

INTRODUCTION

This website is built for students and programmers who want to strengthen their coding skills through clear, high-quality tutorials and in-depth articles.

- Learn C/C++, Python, and other popular languages
- Covers everything from fundamentals to advanced techniques
- Beginner-friendly explanations with practical examples
- All content organized in one easy-to-navigate platform

Inspired by the challenge of learning programming as a student.

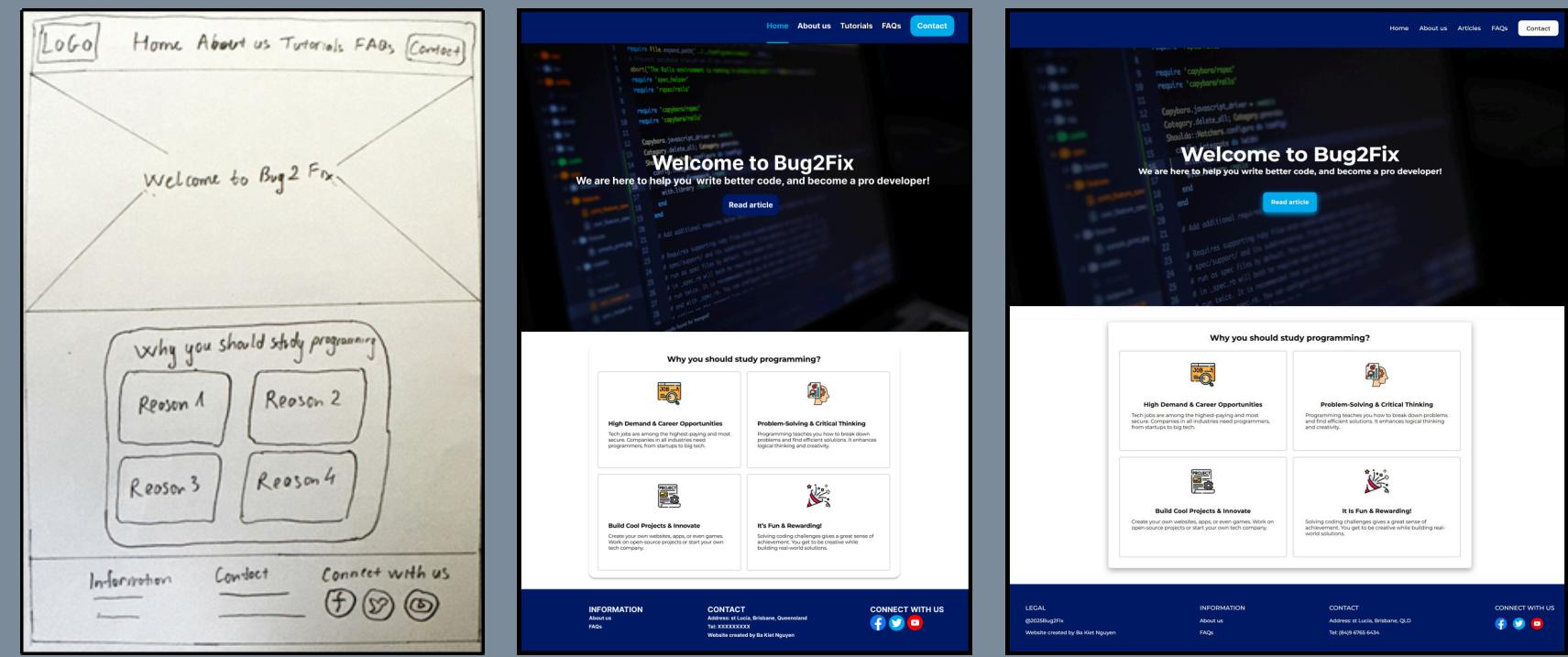
Targer audiences:

- University students learning programming for the first time
- Self-taught learners looking for structured, beginner-friendly tutorials
- Beginner to intermediate programmers who want clear, concise explanations

