



# META MERGE TASKER

---

Assessment 3 Stage 1 - Project Report

<https://github.com/LucaPallas/MetaMergeTasker>

OUA Building IT Systems (CPT111)  
SP3, 2022

---

This project has been done in line with the course work set out in SP3 2022 Building IT Systems CPT111.

As a group we verify the originality of this body of work and all references have been cited where applicable.

Stdout.Systems

by

Adam Mutimer	S3875753
Matthew Wotton	S3905126
Zachary Williams	S3909738
Geoffrey Davis	S3930693
Luca Pallas	S3940012



# META MERGE TASKER

## Table of Contents

<b>Motivation:</b> .....	<b>3</b>
The Team:.....	3
<b>Project aim:</b> .....	<b>5</b>
<b>Project goals:</b> .....	<b>5</b>
<b>Project Progress:</b> .....	<b>7</b>
Description: .....	7
Outcomes to Date: .....	8
Scope Creep:.....	9
Progress: .....	9
Testing:.....	10
Tools and Technology:.....	10
<b>Challenges and Learning</b> .....	<b>12</b>
Project Plan Refinements: .....	14
Timeline Refinements:.....	15
Risks and unexpected events .....	16
<b>Marketing Pitch</b> .....	<b>17</b>
<b>Skills and Jobs</b> .....	<b>18</b>
<b>Reference List</b> .....	<b>22</b>



# META MERGE TASKER

## Motivation:

Upon brainstorming an idea to build as our project for this semester, the team all agreed on an App that would house several other programs that would be beneficial not only in other people's lives, but also in our own personal lives. Several iterations of the idea were discussed, with a short list then posted and a final list of 5 MVF's being agreed upon.

From here, Meta Merge Tasker was born.

Meta Merge Tasker is an organizational app that allows its user to achieve several tasks within one easy to use platform. Tools include an expense manager, a lists program, a password manager, an organizer, and a password generator.

## The Team:

### **Adam Mutimer – Programmer – S3875753 - [S3875753@student.rmit.edu.au](mailto:S3875753@student.rmit.edu.au)**

Adam has been producing code for all 5 MVF's as part of the programming team headed up by Luca. With his extensive background in coding as well as I.T in general, Adam has been able to contribute by building the framework on which the coding is attached to.

Adam's professional background is within I.T, currently working within the education department in an I.T support role. His on-the-job experience and knowledge has been invaluable in providing real world solutions that the other team members have not been aware of.

### **Matthew Wotton – Team Lead - S3905126 - [S3905126@student.rmit.edu.au](mailto:S3905126@student.rmit.edu.au)**

Matthew's knowledge and experience in project management and team management has enabled the team to stay on task and work within the allocated deadlines. Matthew has been involved with report writing and administrative tasks as well as being part of the UI (User Interface) development team, alongside Zach.

Matthew's professional background in project management has allowed other team members to stay focused on tasks by taking on roles outside of the technical work being performed by others. Early in the project, Matthew was involved with developing the UI design and framework. Matthew will move onto working on the final presentation once the current report has been finalized.



# META MERGE TASKER

**Zachery Williams – UI Developer – S3909738 - [S3909738@student.rmit.edu.au](mailto:S3909738@student.rmit.edu.au)**

Zach took on the role of leading the UI Development team in creating the Figma prototype as well as being an early researcher in finding the best tools to create Meta Merge Tasker. Currently, Zach is working in tandem with Matthew in completing the report writing and then moving onto the final presentation.

Zach's background comes from a desire to know more about I.T while working towards gaining a career within the field. Zach's professional background within management roles has led him to have the foresight to stay on task as well as helping and mentoring other team members to stay on schedule.

**Geoffrey Davis – Programmer – S3930693 – [s2920693@student.rmit.edu.au](mailto:s2920693@student.rmit.edu.au)**

Geoff has been somewhat of an all-rounder within the team who has always put his hand up to do more tasks. Presently, Geoff has been working on the programming of several of the MVF's alongside Adam and Luca. Geoff has also taken on lead roles in completing the peer review assessments while ensuring the team reviews his work to maintain a balance in workloads.

Geoff's background in customer service has helped in seeing the Meta Merge Tasker from a user's perspective as Geoff's mindset is geared towards the customers experience and what we can do to improve this experience.

