

FREELANCE SERVICES AGREEMENT (Assignment 2 (continue from Assignment 1))

This Freelance Services Agreement ("Agreement") is made and entered into on Start of Semester, by and between:

Client: CSE3CWA/CSE5006 Staff

Address: La Trobe University

Email: See LMS

AND

Freelancer: You (individual assignment)

Address: Your Address

Email: See LMS

1. Scope of Work

The Freelancer agrees to perform the following services:

This is the first part of a larger project

Problem description:

LTU wants to build a web application that creates HTML + JS to deploy on MOODLE LMS. Current situation LMS has the capacity to used HTML5 such as carousel, tabs, accordion, modal/popup, dropdown, tooltip, progress bar, range slide, date picker, alerts, lightbook, canvas, CSS Animation (rotation), Mermaid code, etc.

Part 1:

You are required to build a NextJs Application bootstrap with create-next-app.

Build your

Pages: home page, about page, etc.

Student Number will be in the top left hand corner of each page.

Header – including a Menu with either Kabab or Hamburger Menu

Themes: Dark Mode/Light Mode (or web browser themes)

Footer – including copyright, Student Name, Student Number and Date

Accessibility – must be compliance with Accessibility Standards

Cookies – Remember which menu tab you were on.

The about page:

Your name, Student number, video of how to use this website.

The homepage:

Build a website that outputs code

Output html5 code with JS and inline CSS (no CSS Classes).

Title	Student No.
Tabs Pre-lab Questions Escape Room Coding Races	About
Tabs	<input type="checkbox"/> Dark Mode
<div style="display: flex; justify-content: space-between;"> <div style="flex: 1;"> <p>Tabs Headers: [+]</p> <ul style="list-style-type: none"> Step 1 Step 2 Step 3 </div> <div style="flex: 1;"> <p>Tabs Content</p> <div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Step 2: <ol style="list-style-type: none"> 1. Install VSCode 2. Install Chrome 3. Install Node 4. etc </div> <div style="border: 1px solid black; padding: 5px; background-color: #f0f0f0;"> <pre>with open('style.css', 'w') as f: f.write('body { font-family: sans-serif; }') f.write('h1 { color: red; }') f.write('h2 { color: blue; }') f.write('p { color: green; }') f.write('a { color: purple; }') if __name__ == '__main__': print('Hello World!')</pre> <p>Output</p> <p>Code generated for the selected tab:</p> <p>Output</p> </div> </div> </div>	

Copyright Name, Student No, Date

Figure 1: Screenshot of the Homepage.

Assignment 1 You are to generate the Front-end (with pages – Escape Room, Coding Races, Court Room) can revert to an empty page/not yet finish.

The output generates the HTML5 + JS code. This code should be able to be copied. If you pasted this code into a blank file – it should open in a Web Browser; Hello.html.

This is an example of the output – after the user have configured the tabs.



Assignment 1 – create the pages and complete the Tabs Page that generate a tabs html output.

Part 2 *NEW*

Continue your assignment 1 (if you haven't completed assignment 1 – don't worry about themes, menu, header, footer, tabs, use a blank page with your student number of the right).

You chose to implement one of the following:

1. Escape Room
 - Students must code their way out of a room.
 - Manual set Timer

- Create an image for the background (escape room)
- Allow several stages
 - e.g. Format Code correctly
 - Click image that allows you to debug code
 - Write a code to generate all the numbers between 0 to 1000.
 - Write a code to port data from one format to another

2. Court Room

- Manual set Timer
 - Create an image for the background (a court room and a work desk)
 - Allow several stages
 - e.g. User is to debug code.
 - They get messages every 20-30 seconds
 - from their boss (e.g. are you done with sprint 1?)
 - from their family (e.g. can you pick up the kids after work?)
 - from agile (e.g. "fix change Title colour to Red")
 - Message such as "fix alt in img1" if they ignore this, it will come back in 2 min with "urgent fix alt in img1" if they haven't done it in 2 min more – a court room appears and you are fine for breaking the Disability Act.
 - Message such as "fix input validation" if they ignore this, it will come back in 2 min with "urgent fix input validation" if they haven't done it in 2 min – a court room appears and you are fine for breaking "Laws of Tort" (after you got hacked – you knew and didn't fix the problem).
 - Other messages:
 - +Fix User login -> don't do it you go to court and declared bankruptcy (no one can use your app and you don't get paid).
- Fix Secure Database -> you got hacked -> go to court breaking laws of tort.

After you created one of the above:

- + Dockerize the App
- + Add APIs for CRUD function to a database
- + Add a save button that saves your output to a Prisma or Sequelize Database
- + Add 2x Tests to autogenerated three examples of your choice above and check the output.

