Research Report for Android Studios Assignment 2:

Name: Aaron Corne

SID: 104600022

Link to code: https://github.com/SoftDevMobDev-2024-Classrooms/assignment02-AaronCorne.git

Assignment 2:

Planning and research: Things that will need to be done for this project

- Sketch layout for desire feel
- I am thinking for a design to have a red curtain backdrop and have each item with a stroke to have a sort of “show” aesthetic with two arrows on either side tha1t act as the next and back buttons for each item
- I think I will have a switch button option for an instrument that could be acoustic or electric
- I like the idea of having the text and feel be quite game-like for one of the styles
- Continuing on from my first assignment I will be using Figma and Figjam to sketch and create the images I need for the items
- Will need to create a development timeline for getting this project done

Core functionalities that need to be done includes:

- sending the data of the chosen musical item to the second activity screen that will showcase the further details of the chosen item through parcelization
- user being able to set the rental length of the item
- User can confirm booking or cancel booking
- When confirming booking, has to be error checked with amount of credits available or if the user agreement box has not been checked
- For user actions a toast bar or snackbar should be displayed

Development plan:

Sketching/wireframe→ understanding UI/UX of app→creation of elements/images/design on figjam (ensuring responsiveness)→setup of workspace/project on AS→Establishment of styles and colours→implementation of layout and code

Code structure:

- Test out passing information using an intent
- Create data model class for musical items
- Create parcelize class for details
- Validation code for buttons (Next, Borrow, Save, Cancel)
- If the item has been booked, update the image to reflect this and change the borrow button to show it has been borrowed

- Need to have credit system for validation but can be reset to get more credits→ could provide a screen upon validation that allows this
- Implementation of Toasts and Snackbars
- User stories and use cases

User stories:

1)

As a music student, I need an app that allows me to rent high-quality musical instruments for monthly periods so that I can practice for exams and performances without having to purchase expensive equipment upfront.

User story 1 + use case: Ruby

Ruby is a 22-year-old music student living in Melbourne and has practical assessments throughout the semester for her course at Swinburne University. She is constantly practicing and is always in need of musical equipment. Swinburne provides some basic instruments, but for her semester, she needs a high-end electric guitar that she can use for the next few months. Purchasing such an instrument would be too costly for her on a student budget, and the Uni's inventory doesn't meet her requirements. She hears about an app that lets users rent musical instruments on a monthly basis.

The app shows various instruments with details such as ratings, descriptions, and rental prices for different periods. Ruby browses through the app, finds the perfect electric guitar, and rents it for three months. She can easily track how many credits she has left and recharge them if needed. With the guitar in hand, Ruby has access to professional-level equipment at a fraction of the cost. The app gives her the flexibility to return the instrument after her recital, allowing her to focus on her studies without worrying about long-term commitments or financial strain.

User story 2+ use case: Noah

2)

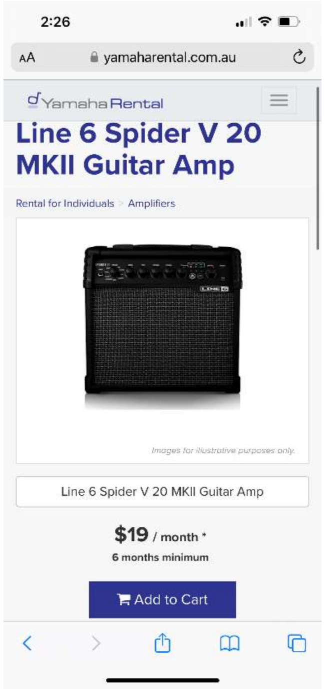
As a musician in a band, I need a platform to rent professional musical gear temporarily for gigs, ensuring that we have the right equipment for live performances without investing in gear we won't use regularly.

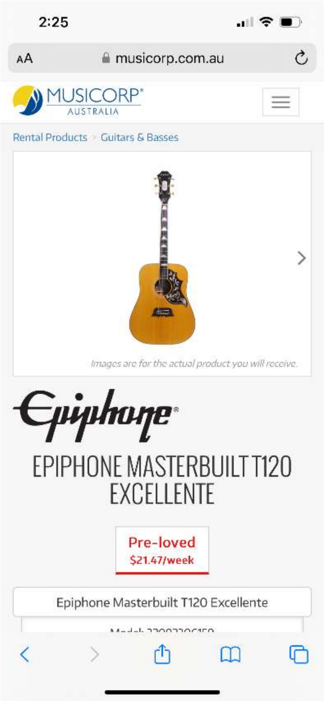
Story:

Noah, a drummer for a local Fitzroy rock band, often performs at small gigs in hole-in-the-wall pubs. The band owns basic instruments, but occasionally, they require extra gear for larger shows, such as an amplifier or high-end drum set, which they don't have the budget to purchase outright. As they have recently garnered acclaim for their performances, Noah and his bandmates decide to rent additional equipment for their performance. This allows them access to the equipment without having to bare the excess of a large upfront fee.