**Luca Pallas – Lead Developer – S3940012 – [s3940012@student.rmit.edu.au](mailto:s3940012@student.rmit.edu.au)**

Without Luca there would be no Meta Merge Tasker. Luca had the initial idea for the project which the team decided to do. As such, Luca has had the final say on which functions the team develops and how each MVF (Minimum Viable Features) should be presented. Luca has been leading the programming team, developing the app along with Adam and Geoff.

Luca has an extensive background in programming, having studied and completed previous programming courses. He has experience in several programming languages which has helped in defining the direction of the app.



# META MERGE TASKER

## Project Aim:

To bring together a project that encompasses every day and widely used applications into one easy to use platform in the shape of a mobile app.

## Project Goals:

### ***Figma Prototype:***

From the start of the project, it was always deemed necessary to have a Figma prototype as part of our artifacts, to show as progress of the project. We achieved this goal by completing the User Interface for our presentation assessment.

As a team we had limited prior knowledge of how to use Figma efficiently, so it was particularly rewarding as the UI design came together and then result was shown as part of the assessment, which we were all immensely proud of.

### ***Team Formation:***

A key goal for the successful completion of this project and the overall semester was to ensure a team was formed who all had skill sets that complimented each other.

When Matthew set out and approached people to form a team, he looked for others that would bring different strengths and skill sets that he thought would be useful to a project success.

Matthew's strength was his background in project management, when Luca was approached, his strengths were in his background in programming and his previous completion of I.T course, Adams strength was his professional I.T background and knowledge of the industry, Zach had previously completed other programming classes and had a desire to learn more, and Geoff brought his knowledge of customer service and engagement as well as his academic experience in I.T.

Overall, a team was brought together with vast and varied experience and traits that would make this project and course a success.

As each person has been able to bring a different perspective on how to approach tasks, problems, and solutions to our end aim.



# META MERGE TASKER

## ***Learning Software:***

In order to bring our project to life, we all had to learn how to use new software in one way or another. The two biggest hurdles were learning Figma and Android Studio.

As previously stated, Figma was used wholly for UI prototype, with Matt having limited experience from a previous project and Zach not having any experience at all.

Our learning experiences included watching and following tutorials through the Figma library as well as researching on our own to determine if a function or idea was possible to do based off what we wanted to show in the prototype.

Overall, the learning experience of Figma and implementing and designing the Meta Merge Tasker prototype was straight forward as Figma is an intuitive platform.

Android Studio has been a learning journey for all during this project. Outside of Adams limited experience with Android Studio, no one else in the team had used the platform before.

Initially, Zach took on the role of researching and watching tutorials to determine if Android Studio was going to be a suitable tool to use in accomplishing our project goals.

Throughout the whole project and as we continue to finalise Meta Merge Tasker during the latter part of the project, the team is still learning and discovering how beneficial Android Studio is in fulfilling our project goals.

Not only has Android Studio been a platform to run and test code; it also had the ability to import our UI design from Figma in which we hoped it would do but did not have a full understanding on how that would be implemented at the start of our journey.

As a team we would say Android Studio has been the biggest challenge and learning curve in order to successfully bring Meta Merge Tasker to life.



# META MERGE TASKER

## Project Progress:

### Description:

#### ***How it began:***

Meta Merge Tasker started off as an idea from our lead developer Luca Pallas, Luca already had a strong idea on how the application would be laid out and its key features, Luca's initial project idea stated: "My idea for a project, is a mobile organizer app with the usual functions such as notes, an appointments calendar, a to-do list etc., but with the added function of having a section dedicated to listing all streaming services/subscriptions the user is signed up to that both provides reminders via calendar but also creates a monthly total cost for all the subscriptions."(Luca Pallas). Following Luca's plan, we brainstormed and laid out a plan which we would follow for the remainder of the project. After the brainstorming was complete, we concluded that the initial app would contain the following five MVF'S as listed below:

- Expense Manager – An expense manager where the user can add, edit, and delete a record. The expense manager will also ask the user to input the cost of each expense and the dates that they will need to be paid, once the new expenses are added the expense manager will calculate the total cost for the month and display this to the user.
- Login Screen – which will allow the user to enter their email address and password, with the data being pulled from an encrypted database.
- Organizer – An Organizer with a Calendar view with the ability to add, edit and delete entries.
- Password Generator – Randomized password generator which pulls from a set of rules set by the user.
- List Application – A list app which will enable the user to add, edit and delete records like a "Todo" list.



