StuBid

Auction, Anytime, Anywhere

National University Of Singapore

CP2106 (Orbital)

Team Members and Student ID:

Bransome Tan Yi Hao (A0234931H)

Anshumaan Tyagi (A0244508E)

Team Name and ID:

StuBid (5005)

Proposed Level of Achievement: Artemis

Difficulty Attempted: Extreme

Year: 2022

ACKNOWLEDGEMENTS

The team would like to express our deepest gratitude to our mentor, Mr Martin Gao, Senior Software Engineer, ByteDane (TikTok), for all of his guidance and support throughout the preparation and completion of our Orbital Project. As well, we would like to thank our advisor, Mr Jordan Seow Hua Jun, an NUS alumnus, for all of his constructive feedback and advice for our project. Our mentor and advisor have been crucial to enabling this project by helping us develop software engineering skills, identifying industrial practices, and contributing greatly towards our learning.

SUMMARY

This report aims to highlight the possibility of creating a one-stop online marketplace, a mobile application for all students studying in Singapore Universities to conveniently buy or sell unwanted items online through an anonymous bidding system, maximizing the benefits gained by both the seller and buyer.

This project is part of NUS School of Computing 1st year summer self-directed, independent work course, also known as Orbital (CP2106: Independent Software Development Project). The difficulty level attempted for this project by the team is Artemis (Extreme). The project is mentored and advised with an industry partner from ByteDance (TikTok) and an experienced NUS alumni.

As part of the brainstorming process, the team considered current and potential university student situations and problems and came up with a motivation to develop “StuBid”, a software application aimed at solving these problems.

StuBid, also known as ‘Student Bidding’ is the product of a team of two 1st year NUS computing students from Computer Science and Information Systems. The mobile application (available for both IOS and Android users) consists of numerous core features notably, “Anonymous Bidding”, “Product Searching/Filtering/Listing/Editing”, “User Profiling/Registration/Login/Edit/Change & Forget Password”, “User Review”, “Real -Time Notifications” and many more.

In this project, the UI/UX Design are created using Figma. The main technology stack used are React Native (Front-End) and Firebase (Back-End). For development purposes, Git is used as a version control system while GitHub is used as a cloud-based hosting service to manage Git repositories. The development environment used is Expo CLI, with Visual Studio Code as the Code Editor. Jest is used for testing purposes.

The software project planning process is demonstrated through the use of “UML/ER diagrams”, “Activity Roadmaps”, “Use Cases”, “Figma prototypes”, and other project management tools. Additionally, “Project management with Notion”, “Version control with Git/GitHub”, “Folder and code structure” of the app are several examples of software engineering practices adopted by the team.

With a clear objective to develop a software application that is optimised towards simplicity and support for the users, “Stubid” aims to reduce online potential scams on products and help sellers/buyers make better informed decisions to sell/buy their products on a safe and conducive platform. At the moment, the application is targeted only at university students, but it can be scaled and expanded to benefit users of all ages worldwide in the future.

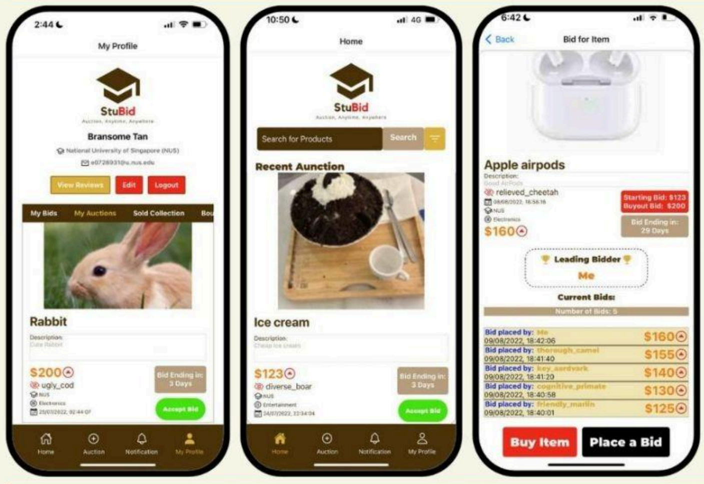
POSTER

PURPOSE/AIM

StuBid, also known as 'Student Bidding', is a one-stop marketplace, a mobile application for all students studying in Singapore Universities to conveniently buy or sell unwanted items online through an anonymous bidding system, maximizing the benefits gained by both the seller and buyer.

StuBid
Auction, Anytime, Anywhere

Tech Stack:



MOTIVATION

- Have you ever wanted to buy or sell unwanted items online through an anonymous bidding system in your university campus?
- Instead of the usual method where an online user interacts with another user to sell/buy items, you could have a bidding system in place where there is no contact or interaction until the final bid is accepted.

WHAT IS ANONYMOUS BIDDING SYSTEM?

All sellers and buyers will be anonymous (e.g personal details are hidden) and will have no interaction during the bidding of the item. When the final bid is accepted, only the seller and buyer will be able to exchange contact with each other.

CORE FEATURES

- User Register/Login
- User Profile (Customizable)
- User Reset/Change Password
- User Review and Rating
- User Contact Exchange
- Anonymous Auction (Price Bidding)
- Product Searching/Filtering/Editing/Deleting
- Bidding Duration/Validity
- Real-Time Product Termination
- Real-Time Product Update
- Real-Time Notifications
- Anti-Spam Bidding Prevention

Anonymous Auction

Product Listing/Searching/etc.

Bidding Duration/Validity

SOFTWARE ENGINEERING PRACTICES

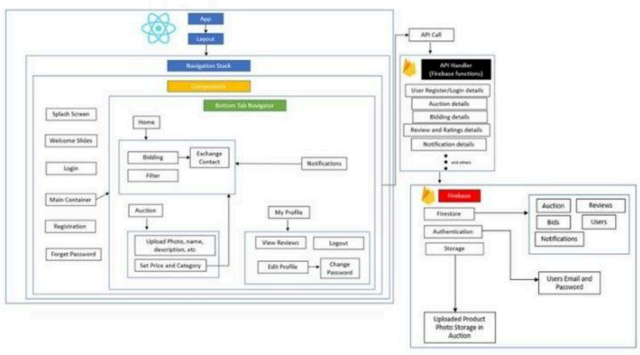
Version Control
- Git and GitHub
- GitHub Issues

Project Management with Notion
- Tasks Tracking

Testing
- Manual Testing
- Unit Testing
- User/Usability Testing

Project Execution
- Agile Software Development
- Sprint Planning

Software Architecture Diagram



SCAN HERE TO FIND OUT MORE

Team Members
Anshumaan Tyagi (Computer Science, Year 1)
Bransome Tan Yi Hao (Information System, Year 1)

NUS National University of Singapore | School of Computing

CONTENTS

ACKNOWLEDGEMENTS

SUMMARY

POSTER

Introduction 7

1. Problem and Situation
2. Motivation
3. Aim
4. Anonymous Auction

User Research and Observations 14

1. Target Audience and Needs
2. Competitors Research
3. Target Audience Observation and Research
4. User Stories

Development Phase 20

1. Project Scope
2. User Sitemap
3. Wireframe (UI/UX)
4. Interactive Wireframe Mockup
5. Technology Stack and Considerations
6. Software Architecture Diagram
7. Software Design Principles
8. Design Patterns
9. Activity Diagrams
10. Database Diagrams

Features **39**

1. Overview of Features
2. App Walkthrough
3. Additional Features
4. Feature Limitations and Constraints

Execution Phase **66**

1. Agile Software Development Framework
2. Sprint Planning
3. Software Testing
    a. Manual Testing
    b. Unit Testing
    c. Limitations of Software Testing
4. User/Usability Testing
    a. Testing for Milestone 2
    b. Testing for Milestone 3

Software Engineering Practices **86**

1. Project Management with Notion
2. Version Control with Git
3. Folder Structure

Response to Milestone Evaluations **94**

1. Milestone 1
2. Milestone 2

Reflection on Orbital Project **99**

References **100**

Introduction

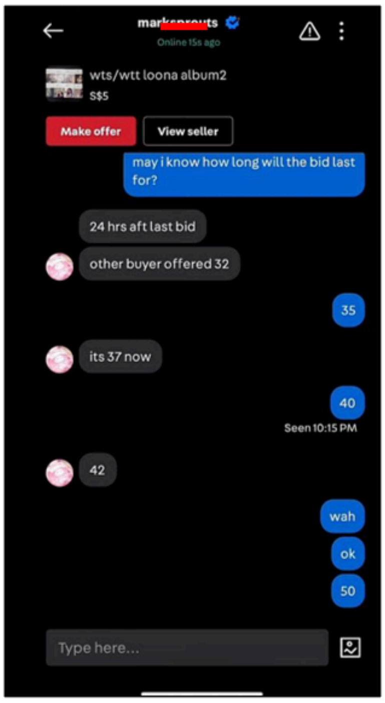
<u>Problem and Situation</u>

Tony is a freshman at NUS who is preparing for the new academic semester. He has recently bought a new MacBook for his studies, and it comes along with a free Apple AirPods. However, he already has an Apple Airpods and does not wish to keep another one. Therefore, he decided to sell the item at a cheaper price of $150 to any other students who wanted it. However, he realises that there is **no online platform catered specifically for students in NUS or other universities to be able to buy or sell unwanted items.** Hence, he went to online platforms like Carousell to look for potential buyers who are interested in the item.

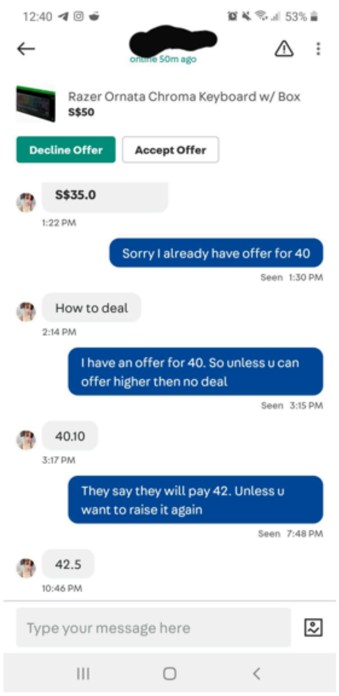
Many buyers have reached out and offered him a range of prices but **not at his targeted selling price** of $150. The highest offer made so far is by Ronaldo at $100, but he **wants to minimise losses** and does not wish to sell it too soon. However, he is **unsure whether there is another potential buyer** who is willing to offer a higher price for his item. Moreover, he **does not want to lose his current highest bidder** if there are no more potential buyers. He is also **afraid of scams** since he is **unsure if the buyer is authentic (truly wants to buy the product) or someone who is wasting his time by giving a fake offer.** Another major concern is that **many buyers pestered him on the current pricing offered by other buyers so that they can readjust their offer** accordingly which is unfair and unjustifiable.

Therefore, he is stuck in a dilemma and unable to come up with a decision. Due to impulse, Tony decided to sell AirPods to Ronaldo at $100 since he was **afraid of losing this offer.** After selling the item, he felt much relieved and happy. However, a few minutes later, Peter made a full offer of $150 to buy the item but it has already been sold. Tony **felt frustrated and disappointed due to not waiting longer and missing out on a better opportunity.**

Note: This situation is based on a real in-person interview conducted with an NUS student voicing his poor experience on an online ecommerce trading platform.


mar[redacted]ts
Online 15s ago
wts/wtt loona album2
S$5

may i know how long will the bid last for?
24 hrs aft last bid
other buyer offered 32
35
its 37 now
40
Seen 10:15 PM
42
wah
ok
50


12:40 53%
[redacted]
online 50m ago
Razer Ornata Chroma Keyboard w/ Box
S$50

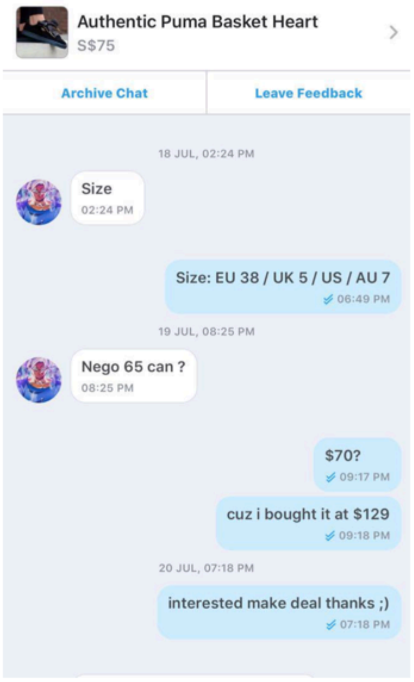
S$35.0
1:22 PM
Sorry I already have offer for 40
Seen 1:30 PM
How to deal
2:14 PM
I have an offer for 40. So unless u can offer higher then no deal
Seen 3:15 PM
40.10
3:17 PM
They say they will pay 42. Unless u want to raise it again
Seen 7:48 PM
42.5
10:46 PM

Seller may provide fake bidding prices for his product which may encourage/force the buyer to increase their offering price.

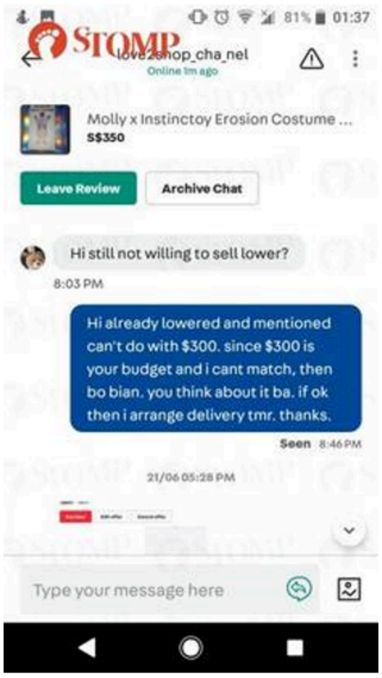
How does the buyer know if the seller is giving misleading bid price to make the buyer increase his bidding price?

In this situation, what could the buyer do to ensure that he/she is not being conned of the true price of the item from the seller?

Problem 1 : Misleading Offers from Sellers


Authentic Puma Basket Heart
S$75

18 JUL, 02:24 PM
Size
02:24 PM
Size: EU 38 / UK 5 / US / AU 7
06:49 PM
19 JUL, 08:25 PM
Nego 65 can ?
08:25 PM
$70?
09:17 PM
cuz i bought it at $129
09:18 PM
20 JUL, 07:18 PM
interested make deal thanks ;)
07:18 PM


81% 01:37

love2shop_cha_nel
Online 1m ago
Molly x Instinctoy Erosion Costume ...
S$350

Hi still not willing to sell lower?
8:03 PM
Hi already lowered and mentioned can't do with $300. since $300 is your budget and i cant match, then bo bian. you think about it ba. if ok then i arrange delivery tmr. thanks.
Seen 8:46 PM
21/06 05:28 PM

Buyer is trying to negotiate for an item from the seller in a much lower price.

However, Seller wishes to get the best price for his/her product but also does not want to lose out on the current deal.

In this situation, should the seller wait for an potential higher offer or risk losing the current buyer?

Problem 2 : Afraid of losing offers from buyers

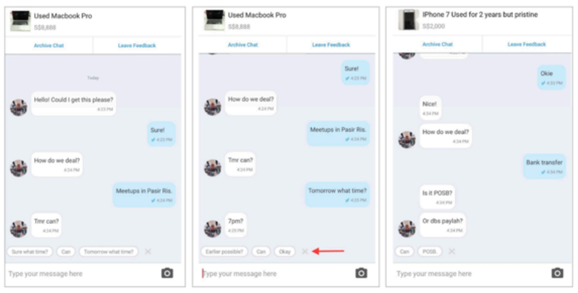
The images shown above are a few examples of complaints and problems encountered by users in an online marketplace shared through media and social networks.

Motivation

Have you ever wanted to buy or sell unwanted items online through an anonymous bidding system in your university campus?

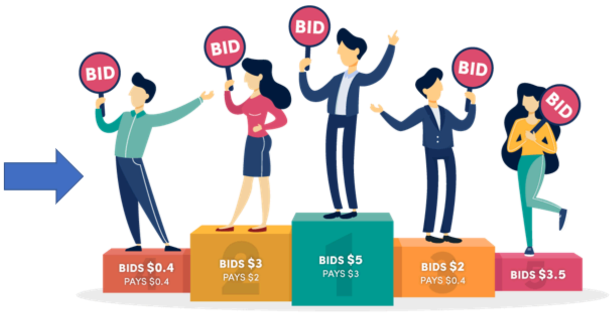
NUS National University of Singapore
NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE
SMU SINGAPORE MANAGEMENT UNIVERSITY
SUTD SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN
SIT SINGAPORE INSTITUTE OF TECHNOLOGY
SUSS SINGAPORE UNIVERSITY OF SOCIAL SCIENCES

Instead of the usual method where an online user interacts with another user to sell/buy items, you could have a bidding system in place where there is no contact or interaction until the final bid is accepted.

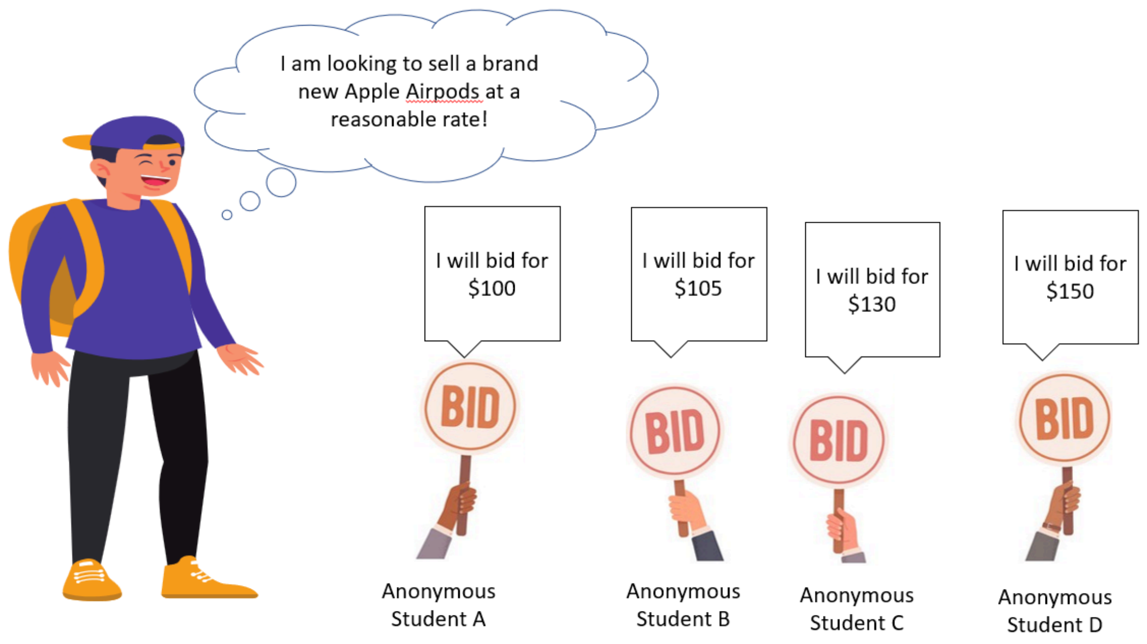

Used Macbook Pro
S$8,888
Archive Chat | Leave Feedback
Today
Hello! Could I get this please? 4:23 PM
Sure! 4:23 PM
How do we deal? 4:24 PM
Meetups in Pasir Ris. 4:24 PM
Tmr can? 4:24 PM
Sure what time? | Can | Tomorrow what time?
Type your message here

Used Macbook Pro
S$8,888
Archive Chat | Leave Feedback
Sure! 4:23 PM
How do we deal? 4:24 PM
Meetups in Pasir Ris. 4:24 PM
Tmr can? 4:24 PM
Tomorrow what time? 4:25 PM
7pm? 4:25 PM
Earlier possible? | Can | Okay
Type your message here

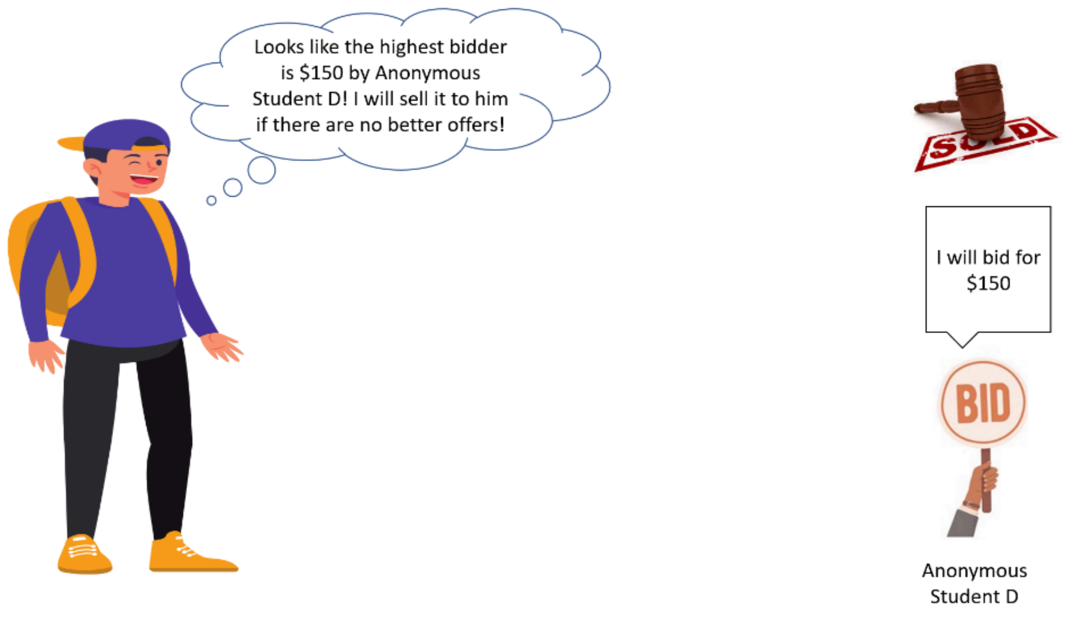
IPhone 7 Used for 2 years but pristine
S$2,000
Archive Chat | Leave Feedback
Okie 4:33 PM
Nice! 4:34 PM
How do we deal? 4:34 PM
Bank transfer 4:34 PM
Is it POSB? 4:34 PM
Or dbs paylah? 4:34 PM
Can | POSB.
Type your message here



Sellers can upload a photo and the relevant details of their item and set a starting bid of the item. Any interested buyers who search for the item can make a bid higher or equal than the starting bid for the item.



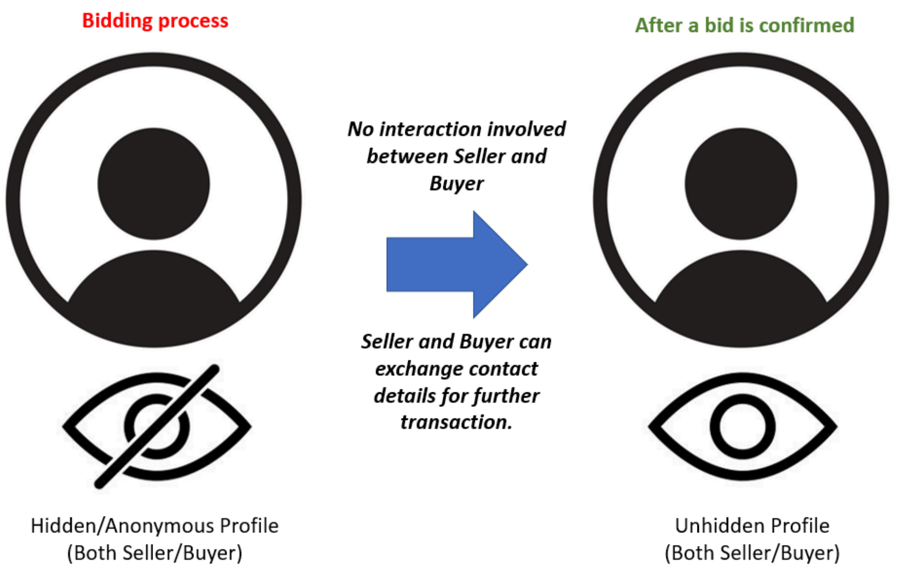
If the Seller is satisfied with the price bid by the item, he or she can accept the bid offered. Conversely, If a buyer wishes to outbid other bidders, he or she can buyout the item.

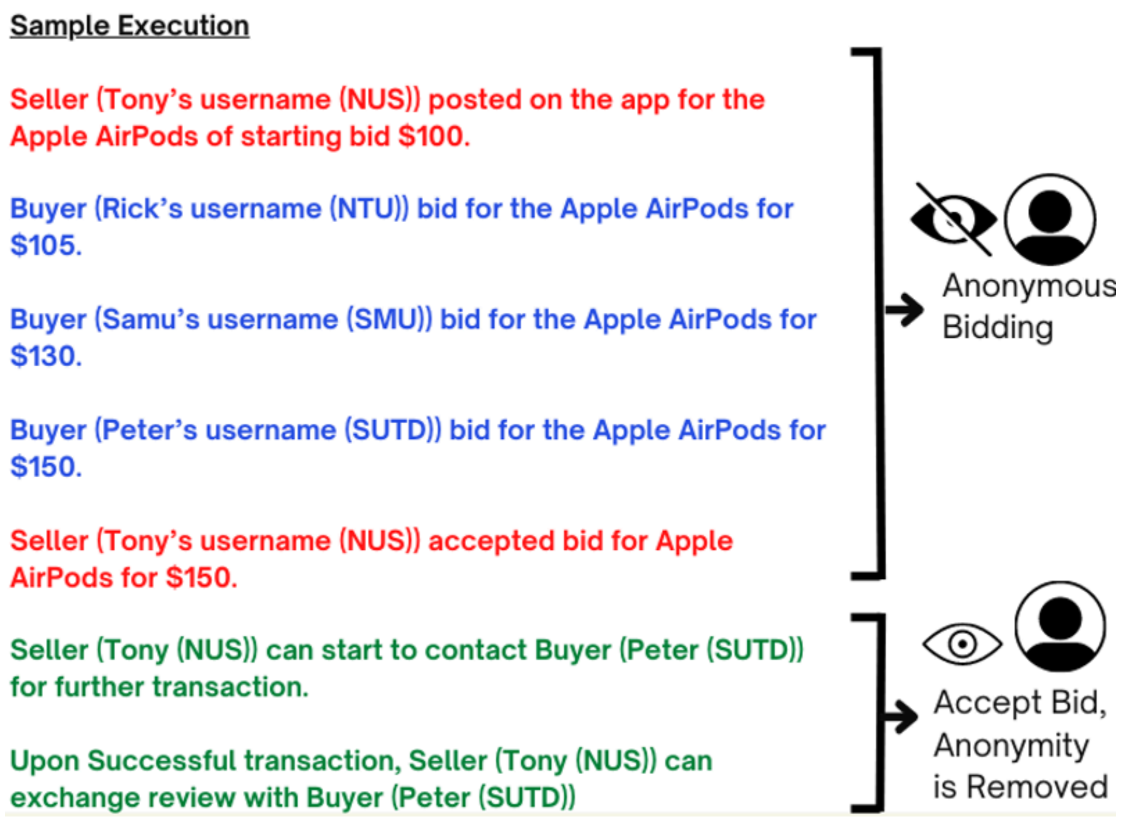

Looks like the highest bidder is $150 by Anonymous Student D! I will sell it to him if there are no better offers!
SOLD
I will bid for $150
BID
Anonymous Student D

Once the bid is successfully accepted, seller and buyer can start to interact and exchange contact information with each other for the transaction to occur. Upon successful transaction, both seller and buyer can leave a review for each other.

Hi! I am Tony from NUS, thanks for submitting your bid of $150 for the Apple AirPods!
Hi! I am Peter from SUTD, thanks for accepting my bid of $150.
Where should we meet to proceed our transaction?
No worries! I can pay a visit to NUS tomorrow and we can meet up for the transaction.

All sellers and buyers will be anonymous (e.g user identity will be hidden) and will have no interaction during the bidding of the item. When the final bid is accepted, only the seller and buyer will be able to contact each other. This anonymous bidding system is beneficial as it helps to maintain privacy and confidentiality of the user's contact information and bidding information.


Bidding process
After a bid is confirmed
No interaction involved between Seller and Buyer
Seller and Buyer can exchange contact details for further transaction.
Hidden/Anonymous Profile (Both Seller/Buyer)
Unhidden Profile (Both Seller/Buyer)



<u>**Sample Execution**</u>

Seller (Tony's username (NUS)) posted on the app for the Apple AirPods of starting bid $100.

Buyer (Rick's username (NTU)) bid for the Apple AirPods for $105.

Buyer (Samu's username (SMU)) bid for the Apple AirPods for $130.

Buyer (Peter's username (SUTD)) bid for the Apple AirPods for $150.

Seller (Tony's username (NUS)) accepted bid for Apple AirPods for $150.

Seller (Tony (NUS)) can start to contact Buyer (Peter (SUTD)) for further transaction.

Upon Successful transaction, Seller (Tony (NUS)) can exchange review with Buyer (Peter (SUTD))

This motivates us to come up with an idea called 'StuBid', also known as 'Student Bidding' which allows students from all universities to buy or sell unwanted items online through an anonymous bidding system. We believe that it will be able to provide much needed convenience, assurance and freedom for students who wish to sell/buy items on an online platform.

StuBid

Auction, Anytime, Anywhere

<u>Aim</u>

We aim to create a one-stop online marketplace, a mobile application which allows students from all Singapore Universities to conveniently buy or sell unwanted items online through an anonymous bidding system, maximizing the benefits gained by both the seller and buyer.

Anonymous Auction

What is an Anonymous Auction?

An anonymous auction is a system of buying and selling products or services by offering them for bidding (allowing people to bid and selling to the highest bidder). All sellers and buyers will be anonymous (e.g user identity will be hidden) to ensure fairness and confidentiality of the auction. The bidders will compete against each other, with each subsequent bid being higher than the previous bid. Once an item is placed for sale and published online, the auctioneer will list down the minimum starting price on the product to attract any potential bidders. The price increases each time someone makes a new, higher bid until finally, no other bidders are willing to offer more than the most recent bid, and the highest bidder takes the item. An auction is considered completed when the vendor accepts the highest bid offered or if the buyer decides to buyout (outbid all other users) for the product. The winning bidder/buyer and the seller hidden profile information will be revealed to both parties for follow-up transaction upon successful auction.

Why Anonymous Auction instead of Normal Auction?

The purpose of making the auction system anonymous is to ensure privacy and confidentiality of user's information and bidding details. This helps to prevent any judgement, discrimination or exclusion of users in any auctions. (For example, a seller may not want to sell/accept a bid for a particular buyer as he dislikes the person OR a buyer may not wish to bid for an item as he dislikes the seller). As a result, this could significantly affect seller/buyer decisions when selling/bidding a product. Therefore, by implementing an anonymous auction, this will ensure fairness and equality for every user during the bidding phase.

How does our idea reduce/prevent any online scams and frauds?

As our target audience are focusing on University Students, users are required to register and login using their own respective university email address. (For example, NUS students are required to register using E1234567@nus.edu.sg). Since every student email is unique to each user, it is easy to identify and track users' real identity based on their student email account if they perform any prohibited actions such as scams (as compared to a gmail account where one user could create multiple new different accounts to register which could be difficult to track). Users may thus be more cautious when using the app and avoid breaking any rules, thereby reducing the risk of online scams and frauds.

User Research and Observations

Target Audience and Needs

University Students

- Students who want to get the best price for their products when selling their items online but are not sure on the ways to approach buyers.
- Students who want to buy items online at their preferred budget or price range but their freedom is restricted as sellers do not want to negotiate.
- Students who encounter scams/low-balling/frauds on products online but are facing problems on tracking down the identity of the perpetrator.
- Students who want to sell or buy items online but want to keep their personal information confidential in the process.
- Students who only wish to sell or buy items to other students and not the general public but do not have a catered platform to do so.

As of now, the table below shows the universities that our app is currently targeted towards. Only current existing students or alumnus who have an official email account from the respective universities are able to benefit from our application.

University
National University of Singapore (NUS)
Nanyang Technological University (NTU)
Singapore Management University (SMU)
Singapore Institute of Technology (SIT)
Singapore University of Technology & Design (SUTD)
Singapore University of Social Sciences (SUSS)

Competitors Research

By conducting competitors research, we can gain an understanding of our competitors' strengths and weaknesses in relation to our solution, as well as find a gap in the market. Moreover, it would assist us in brainstorming, incorporating and improving our ideas by analyzing the solutions of our competitors.

Carousell (https://www.carousell.sg/)	
Purpose	Carousell is a multi platform-classified marketplace for people to buy and sell secondhand goods.
Mission	Carousell inspires every person in the world to start selling and buying to make more possible for one another, on a global scale.
Target Audience	General Public
Notable Features	User Reviews, Chat Function, Like Products, “Recommended for you” Page, etc
Benefits	Fast and easy item publication, better product quality, wide product range to search and filter.
Weakness	Carousell recorded the highest percentile of about 37% among other digital e-commerce platforms such as Lazada or Shopee in 2021 (https://www.channelnewsasia.com/singapore/crime-levels-scams-rise-2021-2501736). Hence, the probability of users experiencing scams and frauds may be higher using Carousell as a marketplace platform. Sellers on Carousell are a mix of those who are selling off second-hand items as well as “shops” that sell a specific type of item. Hence, users who want to buy brand new products may not fancy this platform.

Lazada (https://www.lazada.sg/)	
Purpose	Lazada is a company developing an online shopping and selling marketplace.
Mission	Lazada mission is to bring the world to Southeast Asia and provide a gateway for Southeast Asian brands to reach international consumers
Target Audience	General Public

Notable Features	User Reviews, Convenient Payment Methods, Real-Time Messages, etc
Benefits	Offers a wide range of products, transactions between buyers and sellers are done quickly and easily, have their own delivery service
Weakness	Sometimes the products do not match with what the customers want, they are different from what was displayed. Weakness in the security system with many consumers still experiencing/reporting fraud.

eBay (https://www.ebay.com.sg/)	
Purpose	eBay is the world's online marketplace, a place for buyers and sellers to come together and trade almost anything.
Mission	eBay mission is to provide a global online marketplace where practically anyone can trade practically anything, enabling economic opportunity around the world.
Target Audience	General Public
Notable Features	eBay Wish list, Favourites, Social button on listings, Report Item, Option to auction, etc
Benefits	World’s largest internet marketplace, have their own payment system
Weakness	High fees, No further growth strategy

Learning points from Competitor Research

Inspiration from competitors

1) Carousell offers the ability to make selling as fast and easy as taking a photo, which would not take too long to get our items listed. Moreover, they have many options to search and filter. We intend to replicate this as we believe it will improve the user experience of students who are using the app.
2) Lazada offers the ability for a consumer rating and review system. This provides a new channel that allows customers to feedback their experience on the delivery-related services they received. Therefore, we also intend to build a review and rating system for students to allow them to make better informed decisions on selling or buying items.
3) eBay allows an option of “Auction vs Buy it now”. Buy It Now is the name for a fixed price listing that runs indefinitely until sold. On the other hand, there's an auction, which runs for a specified number of days and lets buyers bid increasing amounts to win the item. Our app is deeply inspired by eBays “Auction vs Buy it now” model and we intend to incorporate it into our application as one of the core features. This allows students to get the best price of their products they are selling, as well as allowing students to bid and buy items and their comfort prices.