Using the app, they browse through an array of musical instruments. The app provides detailed information on each piece, including availability and rental duration. Noah selects an amplifier and a top-tier drum kit, setting the rental period 3 months to align with their busy schedule. The app even allows Noah to track whether the equipment is booked or still available, so there are no last-minute surprises.

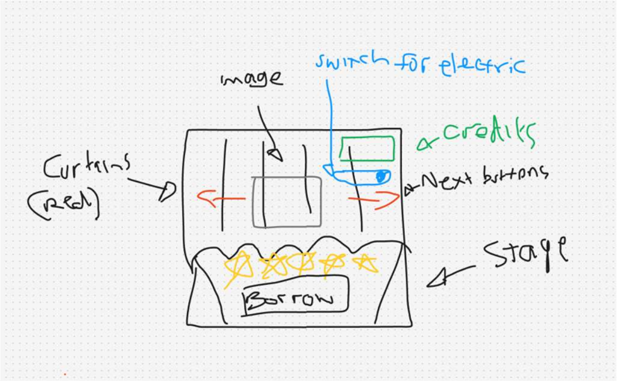
Research images for potential layouts:

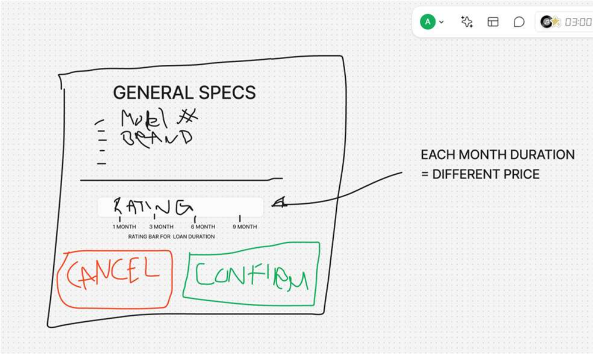

2:26
AA
yamaharental.com.au
Yamaha Rental
Line 6 Spider V 20 MKII Guitar Amp
Rental for Individuals > Amplifiers
Images for illustrative purposes only.
Line 6 Spider V 20 MKII Guitar Amp
$19 / month *
6 months minimum
Add to Cart

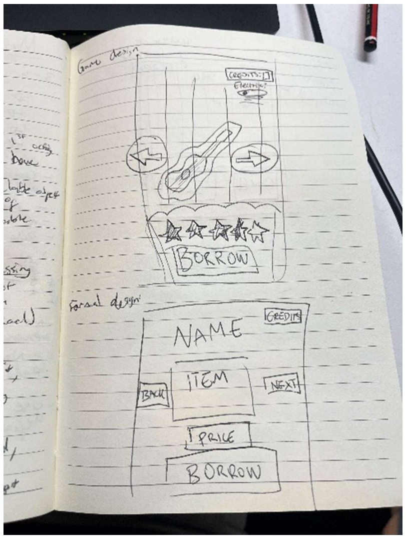

2:25
AA
musicorp.com.au
MUSICORP AUSTRALIA
Rental Products > Guitars & Basses
Images are for the actual product you will receive.
Epiphone
EPIPHONE MASTERBUILT T120 EXCELLENTE
Pre-loved
$21.47/week
Epiphone Masterbuilt T120 Excellente

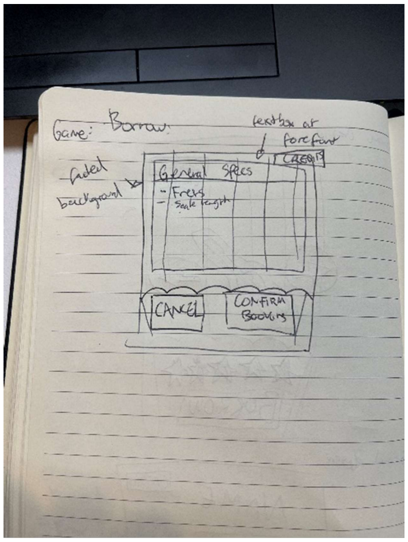
Looking at some different rental websites on my phone, I wanted to get a feel for how the industry displayed their items. Seeing the hierarchy structure of placing the image as the centre and the top of the layout impressed upon me the idea of doing something similar to ensure the user's focus and attention was on the current item.