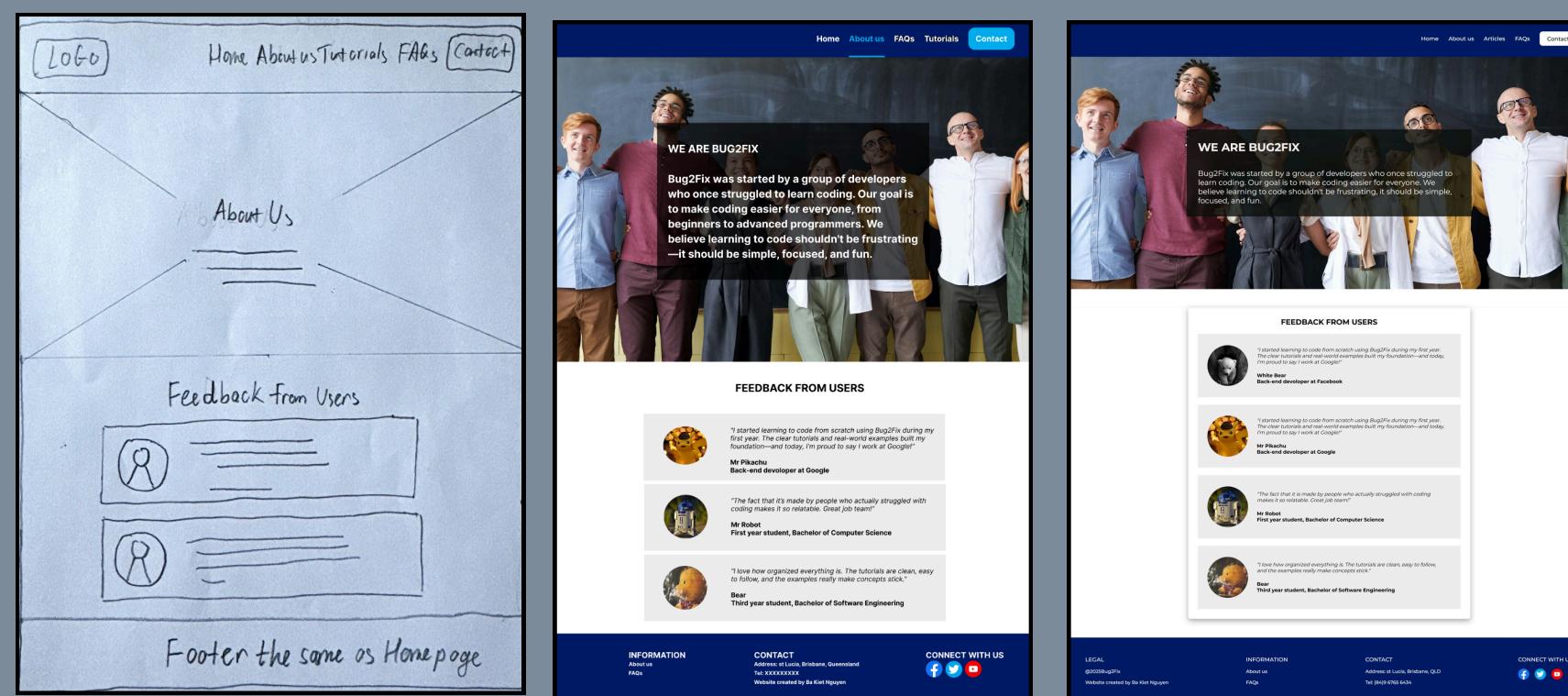
The website has some functionalities such as:

- A search bar that filters articles instantly as users type
- A form validation system that provides error feedback for contact submissions
- A back-to-top button that appears on scroll for easy navigation
- A FAQ toggle system that allows users to expand and collapse questions smoothly