Benefits and Uniqueness of our Project

Although we take inspiration and replicate eBay’s “Auction vs Buy it now” model, we also incorporate anonymity into the model. The purpose of this is to maintain the privacy and confidentiality of user’s information, as well as bidding information. This helps to prevent any judgement, discrimination or exclusion of users in any auctions. Sellers can also minimize losses, while buyers will have the freedom to bid on their own budget. This also applies to the review and rating system model from “Lazada”, where the user feedback will be anonymous. As our target audience is also University students, it is also easier to track down identities of students who commit any illegal actions, resulting in lower probability of scams and frauds.

Target Audience Observation and Research

As a team, on different occasions, we have reached out to our friends from different universities to ask about their opinion and feedback regarding their experience with online e-commerce platforms. More specifically, we asked the students about the challenges they faced frequently on these platforms or why they choose to use e-commerce platforms. The main purpose is to find out more on their problems faced and how we could use their insight to improve our product.

Based on the responses from the university students, we discovered that some students may be reluctant to use e-commerce platforms due to :

1. Students are afraid of scams and frauds.
2. Students may not buy or sell items to users who have lack of favourable reviews
3. Students find the platform too complex with many instructions to upload and sell items online.

Learning points from Target Audience Research

With the feedback of our target audience, we have gained a better understanding of our users' needs and requirements, as well as insights into solving key problems they face. Consequently, this motivates us to provide university students with a safer, more private and more conductive platform so that we can reduce the rate of scams and frauds. Additionally, we strive to make our app easy to use, provide a good user experience, and assist university students in making well-informed decisions when buying and selling items.

User Stories

According to all feedback and research on our target audience, as well as competitor research, we have developed the table below to summarize our project needs. With the user-centric approach, each story is mapped to the development process so we can create our product from a user's perspective.

No.	User Story
1	As a student seller who wants to get the best price for my product, I want to auction it off to potential buyers.
2	As a student buyer who is money conscious, I want the freedom to bid according to my budget.
3	As a student user who is intending to purchase/sell a product on an online platform, I want to have a secure student verified platform, so I know that I am selling/buying from a fellow student and during the process of bidding, my confidentiality will not be breached.
4	As a student user who is afraid of scams on second hand products, I wish to see the previous price history for similar products, so I can make better informed decisions to buy/sell the products.
5	As a student user who wants to be notified of the latest change in bid prices on the products I am selling/buying, I wish to be notified on my account or email for change in bids.
6	As a student user, I should be able to modify my registration and profile details (e.g. edit full name, change password) later.
7	As a potential buyer/seller, I wish to see reviews on another seller/buyer I am dealing with so that I can make more informed decisions. Adding to that, if I am ever denied a sale after guarantee, I wish to report the irresponsible buyer/seller.
8	As a student buyer, I wish to be able to understand more details about the product I’m interested in by asking and receiving a response from a seller (FAQ system).
9	As a potential buyer/seller, I wish to have a bidding timer in place such that at the end of the bidding duration, whoever is the highest bidder will be the winner of the product in place.

Development Phase

Project Scope

The scope of our orbital project [StuBid] can be divided into 6 separate sections as shown below in the following tags:

[Structure/Foundation] (Milestone 1):
Software Architecture Diagrams, UI/UX Wireframes, Activity Diagrams (Flow Chart), Database Diagrams, Technology Stack and Considerations

[Set-up] (Milestone 2):
Splashscreen, Welcome Slides, User Registration and Login, Password Reset

[Main/Core] (Milestone 2 and 3):
Homepage, Anonymous Auction, Initial/Current/Final Bidding States of Product listing, My Account, Notifications, Review System, Bid Duration Mechanism

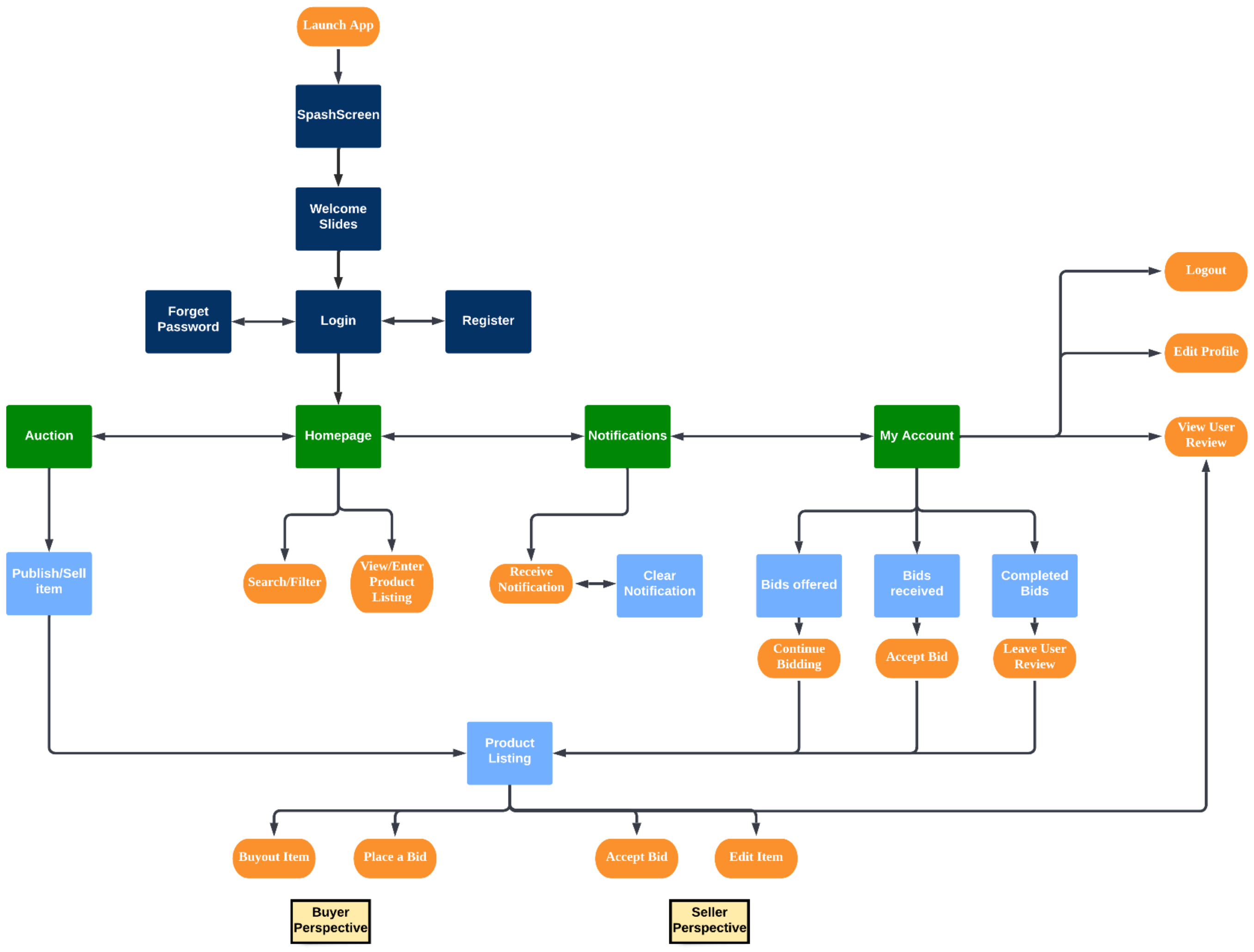
[Further Extensions]:
Item Price History, Reporting system, FAQ system

[In Progress]:
Current Progress for a specific feature.

[Completed]:
Completed production for a specific feature.

User Sitemap

In order to better visualize how users will navigate through our mobile application based on their role (buyer/bidder or seller), we have designed a user sitemap. Additionally, it will serve as a general overview of how our app will operate for those who read our documentation.

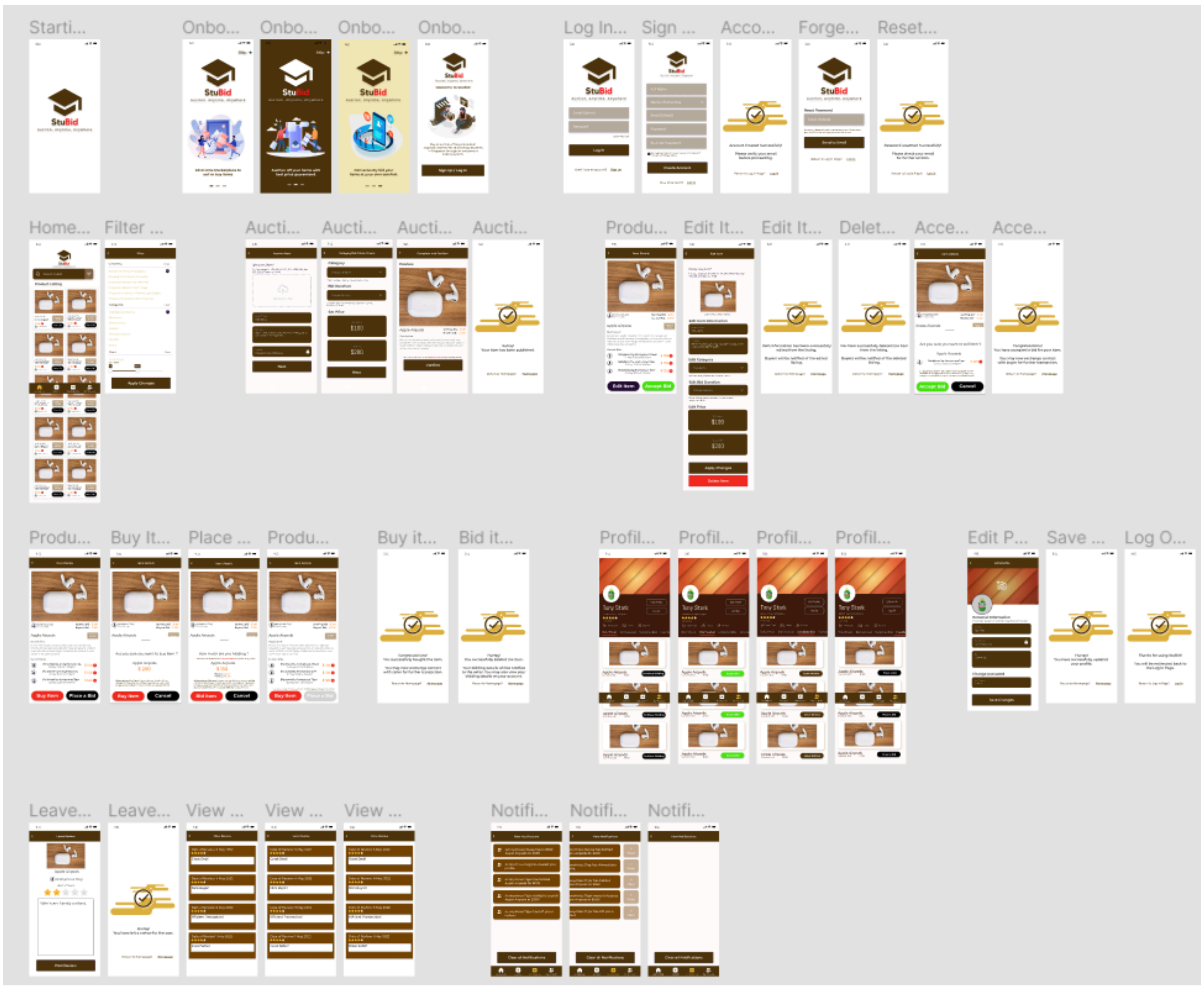


You may visit the link below to access the User Sitemap of our app [StuBid]:
https://lucid.app/documents/view/da2fbad4-9a96-4455-a12b-39aab2ab5d6c

Wireframe (UI/UX)

For overall UI/UX experience, we decided to use Figma as a design tool to create the interfaces for our mobile application. The purpose of using Figma as our primary designing platform is because of several benefits as listed below:

1. Show how screens flow between each other
2. Establish what information should be displayed on a given page
3. Figma prototype can demonstrate how our app will function and look.

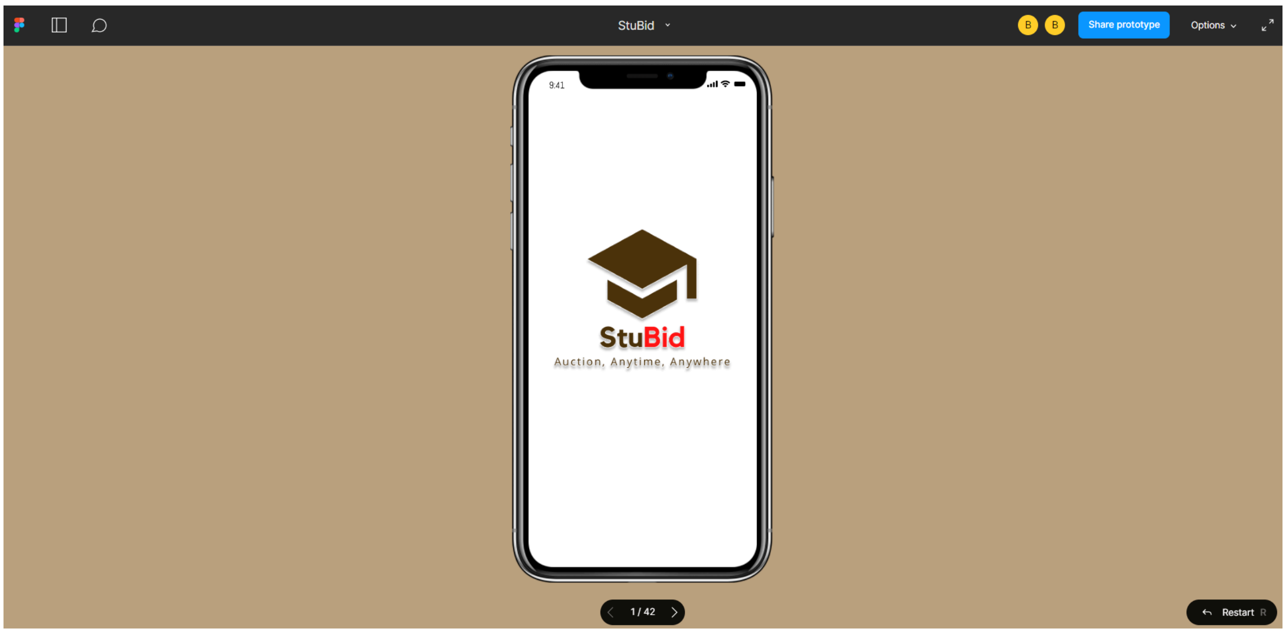


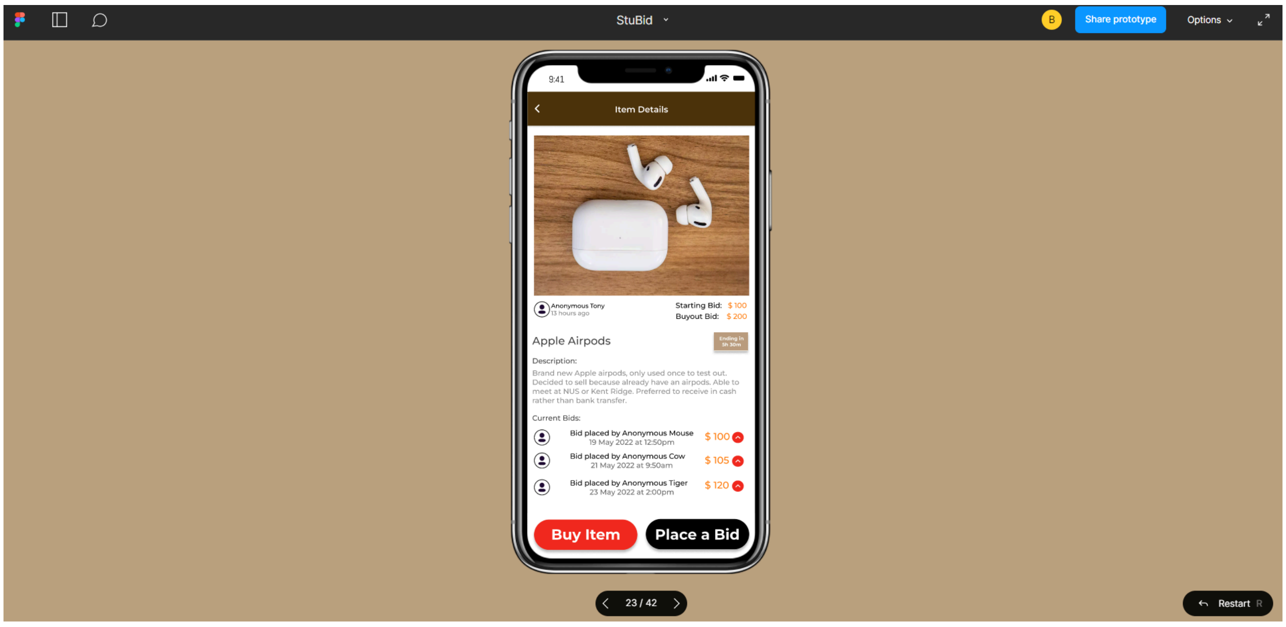
You may visit the link below to access the wireframe/design of our app [StuBid]:
https://www.figma.com/file/ycRVCcj6LPQZd1haL2gPvv/StuBid?node-id=0%3A1

Note: Our Figma prototype is used as a guideline for our app development purposes. The actual design and layout implementation may be subjected to change from the proposed design.

Interactive Wireframe Mockup

Figma was used not only for designing our wireframes, but also for prototyping an interactive version for our application. Prototyping in Figma allows us to explore user interaction with our designs through interactive flows. Moreover, users can gain a sense of how our app will work on a mobile device by previewing how it interacts and flows.





Anonymous Tony
13 hours ago

Starting Bid: $ 100
Buyout Bid: $ 200

Apple Airpods

Ending in 5h 30m

Description:
Brand new Apple airpods, only used once to test out. Decided to sell because already have an airpods. Able to meet at NUS or Kent Ridge. Preferred to receive in cash rather than bank transfer.

Current Bids:

Bid placed by Anonymous Mouse
19 May 2022 at 12:50pm
$ 100

Bid placed by Anonymous Cow
21 May 2022 at 9:50am
$ 105

Bid placed by Anonymous Tiger
23 May 2022 at 2:00pm
$ 120

You may visit the link below to access the Figma Prototype of our app [StuBid]:
https://www.figma.com/proto/ycRVCcj6LPQZd1haL2gPvv/StuBid?node-id=1%3A6&scaling=scale-down&page-id=0%3A1&starting-point-node-id=1%3A6

Technology Stack and Considerations

For our technical design implementations, we would consider three main aspects which includes the Frontend, Backend and CLI. We would list all possible pros and cons for each aspect and make a decision on why we decided on this technical design.

Frontend

There are two technical design/implementation options that we have considered for our frontend which include:

1. React Native
2. Flutter

Using React Native

For our technical design, we have considered React Native Framework for our Frontend Designs.

Pros	**Cons**
● Larger Development Community ● Code reuse and pre-developed components ● Ready-made solutions and variety of libraries available ● Apps in React-native looks exactly the same in both IOS and Android (Cross-Platform) ● Developers like us do not need to build blocks in Java/Kotlin/Swift etc. ● We have prior experience with ReactJS/Javascript and since it is very similar to react-native, it is easier for us to get started with.	● Difficult to debug ● Hard to determine right interfaces ● Does not support many out-of-box components. ● Less resistant to system updates

Using Flutter

For our technical design, we have also considered Flutter Framework for our Frontend Designs.

Pros	Cons
● Rich in Widgets ● Easier to use as more resistant to updates ● Interfaces looks amazing ● Requires less testing ● Easier to debug ● Fast and Efficient ● Apps in React-native looks exactly the same in both IOS and Android (Cross-Platform) ● Developers like us do not need to build blocks in Java/Kotlin/Swift etc.	● Development Community is small but are growing ● Also has a limited set of tools. ● We need to learn Dart, which is not a popular language and we may need to learn from scratch if we wish to use it in our implementation

By weighing the pros and cons, we decided to use **React Native** as our frontend framework for our Orbital Project as we have sufficient experience in Javascript/React which allow us to learn the framework at a faster pace. If we have more time, we could possibly work on using Flutter as our frontend to develop our mobile application.

Backend

There are two technical design/implementation options that we have considered for our backend which include:

1. REST API
2. Firebase

<u>Create an REST API that connects with the database</u>

HTTPS Requests can be made from the app, which will allow the app to connect to the database.

Pros	Cons
● Will allow more functionality and control. ● We own the API, so no overhead charges. ● We are able to learn more about the Backend Functionality.	● Time is constraint and it will take a long time setting up the API. ● Building an API server is new to both of us. ● Makes setting the Environment for the app more complicated. ● Routing for the front end also becomes more complicated as not it requires constant requesting and responding. ● Will need to set up our own authentication, which again requires a lot of time and knowledge.

Use Firebase's Authentication, Cloud Firestore and Storage

In our technical design, we have considered Firebase's server and database to store app and user data.

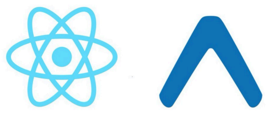
Pros	Cons
● Both of us are experienced in Firebase programming. ● Firebase allows Authentication, admin functionality and cloud storage for data and objects (such as images or files). ● Firebase is easy to use and lightweight, perfect for a prototype app such as ours. ● Firebase database allows for subclassing and stores data as documents (Does not require us to learn a secondary language to retrieve or input data in the database). ● It is easier for Frontend routing as Firebase can be integrated in the Frontend. ● It works well with React Native and Expo, there is a lot of	● Allows only 10k free writes and read of the database ● Storage is limited ● We are restricted in terms of functionality, as we do not own the database or API server

documentation connecting these components and thus we can debug easily.	

By weighing the pros and cons, we decided to use **Firebase** as our backend for our Orbital Project due to the short time period of 3 months. If we have more time, we could possibly work on using REST API as our backend to develop our mobile application.

Command Line Interface (CLI)

After deciding to use React Native for our Frontend development, there are two technical design/implementation options that we have considered for our Command Line Interfaces (CLI):


React CLI vs Expo

1. React Native CLI
2. Expo CLI

<u>With React Native CLI</u>

In our technical design, we have considered React Native CLI for our app development environment.

Pros	**Cons**
• Can integrate native code into application (Java/Swift/Kotlin etc) • More suitable for experienced native users who wants a wide access of components • Third-party plugin support • Modular architecture • Simplified UI	• Require the correct equipment in order to compile app (.apk, .ipa) -> IOS require MAC/X-Code while Android require Android Studio. • Need to deal with complex configurations/settings/signatures/certificates, etc • More time-consuming to set-up/deploy as compared to Expo CLI • Require basic knowledge of IOS and Android

With Expo CLI

In our technical design, we have also considered Expo CLI for our app development environment.

Pros	Cons
• Easier and Faster to develop applications • Enable us to preview our application in real-time while developing • Allow cross-platform mobile development without needing any equipment. • Do not have to install anything in order to start developing mobile applications • Less time-consuming to set-up/deploy as compared to React Native CLI	• May not be suitable for experienced native users who want a wide access to components. • Not all expo modules are compatible with React Native. • Difficult to integrate native code into application (Java/Swift/Kotlin etc) • Some IOS and Android API are not available.

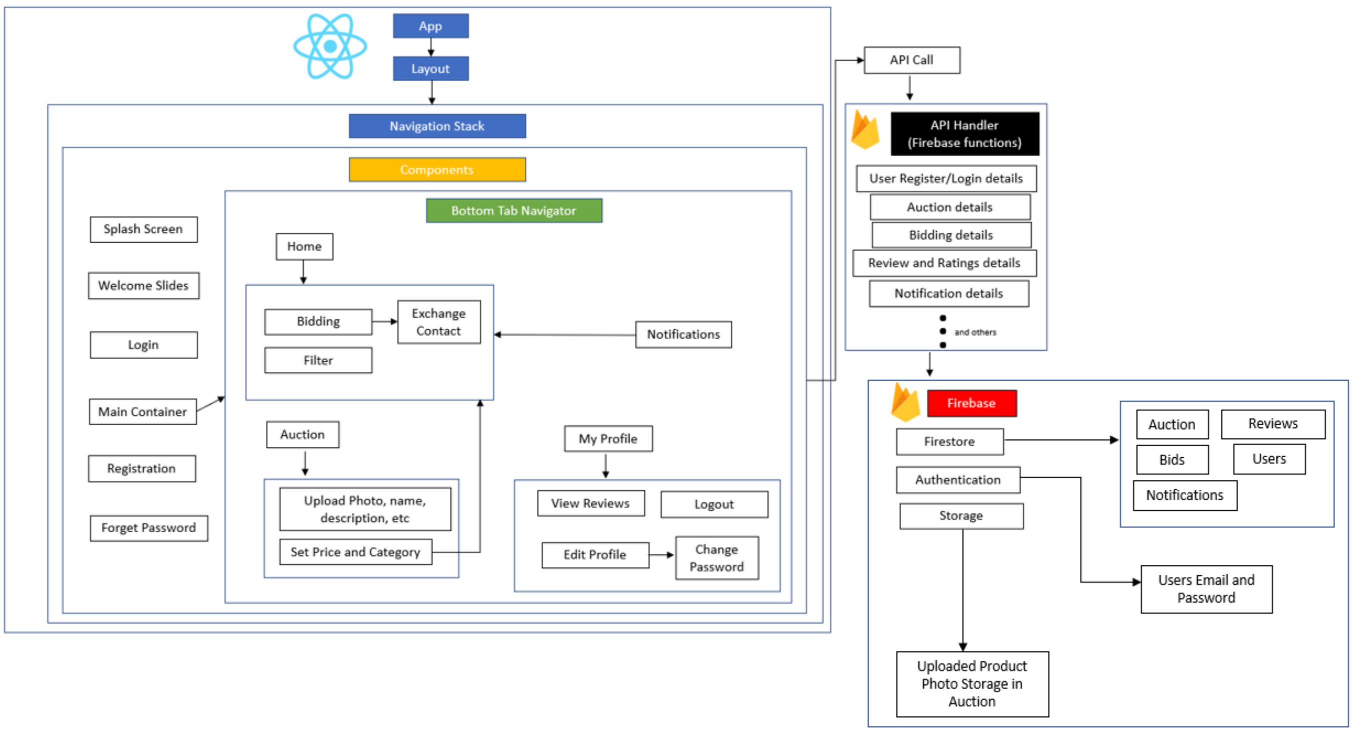
By weighing the pros and cons, we decided to use **Expo** as our Command Line Interface (CLI) for our Orbital Project since it is easier/faster to set-up and deploy in a short time frame of 3 months. Moreover, we are also not requiring complex libraries and API for our mobile app, hence, Expo CLI is our preferred development environment.

Proposed Tech Stack

Technology	Purpose
React Native	Front-end framework.
Firebase	Back-end platform (Utilising Cloud Storage, Authentication and Cloud Firestore database).

Software Architecture Diagram

Our main tech stack consists of React Native (Front-end framework) and Firebase (Back-end platform). Our app is also broken down into different containers and components. All components inside the container are stateful (keep track of data), while containers outside components are stateless (does not store any data). To provide us a way for our app to transition between screens, we utilize the React Native Stack Navigator. For authentication (e.g register, login, forget password) of users, we use the Firebase Authentication library for a secure validation of users. All product photos are uploaded through Firebase Cloud Storage library. We also utilise Cloud Firestore to let us store, sync, and query data (e.g Bidding information, Ratings details) for our mobile app. The purpose of this Software Architecture design is to provide an overall technical view and context of our application to users like us.



You may visit the link below to access a clearer version of our architecture design [StuBid]: StuBid Software Architecture Design.png

Implementation of React Hooks to our application

As users navigate through StuBid, we also needed a means by which to pass their state between screens. In order to render the correct details on each page (user data, product data, bidding data, etc.), we needed a solution. Then we could decide when our data in the backend needed to be refreshed as well. Having researched and discussed, we chose to use **React Hooks**, which allows us to use state and other React features without writing any code. We greatly benefitted from its synergy with React Native and reduced our troubles.

Software Design Principles

For our software design principles, we have decided to utilise the 5 **SOLID** object-oriented principles to ensure that our designs are easier to understand, maintain, and extend. The 5 principles are shown below:

Principles
Single Responsibility Principle (SRP)
Open Closed Principle (OCP)
Liskov Substitution Principle (LSP)
Interface Segregation Principle (ISP)
Dependency Inversion Principle (DIP)

Design Pattern

For our design patterns, we decided to use the **Container and View** pattern as we found it to be the most efficient way of building features in the react environment.

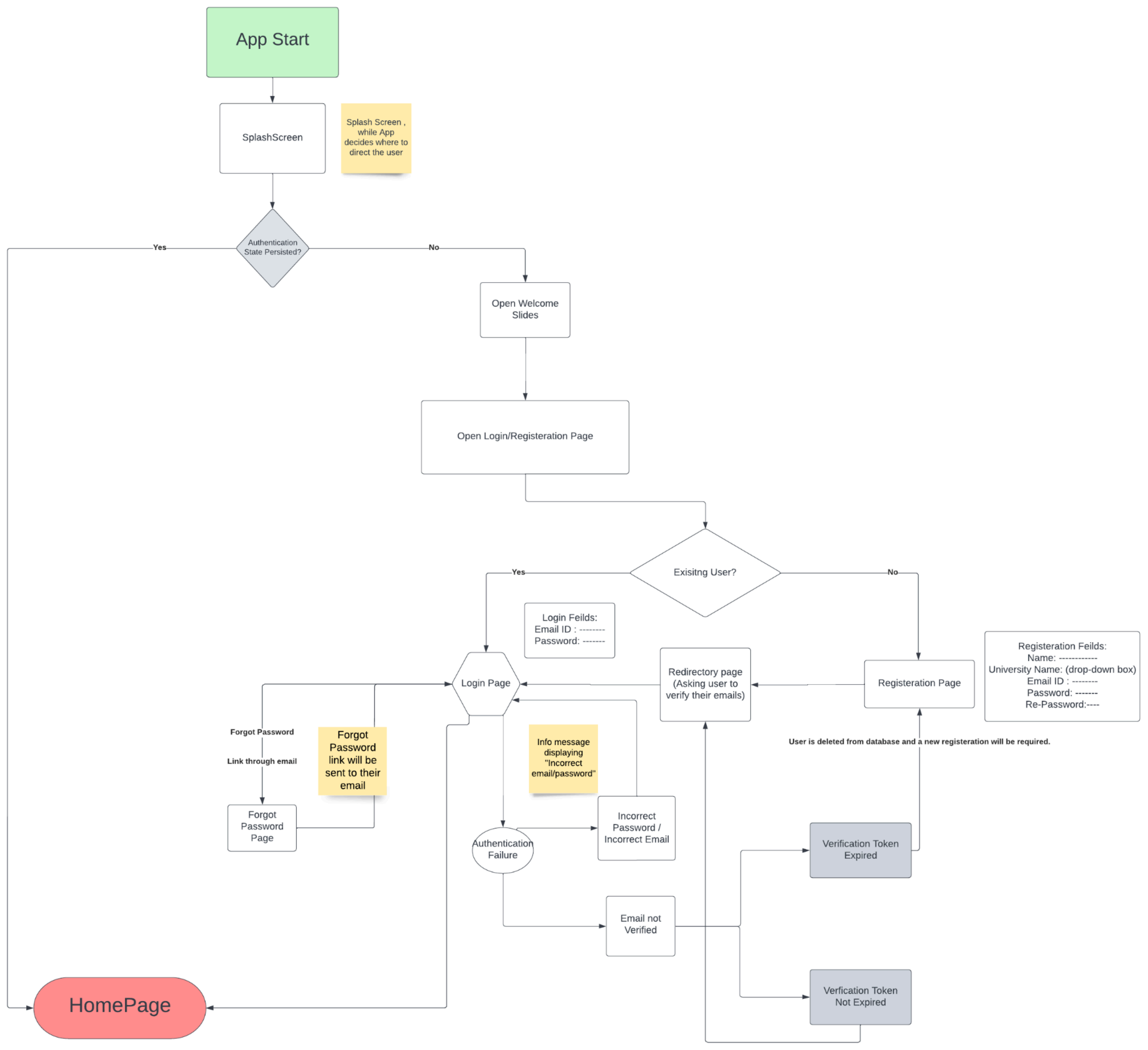
Component	**Purpose**
Container	Screens and features can be accessed through this entry point. ● Data fetching ● Passing required props down to View ● Stateful
View	Only the presentation part is included in it, such as ● The logic behind the user interface/presentation ● The maintenance of more complex components can be simplified by breaking them down into individual components. ● Several components in the presentation are based on props, renders, and contexts. ● The props provided by a presentational component's container or parent are the only sources of data and callbacks for presentational components.

