## Personal Information

Name: Leigh White

Student Number: S3930650

Student Email: [s3930650@student.rmit.edu.au](mailto:s3930650@student.rmit.edu.au)

Hi, my name is Leigh! I grew up in a regional area of Victoria about three hours away from Melbourne, and have recently moved to South East Melbourne. I’m a bit of a gamer, but my biggest hobby is software and web development. Outside of IT, I enjoy music production and DJ’ing, and I am known to be a bit of a Magic the Gather enjoyer.

## Interest in IT

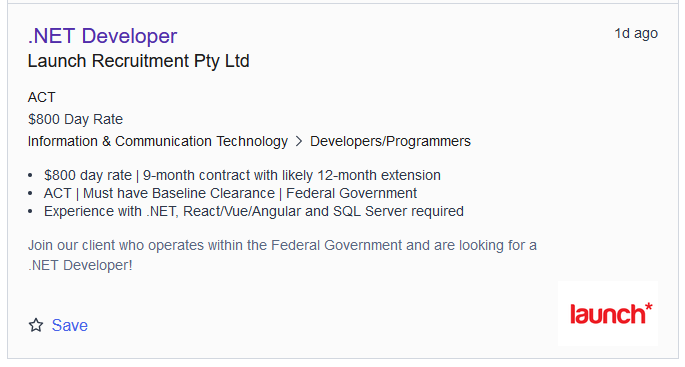
My interest in IT started when I was relatively young. I grew up on a farm and was not the type of kid who enjoyed being outside and I was always looking for reasons to be inside. The family computer was the perfect outlet, even without internet I would spend hours using software like Paint and Microsoft Word just learning how they worked.

I would say moving away from the farm and into the city and actually getting internet probably was the single most defining moment. Instead of wondering how things worked, I could easily find videos and guides in my pursuit to understand the technology I was interested in.

With all the access to information, I found that programming became my primary interest. Being able to solve problems, manipulate data, store data, and display data has always been fascinating, and I enjoy learning about the many different ways that it can be done. Currently C# is my primary programming language.

I would say I am pretty well versed with IT as it has been my primary interest for quite some time. I have experience setting up web servers, email servers, building and upgrading computers, as well as making websites and applications.

## Ideal Job



<https://www.seek.com.au/job/58386138>

As with many job listings in the IT industry, this listing was posted by a recruitment agency so the details are a little bit limited, but the role is to help support a software development project for an organisation in the federal government. The job is appealing to me because I have been interested in software development for a very long time and to turn my hobby into a career is a goal of mine.

The role requires experience with the .NET framework, as well as JavaScript frameworks and SQL databases. There are no required qualifications mentioned, however it does require general work-related skills such as being able to work autonomously, being able to work in teams, being able to communicate effectively with all involved personnel, and understanding of development life cycles and the Agile methodology.

Currently, all of my programming experience is within the scope of their requirements, however I don’t think they’re quite ready for job level as I still find myself getting stuck and looking for guidance. With C# being my primary programming language, the .NET framework is something I’m very familiar with, and in recent times I’ve been learning Vuejs which is a modern JavaScript framework that is also mentioned in the listing. I know some basic SQL from some applications I’ve made, but I’ve never needed to manage a big database so my skills are quite limited there. Hopefully after completing my studies, I can work in a graduate position to get some experience in a real working environment that I will then be able to transition into a role like this one.

I believe I currently have the skills to be able to work autonomously, and in a team, as that is a requirement for my current job. I don’t need to communicate with people who aren’t also in the industry in my current job, so being able to effectively convey messages to people who aren’t as technologically minded is something I would need to work on. Development life cycles and the Agile methodology are something I have come across, but not something I’ve ever looked into, so that is something I also will need to learn. I hope that through my studies I will be exposed to some of these methods and as well as developing the required communication skills.

