COS30043 – Interface Design and Development

Learning Summary Report

HAN THANH CHUNG (StudentID: 104050740)

Self-Assessment Details

The following checklists provide an overview of my self-assessment for this unit.

	Pass (D)	Credit (C)	Distinction (D)	High Distinction (HD)
Self-Assessment (please tick)				☑

Self-assessment Statement

	Included (please tick)
Learning Summary Report	☑
Use of Bootstrap that demonstrates coverage of core concepts	☑
Use of VueJS that demonstrates coverage of core concepts	☑

Minimum Pass Checklist

	Included (please tick)
Progress on Credit Tasks	☑
All Pass Tasks signed off	☑

Minimum Credit Checklist, in addition to the Pass Checklist

	Included (please tick)
Credit and Pass Tasks done, and Progress on Distinction Tasks.	☑
Custom program meets Distinction criteria	☑
Design report with screenshots for a custom program	☑

Minimum Distinction Checklist, in addition to the Credit Checklist

	Included (please tick)
Research report and associated pieces	☑
Custom project meets HD requirements	☑

Minimum High Distinction Checklist, in addition to the Distinction Checklist

Declaration

I declare that this portfolio is my individual work. I have not copied from any other student's work or from any other source except where due acknowledgment is made explicitly in the text, nor has any part of this submission been written for me by another person.

Signature:

Chung Thanh Han

Introduction

This report summarises what I have learned in **COS30043 – Interface Design and Development**. It includes a self-assessment based on the criteria outlined in the unit guide, a justification of the work I have included, an overview of how the unit's intended learning outcomes have been addressed, and a reflection on my learning throughout the semester.

Overview of Pieces Included

This section outlines the pieces that I have included in my portfolio

- **Pass Task 1.1: Data Preparation**

  Submitted a zip file containing a web project with one HTML, one CSS, and one JavaScript file, showcasing either past work or a newly created project.

- **Pass Task 1.2: Hello World**

  Created a simple HTML5 "Hello World" page using VS Code or Brackets, and submitted screenshots of both the code and the rendered page.

- **Pass Task 1.3: Form Accessibility**

  Corrected a web form for accessibility using WCAG 2.0, validated the form using an accessibility checker, and submitted a screenshot showing no errors.

- **Pass Task 1.4: Table Accessibility**

  Fix accessibility issues in the given HTML table using semantic elements and attributes, validate the code, and test with an accessibility checker.

- **Pass Task 2.1: Getting Bootstrap Up and Running**

  Created a Bootstrap-integrated "Hello World" page, demonstrating the correct use of the framework and answering related understanding questions

- **Pass Task 2.2: My Calculator**

  Designed a non-functional calculator layout using nested ** ** elements and Bootstrap's grid system, with proper indentation and structure.

- **Pass Task 2.3: My Bootstrap Template Library**

  Created a one-page Bootstrap corporate template including at least 10 content boxes across 5 sections.

- **Pass Task 3.1: String Test Web App with VueJS**

  Built a VueJS app that displayed a custom message if the entered name matched mine, using **v-model** and **v-show**.

- **Pass Task 3.2: Look-Up Web App**

  Developed a VueJS app that displayed a filtered unit list in a Bootstrap table using **v-for** and directives.

- **Credit Task 3.3: Compute Web App**

  Created a BMI calculator in VueJS that accepts weight and height inputs, calculates BMI, and displays a category message.

- **Credit Task 3.4: Registration Form Web App**

  Create a VueJS app (register1.html) with conditional and loop directives for phone registration, including password validation, OS-based model selection, and a summary display.

- **Pass Task 4.1: Event Handling**

  Build a VueJS number guessing game with check, give up, and restart buttons, showing hints and results based on user input.

- **Pass Task 5.1: Creating Components**

  Created an **app-mypost** VueJS component for posting and deleting status updates.

- **Pass Task 5.2: Parent-Child Communication**

  Create a VueJS component (mymenu) that receives a string array via props and displays it as a **<ul>** list.