Activity Diagrams

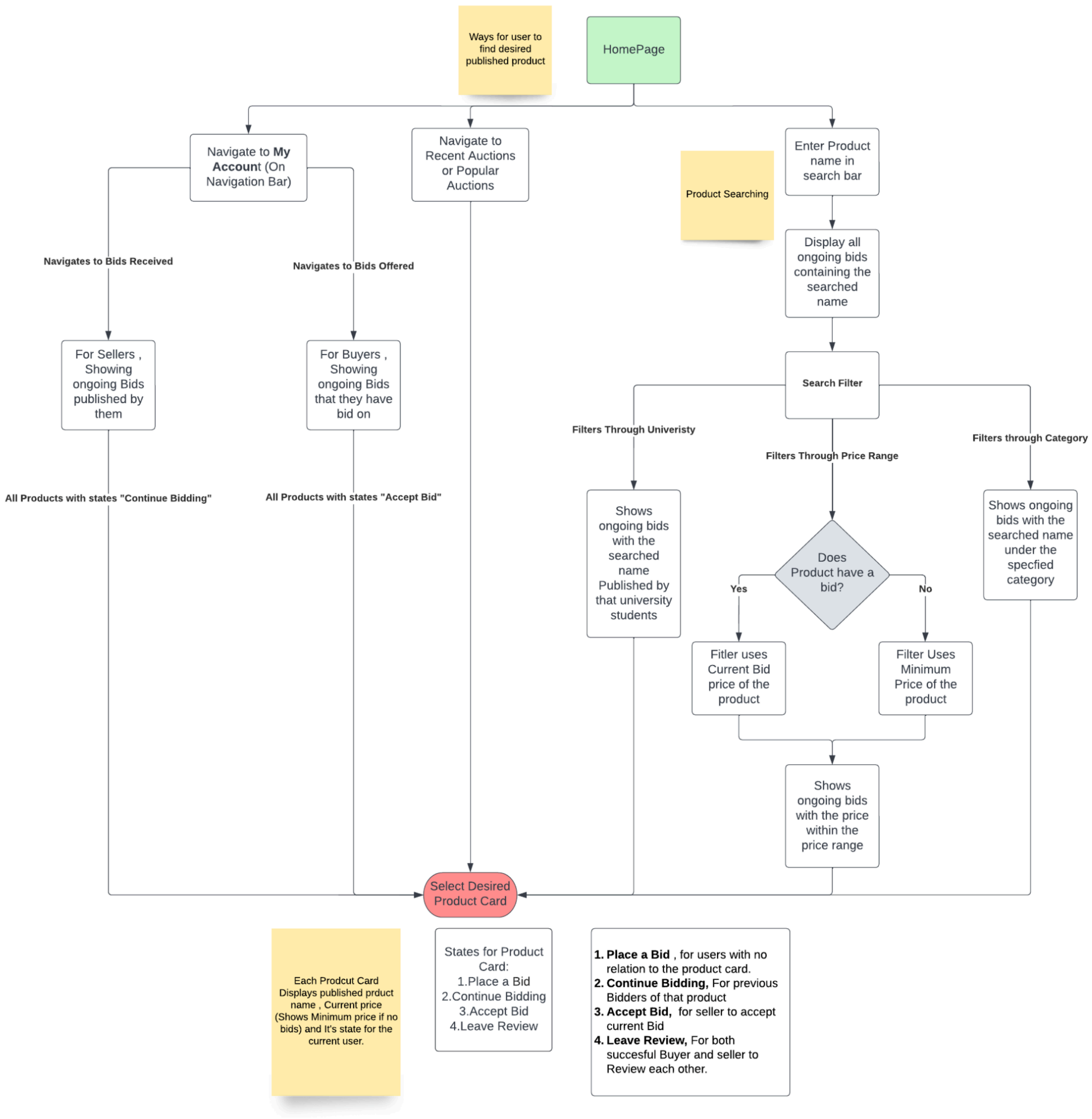
Our Orbital Project [StuBid] has 7 activity diagrams which allow us to explain the key functionality of our application, how the system should behave and brainstorm possible situations/edge cases which may occur in the system. The use cases provided us a way to visualise and define core features to be implemented and resolve any errors that may be encountered.

1) Welcome/Registration/Login
2) Product Searching
3) Auction (Published item to be sold): Initial Bidding State
4) Bidding Process (Seller Perspective): Active Bidding State
5) Bidding Process (Buyer Perspective): Active Bidding State
6) Bid Closure: Final Bidding State
7) User Review System

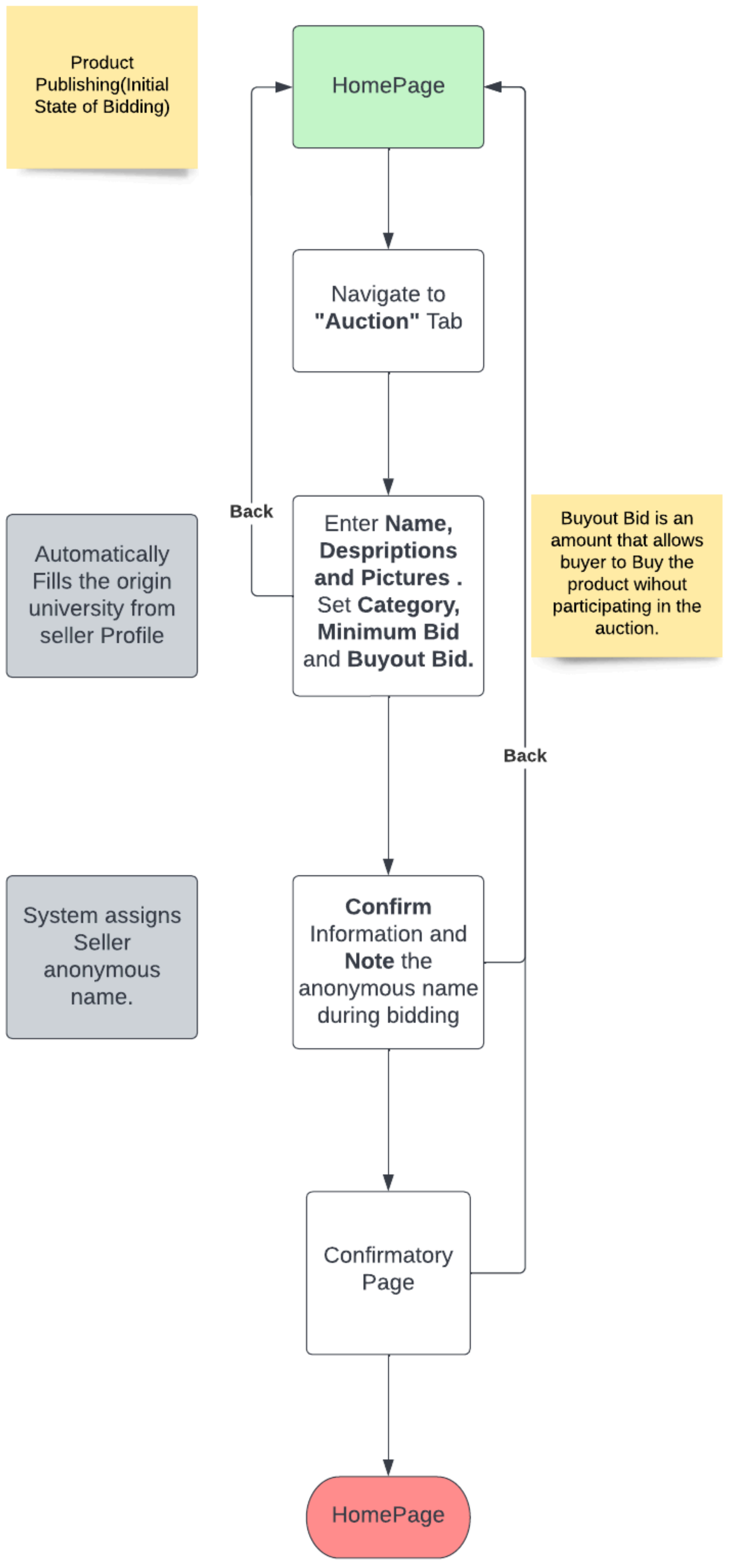
Welcome/Registration/Login


App Start
SplashScreen
Splash Screen , while App decides where to direct the user
Authentication State Persisted?
Yes
No
Open Welcome Slides
Open Login/Registeration Page
Exisitng User?
Yes
No
Login Feilds: Email ID : -------- Password: -------
Login Page
Redirectory page (Asking user to verify their emails)
Registeration Page
Registeration Feilds: Name: ------------ University Name: (drop-down box) Email ID : -------- Password: ------- Re-Password:----
User is deleted from database and a new registeration will be required.
Forgot Password
Link through email
Forgot Password link will be sent to their email
Forgot Password Page
Info message displaying "Incorrect email/password"
Authentication Failure
Incorrect Password / Incorrect Email
Email not Verified
Verification Token Expired
Verfication Token Not Expired
HomePage

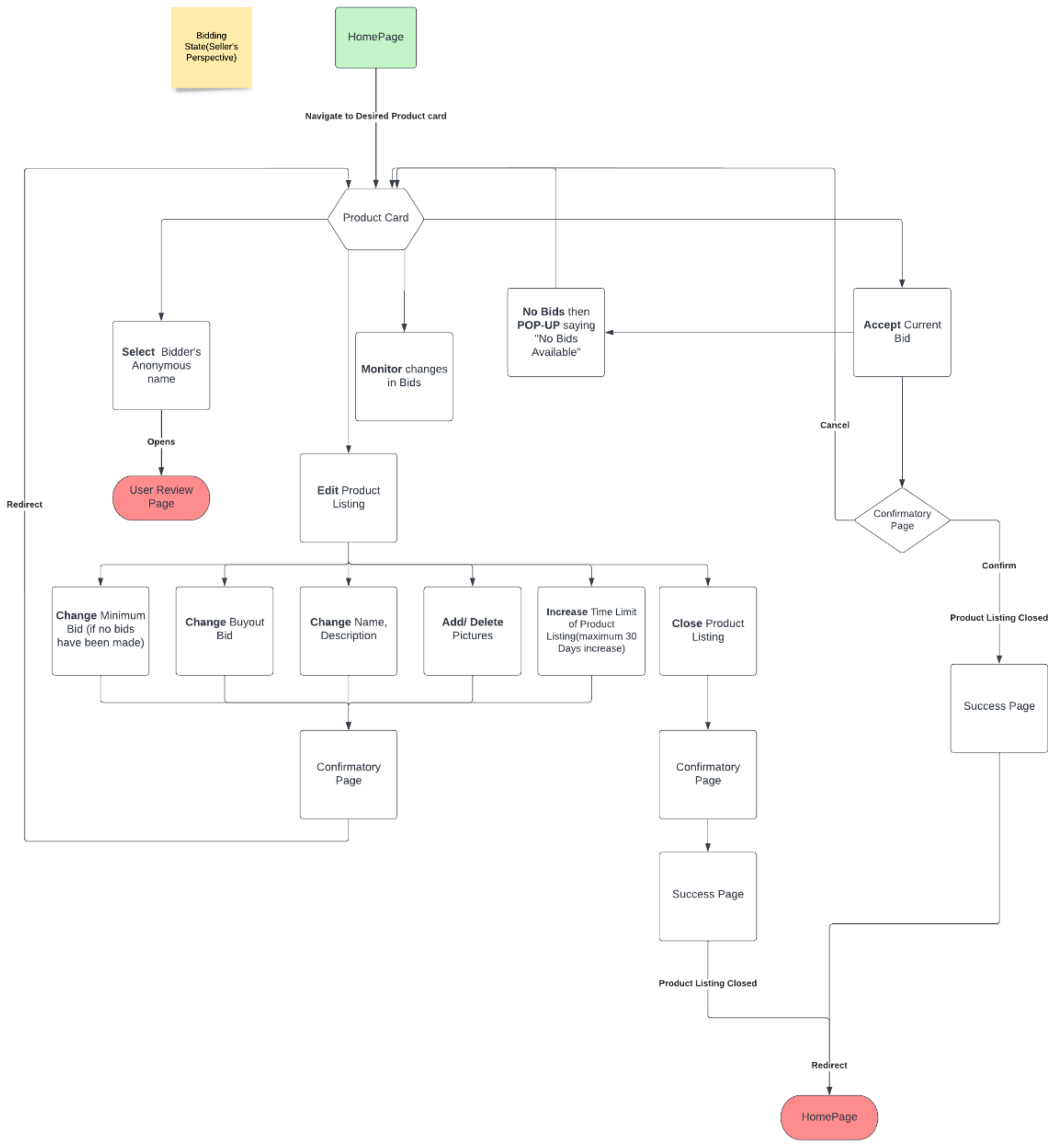
Product Searching


Ways for user to find desired published product
HomePage
Navigate to My Account (On Navigation Bar)
Navigate to Recent Auctions or Popular Auctions
Enter Product name in search bar
Product Searching
Display all ongoing bids containing the searched name
Navigates to Bids Received
Navigates to Bids Offered
For Sellers , Showing ongoing Bids published by them
For Buyers , Showing ongoing Bids that they have bid on
Search Filter
Filters Through Univeristy
Filters Through Price Range
Filters through Category
All Products with states "Continue Bidding"
All Products with states "Accept Bid"
Shows ongoing bids with the searched name Published by that university students
Does Product have a bid?
Yes
No
Shows ongoing bids with the searched name under the specfied category
Fitler uses Current Bid price of the product
Filter Uses Minimum Price of the product
Shows ongoing bids with the price within the price range
Select Desired Product Card
Each Prodcut Card Displays published prduct name , Current price (Shows Minimum price if no bids) and It's state for the current user.
States for Product Card:
1.Place a Bid
2.Continue Bidding
3.Accept Bid
4.Leave Review
1. Place a Bid , for users with no relation to the product card.
2. Continue Bidding, For previous Bidders of that product
3. Accept Bid, for seller to accept current Bid
4. Leave Review, For both succesful Buyer and seller to Review each other.

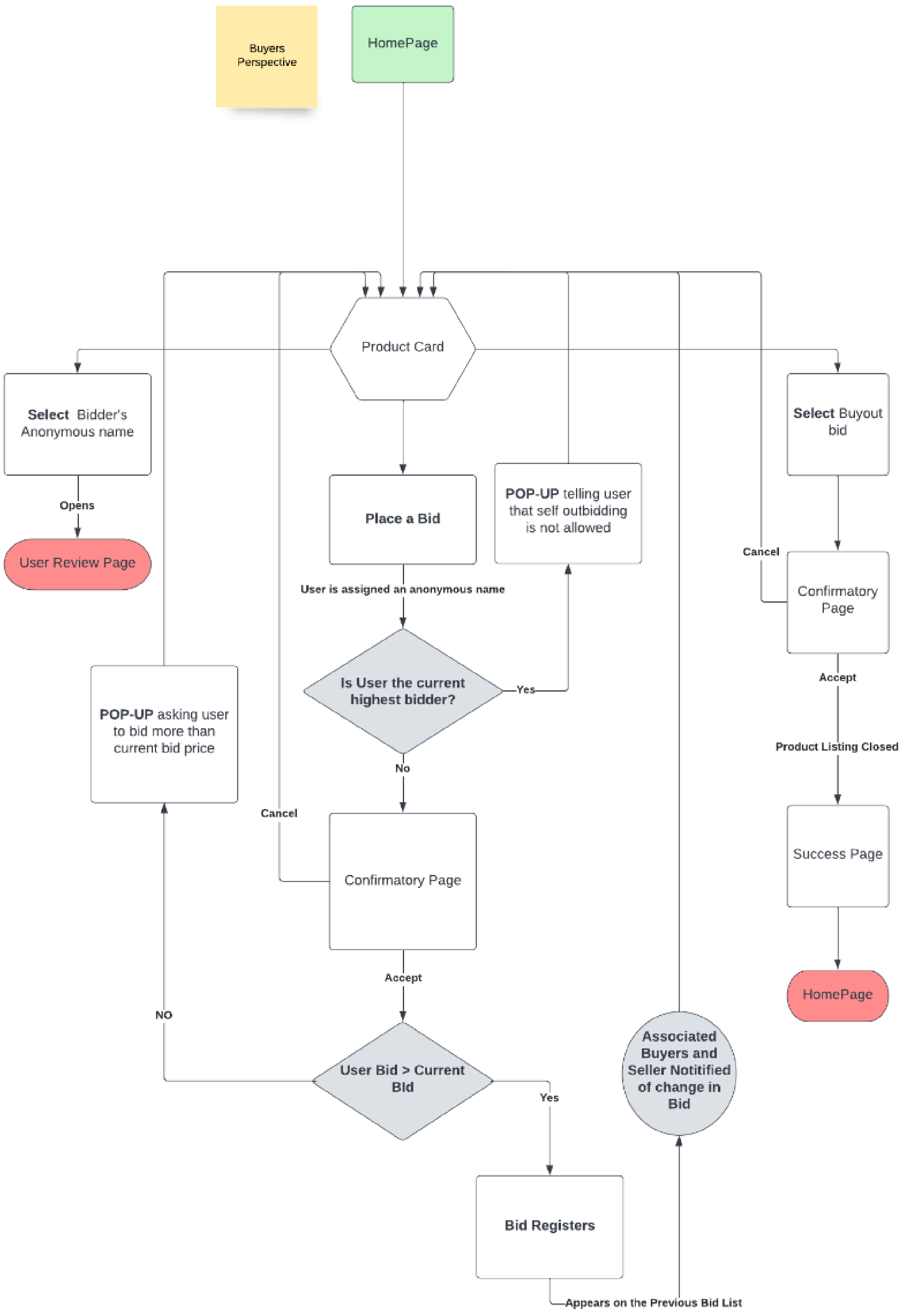
Anonymous Auction (Publishing item to be sold): Initial Bidding State


Product Publishing(Initial State of Bidding)
HomePage
Navigate to "Auction" Tab
Back
Automatically Fills the origin university from seller Profile
Enter Name, Desprriptions and Pictures . Set Category, Minimum Bid and Buyout Bid.
Buyout Bid is an amount that allows buyer to Buy the product wihout participating in the auction.
Back
System assigns Seller anonymous name.
Confirm Information and Note the anonymous name during bidding
Confirmatory Page
HomePage

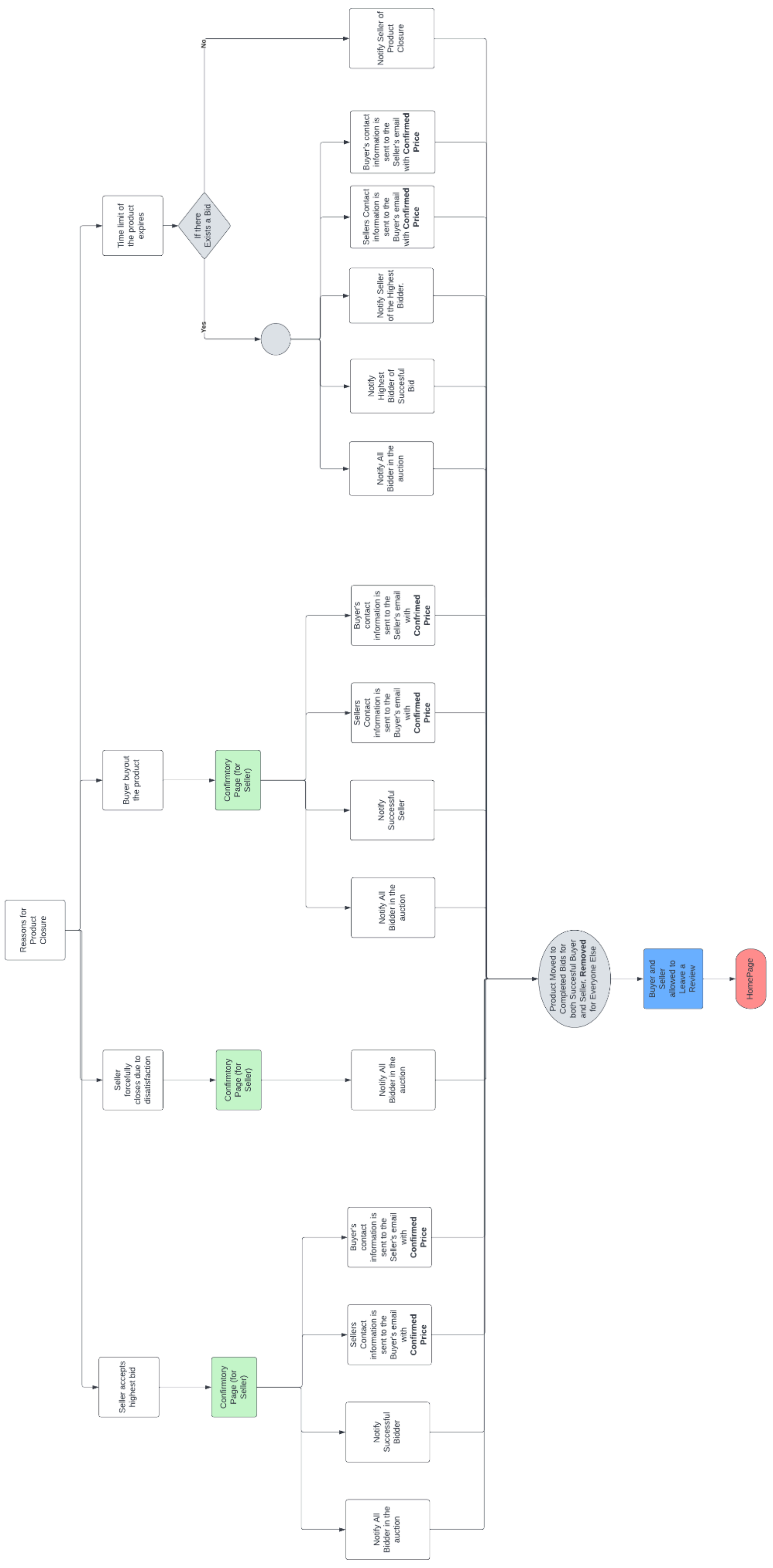
Bidding Process (Seller Perspective): Active Bidding State



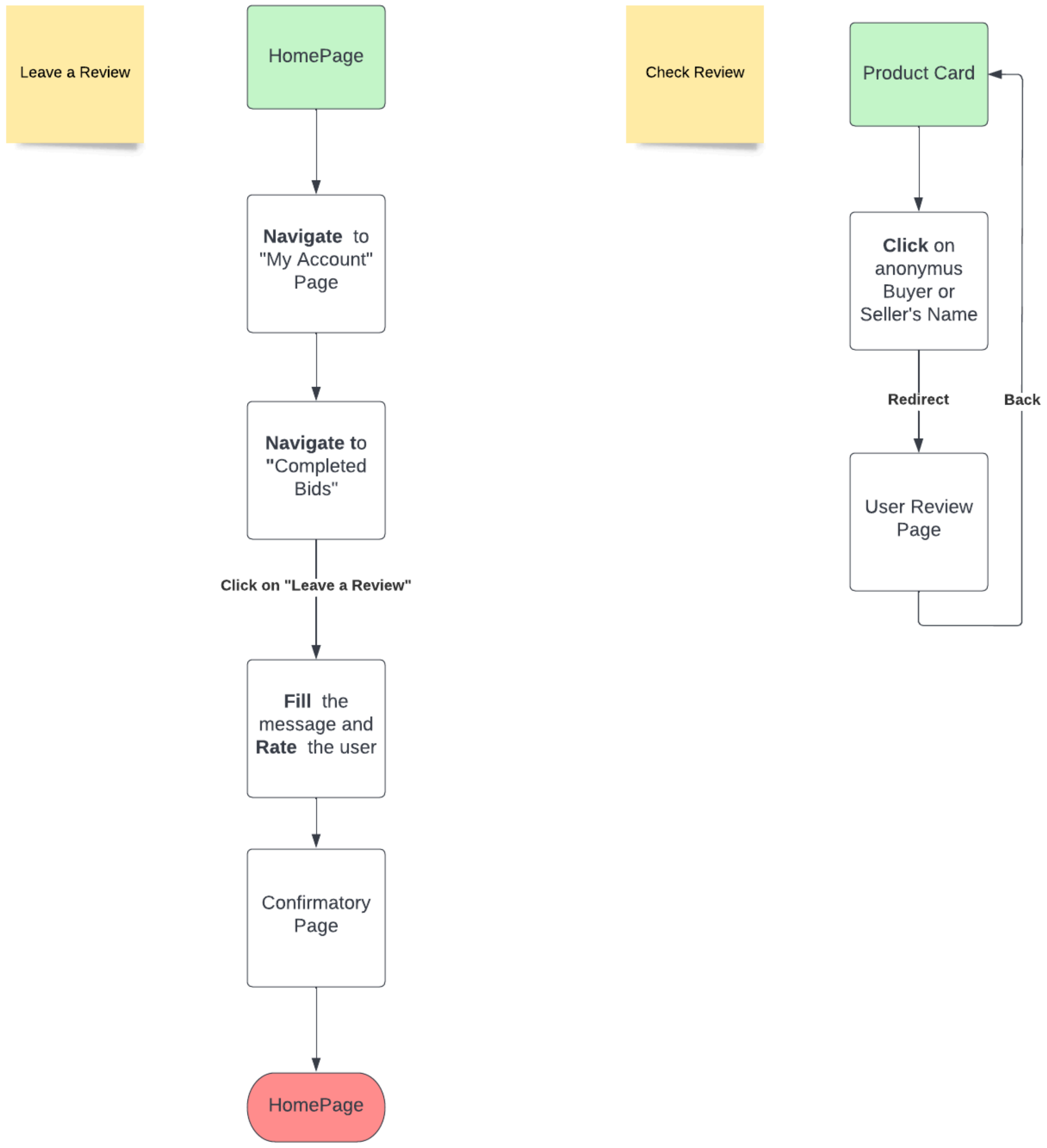
Bidding Process (Buyer Perspective): Active Bidding State


Buyers Perspective
HomePage
Product Card
Select Bidder's Anonymous name
Opens
User Review Page
Place a Bid
POP-UP telling user that self outbidding is not allowed
Select Buyout bid
Cancel
Confirmatory Page
Accept
Product Listing Closed
Success Page
HomePage
User is assigned an anonymous name
Is User the current highest bidder?
Yes
No
POP-UP asking user to bid more than current bid price
Cancel
Confirmatory Page
Accept
NO
User Bid > Current BId
Yes
Associated Buyers and Seller Notitified of change in Bid
Bid Registers
Appears on the Previous Bid List

Bid Closure (Completed Bidding): Final Bidding State

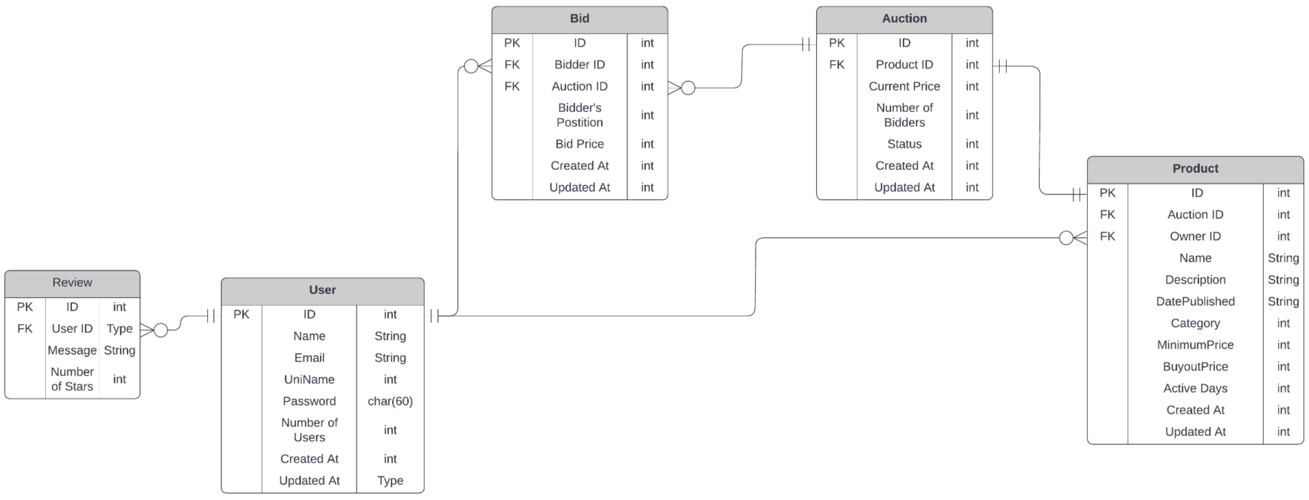

Reasons for Product Closure
Time limit of the product expires
If there Exists a Bid
No
Yes
Notify Seller of Product Closure
Buyer's contact information is sent to the Seller's email with Confirmed Price
Sellers Contact information is sent to the Buyer's email with Confirmed Price
Notify Seller of the Highest Bidder.
Notify Highest Bidder of Succesful Bid
Notify All Bidder in the auction
Buyer buyout the product
Confirmtory Page (for Seller)
Buyer's contact information is sent to the Seller's email with Confrimed Price
Sellers Contact information is sent to the Buyer's email with Confirmed Price
Notify Successful Seller
Notify All Bidder in the auction
Seller forcefully closes due to disatisfaction
Confirmtory Page (for Seller)
Notify All Bidder in the auction
Seller accepts highest bid
Confirmtory Page (for Seller)
Buyer's contact information is sent to the Seller's email with Confirmed Price
Sellers Contact information is sent to the Buyer's email with Confirmed Price
Notify Successful Bidder
Notify All Bidder in the auction
Product Moved to Completed Bids for both Succesful Buyer and Seller, Removed for Everyone Else
Buyer and Seller allowed to Leave a Review
HomePage

User Review System


Leave a Review
HomePage
Navigate to "My Account" Page
Navigate to "Completed Bids"
Click on "Leave a Review"
Fill the message and Rate the user
Confirmatory Page
HomePage
Check Review
Product Card
Click on anonymus Buyer or Seller's Name
Redirect
Back
User Review Page

You may visit the link below to access all the activity diagrams of our app [StuBid]:
https://lucid.app/documents/view/c7481090-8e10-4b6c-bbaa-e60ab06c0ec7

Database Diagrams



Bid		
PK	ID	int
FK	Bidder ID	int
FK	Auction ID	int
	Bidder's Postition	int
	Bid Price	int
	Created At	int
	Updated At	int

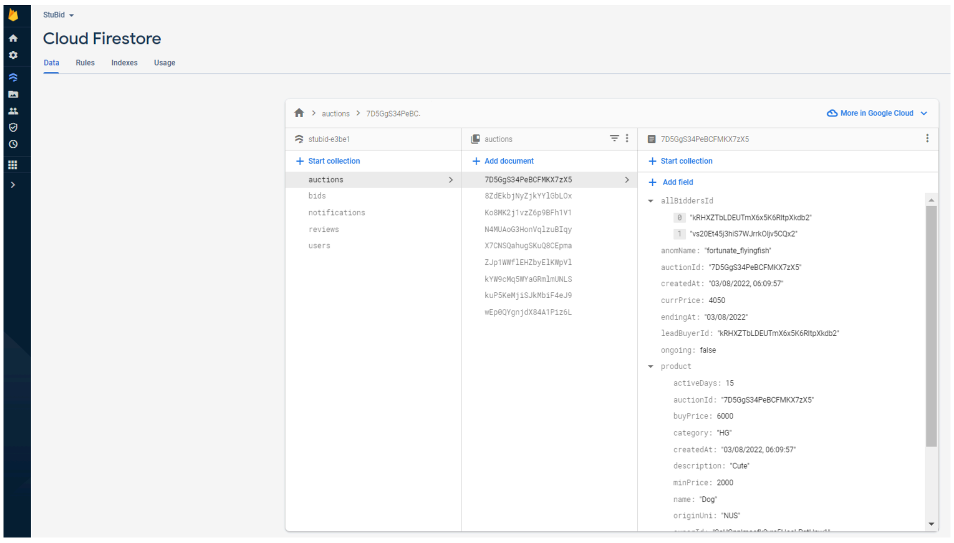
Auction		
PK	ID	int
FK	Product ID	int
	Current Price	int
	Number of Bidders	int
	Status	int
	Created At	int
	Updated At	int

Product		
PK	ID	int
FK	Auction ID	int
FK	Owner ID	int
	Name	String
	Description	String
	DatePublished	String
	Category	int
	MinimumPrice	int
	BuyoutPrice	int
	Active Days	int
	Created At	int
	Updated At	int

Review		
PK	ID	int
FK	User ID	Type
	Message	String
	Number of Stars	int

User		
PK	ID	int
	Name	String
	Email	String
	UniName	int
	Password	char(60)
	Number of Users	int
	Created At	int
	Updated At	Type

You may visit the link below to access the ERD diagram model of our app [StuBid]:
https://lucid.app/documents/view/47955d15-9b77-42dd-b788-74f9bb1c5e29



```
auctions > 7D5GgS34PeBC.

stubid-e3be1        auctions                  7D5GgS34PeBCFMKX7zX5

auctions            7D5GgS34PeBCFMKX7zX5      allBiddersId
bids                8ZdEkbjNyZjkYYlGbLOx        0 "kRHXZTbLDEUTmX6x5K6RItpXkdb2"
notifications       Ko8MK2j1vzZ6p9BFh1V1        1 "vs20Et45j3hiS7WJrrkOljv5CQx2"
reviews             N4MUAoG3HonVqlzuBIqy      anomName: "fortunate_flyingfish"
users               X7CNSQahugSKuQ8CEpma      auctionId: "7D5GgS34PeBCFMKX7zX5"
                    ZJp1WWflEHZbyElKWpVl      createdAt: "03/08/2022, 06:09:57"
                    kYW9cMq5WYaGRmlmUNLS      currPrice: 4050
                    kuP5KeMjiSJkMbiF4eJ9      endingAt: "03/08/2022"
                    wEp0QYgnjdX84A1Piz6L      leadBuyerId: "kRHXZTbLDEUTmX6x5K6RItpXkdb2"
                                              ongoing: false
                                              product
                                                activeDays: 15
                                                auctionId: "7D5GgS34PeBCFMKX7zX5"
                                                buyPrice: 6000
                                                category: "HG"
                                                createdAt: "03/08/2022, 06:09:57"
                                                description: "Cute"
                                                minPrice: 2000
                                                name: "Dog"
                                                originUni: "NUS"
```

For users who wish to view our implementation of Firebase (Backend) database, users can login with the credentials listed below at (https://firebase.google.com/):

Test Account:
Email: stubiduser2122@gmail.com
Password: testuser2122

Features

<u>Overview of Features</u>

Features	**Brief Description**	**Scope**
SplashScreen	The very first screen the user will see that displays while the app is launched.	Set-up
Welcome Slides	Introductory slides which briefly explain what the app is all about for first-time users.	
Login	Users will sign in with their allocated school email address (e.g. tony123@u.nus.edu) and the password they created during registration.	
Register	In the Registration Page, users will fill in their **Full Name**, Select the **University** they are studying at (through a drop down menu), **School Email** and **Password** respectively. <u>Email verification and Authentication:</u> For verification and authentication purposes to confirm if a user is a student, only emails matching the strings after the @ of the respective universities will be accepted during registration.	
Reset Password	In the event where users forget their password, they can opt to reset their password by filling in their respective school emails. Further details will be sent to their emails.	
Homepage	In the homepage, users can 1. Search for products 2. Filter products during searching a. Filter by Universities (e.g. NUS) b. Filter by Categories (e.g. Electronics) c. Filter by Price Range (e.g. $100 - $200) 3. View or participate in biddings of products listed by sellers.	Main/Core
Anonymous Auction (Published item to be sold)	To publish or sell an item for prospective buyers. Fields to be filled up would be 1. Photo of item to be auction 2. Name of the item 3. Description of the item 4. User university email (automatically filled up) 5. Category of item	

	6. Bid duration of the auction 7. Minimum and Maximum price of the item to be auctioned Users will be randomly assigned an anonymous identity.	Main/Core
Product Listing	In the product listing, details filled by the seller prior to the anonymous auction will be shown to the buyer. There will be two perspective in the product listing which will be 1. Seller Perspective a. Edit Item i. Save Changes ii. Delete Item b. Accept Bid c. View Anonymous buyer review 2. Buyer Perspective a. Buyout item b. Place a Bid c. View Anonymous seller review	
Exchange Contact	Successful Auction (Seller and Winning Bidder) will be allowed to exchange contact information. Anonymous Profile will be unhidden and personal details will be revealed to the other party.	
Bid Duration Termination (Mechanism)	As users publish an item to be auctioned, they need to input the bid duration for the item (for e.g. Bid ending in 7 days). If the bid duration ends and user refresh the page, the item will be terminated and there will be two scenarios: 1. There is a leading bidder a. Seller and Buyer will be notified of their successful auction and can exchange contact under profile page. b. All other bidders will be notified of unsuccessful auction 2. There are no bidders a. Only Seller will be notified of unsuccessful auction and can choose to delete listing under profile page.	
My Profile (Account information)	In My Profile, users can 1. View own profile information 2. Edit Profile a. Change Full Name b. Input Handphone (optional) c. Input Personal Bio (optional) d. Change Password 3. View own reviews by other users	

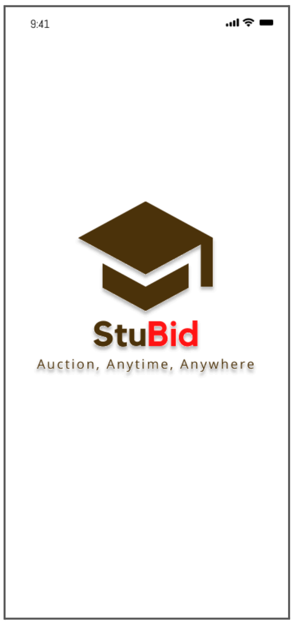
	a. Users can also view other user reviews by tapping on 4. Tab Navigation a. View bids offered (User is Buyer) i. Continue bidding for the item b. View bids received (User is Seller) i. Accept bid for the current highest bidder c. View completed bids (Completed Auctions) i. Leave user review for other party	Main/Core
User Review System	Upon a successful auction (Completed Auction), both parties (Seller and Winning Buyer) will be allowed to exchange reviews between each other. Users will be allowed to leave a rating (1 to 5) and post a comment/feedback for the other party. Clicking on 'Post Review' will navigate the user to the successful review page. Users will be allowed to view their own reviews given by other users in My Profile, as well as viewing other users' reviews by tapping on the anonymous identity in any product listing.	
Notifications	Users can view all the notifications received from the app and clear any notifications if they wish to in the notification page.	