# META MERGE TASKER

These are the 5 MVFs that will be initially added to the application, but the future will also introduce EVFs including; a password manager, audio journal features as well updating and improving the general UI to include more usability.

We also have future plans to implement a web-based version of the application, which would lead to the development of a server environment to provide cloud accessibility for user data.

## ***How it progressed:***

Once the team had a set goal, we moved into the planning phase and determined what tools and applications we would need to use for each part of the application. Over time things were progressing with the project.

There were of course roadblocks and complications like with all big projects. This included things like time management and learning to use unfamiliar development tools, which is discussed in more detail ('Challenges and Learning').

As of week, twelve there has been great progression with the application as we have a working prototype that is currently in the final stages being tweaked and tested to root out any bugs and issues.

The project has followed our initial plan well, considering the goals that we had set ourselves. If there were any changes to the initial plan, they were only subtle and mainly arose due to time constraints.

Some of these changes were caused by instances such as; the team having to learn a whole new program in a short span of time and realizing that initial planned features might be a bit too far from reach at the current point in time.

## **Outcomes to Date:**

As of week twelve, the bulk of the application has been completed. Coding for the 5 MVFs is close to completion but still needs final tweaking. This includes the final testing stages of the frontend and backend code.

This is to be completed by Geoffrey Davis and Adam Mutimer who have been completing much of the code for the application.

Luca Pallas is also finalizing the integration of the prototype UI into the prototype code to ensure the application reflects the identified UI design. Due to time constraints, we were unable to implement any EVFs. Initially going into the project, we did not foresee that learning android studios would set the team back as much as it did.





# META MERGE TASKER

We assumed using the Android Studio IDE would be as simple as adding the java code and linking the buttons to the UI etc. But even so we were able to complete our base 5 MVFs.

In regard to EVF development, the team has come to the understanding of working to improve the mobile application, which would include implementing the password manager feature, and using the mobile application to test and prove concepts before moving on to more advanced features.

## Scope Creep:

During UI development, the design team felt it necessary to explore some refined UI elements and include designs for some EVFs that would not be implemented in the first application prototype iteration. This was helpful in allowing the team to visualize an ultimate end goal and plan for future improvements. However, this resulted in additional design work that may not have been entirely necessary. The team could have planned to explore EVF designs and UI refinements later into development. Some of this additional UI design work may have influenced the coding team to specify additional work for the application prototype. After some team discussion, the team pulled the reins in with a focus on achieving goals within the scope of the project.

Another factor of some scope creep has been unexpected changes to team member availability which has affected project tasks and responsibilities to become delayed and/or redistributed across the team. This has sometimes resulted in inflated workloads and unreasonable deadlines. The team have worked to maintain an open line of communication to help manage these issues and have worked diligently to overcome these challenges.

Even though there has been some scope creep, the team has been able to re-evaluate and delegate to ensure the original project scope is followed.

## Progress:

Progress for the overall project has had its ups and downs. Regarding reports, presentations and assignment documents; the team has done well and presented all work in a timely (mostly) and professional manner, confident with required criteria set by the university.

The team has endeavored to specify reasonable time frames and workloads between the group to complete tasks. Developing the application has been a different story. There have been several setbacks causing the team to fall behind and having to finish things at the last minute. The biggest setback that the team has faced was learning to use the Android Studio IDE.



# META MERGE TASKER

Only two of the team's members were confident with java and then came the task of implementing java into Android Studio. Learning Android Studio has set the development team back about a week (or more) in total. The biggest setback was integrating the java code with the UI, linking buttons with the code, and ensuring that all features run correctly.

Regarding our MVFs, the expense manager feature is not fully coherent with the initial plan that we have set. We initially wanted the expense manager to take the dates of the expenses and be able to calculate the total per month by going off the dates inputted by the user.

Due to time constraints, we were only able to have the expense manager take the name of the expense and its cost per month, add the total of all input expenses, then show the total of all expenses per month.

## Testing:

The testing process has been fairly straight forward and ongoing. As the coding team or programmer completes a workable code block, they send it to another member in the team to run the code as intended within a simulated environment.