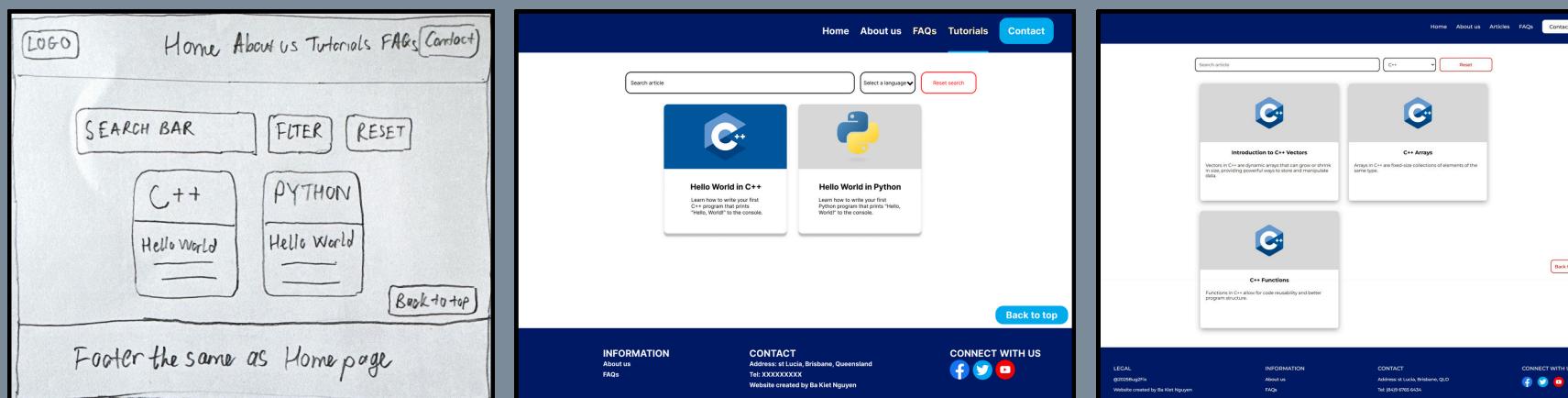
Home Page from low fidelity to final



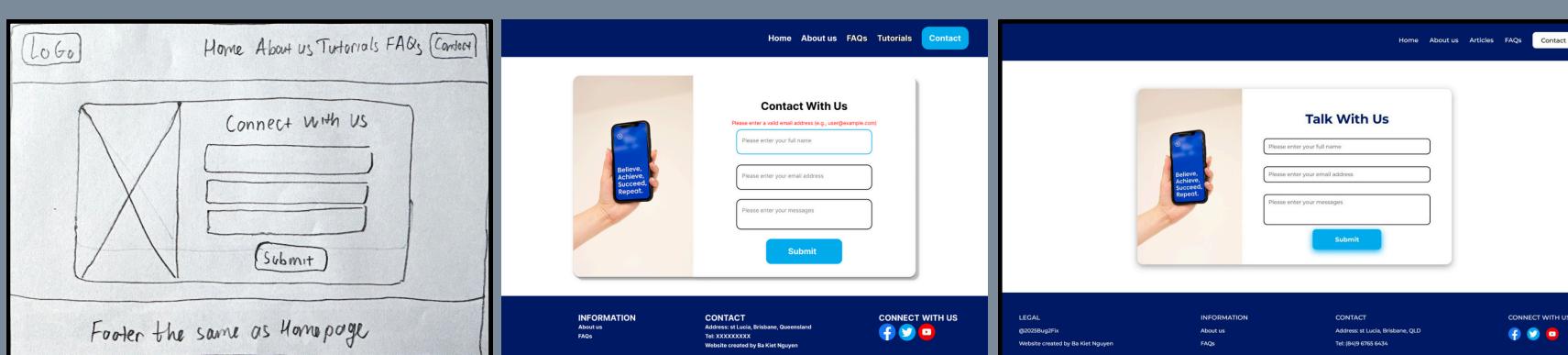
About Us Page from low fidelity to final



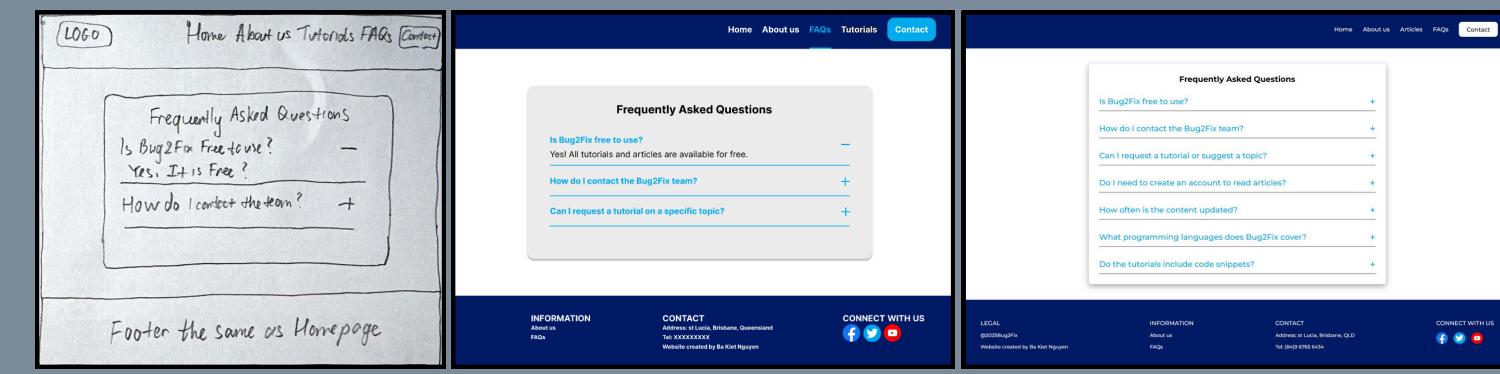
Menu Page from low fidelity to final



Contact Page from low fidelity to final



FAQs Page from low fidelity to final



Lessons learned

Throughout the process of creating this website, I learned how to apply Nielsen's usability heuristics to design a more effective and user-friendly experience. By focusing on principles like consistency and standards, visibility of system status, and recognition rather than recall, I was able to create a platform that feels intuitive - especially for beginners. This approach deepened my understanding of how users think and feel when interacting with a website, and how thoughtful design can reduce confusion. I also significantly improved my technical skills in HTML, CSS, and JavaScript. Also, I learned how to build a clean, responsive interface.

Challenges

While building this website using HTML, CSS, and JavaScript, I faced several technical challenges. Styling elements consistently across different pages was challenging - sometimes the layout didn't appear as I expected, so I spent a lot of time researching solutions and making adjustments. In JavaScript, managing DOM manipulation and user interactions - like handling form validation or dynamically displaying content, proved to be more complex than expected.

Future works

I plan to study front-end frameworks like Angular or React to make the website more dynamic and scalable. Learning these technologies will help improve component reusability and overall performance. I also aim to explore back-end development using Java Spring Boot, so I can build features like user authentication, content management, and data storage. These additions will allow the website to become more interactive and make the web more complete and functional.