The following page [App Walkthrough] will explain more explicitly/specifically/in-depth on the features and functionality of the app. App Walkthrough is designed to help users better understand how our app functions during deployment.

App Walkthrough

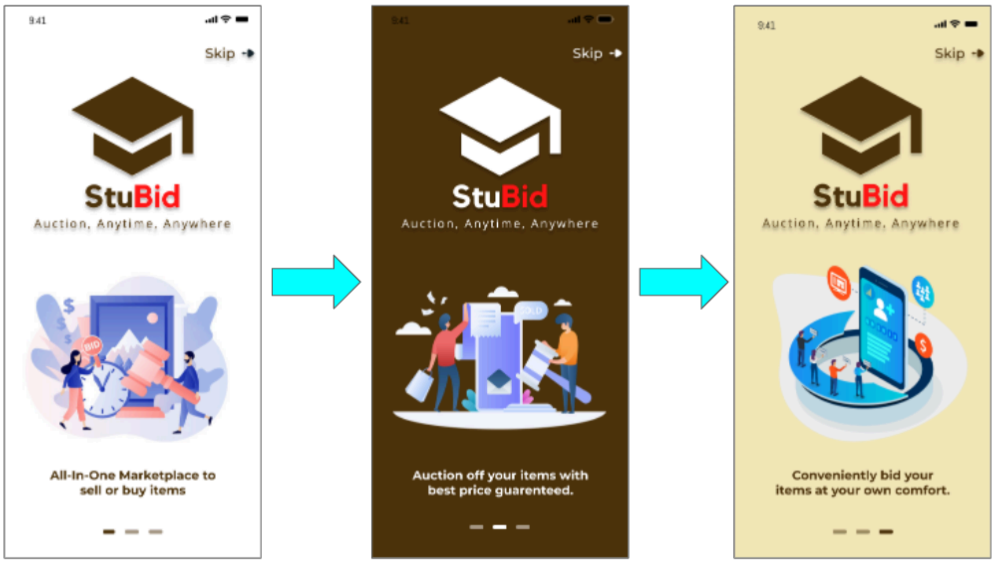
(Refer to "Technical Implementation" under References for more detailed and technical explanation.)

Starting Page **(Completed)**



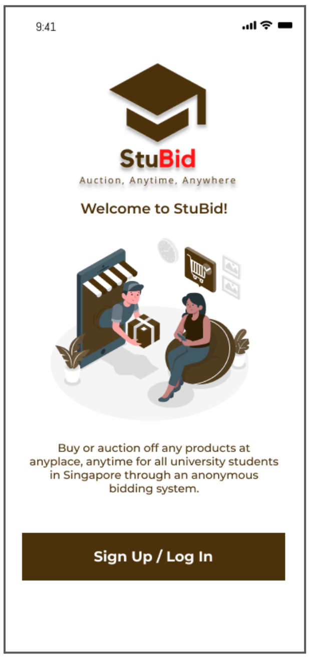
As the user launches the app, a splash screen containing StuBid Logo is the very first screen the user will see that displays while the application or other items are loading.

Introductory Pages **(Completed)**


9:41
Skip
StuBid
Auction, Anytime, Anywhere
All-In-One Marketplace to sell or buy items
Skip
StuBid
Auction, Anytime, Anywhere
Auction off your items with best price guarenteed.
9:41
Skip
StuBid
Auction, Anytime, Anywhere
Conveniently bid your items at your own comfort.

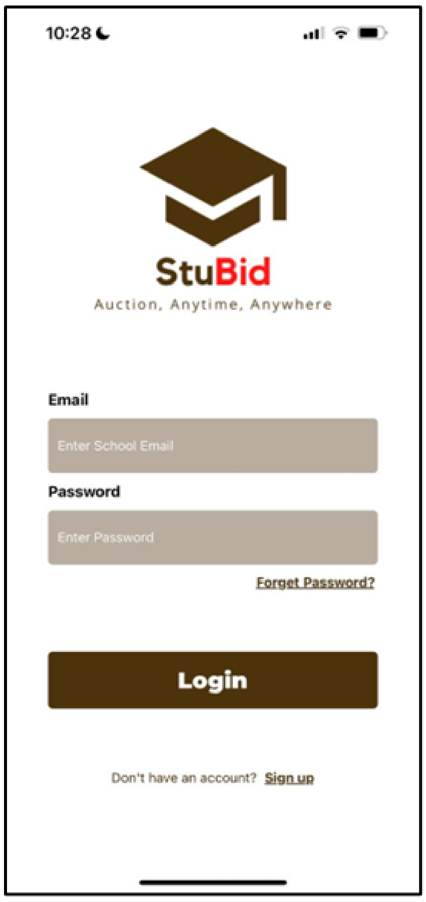
First-time users will be navigated to the Introductory Pages as the app will briefly describe what the app is all about. Users can either choose to swipe through the pages or skip to proceed to the Welcome Page immediately.

Welcome Page **(Completed)**


9:41
StuBid
Auction, Anytime, Anywhere
Welcome to StuBid!
Buy or auction off any products at anyplace, anytime for all university students in Singapore through an anonymous bidding system.
Sign Up / Log In

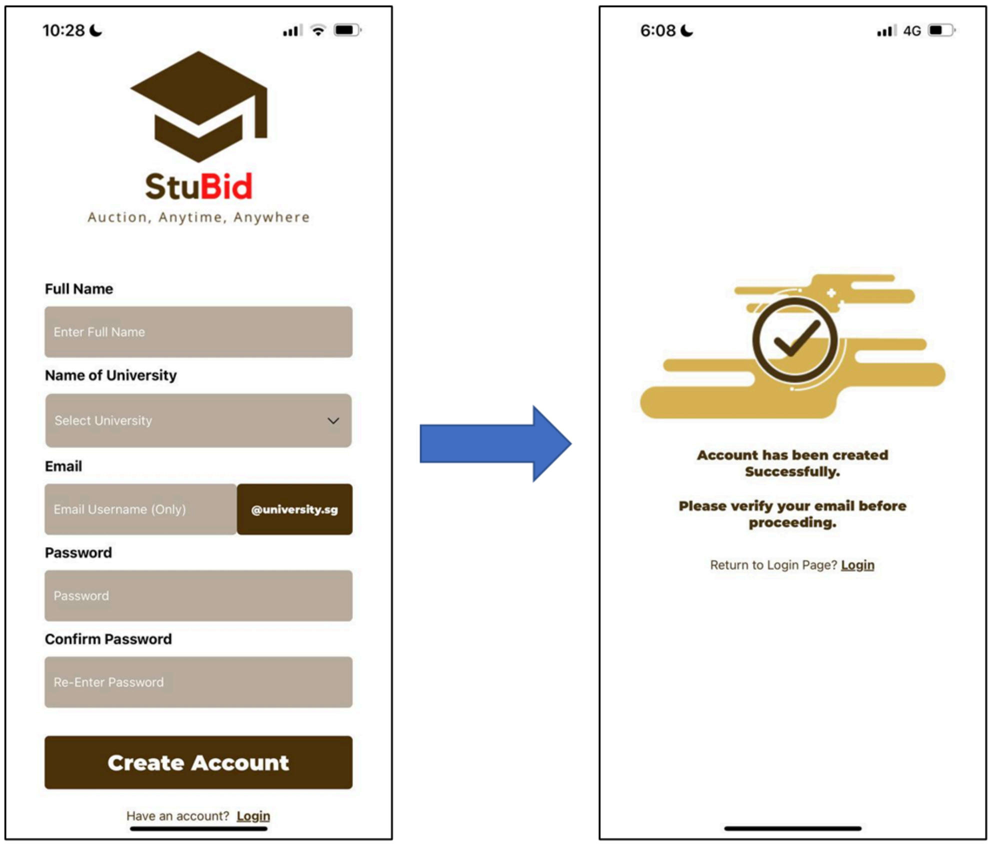
On the Welcome Page, users will receive a Welcome Message from the app and can opt to Sign Up and Login.

Login **(Completed)**



In the Login Page, Users will sign in with their allocated school email address (e.g. tony123@u.nus.edu) and the password they created during registration.
If they do not have an account, they can opt to sign up. Moreover, in any instance where they forget their password, they can click on the Forget Password option to navigate to the Reset Password Page.

Registration **(Completed)**



Account has been created Successfully.

Please verify your email before proceeding.

In the Registration Page, users will fill in their **Full Name**, Select the **University** they are studying at (through a drop down menu), **School Email** and **Password** respectively. Upon Filling up, users can click on Create Account and will be navigated to a ‘Successful Account Creation’ Page if the account is created successfully. A message will prompt the user to verify their email before they can access their account. Users can also opt to return back to the Login Page after creating their accounts.

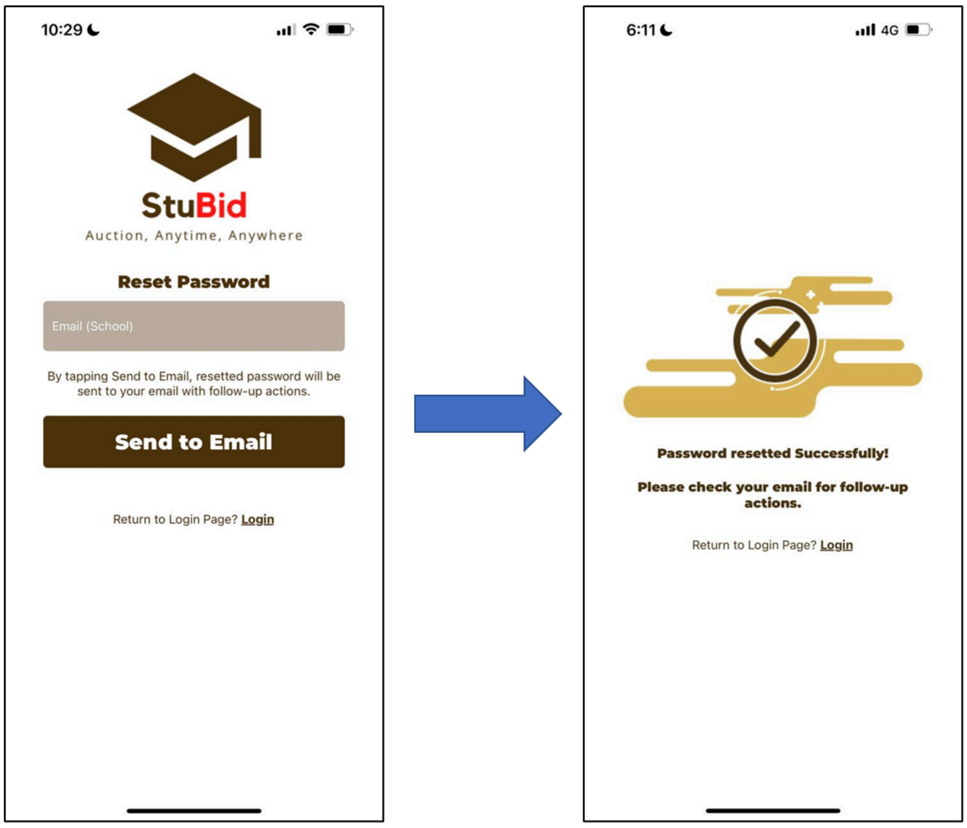
Email Verification and Authentication:

For verification and authentication purposes to confirm if a user is a student, only emails matching the strings after the @ of the respective universities will be accepted during registration. As of now, below is a list of accepted universities for the application.

Table of Accepted Universities and Emails

University	Email
National University of Singapore (NUS)	@u.nus.edu
Nanyang Technological University (NTU)	@e.ntu.edu.sg
Singapore Management University (SMU)	@smu.edu.sg
Singapore Institute of Technology (SIT)	@sit.singaporetech.edu.sg
Singapore University of Technology & Design (SUTD)	@mymail.sutd.edu.sg
Singapore University of Social Sciences (SUSS)	@suss.edu.sg

Reset Password **(Completed)**


10:29
StuBid
Auction, Anytime, Anywhere
Reset Password
Email (School)
Send to Email
Return to Login Page? Login
6:11
4G
Return to Login Page? Login

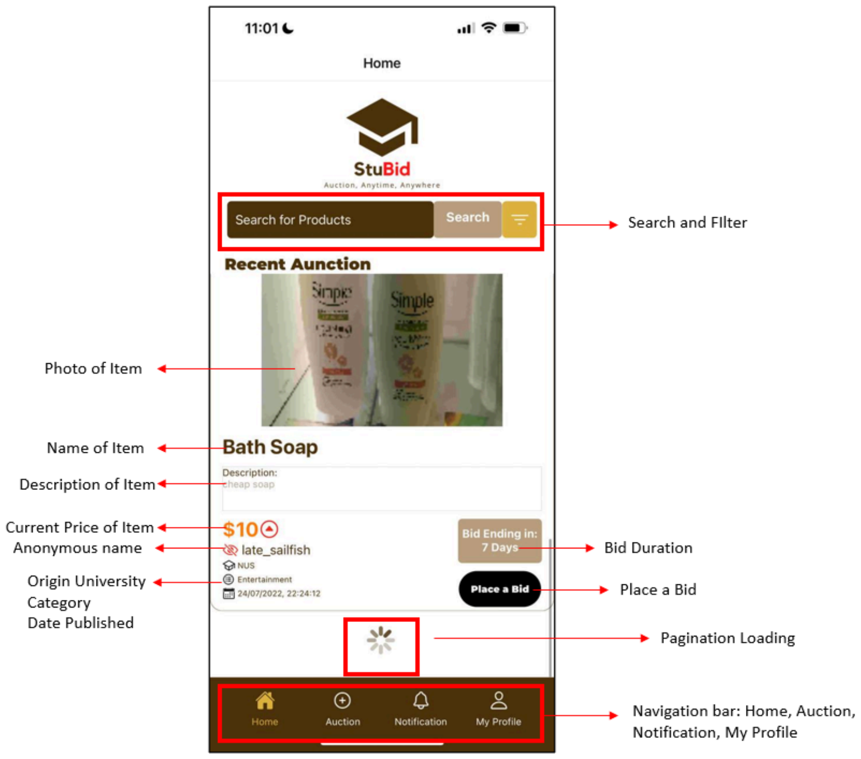
By tapping Send to Email, resetted password will be sent to your email with follow-up actions.

Password resetted Successfully!

Please check your email for follow-up actions.

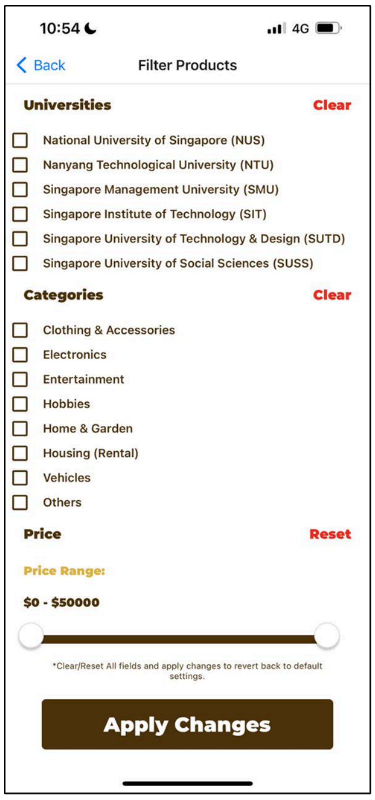
In the event where users forget their password, they can opt to reset their password by filling in their respective school emails. By clicking on Send to Email, an email will be sent to their respective accounts for follow up actions. Moreover, users can choose to navigate back to the Login Page if they do not choose to reset their password.

Homepage (Completed)



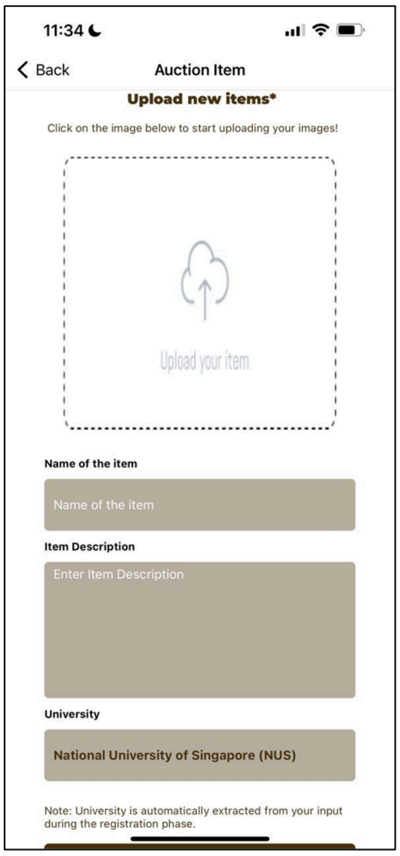
In the Homepage, users can **Search** for their desired products, **Filter** products according to their preferences, View **Product listing** in a Card-based format, and **navigate** through other different features such as Auction, Notifications or My Profile.

Filter Page (Completed)


10:54
4G
Back
Filter Products
Universities
Clear
National University of Singapore (NUS)
Nanyang Technological University (NTU)
Singapore Management University (SMU)
Singapore Institute of Technology (SIT)
Singapore University of Technology & Design (SUTD)
Singapore University of Social Sciences (SUSS)
Categories
Clear
Clothing & Accessories
Electronics
Entertainment
Hobbies
Home & Garden
Housing (Rental)
Vehicles
Others
Price
Reset
Price Range:
$0 - $50000
*Clear/Reset All fields and apply changes to revert back to default settings.
Apply Changes

By clicking on the funnel looking ‘Filter’ button, users can filter their desired products according to different sections which includes **University**, **Categories** and **Price Range**. The ‘tick’ icon represents the user’s filtering selection and they can clear their options by pressing the Clear button. By pressing on apply changes, users will be redirected back to Homepage with their desired change. Moreover, if users do not want to make any changes, they can navigate back to Homepage by clicking the ‘<’ button at the top-left hand corner.

<u>Anonymous Auction (Selling/Publishing Item)</u> **(Completed)**



Upload new items*

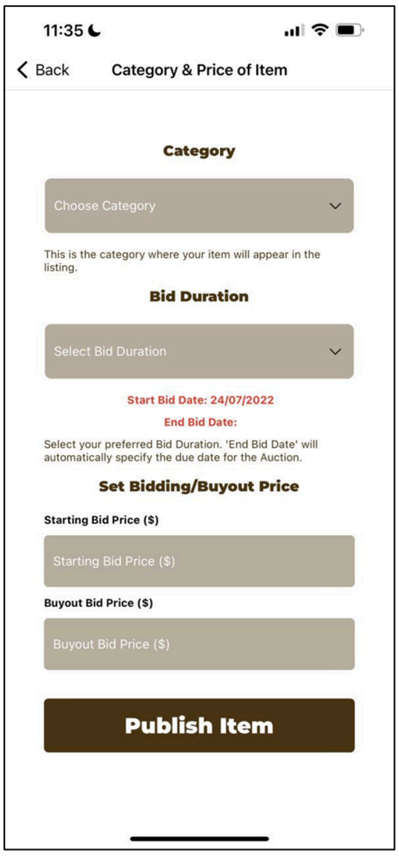
Click on the image below to start uploading your images!

Name of the item: Name of the item

Item Description: Enter Item Description

University: National University of Singapore (NUS)

Note: University is automatically extracted from your input during the registration phase.



Category

Choose Category

This is the category where your item will appear in the listing.

Bid Duration

Select Bid Duration

Start Bid Date: 24/07/2022

End Bid Date:

Select your preferred Bid Duration. 'End Bid Date' will automatically specify the due date for the Auction.

Set Bidding/Buyout Price

Starting Bid Price ($): Starting Bid Price ($)

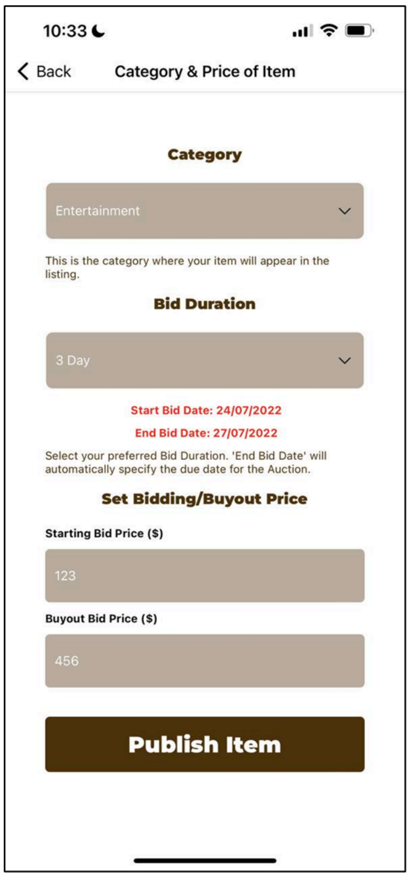
Buyout Bid Price ($): Buyout Bid Price ($)

To publish or auction an item for potential buyers, users can click on ‘Auction’ on the navigation bar. Users will be redirected to the Auction Page and they will fill in the respective fields for their product to be auctioned. Refer to the table below to show the required field needed to be filled in by the user.

<u>Table of fields to be filled up by users when auctioning an item</u>

Component	**Description**	**Layout**
Photo	Set the photo of the item to be auctioned	Image Source (Insert Picture)
Name	Set the name of the item to be auctioned	Text input (Alphanumeric)
Description	Set the description of the item to be auctioned	Text input (Alphanumeric)

School (University)	School of the respective user. (This will be automatically extracted from user profile details)	Text input (Locked)
Category	Select the category of the item (e.g Electronics)	Drop down box
Bid Duration	Set the bid duration of the item to be auctioned	Drop down box
Set Price	Set the starting bid and buyout bid for the item to be auctioned	Text input (Numbers only)


10:33

Back Category & Price of Item

Category

Entertainment

This is the category where your item will appear in the listing.

Bid Duration

3 Day

Start Bid Date: 24/07/2022

End Bid Date: 27/07/2022

Select your preferred Bid Duration. 'End Bid Date' will automatically specify the due date for the Auction.

Set Bidding/Buyout Price

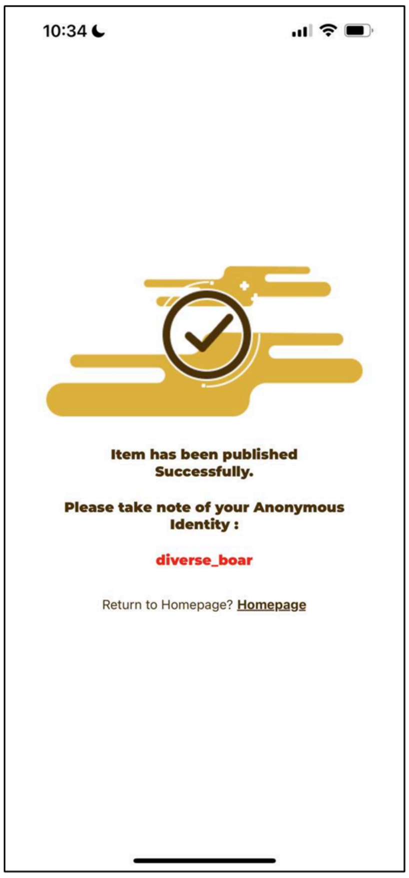
Starting Bid Price ($)

123

Buyout Bid Price ($)

456

Publish Item


10:34

Item has been published Successfully.

Please take note of your Anonymous Identity :

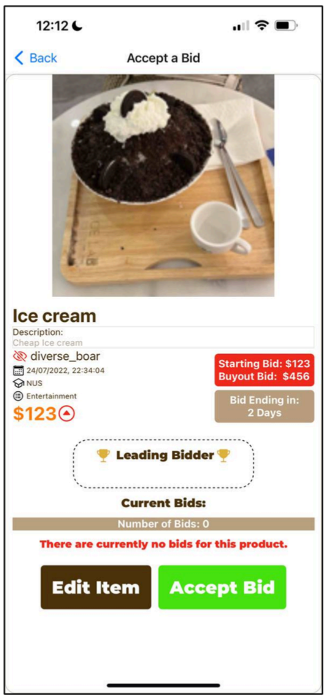
diverse_boar

Return to Homepage? Homepage

After filling up the respective fields, users can click on ‘Publish Item’ to auction their item online to be seen by other users. Users will be automatically assigned an anonymous name (e.g. diverse_boar) if the item is successfully published. Moreover, if users do not want to make any changes or wish to make some edits, they can navigate back to the previous page or Homepage by clicking the ‘<’ button at the top-left hand corner.

Product Information (Completed)

Seller Perspective


12:12

Back | Accept a Bid

Ice cream

Description:
Cheap Ice cream

diverse_boar
24/07/2022, 22:34:04
NUS
Entertainment
$123

Starting Bid: $123
Buyout Bid: $456

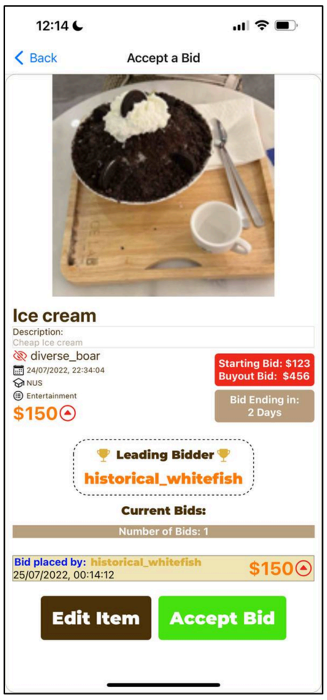
Bid Ending in:
2 Days

Leading Bidder

Current Bids:
Number of Bids: 0

There are currently no bids for this product.

Edit Item | Accept Bid


12:14

Back | Accept a Bid

Ice cream

Description:
Cheap Ice cream

diverse_boar
24/07/2022, 22:34:04
NUS
Entertainment
$150

Starting Bid: $123
Buyout Bid: $456

Bid Ending in:
2 Days

Leading Bidder
historical_whitefish

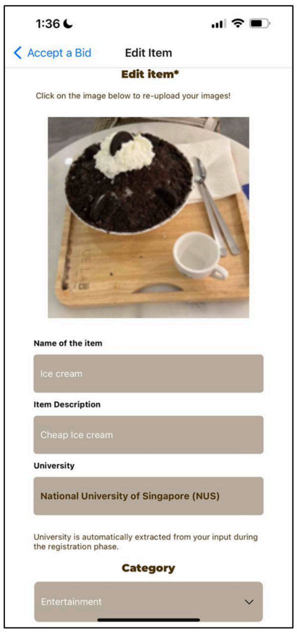
Current Bids:
Number of Bids: 1

Bid placed by: historical_whitefish
25/07/2022, 00:14:12
$150

Edit Item | Accept Bid

Users (Sellers) who have successfully auctioned their item can view their item in the product information page. In this page, Sellers can view their anonymous profile identity, Item name and description, date published, origin university and product category, Starting and buyout bid for item, bid duration, view all current bidders details and leading bidder, number of bids placed, edit item and accept bid. Moreover, if users do not want to make any changes, they can navigate back to the previous page by clicking the '<' button at the top-left hand corner.

Edit Item (Completed)


1:36

Accept a Bid | Edit Item

Edit item*

Click on the image below to re-upload your images!

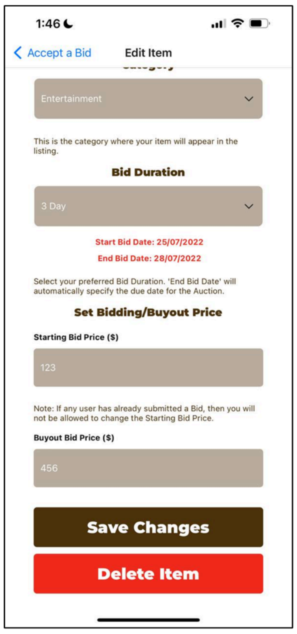
Name of the item
Ice cream

Item Description
Cheap Ice cream

University
National University of Singapore (NUS)

University is automatically extracted from your input during the registration phase.

Category
Entertainment


1:46

Accept a Bid | Edit Item

Entertainment

This is the category where your item will appear in the listing.

Bid Duration
3 Day

Start Bid Date: 25/07/2022
End Bid Date: 28/07/2022

Select your preferred Bid Duration. 'End Bid Date' will automatically specify the due date for the Auction.

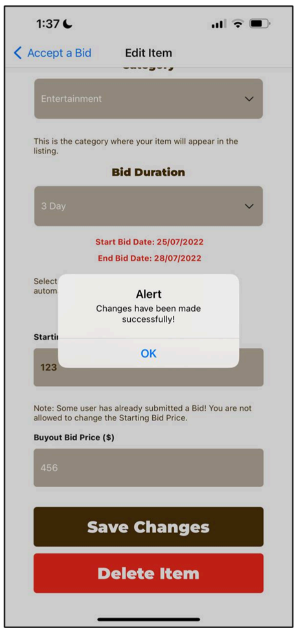
Set Bidding/Buyout Price

Starting Bid Price ($)
123

Note: If any user has already submitted a Bid, then you will not be allowed to change the Starting Bid Price.

Buyout Bid Price ($)
456

Save Changes
Delete Item


1:37

Accept a Bid | Edit Item

Entertainment

This is the category where your item will appear in the listing.

Bid Duration
3 Day

Start Bid Date: 25/07/2022
End Bid Date: 28/07/2022

Alert
Changes have been made successfully!
OK

123

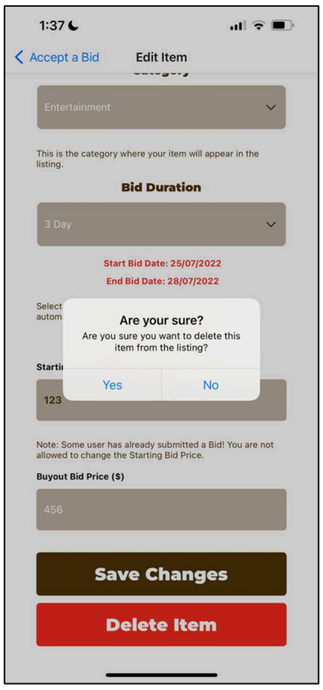
Note: Some user has already submitted a Bid! You are not allowed to change the Starting Bid Price.

Buyout Bid Price ($)
456

Save Changes
Delete Item

Users (Sellers) can edit their item listing if they wish to modify any details before the bidding process. However, if someone has already bidded for the item, Sellers will not be able to modify the 'Starting Bid' price but may still continue to modify any other details.

Delete Item **(Completed)**


1:37

Accept a Bid   Edit Item

Entertainment

This is the category where your item will appear in the listing.

Bid Duration

3 Day

Start Bid Date: 25/07/2022

End Bid Date: 28/07/2022

Select autom

Are your sure?

Are you sure you want to delete this item from the listing?

Yes   No

Starti

123

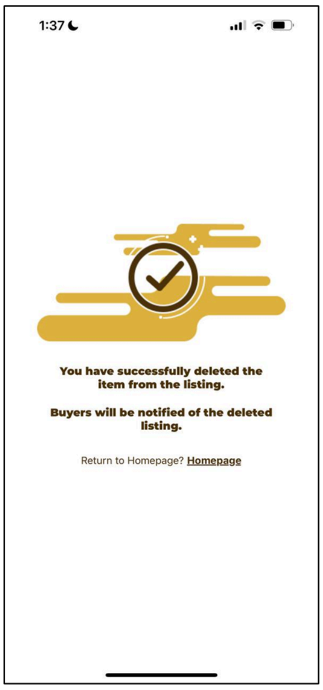
Note: Some user has already submitted a Bid! You are not allowed to change the Starting Bid Price.

Buyout Bid Price ($)

456

Save Changes

Delete Item


1:37

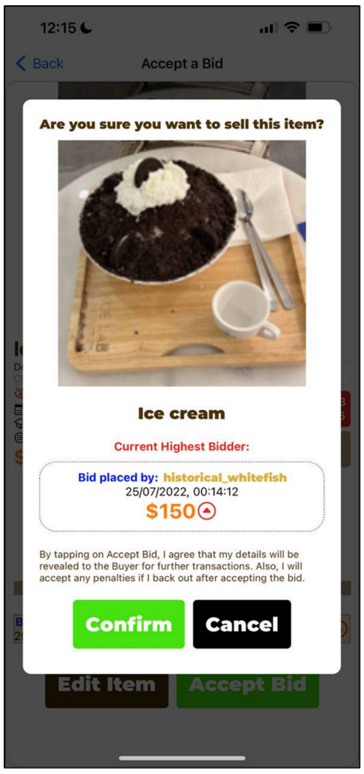
You have successfully deleted the item from the listing.

Buyers will be notified of the deleted listing.

Return to Homepage? Homepage

Sellers can also choose to delete the item if they are unsatisfied with their items. They will be navigated to the respective pages (if successful) upon applying changes or deleting the item.

Accept Bid **(Completed)**


12:15

Back   Accept a Bid

Are you sure you want to sell this item?