If an error is found, the tester will re-run the code block to ensure the error is emulated once more. Upon confirming the error, the tester will provide verbal or written feedback to the originating member to get the code blocked patched.

These important, yet simple steps have been taking place over and over again until each code block becomes free of errors. Errors being either through runtime or by a user entering an unidentified input.

## Tools and Technology:

### **GitHub:**

The team has been using GitHub to collaboratively develop the mobile application prototype. We have Learned how to link the Android Studio IDE with our project GitHub repository to compliment development and assist with testing.

The coding team have been taking advantage of the comments feature for commits, to effectively communicate changes, including some testing information. This has enabled the team to keep up to date with progress on the prototype development.

The coding team have been taking advantage of the comments feature for commits, to effectively communicate changes, including some testing information. This has enabled the team to keep up to date with progress on the prototype development.



# META MERGE TASKER

## ***Microsoft Teams:***

Microsoft Teams has been used as our main forum for communication. We have been using the platform to host our team meetings twice a week through the video chat function, and we have taken advantage of the text chat features with countless back and forth chat sessions.

Mobile access has also enabled team members to be present for meetings when they are unable to be present at their desktop workstations. The platform has also provided a space for the team to collaborate on word documents and reports.

## ***Trello:***

Trello is being used as the team's project hub with important project tasks and activities dated and assigned to team members.

It's also been linked to our Microsoft Teams page, allowing for a smoother workflow between forums.

## ***Android Studio:***

Support of the Java programming language was a major factor in the teams' decision to use this IDE. This is because the Java language makes up the backbone of our project prototype.

This IDE has been a great tool for testing our code, as it provides emulation of android devices and allows easy connectivity to test on real android devices. Android Studio has also supported the team when working across the Windows and Mac platforms.

## ***Figma:***

Figma to date has been used extensively to deliver the prototypes you will see today.

It has allowed multiple team members to work on a single piece of work at the same time without causing file conflicts as it is a web-based development tool.

It has also been of great use in inspiring our designs due to the large array of Ui libraries available.



# META MERGE TASKER

## Challenges and Learning

### ***Redefining our project MVFs:***

After completing our project proposal, we came to realise that our initial MVFs were more intricate than we had realised and we became concerned about completing a project prototype within the project timeline.

With each MVF, like our Organiser feature, requiring multiple sub features to achieve the whole feature, Like a task manager, notes manager and a calendar. We decided to re-evaluate our MVFs and scale them back to their most basic components. We planned to introduce the more refined features of our MVFs into future iterations of our application.

We also realised that login access and a home menu would be important features for our application and decided to include these features as MVFs. Login access would be important to protect potentially sensitive information that our application would record, while a home menu provides a space that easily links to the applications' main features (MVFs).

As a result, we now have 7 MVFs, with the first iteration of our application prototype focusing on basic functionality for each feature. We have also re-focused the main project goal to produce a mobile application prototype within the project timeframe, that demonstrates the relevance and potential of our application idea with basic features.

This was one of the teams' first issues with the project that we worked to resolve. The experience helped the team to become comfortable with sharing constructive feedback as we discussed ideas and reasons for reworking our original MVFs. The team became more confident with our project plans after resolving this issue.

### ***UI Prototypes:***

Some of the challenges faced during the development of UI prototypes were learning our way around the Figma platform, experience using the platform to collaborate, and understanding mobile UX concepts.

We used the following learning resource: (<https://youtu.be/dXQ7IHkTiMM>), to introduce us to the Figma platform.

This was very helpful in getting us started with a new Figma project, but there was still a lot to learn. Only having brief knowledge of using the platform caused some development holdups. It took some time for the UI development team to figure out using the Figma UI libraries and to build and use custom components. but the UI development team worked to increase their understanding of the Figma platform, to break through these barriers.



## META MERGE TASKER

Lack of experience collaborating with others using the Figma platform also caused some delays as the UI development team needed to spend additional time discussing their approach.

The UI development team learned that active communication and setting achievable goals was very important for collaboration.

After showing some UI work to the whole team, we discussed feedback and improvements.

One of the main points of discussion was about the user experience on a mobile device. We then investigated and discussed how we could improve the mobile user experience of the UI for our application.

Everyone had some very insightful feedback for this, and it became a great learning experience into mobile user experience concepts. This caused the UI development team to reorient their plans to include applying this feedback and focus on the mobile user experience as they continued development.

