Project Overview

Project Name: iConcepts Orthodontics - Interactive Textbook (code: IC)

Team Members

- Sangmoon Han: Scrum Master
- Po-Yun Hsiao: Product Owner
- Soyeon Park: Deployment lead
- Man-Hua Chu: Quality Assurance lead
- Yu-Tse Ling: Architecture lead

Project Description

This project aims to develop an interactive textbook regarding orthodontics to improve the studying and teaching quality of students, lecturers, as well as dentists. The first version(1.0) textbook was created last semester, so an advanced version of the textbook will be our focus this semester.

Firstly, the site requires a more stable platform for presenting the context without lags so that users can read efficiently. Secondly, reviewing the current version of the textbook and finding navigational errors that can be improved for usability. After fixing the errors, another 3 chapters of orthodontics knowledge in the textbook will be added to the application to enhance the comprehensiveness. By completing the 3 aspects, take this textbook to a higher level of efficiency, effectiveness, and excellence for its users.

Project Goal

1. Innovation in the Orthontics Learning Method

- Improve students' learning experience by developing interactive online textbooks rather than traditional static images and textbook-oriented learning

2. Building a Student-Driven Self-Paced Learning Environment

- Develop a learning system that allows students to learn concepts on their own and reflect on self-assessment and feedback

3. Establishing safe and sustainable environments

- Ensure successful learning of students and potential expansion into other disciplines

Technologies Used

Area	Tech Stack	Description
Language	TypeScript, JavaScript	TypeScript is used for type safety and scalability, while JavaScript ensures compatibility and flexibility in front-end development.
Framework	React, Next.js	React provides a component-based UI structure, and Next.js enhances it with server-side rendering, routing, and performance optimisation.
Styling	Tailwind CSS, PostCSS	Tailwind offers a utility-first CSS approach for rapid UI development, while PostCSS enables advanced CSS transformations and plugins.
Authentication	Clerk	The application integrates Clerk for user authentication and management, enhancing security and user experience.
Deployment	Vercel	The source code is hosted on Vercel, providing a fast and scalable web application without the need for a backend.

- As we continue to build on the codebase left by the previous team, we plan to develop the webpage based on the framework previously used by.
- **JavaScript** was chosen as the core language for this project due to its versatility in both frontend and backend development.
  In addition, we adopted **TypeScript** to improve code quality and prevent potential errors during the development process.
- For the framework, we selected **React** and **Next.js**.
- **React** enables us to build the application's user interface using a component-based architecture, while **Next.js** is responsible for the overall structure of the web application, including routing, server-side rendering, and static site generation.
- To implement a responsive user interface, we used **Tailwind CSS**, which provides utility-first styling optimised for responsiveness. Along with it, **PostCSS** was integrated to enhance browser compatibility and performance optimisation, allowing us to build a fully responsive web experience across various screen sizes.
- For authentication, we utilised **Clerk**, which allows us to implement core authentication features such as sign-up, and login with ease. **Clerk** simplifies the process of adding secure and reliable user authentication without building it from scratch.
- Finally, the frontend deployment is handled through **Vercel**, the official deployment platform for Next.js. Vercel provides a simple and efficient CI/CD workflow that automatically deploys changes, enabling fast and reliable delivery of updates to end users.

Timeline

- **Phase 1:** Research & Planning (10/03/2025 - 03/04/2025)
- **Phase 2:** Development (04/04/2025 - 30/05/2025)
- **Phase 3:** Testing & Deployment (31/05/2025 - 13/06/2025)