Ice cream

Current Highest Bidder:

Bid placed by: historical_whitefish

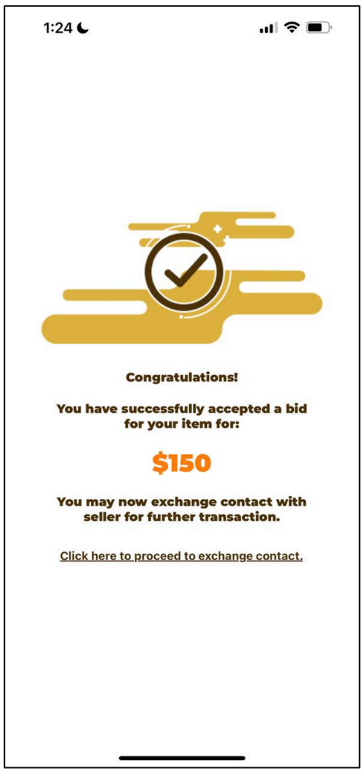
25/07/2022, 00:14:12

$150

By tapping on Accept Bid, I agree that my details will be revealed to the Buyer for further transactions. Also, I will accept any penalties if I back out after accepting the bid.

Confirm   Cancel

Edit Item   Accept Bid


1:24

Congratulations!

You have successfully accepted a bid for your item for:

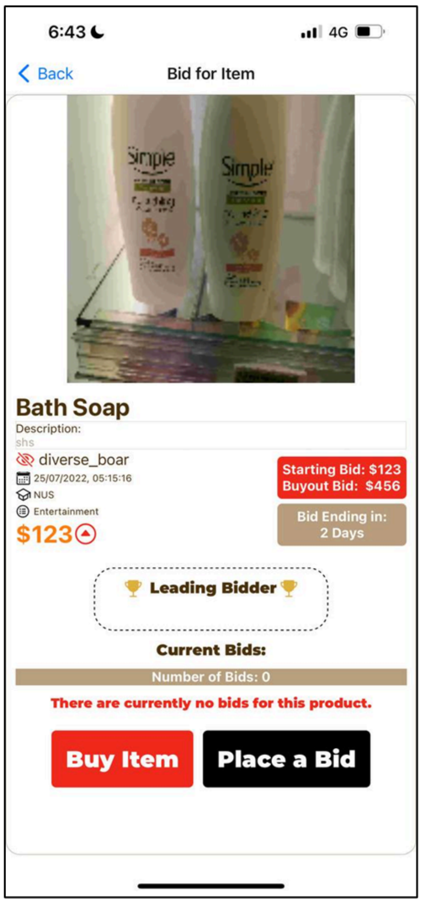
$150

You may now exchange contact with seller for further transaction.

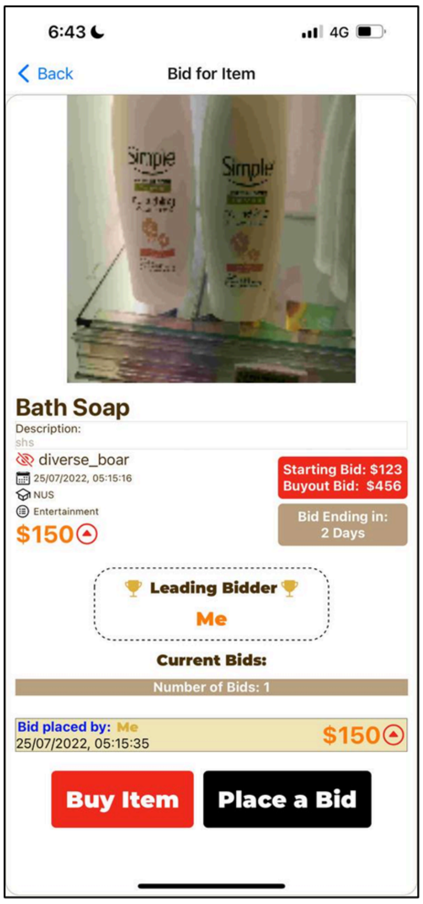
Click here to proceed to exchange contact.

Users (Sellers) can accept bids for their item if they are satisfied with the price or if they wish to sell the item to the buyer. The highest price and anonymous buyer details will be extracted in this process and will be the winner for this auction. Accepting the bid will prompt the Seller again to confirm if the Seller wishes to sell the item. By confirming, Seller will be navigated to the successful auction page and will be able to exchange contact details with buyer for further transactions.

Buyer Perspective



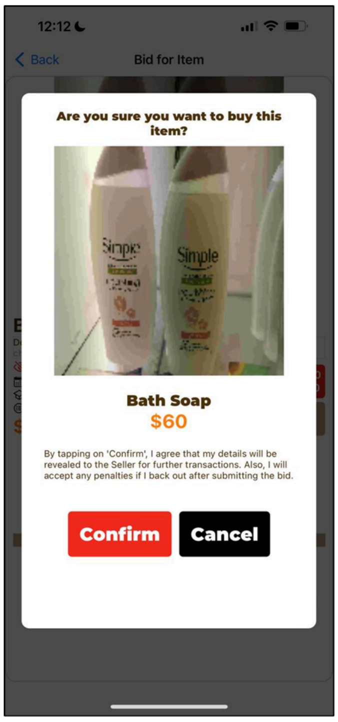
Bath Soap
Description:
shs
diverse_boar
25/07/2022, 05:15:16
NUS
Entertainment
$123
Starting Bid: $123
Buyout Bid: $456
Bid Ending in:
2 Days
Leading Bidder
Current Bids:
Number of Bids: 0
There are currently no bids for this product.



Bath Soap
Description:
shs
diverse_boar
25/07/2022, 05:15:16
NUS
Entertainment
$150
Starting Bid: $123
Buyout Bid: $456
Bid Ending in:
2 Days
Leading Bidder
Me
Current Bids:
Number of Bids: 1
Bid placed by: Me
25/07/2022, 05:15:35
$150

Users (Buyers) who want to buy or bid on their desired item can view the product on the Product information page. In this page, Buyers can view Seller anonymous profile identity, Item name and description, date published, origin university and category, Starting and buyout bid for item, bid duration, view all current bidders details and leading bidder, number of bids placed, buyout item and place a bid for the item. Moreover, if users do not want to make any changes, they can navigate back to Homepage by clicking the ‘<’ button at the top-left hand corner.

Buyout Item **(Completed)**

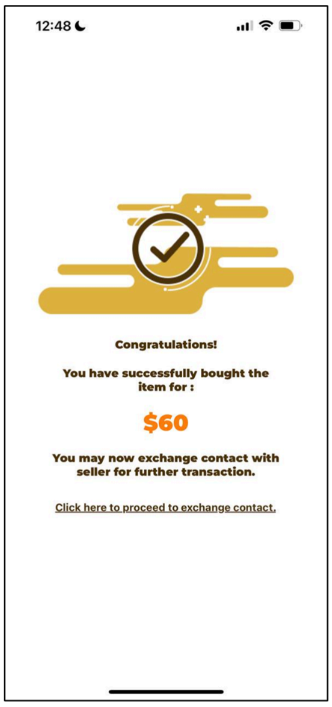

12:12

Back Bid for Item

Are you sure you want to buy this item?

Bath Soap
$60

By tapping on 'Confirm', I agree that my details will be revealed to the Seller for further transactions. Also, I will accept any penalties if I back out after submitting the bid.


12:48

Congratulations!

You have successfully bought the item for :

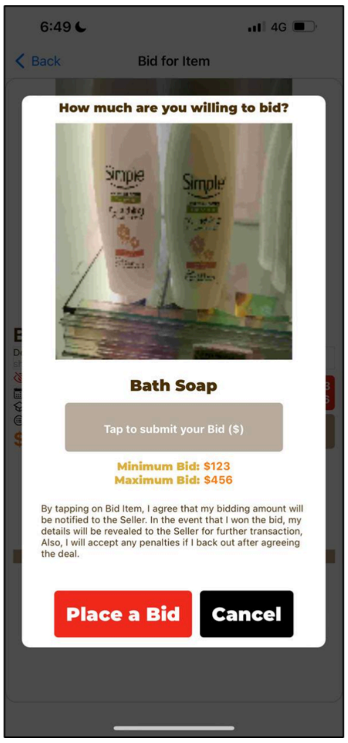
$60

You may now exchange contact with seller for further transaction.

Click here to proceed to exchange contact.

Users (Buyers) who wish to buyout the item immediately can click on the 'Buy Item' button. Buyers will be informed about the buyout price and prompt again whether to confirm buyout. By clicking Confirm, buyers will be navigated to the successful buyout page and will be able to exchange contact with Seller for further transactions. Buyout an item will automatically make the buyer the winner of the auction without any bidding required (outbidding others).

Place a Bid **(Completed)**


6:49 4G

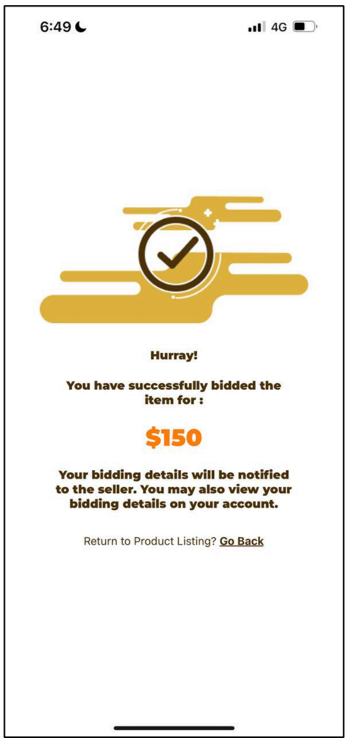
Back Bid for Item

How much are you willing to bid?

Bath Soap

Minimum Bid: $123
Maximum Bid: $456

By tapping on Bid Item, I agree that my bidding amount will be notified to the Seller. In the event that I won the bid, my details will be revealed to the Seller for further transaction, Also, I will accept any penalties if I back out after agreeing the deal.


6:49 4G

Hurray!

You have successfully bidded the item for :

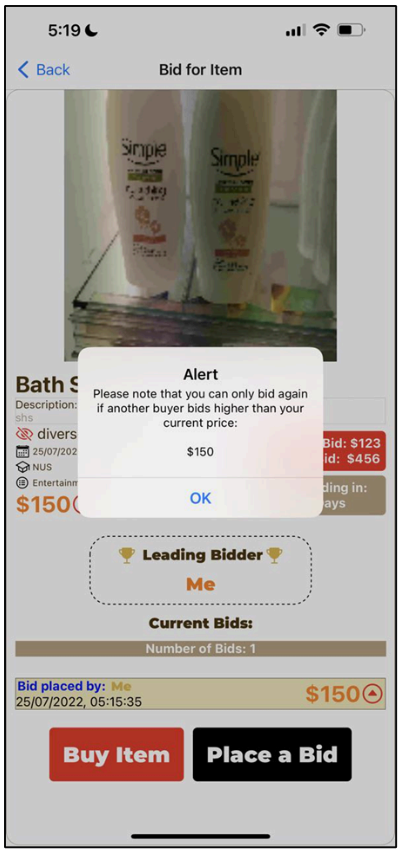
$150

Your bidding details will be notified to the seller. You may also view your bidding details on your account.

Return to Product Listing? Go Back

Users (Buyers) who wish to bid for an item can click on the ‘Place a Bid ’ button. Buyers will be informed of the minimum bidding price and maximum bidding price (Sellout price) and will not be allowed to exceed the bidding range. Buyers can modify their bidding price according to their preference and clicking on Place a bid again will navigate them to the successful bidding page. Buyers will also not be allowed to backtrack on their bid after successful bidding. Buyers can view their bidding as they return back to the product listing.

Anti-Spam Bidding Prevention **(Completed)**



Alert

Please note that you can only bid again if another buyer bids higher than your current price:

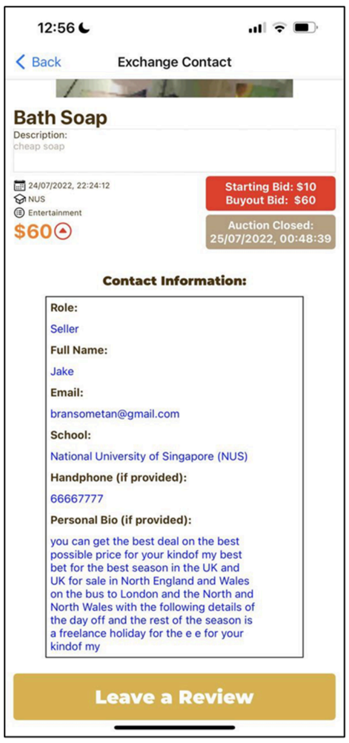
$150

OK

Bid placed by: Me
25/07/2022, 05:15:35 $150

As to prevent fraudulent and spam bidding, users (buyers) will only be allowed to bid again only if another potential buyer increases the current bidding price. The ‘Place a Bid’ button will send a message alert to the user to wait until another buyer has bidded a higher price for the item.

Exchange Contact **(Completed)**


12:56

Back Exchange Contact

Bath Soap

Description:
cheap soap

24/07/2022, 22:24:12
NUS
Entertainment
$60

Starting Bid: $10
Buyout Bid: $60

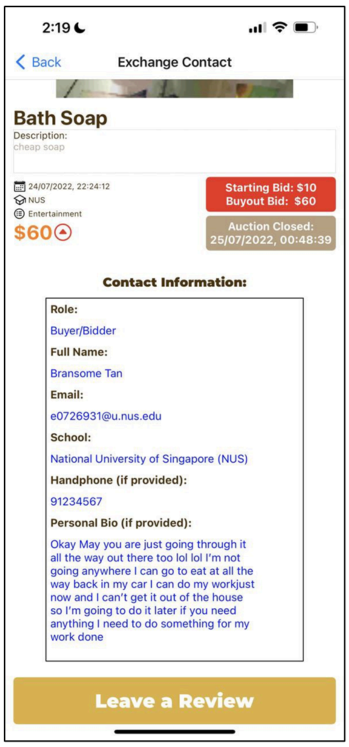
Auction Closed:
25/07/2022, 00:48:39

Contact Information:

Role:
Seller
Full Name:
Jake
Email:
bransometan@gmail.com
School:
National University of Singapore (NUS)
Handphone (if provided):
66667777
Personal Bio (if provided):
you can get the best deal on the best possible price for your kindof my best bet for the best season in the UK and UK for sale in North England and Wales on the bus to London and the North and North Wales with the following details of the day off and the rest of the season is a freelance holiday for the e e for your kindof my

Leave a Review

Exchange Contact (Buyer Perspective)


2:19

Back Exchange Contact

Bath Soap

Description:
cheap soap

24/07/2022, 22:24:12
NUS
Entertainment
$60

Starting Bid: $10
Buyout Bid: $60

Auction Closed:
25/07/2022, 00:48:39

Contact Information:

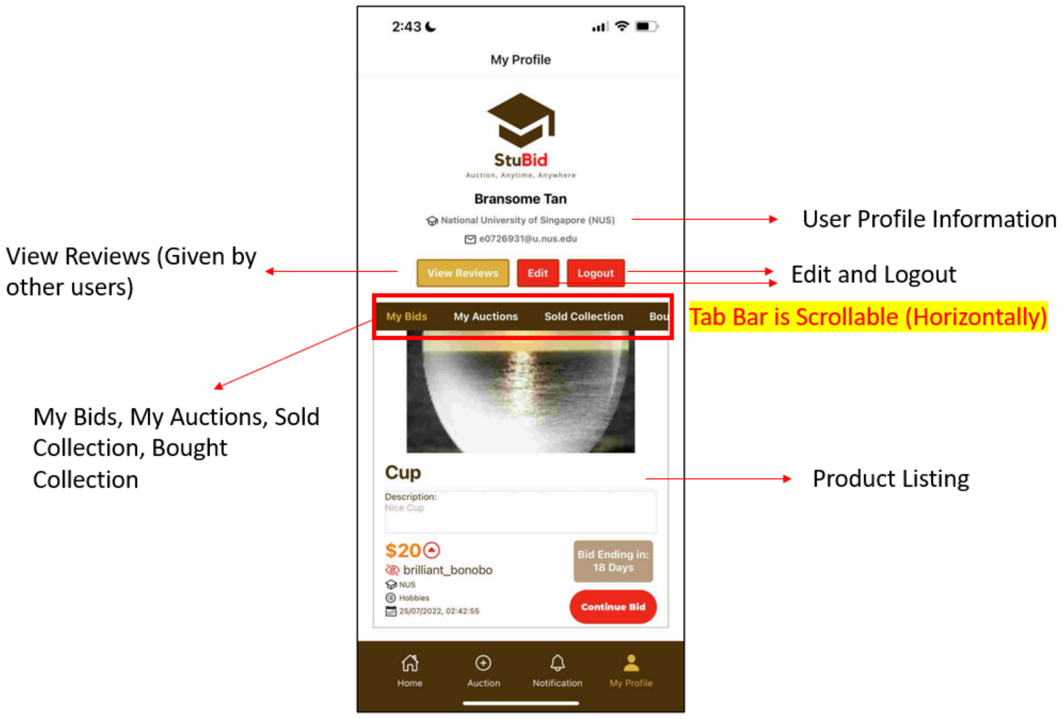
Role:
Buyer/Bidder
Full Name:
Bransome Tan
Email:
e0726931@u.nus.edu
School:
National University of Singapore (NUS)
Handphone (if provided):
91234567
Personal Bio (if provided):
Okay May you are just going through it all the way out there too lol lol I'm not going anywhere I can go to eat at all the way back in my car I can do my workjust now and I can't get it out of the house so I'm going to do it later if you need anything I need to do something for my work done

Leave a Review

Exchange Contact (Seller Perspective)

Successful Auction (Seller and Winning Bidder) will be allowed to exchange contact information. Details that will be revealed to the other party includes Full Name, Email, School, Handphone (if provided), Personal Bio (if provided)

User Profile (My Profile) **(Completed)**

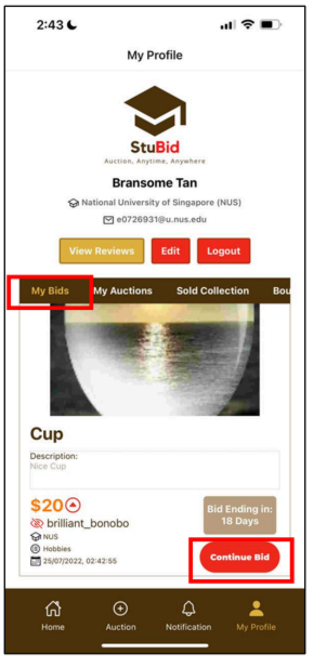


Bransome Tan
National University of Singapore (NUS)
e0726931@u.nus.edu

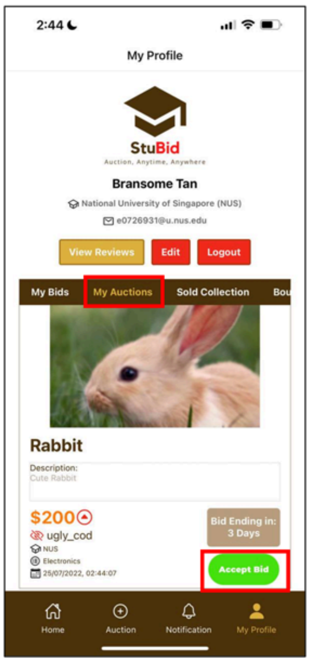
Cup
Description:
Nice Cup
$20
brilliant_bonobo
NUS
Hobbies
25/07/2022, 02:42:55
Bid Ending in: 18 Days

In the My Profile page, users will be able to view their **Profile information**, **View Reviews**, **Edit profile details**, **Logout** of the app, **View current listings** of items bidded, items auctioned and completed bids (Seller Collection and Buyer Collection). Moreover, users can **navigate** through other different features such as Homepage, Auction or Notifications.

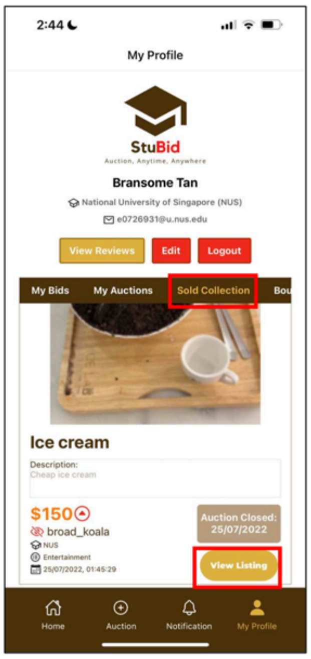
Tab Navigation **(Completed)**



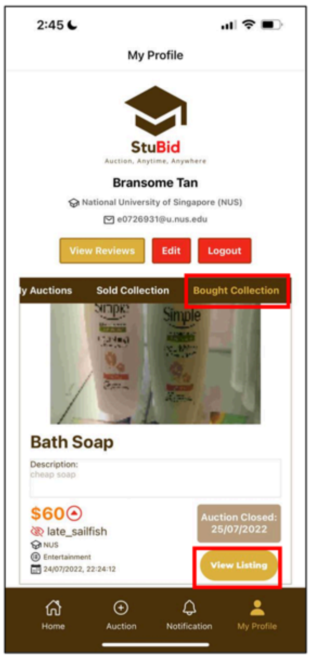
My Bids Tab



My Auction Tab



Sold Collection Tab



Bought Collection Tab

Under My Profile Page Tab Navigation, users will be able to view and perform the following actions as listed below:

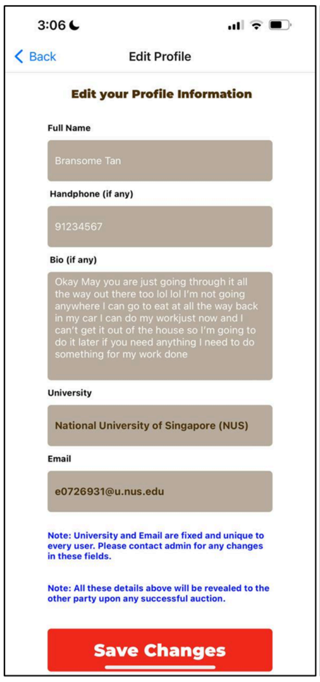
My Bids Tab: View bids offered (User is Buyer)
- Continue bidding for the item

My Auction Tab: View bids received (User is Seller)
- Accept bid for the current highest bidder

Sold Collection Tab: View completed bids as a Seller (Completed Auctions)
- Leave user review for other party

Bought Collection Tab: View completed bids as a Buyer (Completed Auctions)
- Leave user review for other party

Edit Profile (Modify User Details) **(Completed)**


3:06

Back Edit Profile

Edit your Profile Information

Full Name

Bransome Tan

Handphone (if any)

91234567

Bio (if any)

Okay May you are just going through it all the way out there too lol lol I'm not going anywhere I can go to eat at all the way back in my car I can do my workjust now and I can't get it out of the house so I'm going to do it later if you need anything I need to do something for my work done

University

National University of Singapore (NUS)

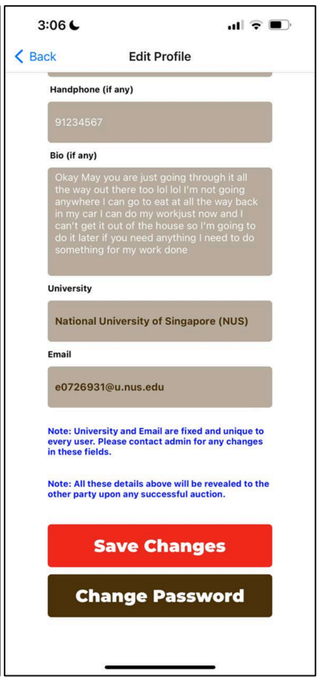
Email

e0726931@u.nus.edu

Note: University and Email are fixed and unique to every user. Please contact admin for any changes in these fields.

Note: All these details above will be revealed to the other party upon any successful auction.

Save Changes


3:06

Back Edit Profile

Handphone (if any)

91234567

Bio (if any)

Okay May you are just going through it all the way out there too lol lol I'm not going anywhere I can go to eat at all the way back in my car I can do my workjust now and I can't get it out of the house so I'm going to do it later if you need anything I need to do something for my work done

University

National University of Singapore (NUS)

Email

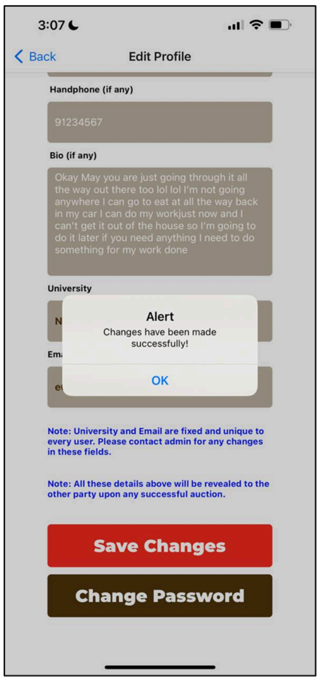
e0726931@u.nus.edu

Note: University and Email are fixed and unique to every user. Please contact admin for any changes in these fields.

Note: All these details above will be revealed to the other party upon any successful auction.

Save Changes

Change Password


3:07

Back Edit Profile

Handphone (if any)

91234567

Bio (if any)

Okay May you are just going through it all the way out there too lol lol I'm not going anywhere I can go to eat at all the way back in my car I can do my workjust now and I can't get it out of the house so I'm going to do it later if you need anything I need to do something for my work done

University

Alert

Changes have been made successfully!

OK

Note: University and Email are fixed and unique to every user. Please contact admin for any changes in these fields.

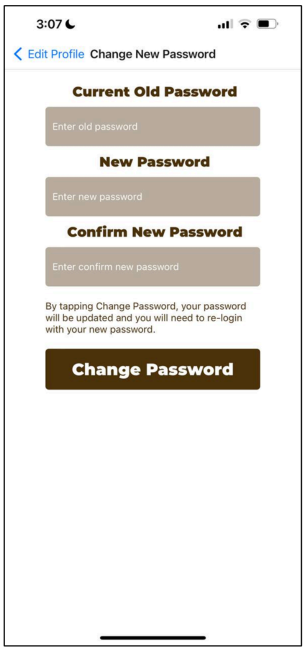
Note: All these details above will be revealed to the other party upon any successful auction.

Save Changes

Change Password

Users will be able to modify the profile details information such as their Full name, handphone or personal description (Bio). Origin University and Email are unique to each user and are uneditable. These are the details that will be revealed to the other party in the event of a successful auction (completed auction).

Edit Profile (Change Password) **(Completed)**


3:07

Edit Profile Change New Password

Current Old Password

Enter old password

New Password

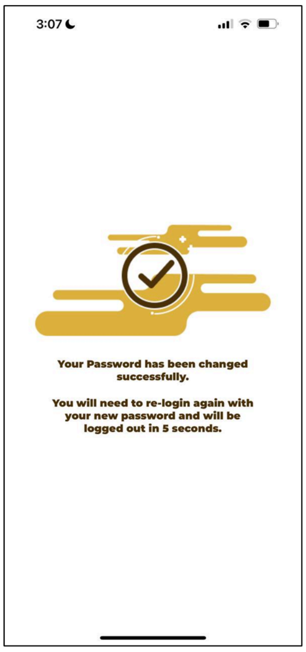
Enter new password

Confirm New Password

Enter confirm new password

By tapping Change Password, your password will be updated and you will need to re-login with your new password.

Change Password

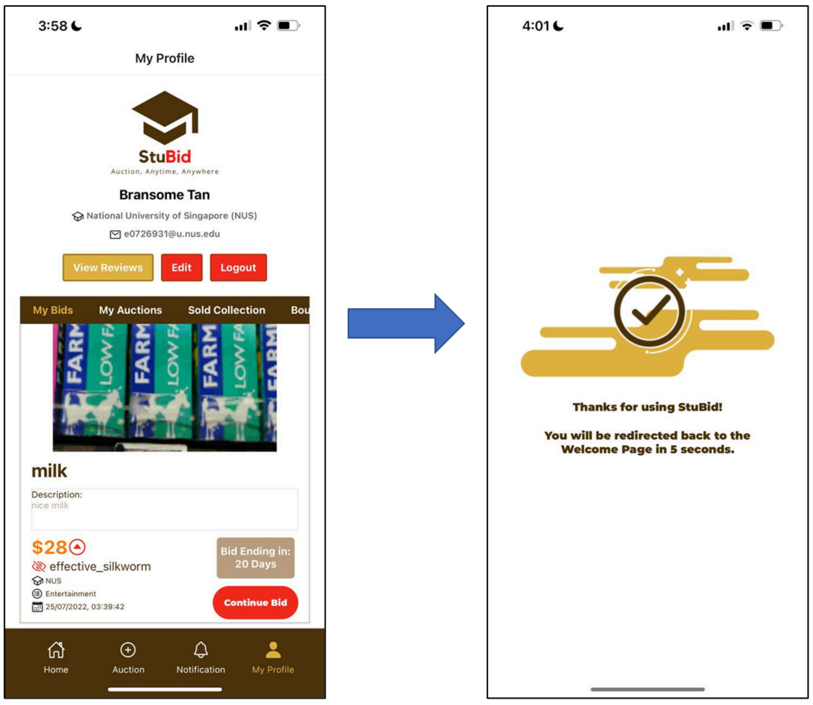

3:07

Your Password has been changed successfully.

You will need to re-login again with your new password and will be logged out in 5 seconds.

Users will be allowed to change their old password to a new password if they wish to do so.

Logout (Completed)

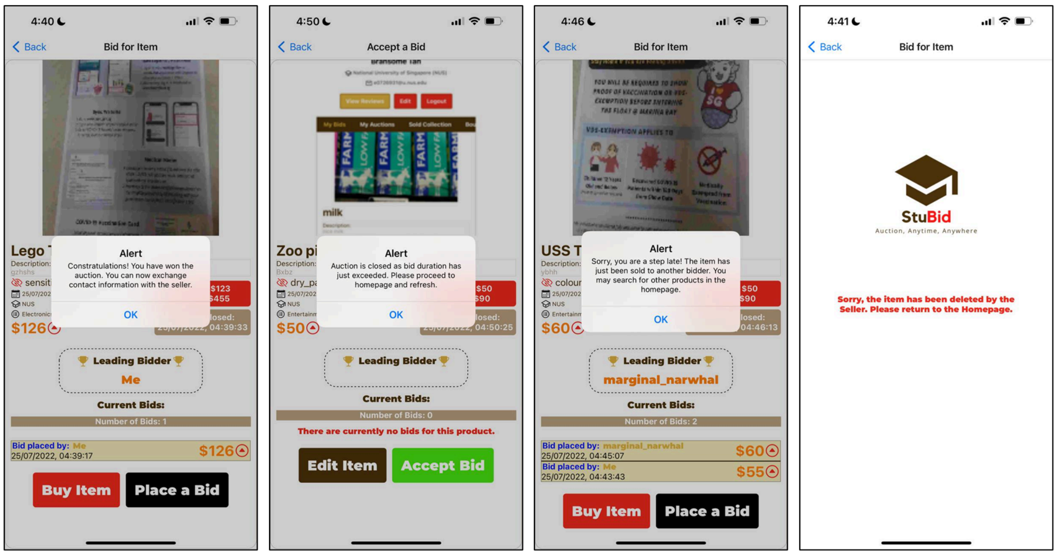


Thanks for using StuBid!

You will be redirected back to the Welcome Page in 5 seconds.

Users will be allowed to Logout of the app and will be redirected back to the Login page if they wish to use another account to Login.

Bid Duration Termination Mechanism (Completed)



Alert
Constratulations! You have won the auction. You can now exchange contact information with the seller.
OK

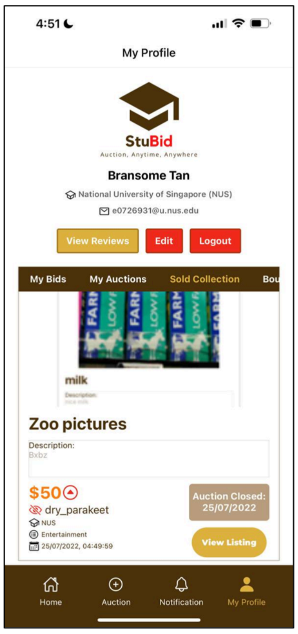
Alert
Auction is closed as bid duration has just exceeded. Please proceed to homepage and refresh.
OK

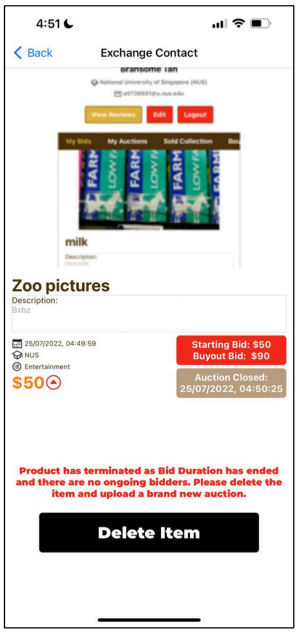
There are currently no bids for this product.

Alert
Sorry, you are a step late! The item has just been sold to another bidder. You may search for other products in the homepage.
OK

Sorry, the item has been deleted by the Seller. Please return to the Homepage.

As users publish an item to be auctioned, they need to input the bid duration for the item (for e.g. Bid ending in 7 days). In the event where the bid duration ends, the item will be terminated immediately and any user currently inside the product listing will be alerted and notified of the closure of the auction. Users will be prompted to redirect back to the homepage..





Product has terminated as Bid Duration has ended and there are no ongoing bidders. Please delete the item and upload a brand new auction.

Users who have their product terminated can check the terminated product in their current product listing under My Profile in either Sold Collection or Bought Collection. There will be two scenarios in which the terminated product will display as listed below :

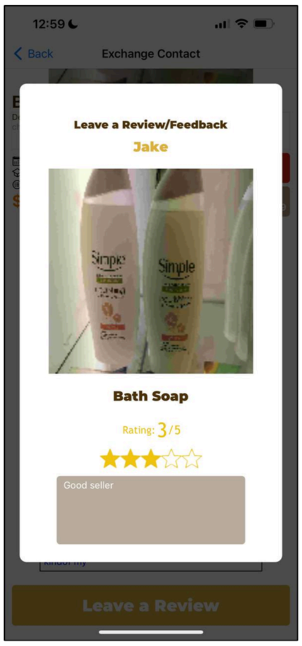
If there is a leading bidder:

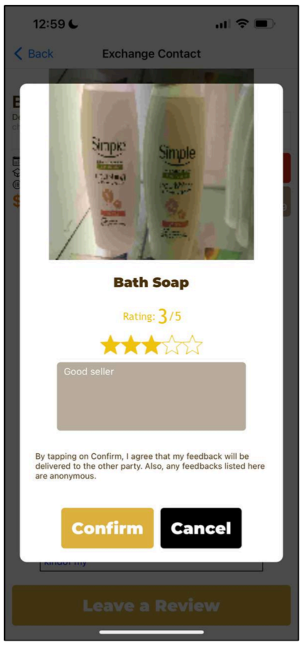
- Seller and Buyer will be notified of their successful auction and can exchange contact under profile page.
- All other bidders will be notified of unsuccessful auction

Else if there are no bidders:

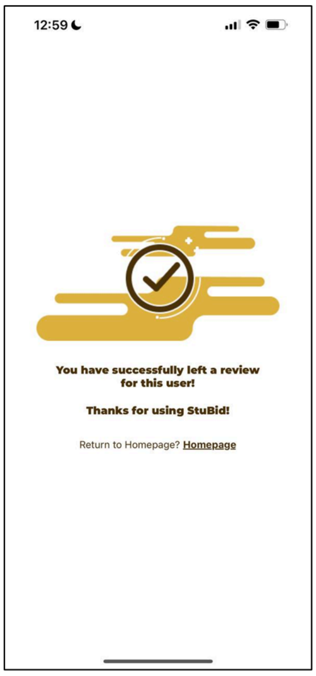
- Only Seller will be notified of unsuccessful auction and can choose to delete listing under profile page.