### ***Android Studio:***

Early in the project, the team decided to use Java as the base programming language for our project. Most of the team have experience using Java and/or have used other object-oriented programming languages like Python.

We then chose to use Android Studio as the main IDE for our mobile application as it supports the use of Java and provides Android emulation for testing.

Using Android Studio has been a big challenge for the team as we are all new to using this IDE. Learning to use Android Studio has been the biggest challenge. It has been difficult to find relevant learning resources and tutorials as Android Studio now focuses on using the new Kotlin programming language for Android.

Using Android Studio also requires some knowledge of Gradle script which we have also been having to learn. Most learning resources we have managed to find have been very involved, complex, and time consuming or outdated. This has caused delays in our work to build a mobile application prototype and has resulted in the team using familiar IDEs like Eclipse to develop coding prototypes as we learn to incorporate code into Android Studio.

Despite these setbacks, the team is becoming more proficient with Android Studio and is progressing through building a mobile application prototype using the IDE.



# META MERGE TASKER

Project Plan Refinements:

## ***Feature research:***

When defining the features for our project, it may have helped to apply more research into other organizational software. This could have helped the team to gain a deeper insight into the features that we would want to include in our mobile application, by analyzing features that are good and not so good, and what could be done to make them better.

This research could have helped us to identify feature trends in organizational applications that we could investigate to make our application more relevant and attractive to users.

The team has discussed and applied some investigation into the project EVFs. However, it may have been beneficial to spend more time on this research and work to create small reports on implementing these features.

This would help the team to more thoroughly understand the tools and work required for these features and how we could plan to introduce these features in future application iterations.

As the team spends the last few weeks to complete our first prototype iteration, the current project plan would become too compressed if we were to include this research with the time we have left.

## ***UI Development:***

Using Figma, it may have been beneficial to the UI development team by working to define basic frame layouts first, then we could define required design components.

This would have helped with collaboration as one or two team members could be identifying library components and building any required custom components (e.g., buttons, text boxes, foreground objects, backgrounds etc...) while other team members could be placing these design components into the UI layouts.

This method of working may have helped to streamline the UI development process, allowed for more team members to be involved and help maintain a consistent theme throughout the UI.

## ***Android Studio:***

Although the team is getting better with Android Studio, the learning challenges have caused many delays. It may have been beneficial to investigate alternate IDEs for our project.





# META MERGE TASKER

There could possibly be more beginner friendly IDEs that we could have used. Alternatively, the team could have looked into learning the Kotlin programming language to better aid with using Android Studio. However, this may have introduced further learning complexities.

Despite the challenges the team has had with this IDE, the team has described a great sense of reward pushing through them and becoming more familiar with the IDE.

## ***Roles:***

It may have been beneficial to give more focus to assigning roles for the project.

Specifically, we could have assigned roles for a UI developer leader and a programming developer leader who would start their roles by actively researching and learning how to use required project tools.

This way, we might have been more prepared for the UI and coding tasks and the team would have specialized leaders more able to provide support.

## ***Timeline Refinements:***

During the first few weeks of the project, it took some time for the team to get passed the forming and storming stages of our new group dynamic.

One of the main challenges was getting familiar with everyone's availability. By the time the team was moving into the norming stage, some of our work was already delayed, which affected our assignment 1 submission to be slightly late. It was early into the project and everyone was still getting to know one another. However, being more timeline focused during this stage, (i.e., evaluating workload timelines and specifying expected deadlines when assigning tasks), could have motivated the team to fall into a more effective and punctual rhythm of work.

Early in the project, within the first 4 weeks, we could have spent some time to define and assign roles to the team. Team members would then be tasked to spend some time each week to research required tools in preparation for their roles. This way, the team would have a network of specialized team members, more capable of providing support in the field relevant to their roles. It could have also helped to identify unforeseen issues, like the complexity of using Android Studio.

During weeks 3 and 4, when we were defining the features for our mobile application, we could have researched other organizational software to gain further insight into the features for our application. We could have also spent more time to consider the implementation of our MVFs and the timeframe we have for the project.



# META MERGE TASKER

This way, we might have been prepared with a more realistic project plan, giving us more time in the upcoming weeks to focus on project and unit tasks.