## Personal Profile

## Project Idea

### Overview (100 words)

**Schoolar** is an application targeted at university students and other academics. It is an article saver and reader designed for keeping record of articles used in studies, or provided in studies. The application will also be able to provide referencing in many of the major referencing styles. There will be multiple reading modes like ‘distraction free’ and a ‘speed reading’ mode. Users will also be able to store their own work within the application to keep everything together. The program will be free for students, and offered as a low-cost subscription to everyone else, or as a one-time, single platform purchase without cloud services.

### Motivation (100 words)

My motivation to make this application is because in my experience so far as a university student, many of the articles I have been provided have been lost after the unit has been completed. By having an application like this one, students and academics will be able to build their own article ‘library’ for use in assignments and projects. Referencing is also something that some people find difficult, and many people use applications to assist them with it. Adding this functionality will be a nice feature for users, allowing them to just use a single application.

### Description (500 words)

Schoolar will be a simple to use article storage and article reader aimed at students and academics. The application will be designed compatibility across computers and mobile devices, managed by a single user account. When a user registers, if they are signing up with a valid ‘.edu’ email address, their account will be free to use all services for a three-year period.

The design of the application will be pretty simple, allowing it to also have a web app following a similar design. The application will have three main sections; one for displaying the list of articles and altering the list, one for viewing the article itself, and one for referencing.

On large devices, the layout will be three columns, with the left-hand column containing the list of articles, as well as shortcut buttons to add new articles, mark/unmark them as read, and remove them. When an article is added, the user will be prompted to categorise it either from a list of default categories, such as ‘Essential Readings’ or ‘Textbook Excerpt’, or their own custom categories. Each article will also be taggable, for example tagged to a unit or topic, so that they can be searched and found easier. The middle column will be the primary column, and it will be where content is displayed. The content can either be able to be displayed ‘as-is’, which will just be a web view of the article, ‘distraction free’, which would strip the article of advertisements and non-essential images, and in a ‘speed reading’ mode, which would allow the user to speed read through the article. The third column will be the referencing column. The user will be able to highlight text, and the program will be able to correctly cite it in their chosen referencing format.

Smaller devices will only work with a student or subscription-based account, all of the functionality will remain the same, however the application will be presented as a single view for each component. The user will be present with the list of their articles and the controls for them. When the user selects from the list, the view will change to the content display and a navigation arrow will appear at the top of the application allowing them to go back. If the user selects text, in the context menu there will be an option for ‘reference’, which will then display a view for referencing the text.

When a user is citing an article, they will have the option to create a running bibliography so that they can keep track of all of the other articles they have referenced, and then when their work is completed, they will be able to upload that as well. Being able to keep everything together is a nice feature, as it provides an extra point of backup as well as the option to access their work from anywhere.

### Tools and Technologies (100 words)

The application will be primarily developed with the .NET MAUI framework using the C# programming language. The data will be stored in a MariaDB, and accessed via API calls made to a Laravel/PHP backend. For the web version of the application, it will be created with Vue.js and styled with Tailwind CSS.

Visual Studio 2022 and Visual Studio code will be required software, and package managers like NPM and Composer will also be needed.

To keep costs down, the application will be built with scalability in mind and hosted on a cloud-based platform. This will ensure the application is always getting the resources that it needs.

### Skills Required (100 words)

Building the application will require frontend as well as backend programming skills. Using .NET MAUI for the application frontend means that the application will be designed using the XAML markup language. The backend for the application will be written in C# so that will require a good amount of knowledge as well. The frontend for the website will be built with Vue.js, so that will require a good understanding of HTML and JavaScript. And finally, with the data service being a Laravel API connected to MariaDB, a good understanding of PHP and SQL is also required.

### Outcome (100 words)

Creating an educational tool that isn’t difficult to get started with or an absolute chore to use is the number one priority. With a fast, easy to use application, the ideal outcome is that it gets picked up by students across the world and increases their quality of work. Partnering with universities or other academic institutes would also be a great goal. The code could easily be modified to have teachers ‘push’ articles to students and monitor if they’ve been opened or not, which could be a powerful tool for them to have.

## Presentation