<u>User Review System</u> **(Completed)**





By tapping on Confirm, I agree that my feedback will be delivered to the other party. Also, any feedbacks listed here are anonymous.

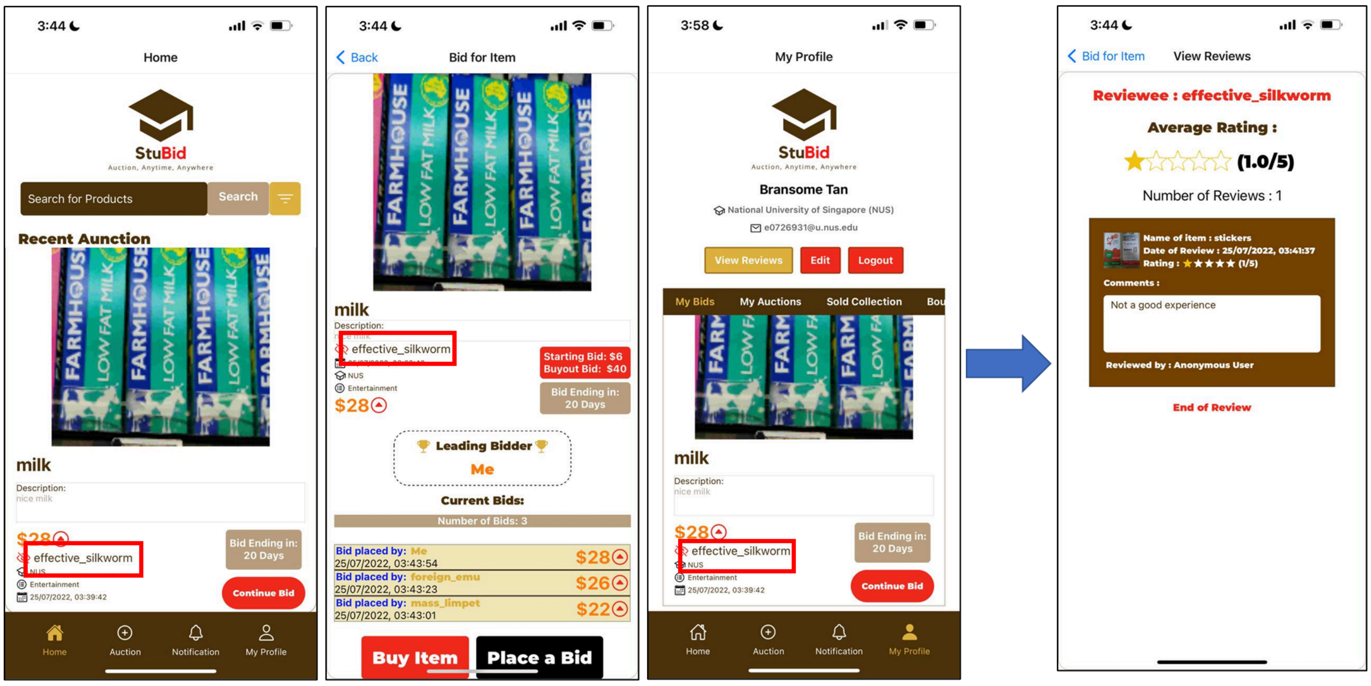


You have successfully left a review for this user!

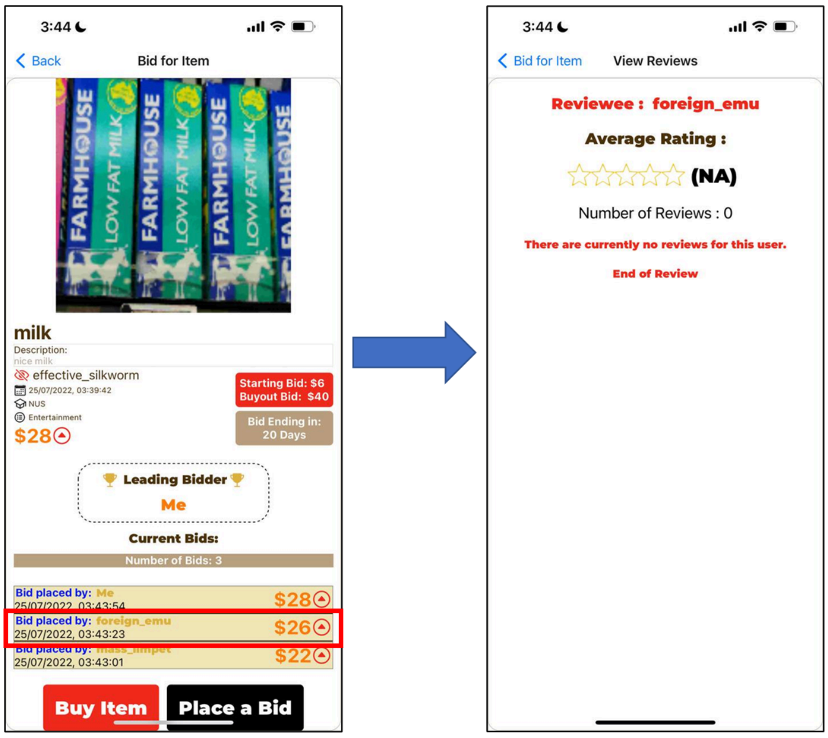
Thanks for using StuBid!

Upon a successful auction (Completed Auction), both parties (Seller and Winning Buyer) will be allowed to exchange reviews between each other. Users will be allowed to leave a rating (1 to 5) and post a comment/feedback for the other party. Clicking on 'Post Review' will navigate the user to the successful review page.

View User Reviews **(Completed)**



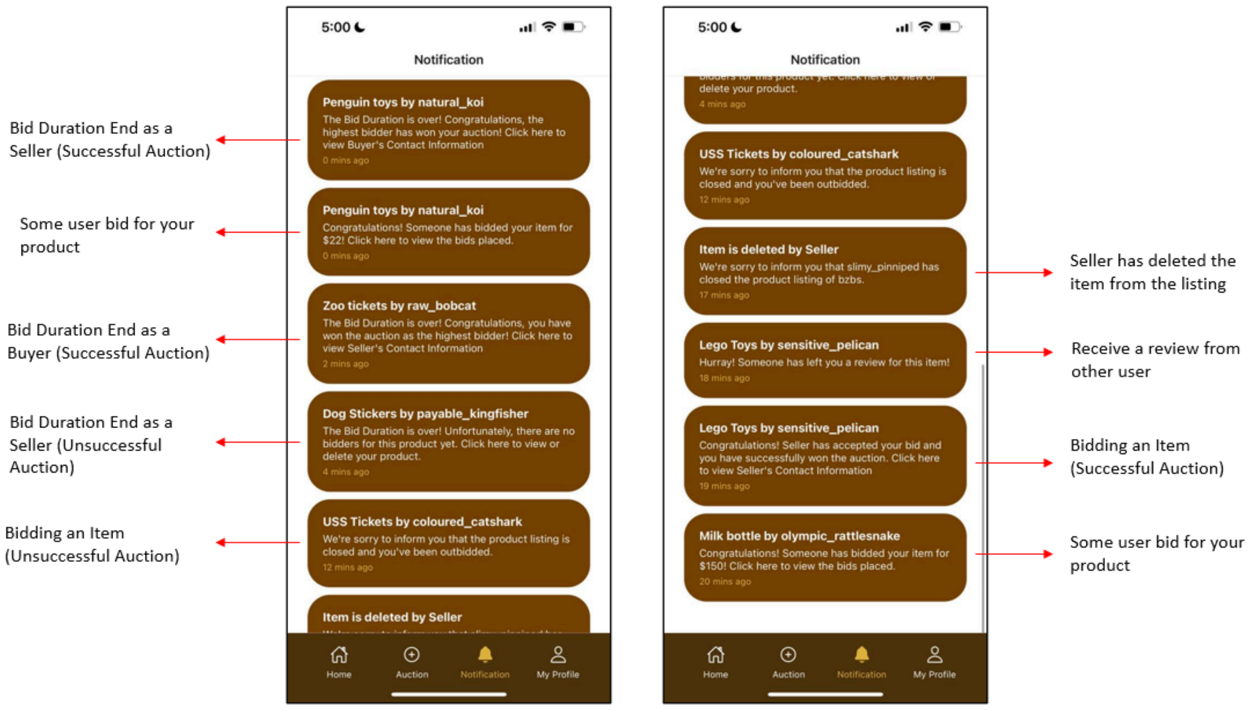
View Reviews of Sellers in Homepage, Product Listing, My Profile



View Reviews of other bidders in Product Listing
(Both Seller and all bidders can view reviews of other bidders)

Users will be able to view their own user reviews on their own respective profile page and other users' reviews during the bidding process. Since this is an anonymous auction, users will only be able to view the ratings/feedback of another user and no profile identities will be revealed. The View Review page will only show the total average rating, date of review, rating given and the feedback provided by a user but NOT the reviewer identity.

Notification System **(Completed)**



Bid Duration End as a Seller (Successful Auction)
Some user bid for your product
Bid Duration End as a Buyer (Successful Auction)
Bid Duration End as a Seller (Unsuccessful Auction)
Bidding an Item (Unsuccessful Auction)
Seller has deleted the item from the listing
Receive a review from other user
Bidding an Item (Successful Auction)
Some user bid for your product
Notification
Home
Auction
Notification
My Profile

Penguin toys by natural_koi
The Bid Duration is over! Congratulations, the highest bidder has won your auction! Click here to view Buyer's Contact Information
0 mins ago

Penguin toys by natural_koi
Congratulations! Someone has bidded your item for $22! Click here to view the bids placed.
0 mins ago

Zoo tickets by raw_bobcat
The Bid Duration is over! Congratulations, you have won the auction as the highest bidder! Click here to view Seller's Contact Information
2 mins ago

Dog Stickers by payable_kingfisher
The Bid Duration is over! Unfortunately, there are no bidders for this product yet. Click here to view or delete your product.
4 mins ago

USS Tickets by coloured_catshark
We're sorry to inform you that the product listing is closed and you've been outbidded.
12 mins ago

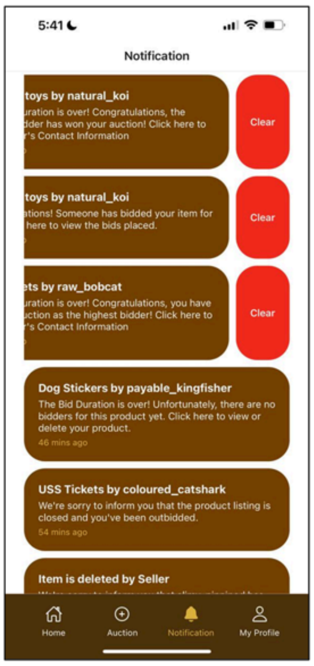
Item is deleted by Seller
We're sorry to inform you that slimy_pinniped has closed the product listing of bzbs.
17 mins ago

Lego Toys by sensitive_pelican
Hurray! Someone has left you a review for this item!
18 mins ago

Lego Toys by sensitive_pelican
Congratulations! Seller has accepted your bid and you have successfully won the auction. Click here to view Seller's Contact Information
19 mins ago

Milk bottle by olympic_rattlesnake
Congratulations! Someone has bidded your item for $150! Click here to view the bids placed.
20 mins ago

Examples of Notifications Received



toys by natural_koi
uration is over! Congratulations, the dder has won your auction! Click here to 's Contact Information

toys by natural_koi
ations! Someone has bidded your item for here to view the bids placed.

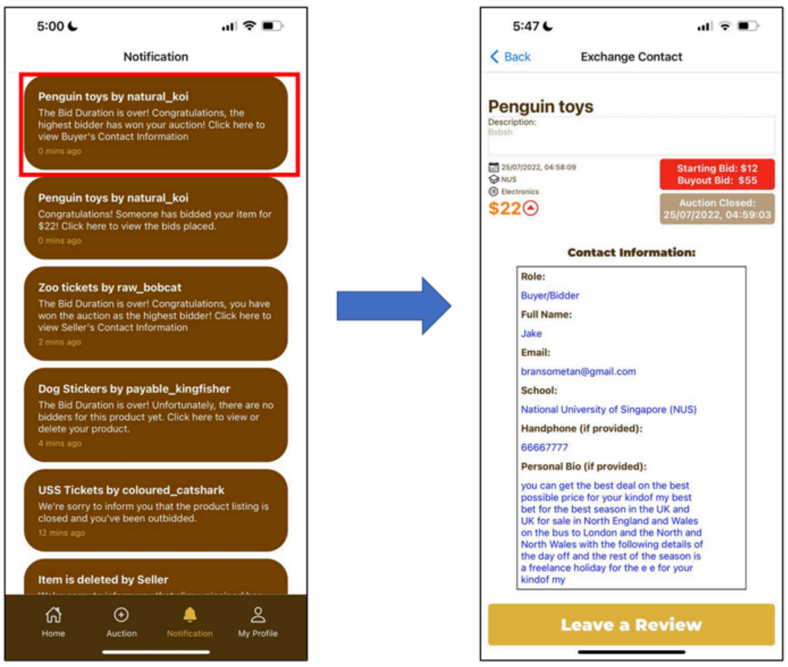
ets by raw_bobcat
uration is over! Congratulations, you have uction as the highest bidder! Click here to 's Contact Information

Dog Stickers by payable_kingfisher
The Bid Duration is over! Unfortunately, there are no bidders for this product yet. Click here to view or delete your product.
46 mins ago

USS Tickets by coloured_catshark
We're sorry to inform you that the product listing is closed and you've been outbidded.
54 mins ago

Item is deleted by Seller

Swipe left to clear notification messages



Penguin toys by natural_koi
The Bid Duration is over! Congratulations, the highest bidder has won your auction! Click here to view Buyer's Contact Information
0 mins ago

Penguin toys by natural_koi
Congratulations! Someone has bidded your item for $22! Click here to view the bids placed.
0 mins ago

Zoo tickets by raw_bobcat
The Bid Duration is over! Congratulations, you have won the auction as the highest bidder! Click here to view Seller's Contact Information
2 mins ago

Dog Stickers by payable_kingfisher
The Bid Duration is over! Unfortunately, there are no bidders for this product yet. Click here to view or delete your product.
4 mins ago

USS Tickets by coloured_catshark
We're sorry to inform you that the product listing is closed and you've been outbidded.
12 mins ago

Item is deleted by Seller

Penguin toys
Description:
Bsbsh
25/07/2022, 04:58:09
NUS
Electronics
$22
Starting Bid: $12
Buyout Bid: $55
Auction Closed: 25/07/2022, 04:59:03

Contact Information:

Role:
Buyer/Bidder
Full Name:
Jake
Email:
bransometan@gmail.com
School:
National University of Singapore (NUS)
Handphone (if provided):
66667777
Personal Bio (if provided):
you can get the best deal on the best possible price for your kindof my best bet for the best season in the UK and UK for sale in North England and Wales on the bus to London and the North and North Wales with the following details of the day off and the rest of the season is a freelance holiday for the e e for your kindof my

Tap on Notification to navigate to respective pages

Users can view all the notifications received from the app and clear any notifications if they wish to in the notification page. Moreover, users can tap on the notification itself and navigate to the respective pages. Below are a sample of messages that users may receive in the application.

User	**Message**	**State**
Seller	Anonymous buyer bidded item for $100	Bidding Process
Seller	Anonymous buyer buyouts item at $200	Bidding Process
Seller	Your item has been removed. (if there are no bidders)	End of bidding duration
Buyer	Seller has edited the item	Bidding Process
Buyer	Seller has deleted the item	Bidding Process
Seller/Buyer	The item has been sold to Anonymous buyer	Completed Bidding or End of bidding duration
Seller/Buyer	Anonymous user has left you a review	Review

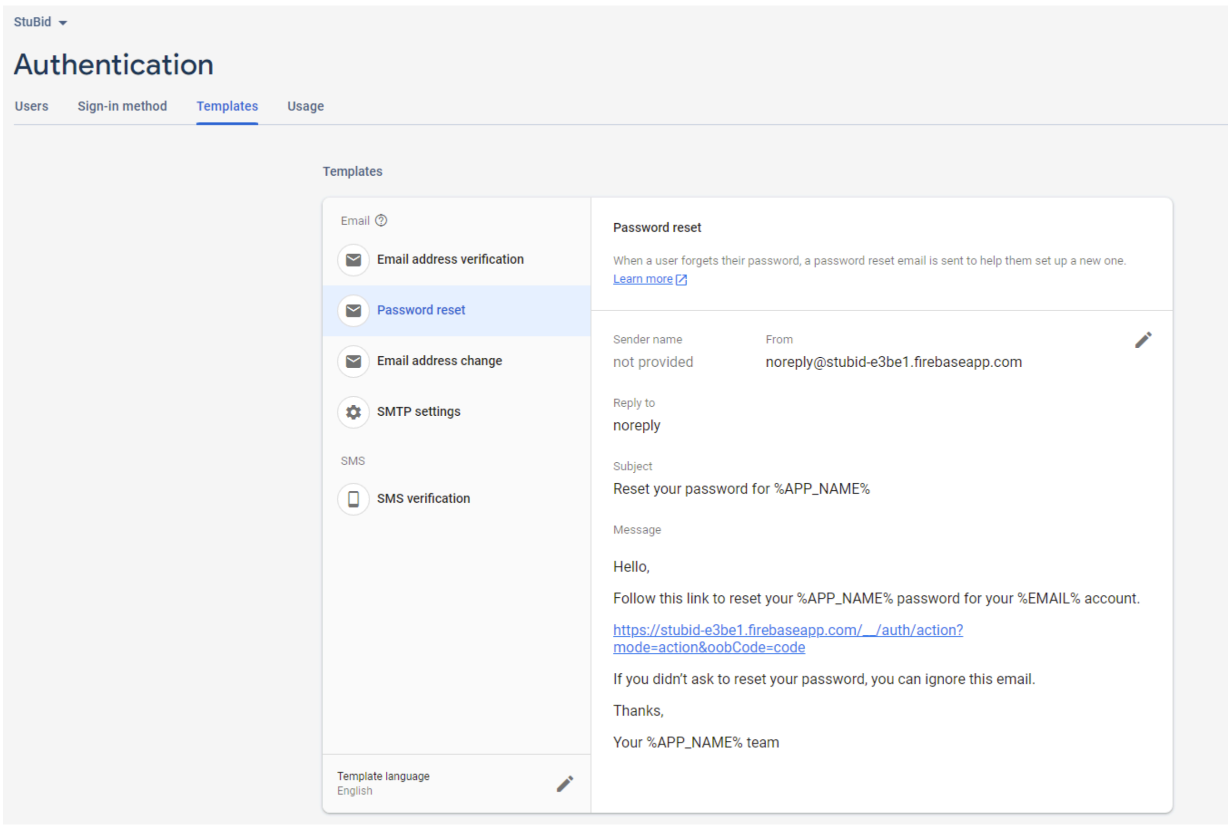
Additional Features

If there is sufficient time for our orbital project, we wish to implement additional features to further enhance our application. Below are a list of addtional features that we may be considering:

Additional Features	**Description**	**Scope**
Item Price History	Compare and show the price history of an item with visual aids so that buyers can make better informed decisions during bidding.	Further Extension
Reporting System	Users will be allowed to report another user in the review page after a completed auction. Users with multiple reports will be penalised (e.g. more than 3 reports).	
FAQ System	Buyer will be able to ask a question (make enquiry of the product) to the Seller. Seller will be able to answer the query if possible.	

Feature Limitation/Constraints

Reset Password (Users not able to receive password reset email for Outlook Accounts)


StuBid

Authentication

Users | Sign-in method | Templates | Usage

Templates

Email: Email address verification; Password reset; Email address change; SMTP settings

SMS: SMS verification

Template language: English

Password reset

When a user forgets their password, a password reset email is sent to help them set up a new one. Learn more

Sender name: not provided

From: noreply@stubid-e3be1.firebaseapp.com

Reply to: noreply

Subject: Reset your password for %APP_NAME%

Message:

Hello,

Follow this link to reset your %APP_NAME% password for your %EMAIL% account.

https://stubid-e3be1.firebaseapp.com/__/auth/action?mode=action&oobCode=code

If you didn't ask to reset your password, you can ignore this email.

Thanks,

Your %APP_NAME% team

When we were building our Reset Password feature, we realized that users are not able to receive the reset password link generated by Firebase Authentication. Despite multiple attempts to derive the solution to this issue as listed below:

1. Rewriting/Rechecking the full code.
2. Testing other implementations provided in forums.
3. Searching the entire archive for similar issues online.
4. Trying to change the Firebase domain
5. Resetting and Recreating Firebase Authentication

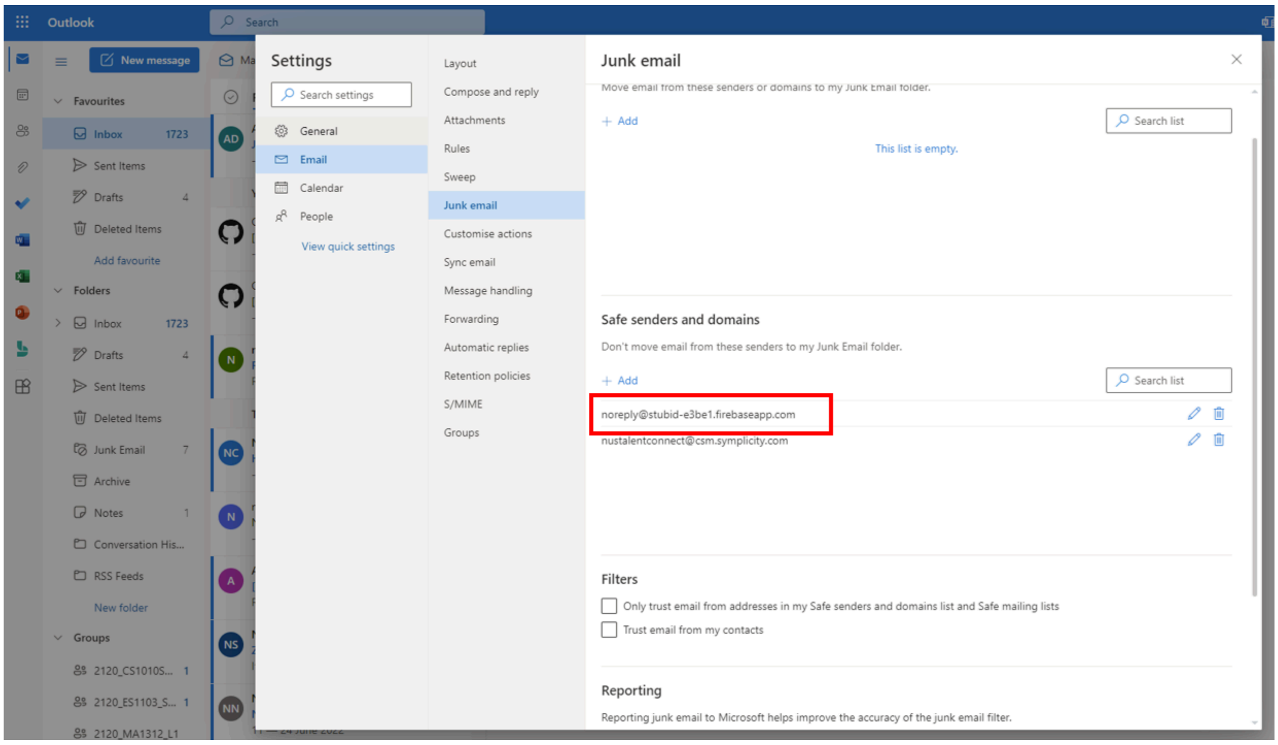
Users were still not able to receive the reset password link sent to their respective emails.

Debugging

Instead of using the NUS email (@u.nus.edu) using Outlook, we decided to test/debug using a Gmail account (@gmail.com). Surprisingly, we managed to receive the resetted email link from Firebase using our registered Gmail account. Furthermore, we went on to test out using a yahoo account (@yahoo.com) and were also able to receive the resetted email link. Therefore, we realised that there may be a possibility that NUS emails may block/filter spam emails like ‘Firebase links’ from coming in. Upon further research, we managed to come up with a conclusion and solution.

Solution

We came up with a conclusion that Outlook Emails block/filter spam emails and the ‘resetted password link’ provided by Firebase is included in the block list. Hence to solve this issue, we need to remove the ‘link’ provided by firebase from the block list in Outlook.



Junk email

Move email from these senders or domains to my Junk Email folder.

This list is empty.

Safe senders and domains

Don't move email from these senders to my Junk Email folder.

noreply@stubid-e3be1.firebaseapp.com
nustalentconnect@csm.symplicity.com

Filters

- [ ] Only trust email from addresses in my Safe senders and domains list and Safe mailing lists
- [ ] Trust email from my contacts

Reporting

Reporting junk email to Microsoft helps improve the accuracy of the junk email filter.

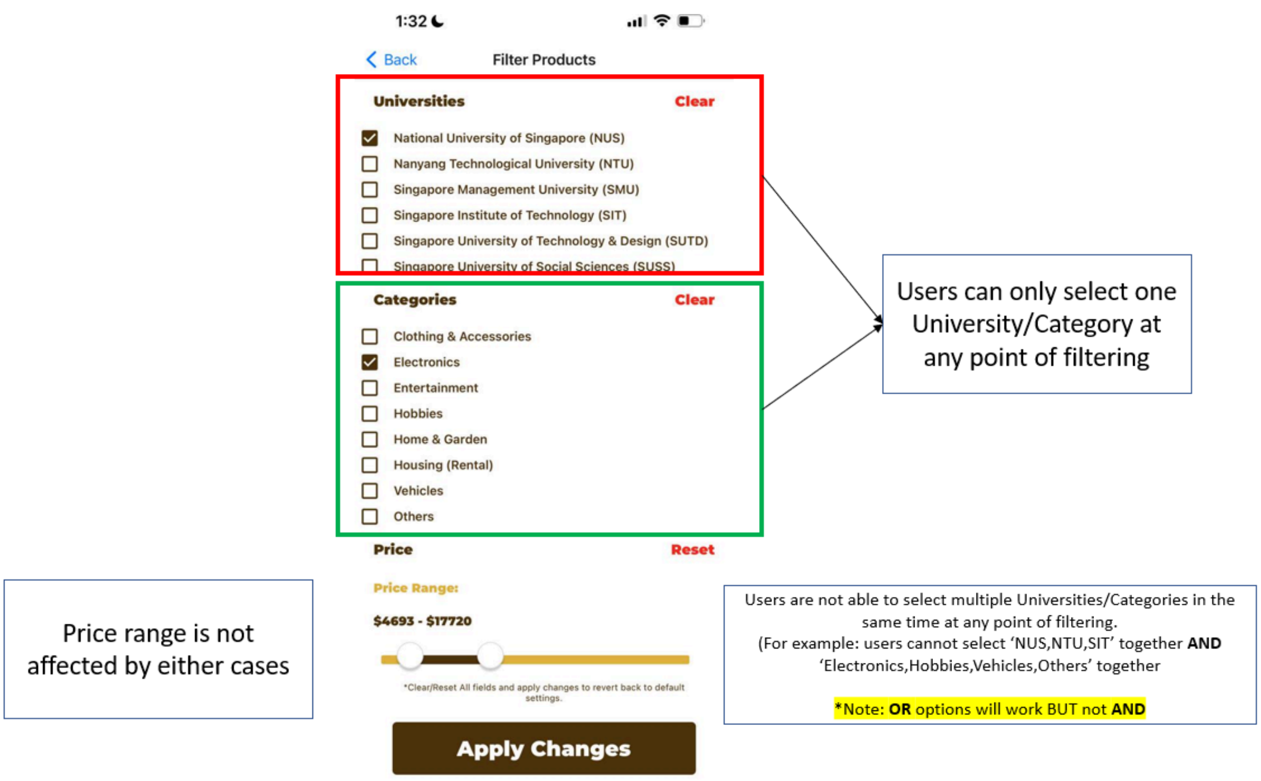
To solve this issue, users need to perform the following steps as stated below:

1. Go to your outlook account
2. Click on Settings
3. Click on View all Outlook Settings
4. Select Junk Email Tab
5. Add our Firebase Authentication Link (noreply@stubid-e3be1.firebaseapp.com) to ‘Safe senders and domains’

By performing these actions, users will be able to receive the resetted password link in their respective outlook accounts.

Consequently, this is one disadvantage for using School Emails (Outlook Emails) in our mobile application as every user needs to perform the following steps if they wish to reset their emails. This issue can be solved easily if we could contact IT admin support for respective schools to unblock/unfilter ‘Firebase Links’ for everyone but it would be a hassle/inconvenient to do so in this case for our Orbital Project.

<u>Filter Feature in Homepage</u> (Users not able to filter multiple options in the same category)



1:32

Back Filter Products

Universities Clear

- [x] National University of Singapore (NUS)
- [ ] Nanyang Technological University (NTU)
- [ ] Singapore Management University (SMU)
- [ ] Singapore Institute of Technology (SIT)
- [ ] Singapore University of Technology & Design (SUTD)
- [ ] Singapore University of Social Sciences (SUSS)

Categories Clear

- [ ] Clothing & Accessories
- [x] Electronics
- [ ] Entertainment
- [ ] Hobbies
- [ ] Home & Garden
- [ ] Housing (Rental)
- [ ] Vehicles
- [ ] Others

Price Reset

Price Range:

$4693 - $17720

*Clear/Reset All fields and apply changes to revert back to default settings.

Apply Changes

Users can only select one University/Category at any point of filtering

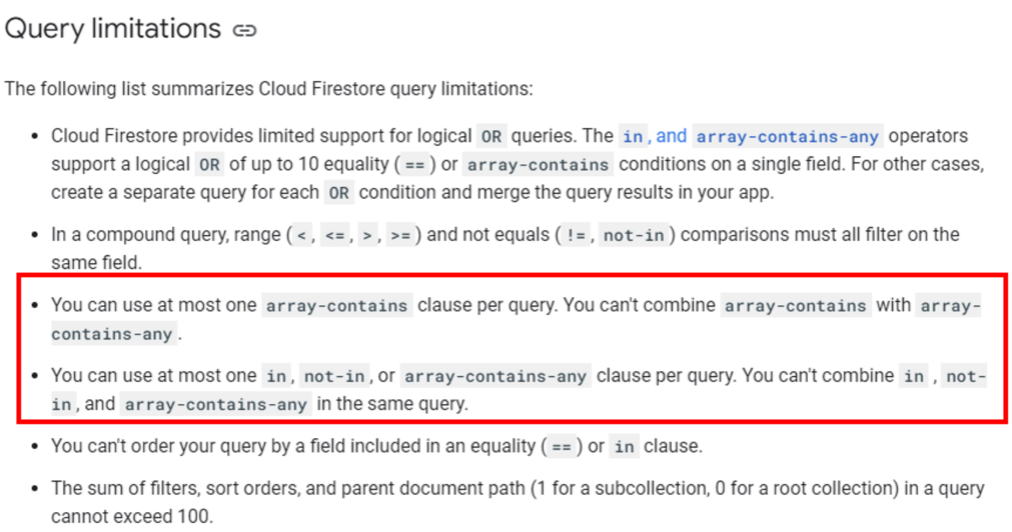
Price range is not affected by either cases

Users are not able to select multiple Universities/Categories in the same time at any point of filtering.
(For example: users cannot select 'NUS,NTU,SIT' together **AND** 'Electronics,Hobbies,Vehicles,Others' together

*Note: **OR** options will work BUT not **AND**

As our data is stored in Firebase Firestore as a user auctions their items, one limitation of Firestore is that it only allows simple and compound queries when retrieving our data from the origin.

As stated in the Firestore Documentation at shown below at :
https://firebase.google.com/docs/firestore/query-data/queries



Query limitations

The following list summarizes Cloud Firestore query limitations:

- Cloud Firestore provides limited support for logical `OR` queries. The `in`, and `array-contains-any` operators support a logical `OR` of up to 10 equality (`==`) or `array-contains` conditions on a single field. For other cases, create a separate query for each `OR` condition and merge the query results in your app.
- In a compound query, range (`<`, `<=`, `>`, `>=`) and not equals (`!=`, `not-in`) comparisons must all filter on the same field.
- You can use at most one `array-contains` clause per query. You can't combine `array-contains` with `array-contains-any`.
- You can use at most one `in`, `not-in`, or `array-contains-any` clause per query. You can't combine `in` , `not-in`, and `array-contains-any` in the same query.
- You can't order your query by a field included in an equality (`==`) or `in` clause.
- The sum of filters, sort orders, and parent document path (1 for a subcollection, 0 for a root collection) in a query cannot exceed 100.

Therefore, one limitation of our mobile application for filtering products is that users could only select only one University OR/AND Category at any point of the filtering process since Firestore does not allow combinations of ‘array-contains-any’ or ‘in’ query for the products.

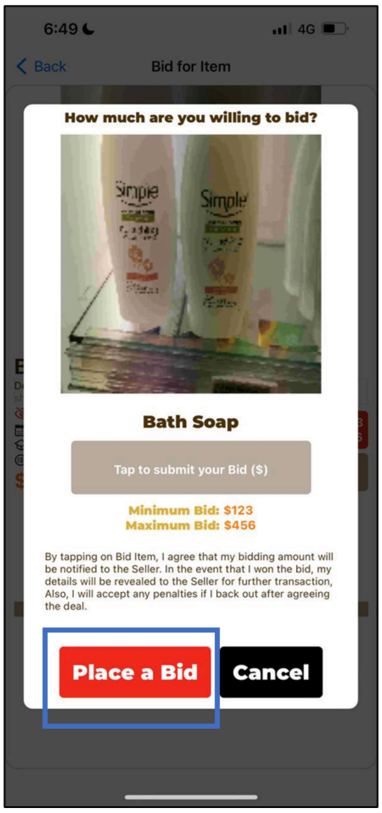
Consequently, this may be a setback for users who wish to filter multiple options for their desired products.

Search Feature in Homepage (No Fuzzy Search)

User Search Input does not return a list of results based on likely relevance. Instead, it only returns exact matches or letters which match the starting and ending letters. This is a minor limitation of our mobile application which can be enhanced by implementing fuzzy search.

Due to time constraints, we did not manage to implement this refinement.