During development of the UI prototypes, involving more team members into this task may have reduced the workload and made the week 3 timeline we had for this task feel more achievable. It might have also helped to start this task a week earlier, which would have been week 5, but during week 5 we were busy finishing up assignment tasks and learning to use our project tools.

The original project plan (detailed on Trello with cards in the 'Project Stream' list) includes a detailed plan of how the team initially intended to work on our project during the timeframe of this unit. This included outlined tasks delegated across the team for each week, team meeting minutes and sub-group dynamics. This plan was designed to prioritize important tasks while allowing time for research and lower priority tasks.

Due to how the team dynamic has evolved and how we have collectively decided to pursue project goals; the initial project plan soon became obsolete. Despite this, the initial plan has guided the team as we actively planned project objectives. We have made some time and progress to research project tools and refine our project features.

We have used subgrouping to work on UI prototypes and coding development while other subgroups could lead research, reporting and some assignment tasks. Although we have provided some updates to our Trello board, the team could have worked to implement these updates and changes into our project plan on Trello.

## Risks and unexpected events

### ***Inadequate Time Consideration:***

The team has had some issues falling behind due to underestimating workload and timelines. This was mainly prominent during the creation of our project proposal and development of UI prototypes.

As a result, the team has been learning to give more consideration to workloads and timeframes when discussing and assigning tasks.

### ***Miscommunication of Ideas/Task:***

The team has experienced some issues with communication and understanding some individual ideas. This occurred early into the project when the team was still getting to know and understand each other. This caused some delays in defining the type of project the team would work on.





# META MERGE TASKER

When discussing the basis of our project, everyone had their own concepts of what the team could achieve based on their own IT experience, knowledge and ideas. This made it difficult for individuals to accept ideas that they were not familiar with. After discussing everyone's experience and capabilities, the team became more comfortable in considering and accepting ideas. We ran into similar issues when defining the main project idea and the project features.

Researching ideas was helpful in understanding how they may be incorporated and what challenges we would face.

It was also helpful to have some team members who are more experienced in programming who could suggest how things might be implemented and what tools might be required.

## ***Team Member Dropouts:***

We initially began as a team of 6 members but early into the project (within the first 3 to 4 weeks) we had one team member become inactive, not showing up to team meetings and not responding to written communications.

This caused some delays as the team worked to resolve this issue and take on the vacated tasks, which also affected a slight increase to the teams' workload. We did our best to try and rekindle communications before having to make the issue known to our mentor and course coordinator.

This team member never indicated any issues or concerns before going inactive and the team were not aware of any issues until it was too late.

Unfortunately, this team member is no longer a part of our team.

## Marketing Pitch

The Meta Merge Tasker project aims to build an organisational Android application with plans to incorporate a web-based version of the application. The application is designed to bring important organisational applications together, with a focus on simplicity and usability. This includes task management, documenting, basic expense management and password management features. Further development would lead to the creation of a subscription-based cloud platform for users. Targeted towards self-employed workers, people working from home and students. We have worked to keep features simple, reducing intricacies and helping to create an application that is easy to navigate and use for first time users. This will help to potentially broaden the application's user base to include more general users, including parents who may be attracted to the lists, expense manager and password manager features.



# META MERGE TASKER

We are currently in development of the mobile application component of the project.

We plan to use the mobile application to test our user base, identify improvements and refine basic features and concepts before commencing with the web application development. This will help guide smooth development as we work to translate the Meta Merge Tasker features into a web application. As we develop the web-based application, we also plan to create a database and server environment to store and access user information. This would introduce the possibility of devising a subscription service for users to store and access their application data in the cloud. Allowing users to store and access notes, lists, tasks, expense data and passwords between mobile and web applications.

## Skills and Jobs

### **Full Stack Java Developer**

(seek.com.au, *Fullstack Java Engineer - WFH options- medical industry job in Melbourne vic 2022*)

Having a full stack Java developer on the team would enable the team to expand the login features of our application and expand application features to be accessed from a server. This would help to integrate the mobile application with a web-based application.

The main responsibilities for this role would include setting up a server space that contains a database for user information, including login details and user data. This role would also include refining front end code. This role would lead to working with web development team members to help build a web-based version of the application.

It would be preferable to have experience in android development, in particular, with using Android Studio.