Sketches: Using Figma draw tool to sketch basic ideas







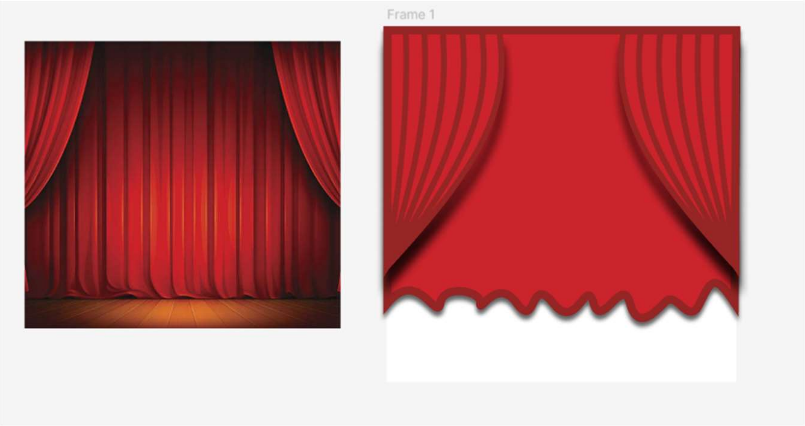


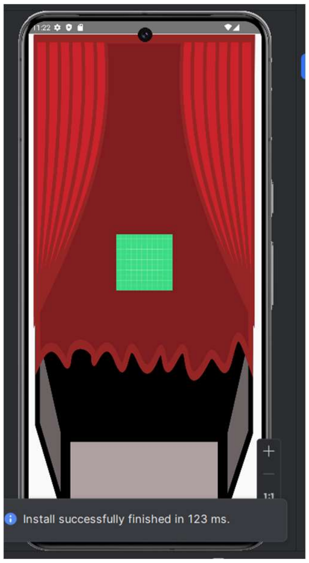
In the above left image, there are two possible layouts that I sketched out to see which direction I wanted to go in. The first is the game design aesthetic that was mentioned earlier and that includes a much more stylistic endeavour for the layout and feel of the app. This design makes use of customising buttons and instead of doing Next and Previous buttons, designing arrows that do the same thing. Then in using a switch, I could change the instrument to be a different variation, for instance switching from an acoustic to an electric guitar.

Following that design, I decided I wanted to also have an opposing design called the 'formal design' which more focused on the bare and sort of minimalistic layout and feel for the app. This had its advantages as it was a stripped back version of the game design style and feel and felt more straightforward. This design would leverage the standard buttons, standard TextViews and ImageViews.

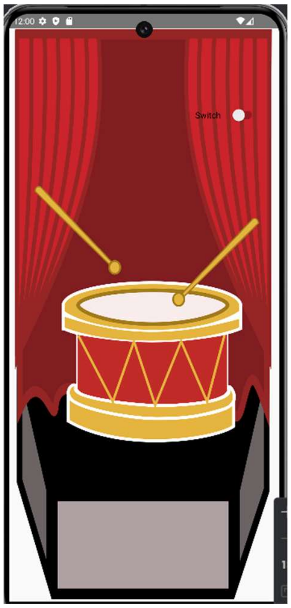
Element creation:

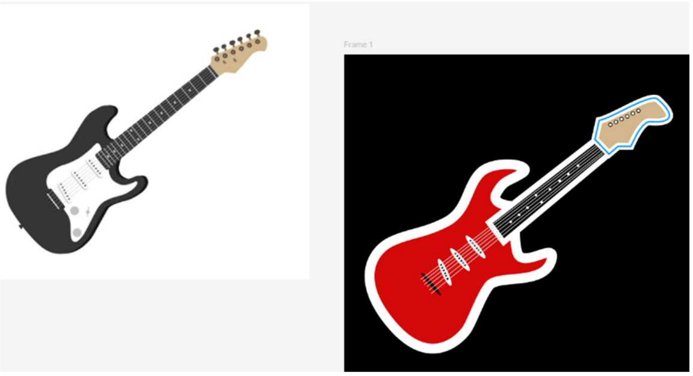
Backdrop for game design layout.