- **Credit Task 5.3: Creating a Router**

  Implemented routing in a VueJS app to display unit details upon user interaction.

- **Credit Task 6.1: My Registration Form**

  Designed a Bootstrap-styled registration form in VueJS, featuring validation for required fields and submission handling.

- **Pass Task 7.1: Requesting External Data**

  Create a VueJS app that uses jQuery's getJSON() to fetch data from https://jsonplaceholder.typicode.com/posts and display only the id and title of each item.

- **Pass Task 7.2: Retrieving Data from a Text File**

  Created a VueJS app that uses `fetch()` to load data from a local JSON file and display it in a styled table.

- **Pass Task 8.1: Student Marks**

  Created a VueJS and Bootstrap app that displays a table of 26 student marks with accessibility features and paginates using **vuejs-paginate**.

- **Credit Task 8.2: My Table**

  Developed a VueJS app that loads and paginates unit data from a JSON file, presented in an accessible table using Bootstrap.

- **Pass Task 9.1: Single Page Application**

  Create a VueJS single-page application using Vue Router and Bootstrap with three components: Name Test, Post Management, and Student Marks, each rendered via navigation, and include pagination where required.

- **Credit Task 9.2: Single Page Application**

  Developed an SPA with login and dashboard views, allowing users to view, add, update, and delete unit records with pagination.

- **Pass Task 10.1: Creating a Single-Page Application Using Vue CLI**

  Created a Vue CLI SPA with three components (Home, Tasks, Units), integrated with routing, and displayed JSON data in the Units section.

- **Pass Task 11.1: Draft Learning Summary Report**

  Submitted a draft report outlining targeted grades, reflections on the work, and how outcomes were achieved.

- **Distinction and High Distinction Task 6.2 Custom Web Application**

  Developed a responsive Vue + Bootstrap app featuring login, search, and CRUD interactivity. The HD version included external data and pagination.

- **High Distinction Task 10.2: High Distinction Project**

  Completed an advanced project such as a tutorial or media product, demonstrating a deeper understanding of the unit beyond the custom app.

Coverage of the Intended Learning Outcomes

This section outlines how the pieces I have included demonstrate the depth of my understanding of each of the unit's intended learning outcomes.

ILO 1: Apply Design

I demonstrated fundamental design skills by creating responsive layouts using Bootstrap's grid system, applying consistent styling, and maintaining clear structure and indentation throughout my work. These skills are evident in:

+ **Pass Task 2.2: My Calculator**
+ **Pass Task 2.3: My Bootstrap Template Library**
+ **Distinction & High Distinction Task 6.2: Custom Web Application**

ILO 2: Use Frameworks

My portfolio shows effective use of contemporary frameworks such as VueJS and Bootstrap. I developed dynamic user interfaces using Vue components, applied Vue Router for SPA development, and consistently integrated Bootstrap for layout and design. Key examples include:

- **Pass Task 5.1: *Creating Components***
- **Pass Task 9.1: *Single Page Application***
- **Pass Task 10.1: *Creating a Single Page Application Using Vue CLI***

ILO 3: Develop User Interfaces

I designed and built user interfaces that work well across devices by applying mobile-first and accessibility principles. This is demonstrated in:

- **Pass Task 7.1: Requesting External Data**
- **Pass Task 7.2: Retrieving Data from a Text File**
- **Pass Task 8.1: Student Marks**
- **Credit Task 8.2: My Table**
- **Credit Task 6.1: My Registration Form**

These tasks involved fetching external data, managing state, paginating tables, and implementing form validation with VueJS.

ILO 4: Evaluate User Interfaces

I improved usability and accessibility across my projects by following WCAG standards and applying validation tools. My attention to accessibility is demonstrated in:

- **Pass Task 1.3: Form Accessibility**
- **Pass Task 1.4: Table Accessibility**

Throughout the unit, I consistently applied best practices to enhance user experience and interface quality.