# META MERGE TASKER

Required skills and experience:

- Java, with a good knowledge of Java classes
- Node.js, for building a server environment
- JSON, for transferring Java objects from the server
- SQL, for building a database to store user data
- Android Development experience
- Experience using Android Studio
- Familiarity with Gradle code and React.js for mobile development
- Team project experience
- GitHub collaboration experience
- HTML
- CSS
- Some web development experience

An individual who is comfortable working as part of a team and capable of working within basic scope boundaries would be essential. We would also value someone who is able to work with less experienced team members and comfortable acting as a mentor to help build the teams' skills.

## **Senior Software Developer**

(seek.com.au, *Senior software engineer job in Canberra act 2022*)

The Senior Software Developer would be responsible for overseeing the application software development as a whole. They would be in charge of leading and guiding the development team. The Senior Software Developer would also play a major role in software testing. Overseeing testing plans and evaluating testing results, advising the team with software refinements. This role would require similar skills and experience as a full stack Java developer, with a specialization in development of the Android platform.

Required skills and experience:

- Ability to lead a software development team
- Collaborative software development experience, with experience using Android Studio and GitHub
- High level Java frontend and backend development experience
- Experienced in Android application development
- Experience in software testing

Given their leadership role, the Senior Software Developer would need to be able to guide and mentor the development team. Assisting the team with development tools and guiding research into overcoming development challenges.



# META MERGE TASKER

## **Security Software Developer**

(seek.com.au, *Security software developer job in Canberra act 2022*)

The Security Software Developer would work to ensure that all user data passed through the application and stored in the database is appropriately secured. They would be responsible for planning and overseeing security implementations. The role would also include penetration testing to ensure early detection of security vulnerabilities.

The Security Software Developer would work closely with the Senior Software Developer to plan implementation of data encryption, and security methodologies and technologies.

In addition to this, they would also work closely with the Full Stack Java Developer and Web Developer to implement these security technologies into the mobile and web-based applications. The Security Software Developer would be familiar with the latest security technologies and work to improve security methods throughout the development.

Required skills and experience:

- SQL Transparent Data Encryption
- Java
- Web development skills including HTML and CSS
- Some familiarity with Android Studio
- Experience using GitHub to collaboratively develop software
- Ability to maintain up to date knowledge in security technologies

The Security Software Developer would also take on a leadership role and guide the team to resources that may further enhance security features.

## **Web Developer**

(seek.com.au, *Web developer job in Thebarton, Adelaide Sa 2022*)

The Web Developer would be responsible for overseeing development of a web-based version of our mobile application. They would interpret the mobile applications' features, themes and UI into a HTML website. They would work closely with the Full Stack Java Developer and Security Software Developer to build a website that can securely access server and database features.

They would also actively communicate with the Senior Software Developer and mobile development team to ensure consistency between the mobile application and the web-based application.



# META MERGE TASKER

Required skills and experience:

- High level HTML and CSS knowledge
- Some experience using Node.js and SQL – using servers and accessing databases
- Experience working with design briefs
- Ability to collaborate within a team
- Preferable experience using GitHub
- Preferable experience working on projects with paired mobile and web applications

The Web Developer role would include leadership opportunities, with potential for leading less experienced web development team members. This would require the web developer to be comfortable with acting as a mentor and guiding team members with web development tools and leading them to relevant learning resources if required.



# META MERGE TASKER

## Reference List

*Download Android Studio & App Tools* (no date) *Android Developers*. Available at: <https://developer.android.com/studio> (Accessed: November 20, 2022).

*The Collaborative Interface Design Tool*. (no date) *Figma*. Available at: <https://www.figma.com/> (Accessed: November 20, 2022).

*Web developer job in Thebarton, Adelaide Sa* (no date) *SEEK*. Available at: <https://www.seek.com.au/job/59155144> (Accessed: November 20, 2022).

*Security software developer job in Canberra act (2022)* *SEEK*. Available at: <https://www.seek.com.au/job/59204136> (Accessed: November 20, 2022).

*Senior software engineer job in Canberra act (2022)* *SEEK*. Available at: <https://www.seek.com.au/job/59068586> (Accessed: November 20, 2022).

*Fullstack Java Engineer - WFH options- medical industry job in Melbourne vic* (no date) *SEEK*. Available at: <https://www.seek.com.au/job/59119254> (Accessed: November 20, 2022).