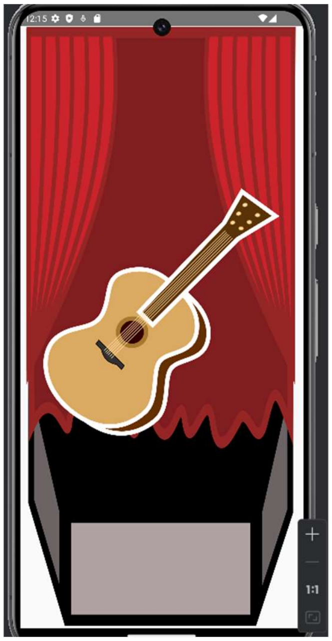


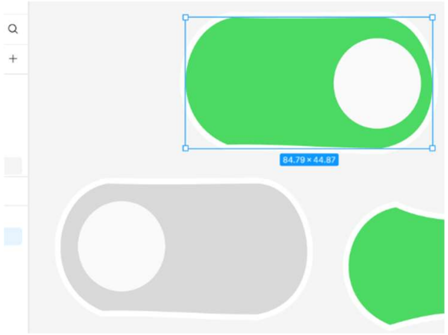


Install successfully finished in 123 ms.


12:00
Switch


Frame 1


12:15
1:1


84.79 × 44.87

Key design decisions:

- Combination of both the game design and formal layout, utilising the Next and Previous buttons
- Creation of elements in Figma to give the items a more stylistic and illustration aesthetic
- Allowing for a strap for the guitar option
- Allowing for insight into the music item by clicking on the image. This then updates accordingly to see if the item has been borrowed
- Using the primary colours of Red, Blue, Yellow as a colour motif throughout. This also allowed an appropriate colour contrast throughout the app which is important to a good User Experience
- Implemented validation checks for credit purchases, as well as other validation relating to ensuring the checkboxes of the privacy and terms have been clicked
- Updating the image to demonstrate it has been booked. I used a green filter overlay on the original image to signify that the rent of the item was a success
- Using the rating bar to demonstrate previous users reviews of the item
- Utilising the parcelisation feature as well as additional Intent features

When it came to Toasts vs Snackbars, I instigated the thought and design process as this: a Snackbar would demonstrate a directive and a Toast would signal an action, working as a descriptor to the events within the app. This is seen in the onCreate initialisation when the app starts as a Snackbar notifies the user that by clicking on the image, more details will come. This is also seen in the second activity where the user is notified that if they are out of credits to rent the item, then by pressing the save button, it would allow them to recharge their credits.

Contrastingly, the Toasts are used to describe things like when an item is rented or if the action is cancelled. I liked categorising the Toasts and Snackbars like this as it allowed me to know when to use each function and led to a smoother development process as a result.

Explanation of components of my Intent and advantages of using Parcelable objects in Android:

The components in my parcelable intent are:

- Item name

- If the item has a strap (for guitars)
- Rating
- Price
- The default image
- The booked image
- Description
- If the strap was selected
- If the item had been booked

The last two components were initialised as a **var** due to the nature of the data changing, whilst the rest were not so they could stay as **val.**

I thought these were the most crucial elements of each item to include and they allowed me to manipulate programmatically the user experience and to an extent the layout (as when the recharge TextView was visible or not).

The advantages of using Parcelable objects in android is that it allows a fast and memory efficient transfer of complex data between two activities. In my design, I actually implemented the passing of data to two activities in which only one launcher was required. This meant that I could create a seemingly complex (all relative to my knowledge and understanding) application as I could integrate the passing of data to display some cool features (such as the DetailsActivity).

Issues encountered and their solutions:

Some of the issues encountered was making executive decisions regarding the design of the activities. I spent a lot of time playing around with endless design choices only to realise that they were a burden and costing me time to work on other areas of the app. This was seen in things like trying to get the Electric guitar to be changeable via the switch like in the original design sketch. However, due to the structure of the code in the main activity I realised that instead I would have to use the switch for something else. This was because I was putting the Electric guitar as an indexable item in the arrayList but then wanting to put conditions on that specific index and that caused me a lot of issues with the other elements in the list.

Further, small technical issues plagued me throughout the development process, such as setting up the Espresso tests. I had had trouble in the first assignment trying to work this out and had instead manually created the espresso tests, but this time I wanted to work out what was going on. After some digging around and asking the inbuilt Gemini AI what was going on, I was directed to editing out the Jetpack Compose functions that were in my system. After getting rid of any reference to Compose functions, I was able to get Espresso testing up and running.

Testing justification

With the Espresso testing, I think I am still trying to figure out its best use cases and implementations. For this project I wanted to mainly focus on the testing the TextViews and the buttons like it was suggested in the project brief. This meant that I could focus on ensuring that the text matched what was intended and the button worked when clicking.