+ Request feedback from two from each of the following; family, friends and industry and ask them to complete an ethical survey

<https://redcap.latrobe.edu.au/redcap/surveys/?s=PPEKFTMPXF4KKEFY>

- + Instrument your app (run the tests again and observe your instrumentation)
- + Deploy it in the Cloud
- + Add a lambda function that creates dynamics pages of your html output.

The following Schedule is a break down of where you should be up to per week.

Weeks	Feature	Expected Progress (should be evidence on GitHub)
Week 1	Lab: Project Setup (must use NextJs – a v22+)	Software Setup

	Lecture: HTML, CSS and JS.	
Week 2	Lab: Cookies Lecture: Cloud Setup	Assignment 1: Create an About Page
Week 3	Lab: Carousel, Hamburger Menu Lecture: Usability: Themes; local storage, React (Hide/Show blocks)	Assignment 1: Create a Header, Footer, Menu, Cookies (for navigation) with Breadcrumbs.
Week 4	Lecture: Advanced React Lab: List + Dynamic Pages	Assignment 1: Create Interactive links
Week 5	Lecture: API Lab: API	Assignment 2: Create Routes
Week 6	Lecture: Docker Lab: Docker + API	Assignment 2: Dockerize your project
Week 7	Lecture: Database Lab: Prisma + Sequelize	Assignment 2: Add a Database
Week 8	Lecture: Testing Lab: Load Testing, Performance Testing, Accessibility Testing, etc	Assignment 2: Add Automatic Testing
Week 9	Lecture: Instrumentation and Observability Lab: Instrument Next.Js	Assignment 2: Add Instrumentation and Observability
Week 10	Lecture: Cloud Native Components Lab: S3, Functions + Web App	Assignment 3: Convert API and Storage to Cloud
Week 11	Lecture: Authentication Lab: Authentication	Assignment 3: Add Authentication
Week 12	Revision	

Assignment 1 Handover:

Video Recording: You will record your screen with your face and audio visible doing a walk-through of your application and code. (3-8 minutes max).

Code: Zip copy of your code – delete Node_Modules before you upload.

Screenshots: Provide Screenshots of your GitHub Commits.

Assignment 2 Handover:

Video Recording: You will record your screen with your face and audio visible doing a walk-through of your application and code include your playwright tests. (3-8 minutes max).

Code: Zip copy of your code – delete Node_Modules before you upload.

Screenshots: Provide Screenshots of your GitHub Commits, Screenshots of your Lighthouse, JMeter test results and feedback from users.

2. Remuneration

The Client agrees to pay the Freelancer:

Assignment 1

- Amount: 25%
- Payment Schedule: In terms of Marks See LMS for Marking Criteria
- Method of Payment: Marks for Assignment

Assignment 2

- Amount: 25%
- Payment Schedule: In terms of Marks See LMS for Marking Criteria
- Method of Payment: Marks for Assignment

Remuneration will be pro-rata. Marks will be per each section on the marking guide.

Fines (marks deduction) as part of this agreement you will be fine for breaking the law. (In other words, if you developed an application that breaks the Disability Act, Crimes Act, etc you will lose marks).

Fines (marks deduction) if you failed to use a Next-Js App bootstrap with create-next-app (typescript only) using App Router and other current technologies.

CoPilot, ChatGPT and others Generative AI can be used, but you need to reference it in their code. Please note Generative AI does not work well for the latest technology, do and refer to the laboratory.

Help/Assistant: You will be first asked if you have completed the appropriated lab.

3. Timeline

The Freelancer agrees to deliver the work according to the following timeline:

Semester Mode: Start: Week 1 End: Week 4

Semester Mode: Start: Week 4 End: Week 11

See LMS for exact details.

4. Revisions

The Client is entitled to 2 rounds of revisions. Additional revisions will be billed at \$1000/revision.

5. Independent Contractor Relationship

Freelancer is an independent contractor, not an employee of the Client. Nothing in this Agreement creates a partnership, joint venture, or employer-employee relationship.

6. Confidentiality

Both parties agree to maintain confidentiality regarding sensitive or proprietary information exchanged during the project.

7. Intellectual Property

Upon full payment, the Client will own the rights to the final deliverables. The Freelancer retains the right to showcase work in their portfolio unless otherwise agreed in writing.

8. Termination

Either party may terminate this agreement with 3 days' notice. The Client agrees to pay for all work completed up to the termination date.

9. Governing Law

This Agreement shall be governed by the laws of Victoria, Australia

10. Entire Agreement

This Agreement constitutes the entire understanding between the parties and supersedes all prior discussions or agreements.

IN WITNESS WHEREOF, the parties hereto have executed this Freelance Services Agreement as of the date first written above.

Client Signature: _____

Name: Dr. Tony de Souza-Daw

Date: _____

Freelancer Signature: _____

Name: [Freelancer Name]

Date: _____