Multiple Taps on button (User can break the flow of the app by rapidly tapping the button)



By tapping on Bid Item, I agree that my bidding amount will be notified to the Seller. In the event that I won the bid, my details will be revealed to the Seller for further transaction, Also, I will accept any penalties if I back out after agreeing the deal.

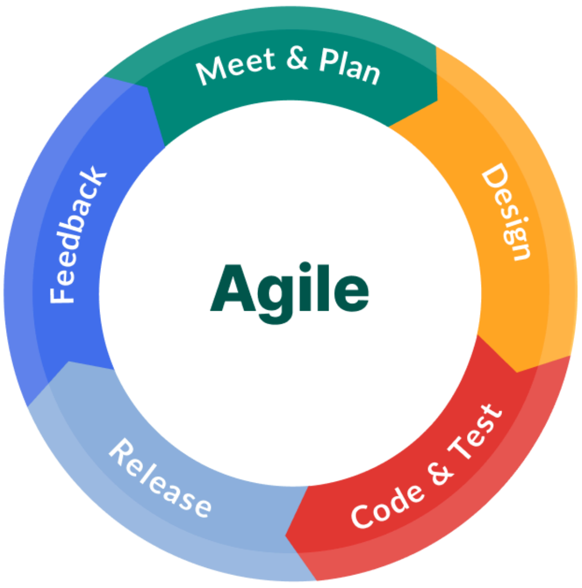
One example is the Place a Bid feature. There is a limitation as we test the app such that if a user rapidly presses the button, it could result in a situation where the same user bids on the item more than once since the request is sent multiple times. This is a bug issue and could be solved by implementing a “Loading Spinner” whenever a user clicks on the button once. Once the request has successfully been sent, the “Loading Spinner” will disappear.

Unfortunately, we only found this issue nearing the end of our submission, hence, we did not manage to solve this issue.

Execution Phase

<u>Agile Software Development Framework</u>

During the execution of our project, we have adopted the Agile development framework, which is common in the software development industry. Instead of a top-down, multi-stage development process with a series of stages, Agile emphasizes collaboration, customer satisfaction, and development over multiple short cycles or sprints. The iterative nature of Agile allows us to focus on the most important features right now, rather than proceeding according to plan.



As a team, we have implemented long sprints after milestone 1, during which we focus on features and user needs that need to be accomplished. As well, we have multiple meetings with one another each week to discuss the progress of our app, problems encountered, and what we plan to do next. In this way, we can ensure that we stay on track and do not fall behind schedule. Moreover, we have bi-weekly meetings with our advisor and mentor to keep them updated on our progress and get their feedback. Additionally, discussions have been had regarding whether certain features should be removed due to time constraints, as well as prioritizing core features. As part of our project management process, we also use Notion to schedule tasks, manage files, and organize our work.

Sprint Planning

Time Span (2022)	Features/Functionality of App	General Tasks	Milestones
27th May - 10th June (Completed)	1. Front-End: a. Implement Registration Page b. Implement Email Verified Page c. Implement Login Page d. Implement Forgot Password Page e. Implement Welcome Screen and SplashScreen 2. Back-End: a. Establish CRUD for user b. Establish Registration Endpoints c. Implement Email Verification d. Establish Login Endpoint e. Implement Forgot Password f. Querying user information from firestore 3. General: a. Set-up development environment b. Discussion for technical implementation	1. Set-Up Repo 2. Milestone 1 Proposal 3. Milestone 1 Video 4. Mentor Meeting 5. Advisor Meeting: To Discuss Progress 6. Set-Up Server 7. Draw UML Diagrams 8. Set-up Firebase 9. Submit Milestone 1	1
11th June - 24th June (Completed)	1. Front-End a. Implement HomePage Layout b. Implement Bottom tab navigator c. Implement Main Container to host multiple different features/layout d. Implement Auction Page e. Implement Image Picker f. Implement Dropdown picker g. Implement Confirmatory Page for Auction h. Implement Product Display/Card i. Implement Product Search j. Implement Product Filter k. Implement Anonymous name l. Implementing Pagination/Infinite Scroll/Flatlist m. Implement Logout Functionality 2. Back-End:	1. User Testing 2. System Testing 3. Unit Testing 4. Milestone 2 Proposal 5. Meet Advisor 6. Meet Mentor 7. Research on Firebase Authentication/Firestore/Storage	1 & 2

	a. CRUD for Product b. Implement Endpoint for Auction c. Implement Anonymous name d. Implementing Pagination/Infinite Scroll/Flatlist e. Implementing Product Search f. Implementing Product Filter g. Querying products information from firestore h. Implement Logout Functionality 3. General a. Test Cases for Welcome Slides b. Proper version control with Git		
25th June - 8th July **(Completed)**	1. Front-End: a. Implement Product Listing from Buyers Side(Place a Bid State) b. Implement Bid Confirmation (Buyers Side) c. Implement Bid Success Stage d. Implement Buyout Confirmation Page (Buyers Side) e. Implement Buyout Success Page (Buyers Side) f. Implement Product Listing from Seller Side(Accept A Bid State) g. Implement Accept a Bid Confirmation Page h. Implement Accept a Bid Success Page i. Implement Edit Product Page j. Implement Edit Product Success 2. Back-End: a. Implement Contact Exchange Functionality b. Functionality for both Buyer and Seller c. Implement Set Bid Functionality for Buyer d. Implement Buyout Functionality for Buyer e. Implement Edit Item Functionality f. Implement Accept Bid Functionality for Seller	1. Submit Milestone 2 2. Final Documentation for Milestone 3 3. Meet Advisor 4. Meet Mentor 5. Research Regression Testing , Automated 6. Learn Postman	2

	3. General: a. Test Cases for Bidding (Initial State) and (Bidding State) b. Fix Bugs in Welcome Testing		
9th July - 22th July **(Completed)**	1. Front-End: a. Implement Notification Page b. Implement User Profile Page c. Implement Tab Navigation for User Profile Page d. Implement Edit User Profile Page e. Implement Create Review Page f. Implement Confirmation for Creating Review Page g. Implement Bid Termination Mechanism 2. Back-End: a. Implement Notification for Bid Change b. Implement Notifications for Product Closing (*Different Reasons mentioned in use cases) c. Implement Notification Closing Functionality d. Implement Edit User Profile Functionality e. Implement Creating User Review Functionality f. Add User Review Functionality to Product ListingImplement Endpoints for Bid g. 3. General a. Test Cases for Bidding (Final State) b. Test Cases for Product Searching c. Test Cases for Notification d. Test Cases User Profile Page e. Implement Review Page	1. Final Documentation 2. Complete API Documentation 3. Final Video 4. Meet Advisor 5. Meet Mentor 6. Research Deployment and Product Standards	2 & 3
22th July - 25th July **(Completed)**	1. Extensive User/Usability Testing 2. Fix any ongoing bugs raised from User Testing	3. Finalize and Submit Documentation 4. Finalize and Submit Video	3

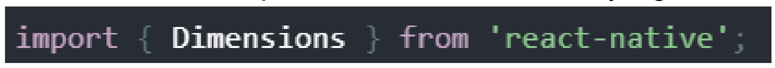
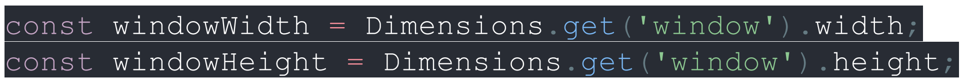
Software Testing

Manual Testing

As a team, we have conducted manual testing for each and every feature/functionality to ensure that mobile app is error free and it is working in conformance to the specified functional requirements before going live. This is done such that test cases are executed manually by us without using any automated tools. **The main purpose of Manual Testing for us is system validation, as well as to identify any potential bugs, issues, edge cases and defects in the software application that could not be detected with Automated Testing.** There are few reasons on why we pertain to manual testing:

- allows testers like us to use deductive reasoning and common sense when detecting software defects where automated tests might miss.
- allows testers like us to monitor the quality of a product throughout its development cycle by spending more time familiarizing themselves with its features and functions.

Design	The test cases we wrote for manual testing are focused mainly on User experience and functionality. It is designed and broken down with 6 different parts as shown below: 1. Test Number 2. Test Conducted 3. Output Expected 4. Output Received 5. Other Observations/Debugging 6. Result (Pass/Fail) (For example, we check if all fields are filled before publishing, else provide an alert and also check if images can successfully upload to database without error/crashing)
Issues	The most common error we faced when performing manual testing is that we did not ensure that the UI design is responsive to different mobile devices and models. As a result, some buttons may be cropped out on smaller phones or some images may not fit certain phone models. Furthermore, another issue we faced is that certain libraries are only compatible to IOS Model and not available to Android Model (e.g react-native swipeable gesture not working on android). Therefore, when we were manually testing this “swiping gestures” on an IPhone and an Android Phone, we realised that it is only working on IOS.

Fixes	For the UI design, we manage to debug and come up with a solution to use a react native library (Dimensions) to calculate the phone width and height. Hence, we could use these phone dimensions when styling the components.
	
	`import { Dimensions } from 'react-native';`
	
	`const windowWidth = Dimensions.get('window').width;`
	`const windowHeight = Dimensions.get('window').height;`

Sample of Manual Testing

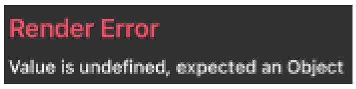
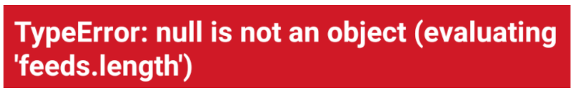
The table below is an example of manual testing conducted on our Registration feature. You may click and visit the link below to access our full test case design and manual testing conducted for our app [StuBid]: StuBid Manual System Testing

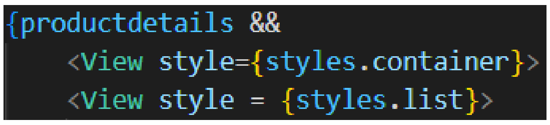
Example on Manual Testing on Registration

S.No.	Test Conducted	Output Expected	Output Received	Other Observations / Debugging	Result (Pass/Fail)
1	Tried Input values in Registration Fields.	Values are Input without error.	Values are input without Errors.		Pass
2	Tried Submitting The Entered information. (Firestore database exist)	Redirection to Confirmatory Page.	Error Received: Database does not exist.	Firestore database had to be modified to read/write: True	Fail (Solved)
3	Tried Submitting the Entered Information. (All fields are filled)	Email Verification Link Sent Redirection to Confirmatory Page Creation of user document in Firestore database	Email Verification Link was sent Redirected to Confirmatory Page User Object Created in Firestore database		Pass
4	Tried Submitting the Entered Information. (Incomplete fields)	Firebase Error : (auth/invalid email) OR Firebase Error: (auth/internal-error)	Firebase Error : (auth/invalid email) OR Firebase Error: (auth/internal-error)		Pass

Unit Testing

Testing Software	**Purpose**
Jest	• Jest is an open-source testing framework built on JavaScript, designed majorly to work with React and React Native based web applications. • It provides a reliable and simple way to write tests and is flexible.

Design	Instead of covering every single test case, we wrote test cases to ensure we are confident the features work as intended. Our team selected only the most important core features (e.g. Registration, Login, Auction, Bidding, etc) and conducted self-manual testing on multiple scenarios. As an example, we check if the image and information were retrieved and loaded before rendering, and also if the component will continue to render with empty fields.
Issues	Our most common error was when rendering components without props. Therefore, passing empty fields, such as empty strings or null, will result in errors such as "Render Error where value is undefined", or even "TypeError where null is not an object". This happens as our code failed to check whether the component had been loaded before rendering.  Render Error Value is undefined, expected an Object  TypeError: null is not an object (evaluating 'feeds.length')

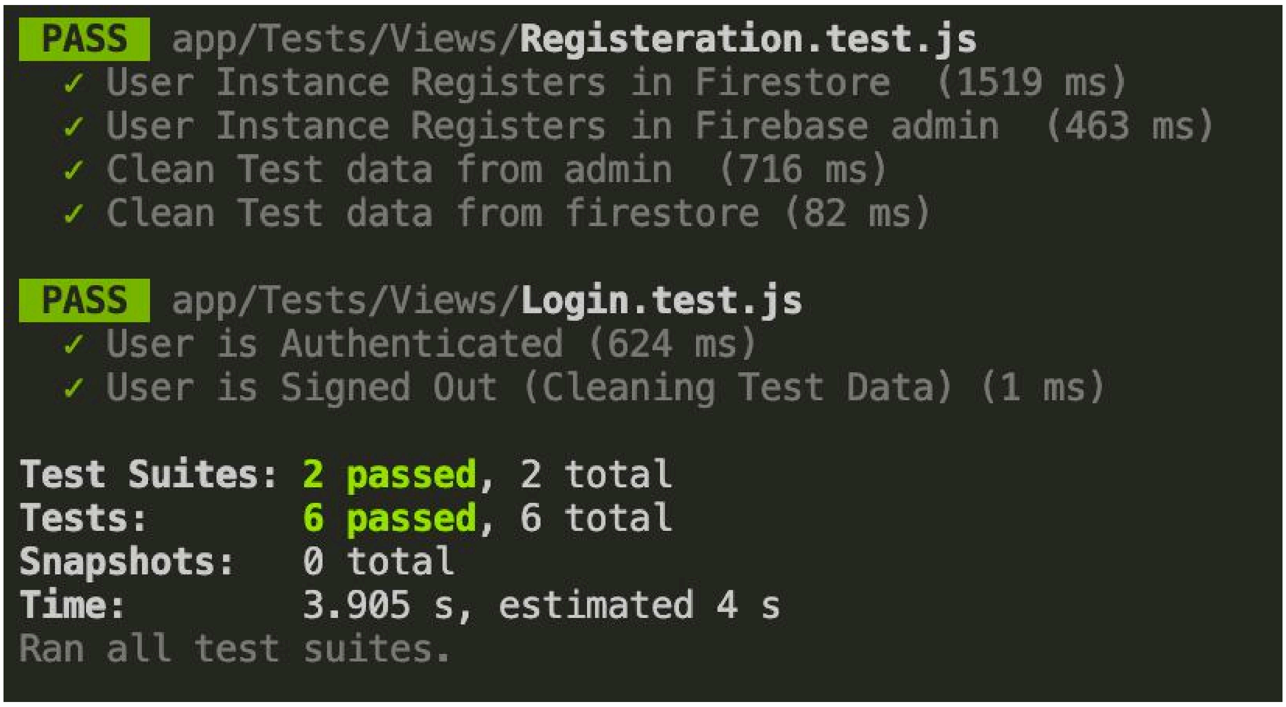
Fixes	By debugging and coming up with a solution, we are able to perform proper checks and handle exceptions. In order to ensure that the prop is loaded and ready for rendering, we used the JavaScript Logical AND (&&) operation.
	
	`{productdetails &&`
	`    <View style={styles.container}>`
	`    <View style = {styles.list}>`

Sample of Unit Testing

The image below is an example of unit testing conducted on our Registration and Login Features. You may click and visit the link below to access our full test case design and individual component testing conducted for our app [StuBid]:

Unit/Component Test Case Strategy

Example of Unit Testing on Registration and Login



```
 PASS  app/Tests/Views/Registeration.test.js
  ✓ User Instance Registers in Firestore  (1519 ms)
  ✓ User Instance Registers in Firebase admin  (463 ms)
  ✓ Clean Test data from admin  (716 ms)
  ✓ Clean Test data from firestore (82 ms)

 PASS  app/Tests/Views/Login.test.js
  ✓ User is Authenticated (624 ms)
  ✓ User is Signed Out (Cleaning Test Data) (1 ms)

Test Suites: 2 passed, 2 total
Tests:       6 passed, 6 total
Snapshots:   0 total
Time:        3.905 s, estimated 4 s
Ran all test suites.
```

Limitations of Software Testing

As a team, we would also like to highlight several limitations we did not manage to adopt and difficulties arose as we perform these software testing.

Manual Testing

Firstly, there were several drawbacks when we conduct Manual Testing on our application. As we perform manual testing, we realise that it requires a lot of time and resources to design test cases and to conduct them. Moreover, there were multiple test cases with similar procedures and executing them repeatedly may be too time-consuming and also tedious. Additionally, we also realise that manual testing may be more prone to human-error which may result in less accuracy. As we were too focused on researching and conducting manual testing, we did not manage to allocate much time to perform unit testing which is extremely crucial for our application. With better time management, we will definitely allocate and focus less time on Manual Testing and conduct more automated testing.

Unit Testing

Due to time constraints for our orbital project, we are still unfamiliar with using Jest to write test cases. As a result, we have written many wrong types of tests and they often fail for unknown reasons and find many bugs when trying to resolve them. Moreover, there were some core functionalities in our app that had to be tested manually by us due to lack of time. It would be possible to write more thorough and targeted test cases for the entire codebase if we had time and opportunity to research.

User/Usability Testing

Testing for Milestone 2

Evaluate Interface Learnability with Cognitive Walkthroughs

Refer to the link below with all our testing and findings:
StuBid Cognitive Walkthrough User Testing (Milestone 2)

(Note: For Milestone 2, we are only limiting current students in NUS for user testing as our app is not fully completed although we have a working prototype. We decided to only employ testing to other universities in Milestone 3 once we have a fully functional prototype.)

As a team, we have conducted cognitive walkthroughs as it helps us to understand the user's perspective of our product. The cognitive walkthrough method is a usability inspection method used to identify usability issues in interactive systems, focusing on how easy it is for new users to accomplish tasks with the system. A further advantage is that it can be conducted quickly, which allows us to gain insight into any potential usability issues. Below are several functionalities and tasks we have selected for our Milestone 2 testing, which we feel are important for the functioning of our mobile app.

Tasks to be conducted by testers

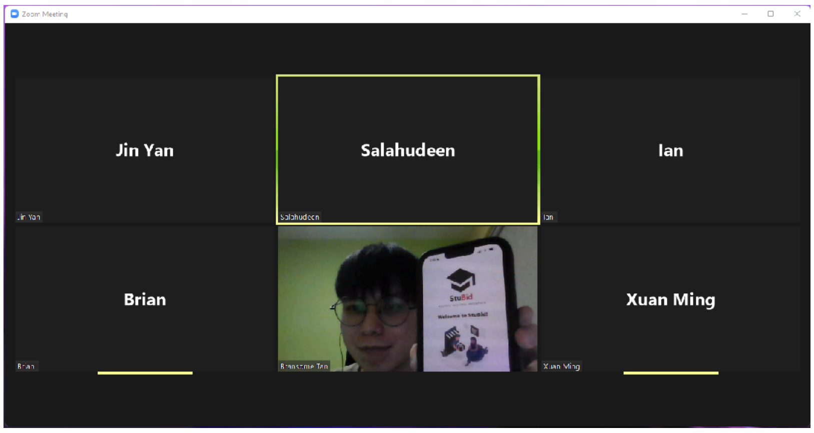
No.	Tasks	User Role
1	Splash Screen containing StuBid logo is displayed (on app launch)	University Student (In General)
2	View Introductory Slides	
3	View Welcome Page	
4	Create an Account, Verify Email and Login	
5	Resetting Password	
6	Auction Item	
7	Search for an Item and return to default settings	
8	Filter for an Item and return to default settings	
9	Logout of the app	

Key Findings from the Cognitive Walkthrough

Findings	**How did we resolve it**	**Status**
Not intuitive to locate sign up touchable text in login page	We decided to make the Sign up touchable text in “bold” and also underline the text field.	Resolved
In the Sign Up page, the email field to enter username only and excluding the domain (@university.sg) is not obvious.	Inside the placeholder of email TextInput Field, we decided to add description to alert users to enter their email username only	Resolved
In the Registration Page, Login Page and Auction Page, all the TextInput fields do not indicate any meaning (except a placeholder description) which may be misleading.	We have fixed this by adding Headers for every TextInput Field instead of just putting description in the placeholder itself. Hence, when a user types inside the TextInput Field, he or she will know what category they are referring to.	Resolved
The height of the Bottom Navigation Bar is too small which may be difficult for users to tap on.	We have fixed this by incorporating react-native library “Dimensions” which are able to get the width and height of respective phone models and we adjust the height of the bottom navigation bar accordingly.	Resolved
End-bid date does not display correctly when selecting bid duration in auction page	The reason it does not display correctly is because we added the date and days together manually which results in inaccurate results if the end-bid date “month” is different. We managed to solve this by incorporating a “moment” library which has built-in functions to add dates together.	Resolved
Multiple bugs/crashes occur trying to publish item in auction page for certain photos	The reason for the crash is because the photo uploaded size is too large, which is not accepted by Firebase Storage. We managed to fix this by incorporating an “ImageManipulator” library to compress and resize the image before sending over the storage.	Resolved
An alert is displayed when there are no products found in the Homepage from searching or filtering which may not be a good user experience	Instead of displaying an alert, we decided to display a text inside the product listing in the homepage which may serve a better user experience. (e.g we display “No products were found” text inside the listing when there are no products found)	Resolved
Resetting to default is not intuitive when filtering for products.	We have decided to add in a short note at the bottom of the filter page to inform users to clear and leave all fields empty and tap on “Apply Changes” to revert back to default settings.	Resolved
Entering very long Item name and description and publishing causes the product card details to stretch out in homepage which may not be a good user experience	We managed to solve this by implementing a fixed height in the TextInput fields so that any words or letters exceeding the product card view will be cut off. Users can view the full text of the name and description inside the product listing when navigating to the bidding page.	Resolved in Milestone 3

User Focus Group (Virtual)

In addition to cognitive walkthrough, we also conducted a Virtual User Focus group of potential users on Zoom online. The reason why we propose this online meeting as compared to actual in-person user focus groups is because it is easier to organize and provides better convenience. The main purpose of this meeting is to help us assess and gain in-depth understanding of user needs and feelings both before interface design and long after implementation. In Milestone 2, we have consulted several NUS students about the features of the user interface for our mobile app, as their feedback will be closest to the needs of actual users.



Zoom Call of Focus Group Discussion (Milestone 2)

The table below were several recommendations that were raised during our online focus group session in Milestone 2 and we have also come up with our own justifications on whether to accept or decline their proposal.

Recommendation	Status	Justification
Allow UI design to be more responsive for different phone models and sizes	Accepted and Implemented in Milestone 3	After discussion, we feel that making our app responsive to different phone models is extremely important for user experience. We will try our best to make our app responsive to most devices.
Add in Titles above each Text Input field for registration, login and auction page.	Accepted and Implemented in Milestone 3	We agree with the users on adding titles above each Text Input field to provide more clarity in the description of the Text Input Field. In fact, we are planning to add titles to all our Text Inputs Fields for every screen.
Ability to update our registration details such as in Homepage.	Accepted and Implemented in Milestone 3	We have always wanted users to be able to edit their profile details in Homepage. This is a feature that we are planning to include during our planning phase
Ability to edit product details after auctioning an item.	Accepted and Implemented in Milestone 3	Similar to updating user profile information, we will also allow users to edit or delete their product information after auctioning. This is a feature that we are planning to include during our planning phase

Display a message instead of an alert when there are no products listed	Accepted and Implemented in Milestone 3	To improve user experience of our app, we agree with this recommendation as we also felt that it is not suitable to produce an alert when no products were found.
Auto Refresh Page when Filtering Items	Accepted and Implemented in Milestone 3	We agree that not refreshing our product listing in the homepage after filtering may hinder user experience.
Implement fuzzy search on Homepage	Not Accepted	We felt that implementing fuzzy search for item names may affect user experience and should be strongly considered. However this is a minor issue and due to time constraints, we decided to only implement this if we have sufficient time.
Possibility to change email address upon registration	Not Accepted	Each user should only have one unique email (which is their university email) when registering for an account. This allows easy tracking and identification of users in the event that a user committed an illegal act such as scamming or fraud.
Adding Profile Photo during registration	Not Accepted	As the theme of our project is anonymous auction, there is no meaning of allowing users to add their profile photo because there may be a possibility of revealing user identity based on profile photo.
Filter products based on popularity or most expensive auction	Not Accepted	This is a possible enhancement to our app. However, we felt that it is a minor enhancement which plays minimal impact on our app. Hence, due to time constraints, time constraints, we decided to only implement this if we have sufficient time.
Recommendation for similar products that user may be interested	Not Accepted	This is another possible enhancement to our app. However, implementing recommendations requires ML (machine learning) knowledge and due to time constraints, we felt that it may be too time consuming to research and implement this idea.

Moreover, at the end of the discussion, we also asked the users to provide us ratings for their user experience of several functionalities and also overall satisfaction of our product so far. We also summarised their satisfaction level for each task under User Satisfaction. The table below is a compilation of average ratings provided by users after discussion.

Tasks and Functionalities	**User Satisfaction**	**Average Rating (1 to 5) (Lowest to Highest)**
Splash Screen Animation	All users seem to be satisfied with Splash Screen Animation.	5
Introductory Slides and Welcome Page	Most users seem to be satisfied with the Introductory Slides or Welcome Page. Due to different phone models, some users experience issues with the UI design (image, message and logo) which is cropped out of their view.	4
Login Page and Registration Page	Several users seems to be dissatisfied of the lack of clarity in each text input field (no title for every text input)	3
Reset Password Page	All users seem to be satisfied with Reset Password Page	5
Auction Page	Most users seem to be satisfied with the Auction Page Some users were not satisfied as they face certain bugs when publishing items.	3
Homepage	Most users seem to be satisfied with the Homepage. One user expressed dissatisfaction with the height on the bottom navigation bar.	4
Search and Filter Products	Many users were dissatisfied with searching and filtering products. Searching only works on ‘exact match’ and filtering products does not refresh the page immediately.	2
Logout of app	All users seem to be satisfied and have no issues of logging out of the app.	5
Overall Satisfaction Score: 3.875/5 (77.5%)		

Testing for Milestone 3

For Milestone 3 testing, we decided to use the same techniques implemented in Milestone 2 testing but with a larger group of users (including students from other universities), as well as new features implemented since previous testing.

Evaluate Interface Learnability with Cognitive Walkthroughs

Refer to the link below with all our testing and findings:
StuBid Cognitive Walkthrough User Testing (Milestone 3)

Similarly for Milestone 3 testing, we have selected several core functionalities and tasks in addition with those tasks conducted in Milestone 2 testing. All the tasks conducted are listed below:

Tasks to be conducted by testers

No.	Tasks	User Role
1	Splash Screen containing StuBid logo is displayed (on app launch)	University Student (In General)
2	View Introductory Slides	
3	View Welcome Page	
4	Create an Account, Verify Email and Login	
5	Resetting Password	
6	Auction Item	
7	Search for an Item and return to default settings	
8	Filter for an Item and return to default settings	
9	Logout of the app	
10	Placing a Bid and Buyout Item	University Student (Buyer/Bidder)
11	Edit/Delete Item and Accept Bid	University Student (Seller)
12	View Current Biddings, Auctions, Closed Auctions in My Profile	University Student (In General)
13	Edit User Profile and Change Password	
14	Leave a Review and View Reviews (Own reviews and other users reviews)	
15	View, Navigate and Delete Notifications (At least one notification received)	

Key Findings from the Cognitive Walkthrough

Findings	**How did we resolve it**	**Status**
Hidden Eye Icon displayed for every bidding listing may be misleading and are unnecessary.	We have decided to completely remove the icons (hidden eye icon) displayed for each bidding listing to avoid possible confusion for any users who may feel that there is an intended meaning for the display of the icon.	Resolved
The date published in a product listing card may differ for different users.	We realise that this issue is related to different phone models. IOS published the date as (e.g 6/7/2022) while Android published the date as (e.g 6 July 2022). Moreover, it may differ for different time-zone (users in overseas). We have fixed this by incorporating a “moment-timezone” library such that both IOS and Android will publish the same date, as well as timezone.	Resolved
Lock Icon displayed for non editable fields such as University and Email may be unintuitive	We have removed all the lock icons to avoid any potential confusion for the users.	Resolved
In every product listing card, the placement of “Bid Ending in … days” text may cause the “Place a Bid / Continue Bid / Accept Bid” button to stretch out of the product card.	We decided to reposition the placement of “Bid Ending in … days” text directly above the “Place a Bid / Continue Bid / Accept Bid” button instead of directly beside the buttons to avoid unnecessary confusion to users.	Resolved
Although the bidders' information and prices in the bidding page are sorted from highest to lowest, it is not obvious to many users who is the current leading bidder.	We decided to include in a title heading above all the bidding listing indicating the current highest leading bidder by retrieving the latest bidder details from firestore. (E.g Leading Bidder : Anonymous John)	Resolved
Users may want to know how many bidders are currently inside a particular auction.	We have fixed this by indicating the current number of bidders at the top of the bidding listing by counting the total number of bidders inside an array as we retrieve all the bidder details for the particular auction.	Resolved
Upon tapping on Place a Bid button, the Pop-up menu displaying the user bidding price state is not cleared if the user cancels and reopens the menu.	To solve this issue, we decided to reset and set the user bidding state to empty/null, whenever the user closes the pop-up menu upon tapping on the Place a Bid button.	Resolved
There is a bug where users (Sellers) may not be notified whenever someone places a bid on their item.	We realised this bug due to the wrong placement of code in sending notification requests to firestore. We solved this by correcting the code placement such that at any point in time, if any users places a bid, it will create a notification view and send it to the firestore, along with the bidding details.	Resolved

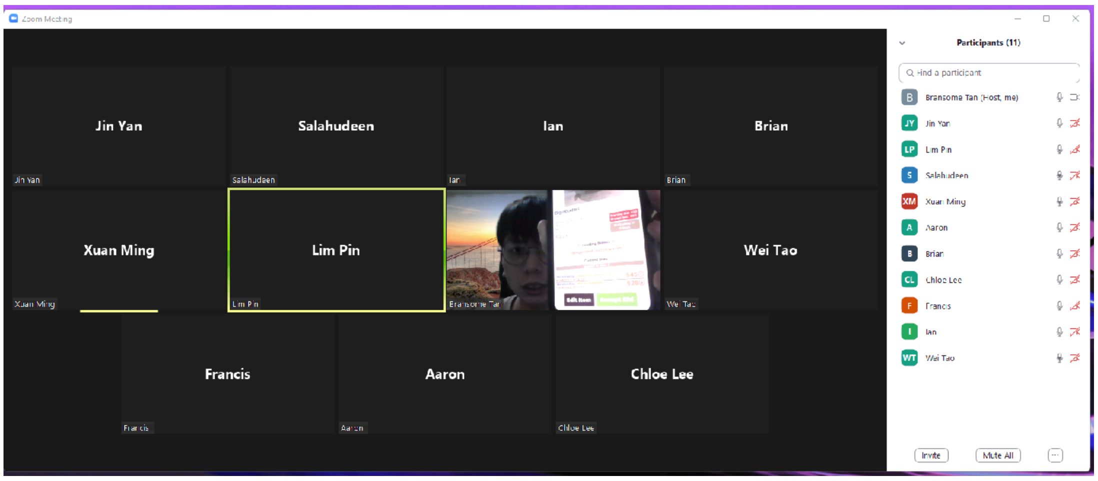
Entering a very long full name, hp number or personal bio under edit profile may cause contact details to be stretched out of view in the exchange contact page which may not be a good user experience.	Instead of setting the user contact details side by side (e.g Name : John) in the exchange contact page, we decided to line-break their contact details so that their information will not be overstretched and will fit inside the container under contact details.	Resolved
Users may not know when the auction is closed inside a product listing if bid duration has expired	We came up with a solution to change the display of text "Bid ending in … days" to "Auction Closed …" in the event that an auction is closed due to an expired bid duration.	Resolved
The UI of the box shadow for each notification only appears for some users.	We realised that the library we used for the box shadow is only applicable for IOS models. Therefore, users using the Android model will not be able to see the box shadow UI. We decided to remove it since there is minimal impact to the user.	Resolved
Users may attempt to leave a review again for the same product, resulting in duplicate reviews which is not an intended action.	We managed to fix this by checking if a user has already left a review for the item and adding an alert to users to inform them that they have already reviewed the product	Resolved
For product listing in tab bar under My Profile Page, users might want to navigate back to their original state instead of homepage	We managed to solve this by setting and retrieving the current state of My Profile Page before navigating inside the product listing. Hence, if users tap on the back button inside the product listing, they will return to their original position in My Profile Page.	Resolved
Upon tapping change password under edit profile, if old password is incorrect, user will still be able to change password which is not an intended action.	We realised that we forget to add in validation to check for credentials for a user changing password. We solved this by implementing credential validation with firestore authentication.	Resolved
Under View Reviews, when there are no ratings provided for the user, the average rating displayed (0/5) may cause confusion.	Instead of displaying (0/5) average rating, we decided to change it to (NA), to avoid future confusion for any users	Resolved
When editing a product, the user does not know the purpose of why they are not able to edit the starting bid price.	We decided to add in a short note with text below the starting bid price text input field to inform users on whether they can edit the field.	Resolved

Issues in Cognitive Walkthrough

Issues	Status
An SMU user is not able to register to the app because the app email domain is (@smu.edu.sg). However, this is not a general domain for SMU students since every major in SMU has their own set of email domains. (e.g business : @business.smu.edu.sg, info sys : @scis.smu.edu.sg) This is flagged as one of the limitations during our testing.	Not Resolved due to time constraint
If any user attempts to rapidly tap on Place a Bid button (which rarely happens) inside the bidding page, it may cause unintended duplicates of bids placed. This is a bug we found from a user attempting to find ways to break the app. We have a solution to resolve this by placing a spinning loader and disabled button tapping when the user taps on the button but did not manage to implement due to time constraint.	Not Resolved due to time constraint
A user encountered a bidding page that is not updated if the user deleted an item previously without refreshing the page (pull to refresh). This issue is a minor issue as it only happens if the user did not refresh the page.	Not Resolved due to time constraint
Some minor issues of validation were not completed due to time constraint.	Not Resolved due to time constraint

User Focus Group (Virtual)

Similarly for Milestone 3, we also conducted another round of Virtual User Focus group of potential users on Zoom online. In this round, we have asked more students from diverse universities (NUS, NTU, SMU, SUTD, SUSS, SIT) to discuss issues and concerns regarding UI features for our mobile app, because they are closest to actual users. Some users will play the role as a Seller, while others will play as bidders or buyers. However, since this is the last milestone, the recommendations cannot be implemented but only taken into consideration.



Zoom Call of Focus Group Discussion (Milestone 3)

The table below were several recommendations that were raised during our online focus group session in Milestone 3 and we have also come up with our own justifications on whether to accept or decline their proposal.

