**Design, Analysis and Implementation Decisions for Make Waves Recording Studios**

**Initial Design Thoughts**

For Make Waves, the idea was to build a slick, responsive web app for a modern recording studio — something that looked creative but still professional. I wanted the design to feel bold and stylish, with a strong focus on usability. The layout is mobile-first and the whole experience is designed to feel smooth and polished on any device. The design evolved with ongoing experimentation.

**Home Page**

The landing page makes an impact with a clean background video, a strong logo, and clear calls to action. It’s built to adapt smoothly on mobiles, tablets, and desktops — so it looks great whether you’re browsing on your phone or a full-sized screen.

**Services Page**

This page breaks down everything the studio offers. Each section is well-spaced and easy to skim, with a clear visual structure that helps guide users through the content. Everything’s optimised to look tidy and readable, especially on smaller screens using cards.

**About Page**

This tells the story behind Make Waves. I’ve layered text and images so it feels engaging without being cluttered. It’s styled to keep the focus on the message — who we are and what we’re about whilst implementing a clean slider gallery.

**Login & Registration**

The login and sign-up forms use Flask-WTF for security and validation with a clean design implementing bootstrap styles.

**Booking Page**

Once you’re logged in, you can book a studio session using a clean form built with Flask-WTF. Everything’s protected with CSRF and there’s built-in validation to keep things secure.

**User Dashboard**

Once logged in, the user has a view of all of their details, upcoming bookings, options to edit/delete their bookings and an edit profile option.

**Edit Profile**

From here, users can update their personal details for their profile. I have kept this page simple for ease of access.

**Admin Dashboard**

Admins get access to a private dashboard where they can view and manage all user’s bookings. It’s basic but functional — enough to support real use cases without being over-complicated, given the size of the web application.

**Custom Error Pages (403, 404, 500)**

Instead of boring default error pages, I’ve added fully styled versions that match the rest of the app. Navigation is hidden to keep things clear, and users get simple routes back to the home page or login.

**Main design features**

**My Development Process**

**Challenges faced during development**