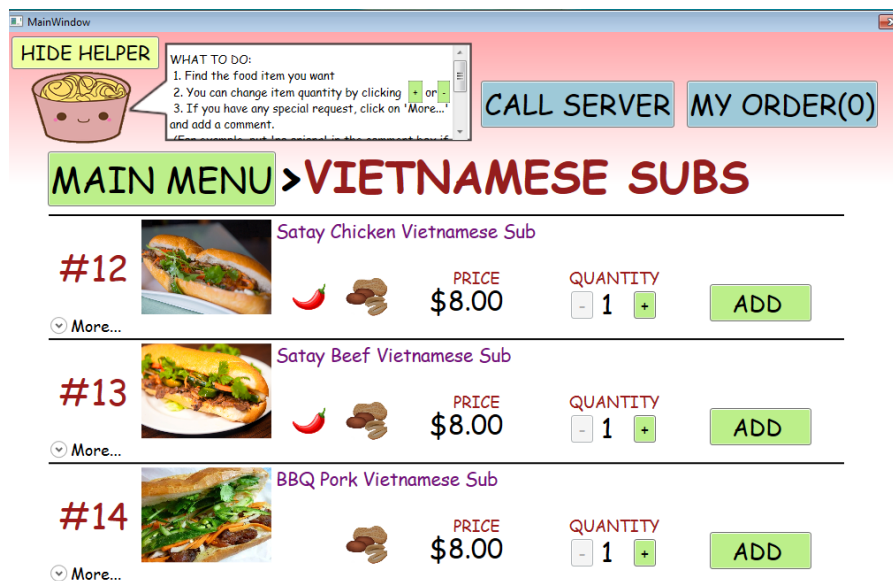
5. Final project example (screenshots, please see also final project videos, shared in the recorded lectures)

NohonGo! Japanese lessons example:





Vietnamese restaurant ordering interface example:



6. Project Grading Guidelines

Students

Names _____

Emails _____

Note: These are just guidelines detailing "convenience" checkpoints. Getting many satisfactory checks does not necessarily indicate a good project (or vice versa).

Completeness of Project	Missing	Incomplete portions	Satisfactory
Original submission for the first assignment	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Part I, A2: Redesign writeup: Screens & redesign rationale	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Part II, A2: Heuristic evaluation	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Part II, A2: Redesign rationale & final design critique	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Working end of term demo	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
All team members completed all weekly 'Most Valuable Team Member' surveys	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
To portfolio final due date			
Horizontal prototype			
Quality and completeness of the redesign rationale	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Screen snaps/final design rationale	Poor	Ok	Great
Fixes major flaws in horizontal prototype	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Good rationale behind design	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Design critique indicate major problems?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Design critique indicates how these problems may be solved.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Your Heuristic Evaluation	Poor	Okay	Great
Problems categorized by heuristics	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Major problems detected	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Severity ratings are reasonable	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Main points of the evaluation are summarized	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Sophistication and quality	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Completed system	Poor	Okay	Great
Depth of interface shown	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Breadth of interface shown	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Non-interface aspects	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Graphical design	Poor	Okay	Great
Visual appearance	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Sensibility of layouts	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Evaluating the <i>project</i>	Poor	Okay	Great
Simple and natural dialog	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Speaks the users language	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Minimizes memory load	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Consistent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Provides feedback	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Clearly marked exits	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Shortcuts for experts	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
User error handling	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Provides relevant help	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Technical aspects	Major problems	Adequate	
Robust/bulletproof	<input type="checkbox"/>	<input type="checkbox"/>	
Final video figure and project demonstration	Unacceptable	Adequate	
	<input type="checkbox"/>	<input type="checkbox"/>	
Gave a good feel of system?	<input type="checkbox"/>	<input type="checkbox"/>	
	<input type="checkbox"/>	<input type="checkbox"/>	
Overall impression	Poor	Okay	Great
Final design	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Design evolution	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Portfolio	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Heuristic evaluation	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Implementation	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Final demo and video figure	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>