Reflection

The most important things I learnt:

I gained a strong foundation in building modern web applications using Vue.js, covering key concepts like data binding, component creation, routing, and form handling. Understanding the separation between the View and the ViewModel allowed me to write cleaner, more maintainable code. Using Vue CLI and the Composition API gave me the confidence to build single-page applications from scratch. I also learned how to connect to external APIs and implement pagination to display data dynamically. Beyond the technical side, I developed better time management and improved how I present my work by organizing and documenting everything clearly in the portfolio.

The things that helped me most were:

The step-by-step nature of the tasks helped me build skills gradually, without feeling overwhelmed. Using the browser's debugging tools gave me immediate feedback and helped me learn faster. Incorporating Bootstrap also saved time and ensured my app was responsive. Vue CLI streamlined the setup process and gave me a more efficient way to manage projects.

I found the following topics particularly challenging:

I found the following topics particularly challenging. I had a tough time with components, routing, APIs, and pagination, especially in Pass Task 5.1, Credit Task 5.3, and Pass Task 10.1. Passing data between components and setting up Vue Router was confusing at first. Using vuejs-paginate and understanding async data added to the challenge. The Vue CLI folder structure also felt overwhelming. I also struggled with the Custom Web App idea and ran into errors after mixing ES6 modules with CDN scripts. Despite the setbacks, I kept reviewing lectures, researching, and testing until things started to make sense. It was frustrating, but I learned a lot along the way.

I found the following topics particularly interesting:

Single-page applications stood out to me as one of the most interesting topics. It was eye-opening to see how modern web apps can offer seamless experiences without full page reloads. Learning how routing works behind the scenes gave me a new appreciation for how smoothly user interactions are achieved. I also enjoyed working with Vue and Vue CLI, as they helped me understand how professional development environments are structured. The component-based approach made it easier to manage and scale applications, and once I got used to it, it felt very intuitive.

I feel I learnt these topics, concepts, and/or tools really well:

I became confident in handling forms, validation, and dynamic data. Setting up user input, displaying error messages, and ensuring proper data handling became more natural with practice. I also improved in pagination and data binding, especially after seeing how everything connected in my weekly projects. What once felt overwhelming eventually became manageable, and I could see real progress in how I approached and solved problems.

I still need to work on the following areas:

I still want to build more confidence with APIs and routing, especially nested routes and working with different types of data. There were moments when I was not sure how things were supposed to connect, and I often had to rely on trial and error. I also want to get more comfortable with the overall structure of Vue projects so that starting a new one does not feel so overwhelming.

My progress in this unit was …:

My progress in this unit was gradual but rewarding. At first, I found many of the concepts overwhelming, especially with how much was new to me. But as I kept working through the weekly tasks and pushing myself to understand how everything connected, I started to build more confidence. Each piece of the puzzle made the next one easier to grasp. By the end, I felt more capable of building a complete app and solving problems on my own. It was not always smooth, but I can see how far I've come.

This unit will help me in the future:

This unit will definitely help me in the future, both in my studies and in my career. Learning how to build single-page applications and work with tools like Vue, Vue CLI, and APIs has given me a strong foundation in front-end development. These are real-world, in-demand skills that directly apply to modern web projects. Along the way, I also learned how to break down complex problems, manage my time more effectively, and keep going even when things didn't work right away. These lessons will stay with me and be valuable not just in future units, but in any development work I do moving forward.

If I did this unit again, I would do the following things differently:

If I did this unit again, I would start planning my custom web app idea much earlier instead of leaving it until later in the unit. I also would have spent more time practicing key concepts like routing, props, and working with APIs early on, rather than waiting until I needed them for a task. Looking back, I can see that the times I made the most progress were when I stayed consistent with my learning and asked questions when I was stuck. In the future, I will try to be more proactive and organised, break down tasks into smaller steps, and make better use of the resources available to me from the start.

Other…:

This unit has boosted my confidence in creating interactive, professional-grade web applications. It also helped me learn to manage my time better and plan project milestones based on weekly topics.