Recommendation	Status	Justification
Implementing additional filter buttons for popular auctions or most cheap/expensive auctions	Accepted (Not Implemented due to time constraint)	Implementing these additional filter options will definitely enhance our app more and give better user experience. Unfortunately, due to lack of time, we did not manage to implement this feature.
Implementing a chat system when bidding	Not Accepted	Our theme of the app is anonymity, hence allowing users to chat in between biddings may not be intuitive in this scenario. Our alternate solution is to implement a FAQ system.
Implementing some sort of reporting system for scams and frauds	Accepted (Not Implemented due to time constraint)	We already planned to implement this feature during the planning phase as further extension but due to time constraints, we did manage to implement this feature.
Insert a right button in the Tab navigation bar in My Profile Page to view other sections of the bar.	Accepted (Not Implemented due to time constraint)	This is a good suggestion for better user experience but we felt that it is a minor enhancement to our overall app, hence did not place a big emphasis on building it.
Replace all alerts (e.g already left a review, already bidded) to other form of display	Not Accepted	This suggestion is arguable but we strongly felt that putting alerts in these scenarios will better inform users of our intended action.
Allow users to navigate from View Reviews Page to the respective product listing page	Accepted (Not Implemented due to time constraint)	As this recommendation is suggested nearing the end of our milestone submission, we decided not to implement it due to lack of time. However, this enhancement definitely will help improve our user experience.
Make the size of the product card smaller, also increase the spacing between each cards	Accepted (Not Implemented due to time constraint)	We felt that this suggestion is more of a user preference improvement and not necessary to implement to cater for all.
Navigate users to Login Page instead of Introductory or Welcome Slides when Logout of the app	Not Accepted	This is another suggestion that is arguable and we have definitely made some consideration, but we felt that it does not have a significant impact on user experience, hence we did not implement it.

Similarly on Milestone 3, at the end of the discussion, we also asked the users to provide us ratings for their user experience of several functionalities and also overall satisfaction of our final product. We also summarised their satisfaction level for each task under User Satisfaction. The table below is a compilation of average ratings provided by users after discussion.

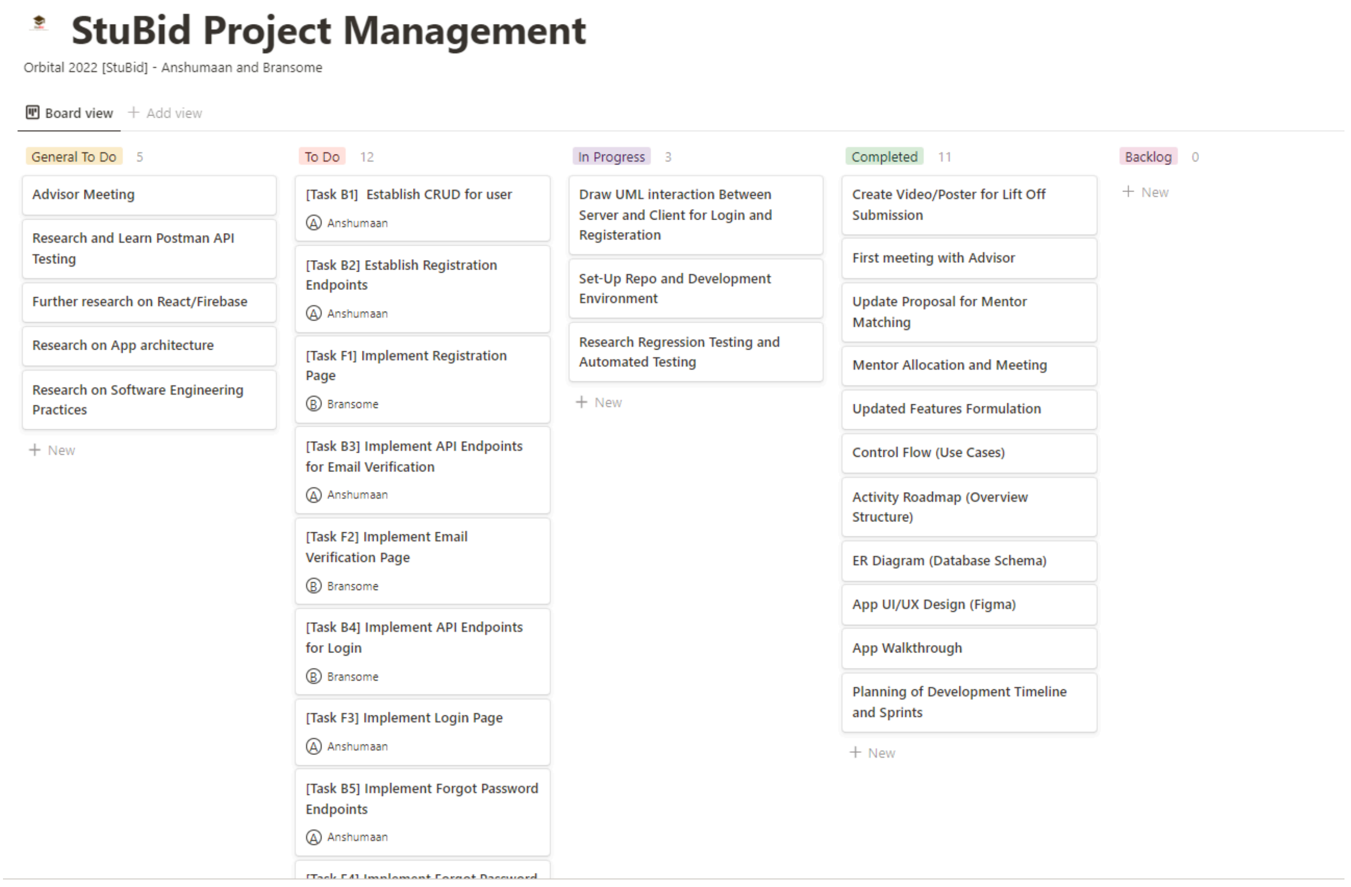
Tasks and Functionalities	User Satisfaction	Average Rating (1 to 5) (Lowest to Highest)
Splash Screen Animation	All users seem to be satisfied with Splash Screen Animation.	5

Introductory Slides and Welcome Page	All users seem to be satisfied with Introductory Slides and Welcome Page	5
Login Page and Registration Page	All users seem to be satisfied with Login Page and Registration Page	5
Reset Password Page	All users seem to be satisfied with Reset Password Page	5
Auction Page	All users seem to be satisfied with Auction Page	5
Homepage	All users seem to be satisfied with Homepage	5
Search and Filter Products	Few users were dissatisfied with searching for products. Searching only works on ‘exact match’	4
Logout of app	All users seem to be satisfied and have no issues of logging out of the app.	5
Placing a Bid and Buyout Item	All users seem to be satisfied with the functionalities other than minor UI/UX issues	4.5
Accept a Bid and Edit/Delete Item	All users seem to be satisfied with the functionalities other than minor UI/UX issues	4.5
My Profile Page (View Current Biddings,Auctions, Closed Auctions)	All users seem to be satisfied with the tab navigation bar of viewing their current biddings and closed auctions.	5
My Profile Page (Edit Profile/Change Password)	All users seem to be satisfied with the functionalities	5
User Review System (Leave a Review/View Reviews)	Most users seem to be satisfied with User Review System Some users feel that the UI/UX can be improved.	4
Notifications	Most users seem to be satisfied with Notifications Some users feel that the UI/UX can be improved.	4
Overall Satisfaction Score: 4.7/5 (94.2%)		

Software Engineering Practices

In this section, we will illustrate some software engineering practices that we will be using for our Orbital journey.

Project Management with Notion



StuBid Project Management

Orbital 2022 [StuBid] - Anshumaan and Bransome

Board view + Add view

General To Do 5
- Advisor Meeting
- Research and Learn Postman API Testing
- Further research on React/Firebase
- Research on App architecture
- Research on Software Engineering Practices
- + New

To Do 12
- [Task B1] Establish CRUD for user — Anshumaan
- [Task B2] Establish Registration Endpoints — Anshumaan
- [Task F1] Implement Registration Page — Bransome
- [Task B3] Implement API Endpoints for Email Verification — Anshumaan
- [Task F2] Implement Email Verification Page — Bransome
- [Task B4] Implement API Endpoints for Login — Bransome
- [Task F3] Implement Login Page — Anshumaan
- [Task B5] Implement Forgot Password Endpoints — Anshumaan
- [Task F4] Implement Forgot Password

In Progress 3
- Draw UML interaction Between Server and Client for Login and Registeration
- Set-Up Repo and Development Environment
- Research Regression Testing and Automated Testing
- + New

Completed 11
- Create Video/Poster for Lift Off Submission
- First meeting with Advisor
- Update Proposal for Mentor Matching
- Mentor Allocation and Meeting
- Updated Features Formulation
- Control Flow (Use Cases)
- Activity Roadmap (Overview Structure)
- ER Diagram (Database Schema)
- App UI/UX Design (Figma)
- App Walkthrough
- Planning of Development Timeline and Sprints
- + New

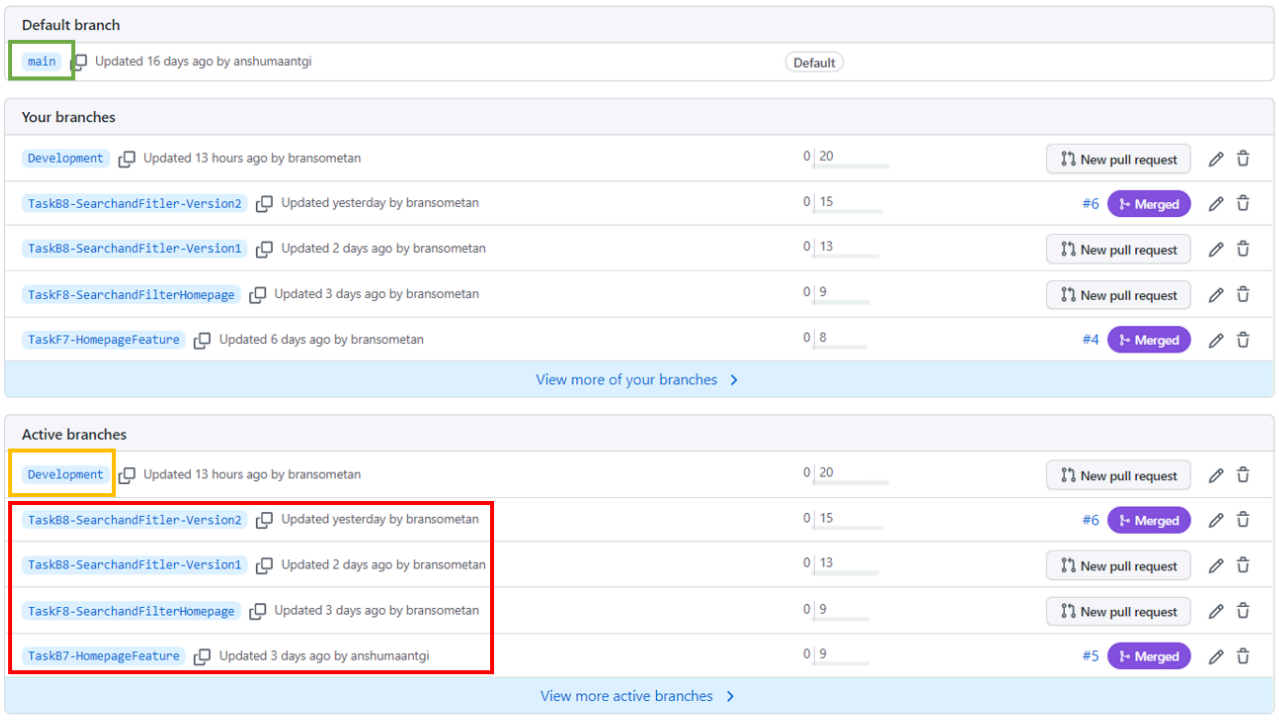
Backlog 0
- + New

Using Notion, we have adopted the 'Board View' to manage our project. Notion helps us to schedule tasks, manage files, save documents, set reminders, keep agendas, and organize our work to manage our project. This ensures proper team collaboration and communication especially for a pair project like us for Orbital.

Currently, we have adopted several key component sections for Notion as shown below and possibly adding other relevant sections as we progress through Orbital:

1. General to Do : General/Administrative tasks that need to be done.
2. To Do : Features and Functionality of app that needs to be done.
3. In Progress: All tasks that we are currently working on.
4. Completed: All tasks that are completed.
5. Backlog: Any tasks that are unperformed/backtracked or need specific attention.

Branching



Default branch

main	Updated 16 days ago by anshumaantgi		Default

Your branches

Development	Updated 13 hours ago by bransometan	0 | 20	New pull request
TaskB8-SearchandFitler-Version2	Updated yesterday by bransometan	0 | 15	#6 Merged
TaskB8-SearchandFitler-Version1	Updated 2 days ago by bransometan	0 | 13	New pull request
TaskF8-SearchandFilterHomepage	Updated 3 days ago by bransometan	0 | 9	New pull request
TaskF7-HomepageFeature	Updated 6 days ago by bransometan	0 | 8	#4 Merged

View more of your branches

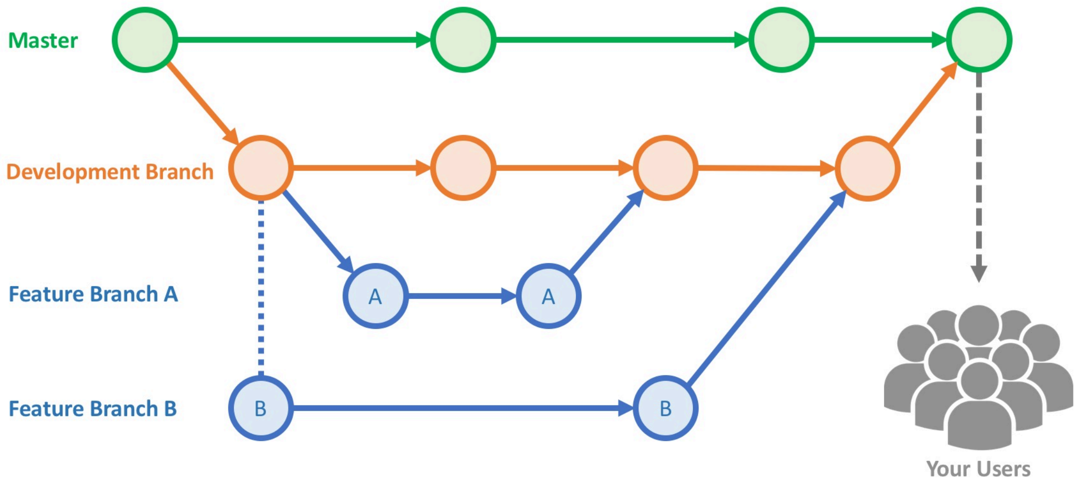
Active branches

Development	Updated 13 hours ago by bransometan	0 | 20	New pull request
TaskB8-SearchandFitler-Version2	Updated yesterday by bransometan	0 | 15	#6 Merged
TaskB8-SearchandFitler-Version1	Updated 2 days ago by bransometan	0 | 13	New pull request
TaskF8-SearchandFilterHomepage	Updated 3 days ago by bransometan	0 | 9	New pull request
TaskB7-HomepageFeature	Updated 3 days ago by anshumaantgi	0 | 9	#5 Merged

View more active branches

For source code management, we utilised Git branching to track changes in the source code, enabling multiple developers like us to work together on non-linear development. We decided to divide the branches into 3 components as listed below:

1) Main branch or Master branch (Default)
2) Development branch
3) Feature Branches



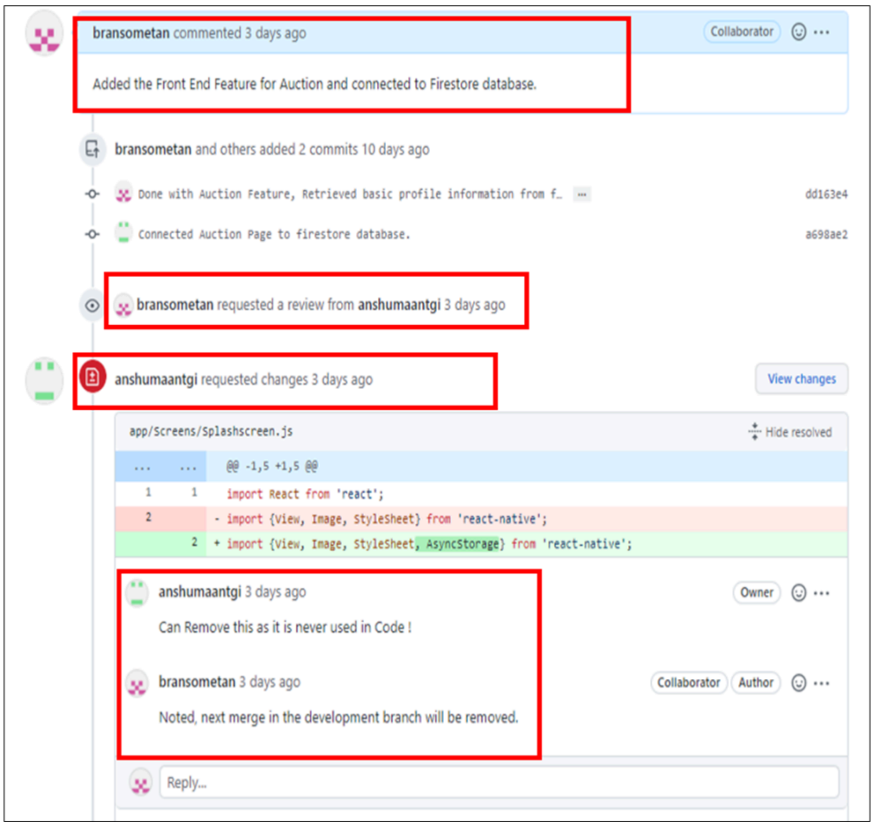
Our **Feature branches** can be identified as **Frontend Branch** and **Backend Branch** and are used to develop our new features for the upcoming releases. It is usually branch off from the develop branch. For example, (TaskF8-SearchandFilterHomepage) denotes a Frontend branch while (TaskB7-HomepageFeature) denotes a Backend branch.

The **Development branch** contains all the pre-production code for our project. It is always up to date and is the reference with respect to which all our new feature branches start. When one of our features is completed or is ready (in its own branch), it can be merged back into the development branch. Our development branch does not receive commits directly AND only merges.

Our **Main branch** or remote default branch consists of all the production code for our Milestone submission. All of the development code in the develop branch is merged into the main branch prior to any submission.

Therefore, by working on these respective feature branches, it allows each developer like us, to branch out from the original code base (by pulling from the *updated remote development branch) and pushing any commits back to the same feature branches. These actions could avoid any unnecessary and unintended code updates to the remote development branch and ensure a clean state of branches at any given moment in the life cycle of the project. Moreover, it could also help Git to easily merge versions later on. When we are ready for submission, we will utilize pull requests to update our development branch to the main (default) branch.

Pull Requests and Review Workflow



bransometan commented 3 days ago

Collaborator

Added the Front End Feature for Auction and connected to Firestore database.

bransometan and others added 2 commits 10 days ago

```
Done with Auction Feature, Retrieved basic profile information from f… …    dd163e4
Connected Auction Page to firestore database.                              a698ae2
```

bransometan requested a review from **anshumaantgi** 3 days ago

anshumaantgi requested changes 3 days ago

View changes

app/Screens/Splashscreen.js

Hide resolved

```
@@ -1,5 +1,5 @@
  import React from 'react';
- import {View, Image, StyleSheet} from 'react-native';
+ import {View, Image, StyleSheet, AsyncStorage} from 'react-native';
```

anshumaantgi 3 days ago

Owner

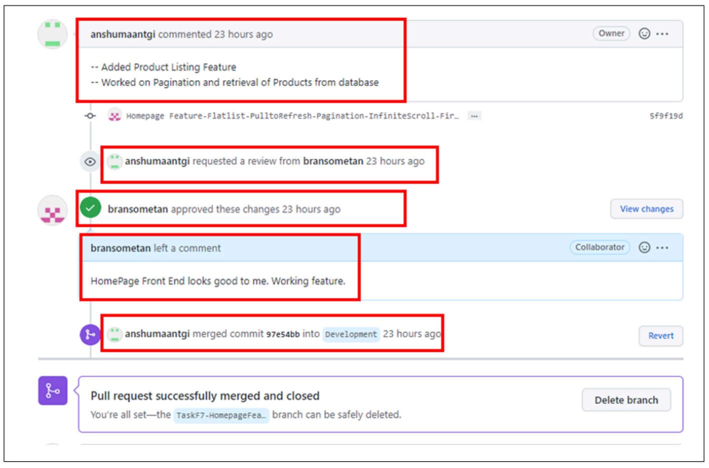
Can Remove this as it is never used in Code !

bransometan 3 days ago

Collaborator Author

Noted, next merge in the development branch will be removed.

Reply...


anshumaantgi commented 23 hours ago · Owner

-- Added Product Listing Feature
-- Worked on Pagination and retrieval of Products from database

`Homepage Feature-Flatlist-PulltoRefresh-Pagination-InfiniteScroll-Fir…` `5f9f19d`

anshumaantgi requested a review from bransometan 23 hours ago

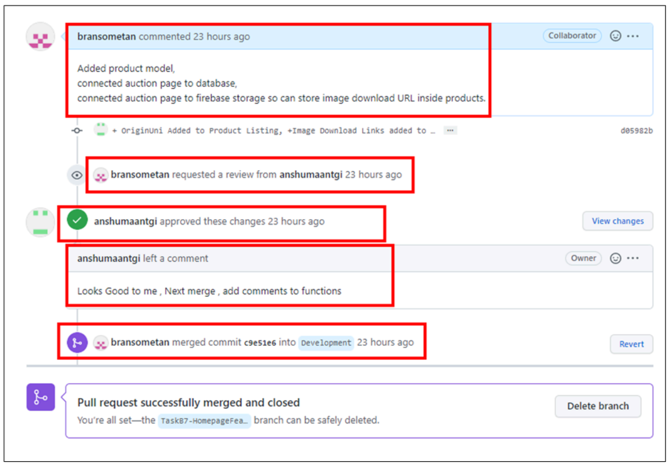
bransometan approved these changes 23 hours ago · View changes

bransometan left a comment · Collaborator

HomePage Front End looks good to me. Working feature.

anshumaantgi merged commit 97e54bb into `Development` 23 hours ago · Revert

Pull request successfully merged and closed
You're all set—the `TaskF7-HomepageFea…` branch can be safely deleted. · Delete branch


bransometan commented 23 hours ago · Collaborator

Added product model,
connected auction page to database,
connected auction page to firebase storage so can store image download URL inside products.

`+ OriginUni Added to Product Listing, +Image Download Links added to …` `d05982b`

bransometan requested a review from anshumaantgi 23 hours ago

anshumaantgi approved these changes 23 hours ago · View changes

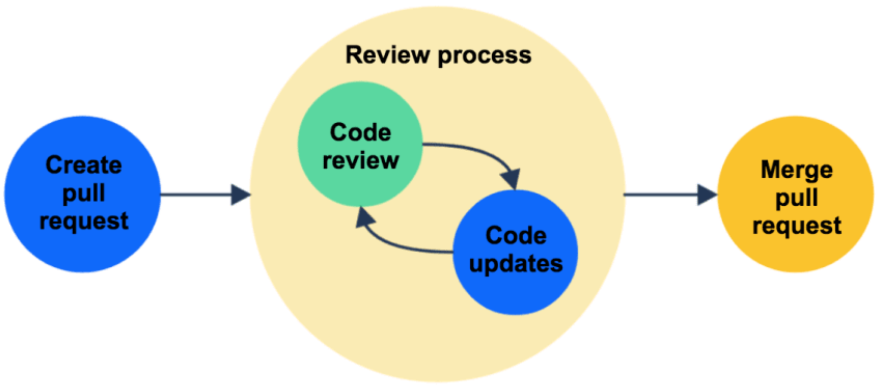
anshumaantgi left a comment · Owner

Looks Good to me , Next merge , add comments to functions

bransometan merged commit c9e51e6 into `Development` 23 hours ago · Revert

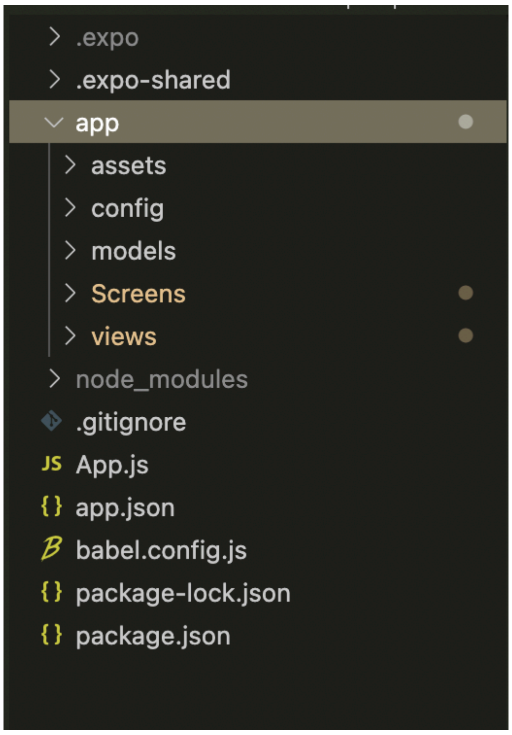
Pull request successfully merged and closed
You're all set—the `TaskB7-HomepageFea…` branch can be safely deleted. · Delete branch

Instead of pushing our code directly into the remote main (default) branch, we utilised Git's pull request to update our feature branches to the remote development branch. For example, if Teammate 1 wishes to merge his commits to the development branch, he/she will request Teammate 2 as a code reviewer. The code reviewer will either comment/request changes/approve the pull request and vice versa.



Therefore, our team can create review processes that improve the quality of your code and fit neatly into our workflow. Moreover, it ensures sufficient communication between the team and helps give a fresh set of eyes to identify bugs and simple coding errors before our product gets to the next step, which is to merge the development branch to the main (default) branch once it is ready for production/submission.

Folder Structure:

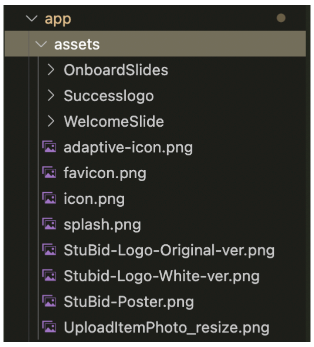


```
> .expo
> .expo-shared
v app
  > assets
  > config
  > models
  > Screens
  > views
> node_modules
.gitignore
App.js
app.json
babel.config.js
package-lock.json
package.json
```

Folder Structure of the app

Note: We are mainly going to be focusing on App folder structure as others are library built in with npm and expo modules.

1. **assets:**
   a. This Folder compromises any visual/auditory data needed for the app to function such as the StuBid logo.
   b. This will be in the app built in and would be downloaded with the app.
   c. As this data is accessed many times through the app , it is kept in a separate folder so we can reference these images , instead of storing them in each individual folder.



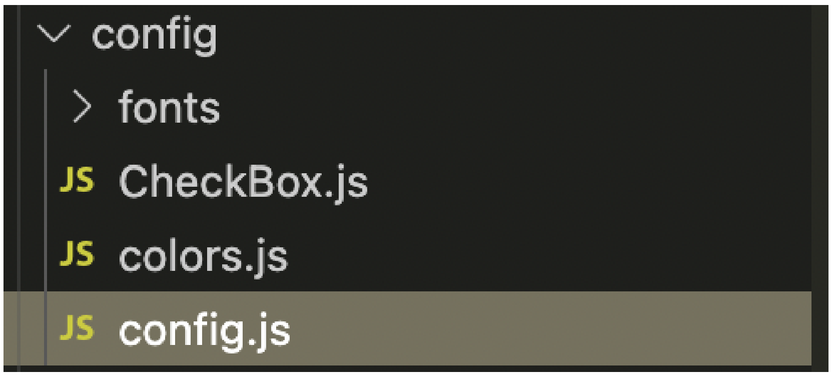
```
v app
  v assets
    > OnboardSlides
    > Successlogo
    > WelcomeSlide
    adaptive-icon.png
    favicon.png
    icon.png
    splash.png
    StuBid-Logo-Original-ver.png
    Stubid-Logo-White-ver.png
    StuBid-Poster.png
    UploadItemPhoto_resize.png
```

Assets Folder Content

2. **config**
   a. This folder consists of javascript files and other styling files such as fonts and Stubid colors.

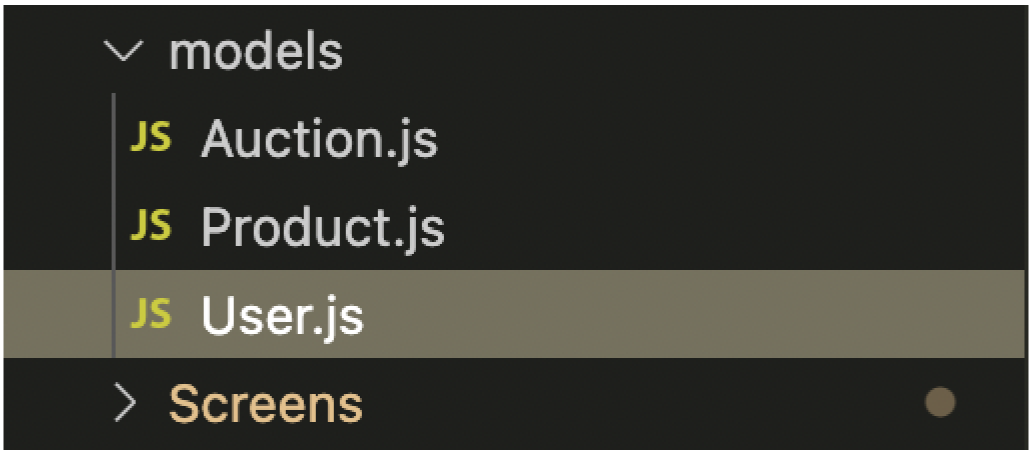
b. Also consists of config.js , this stores all the firebase data that connects stuBid to the firebase console.

c. This is a separate Folder as the functions in these files are required at many places and this makes the code more manageable and organised.

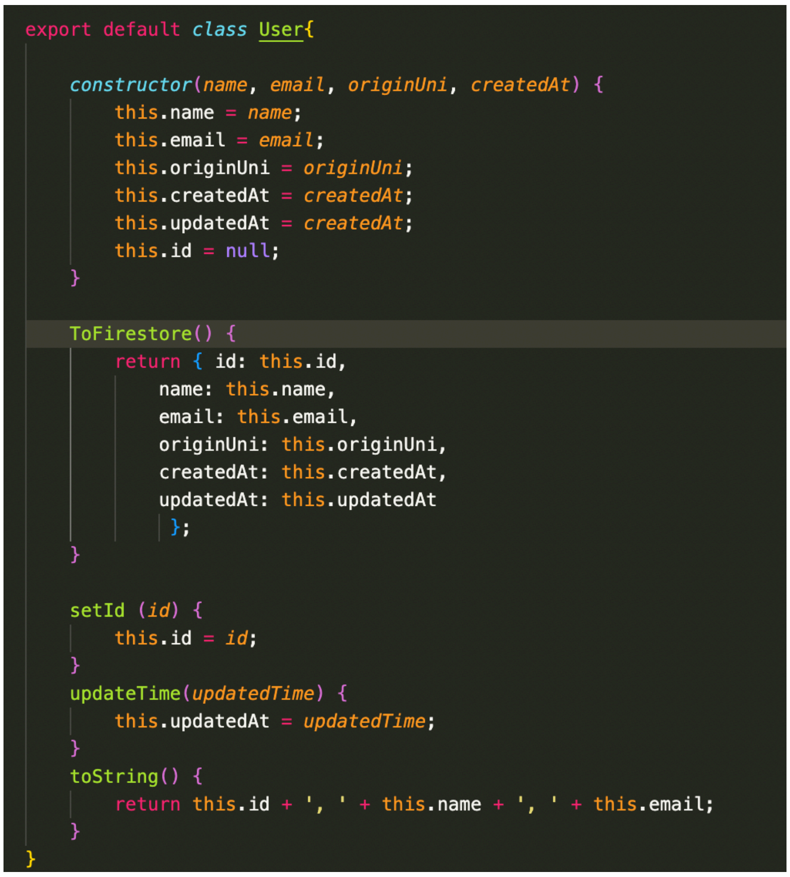


config Implementation

3. **model**
   a. This stores the object classes that are used to store information in the Firestore database.
      i. Each Object class stores the field to be transferred to a firestore database and a function to convert that data into an object that can be parsed by the firestore function **setDoc.**
   b. This is required as Object information is constantly being transferred between screens and thus makes debugging / adding to these objects simpler and more convenient for us as developers.
   c. For Example, **User** class stores information that is to be stored in the firestore database.



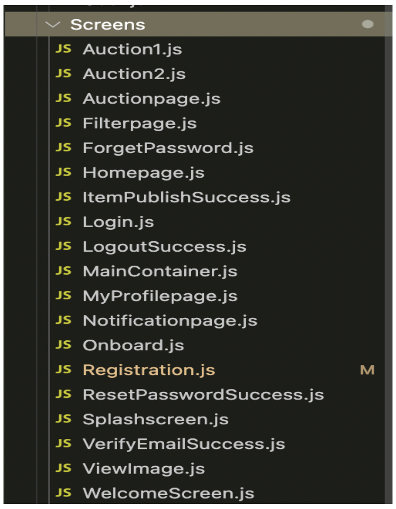
model Implementation



```
export default class User{

    constructor(name, email, originUni, createdAt) {
        this.name = name;
        this.email = email;
        this.originUni = originUni;
        this.createdAt = createdAt;
        this.updatedAt = createdAt;
        this.id = null;
    }

    ToFirestore() {
        return { id: this.id,
            name: this.name,
            email: this.email,
            originUni: this.originUni,
            createdAt: this.createdAt,
            updatedAt: this.updatedAt
            };
    }

    setId (id) {
        this.id = id;
    }
    updateTime(updatedTime) {
        this.updatedAt = updatedTime;
    }
    toString() {
        return this.id + ', ' + this.name + ', ' + this.email;
    }
}
```

User Class

4. **Screens**
    a. This Folder stores the screens.
    b. Screens are what the user sees and where the Frontend operations are performed. It is also where mostly all the react native code is written.

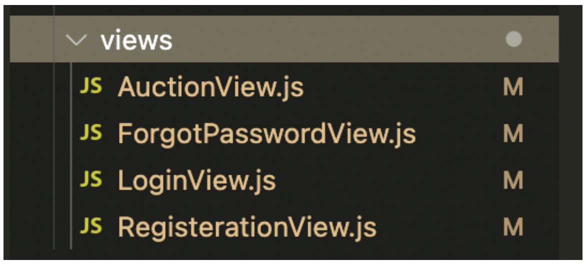


```
Screens
  Auction1.js
  Auction2.js
  Auctionpage.js
  Filterpage.js
  ForgetPassword.js
  Homepage.js
  ItemPublishSuccess.js
  Login.js
  LogoutSuccess.js
  MainContainer.js
  MyProfilepage.js
  Notificationpage.js
  Onboard.js
  Registration.js        M
  ResetPasswordSuccess.js
  Splashscreen.js
  VerifyEmailSuccess.js
  ViewImage.js
  WelcomeScreen.js
```

Screen Folder Implementation

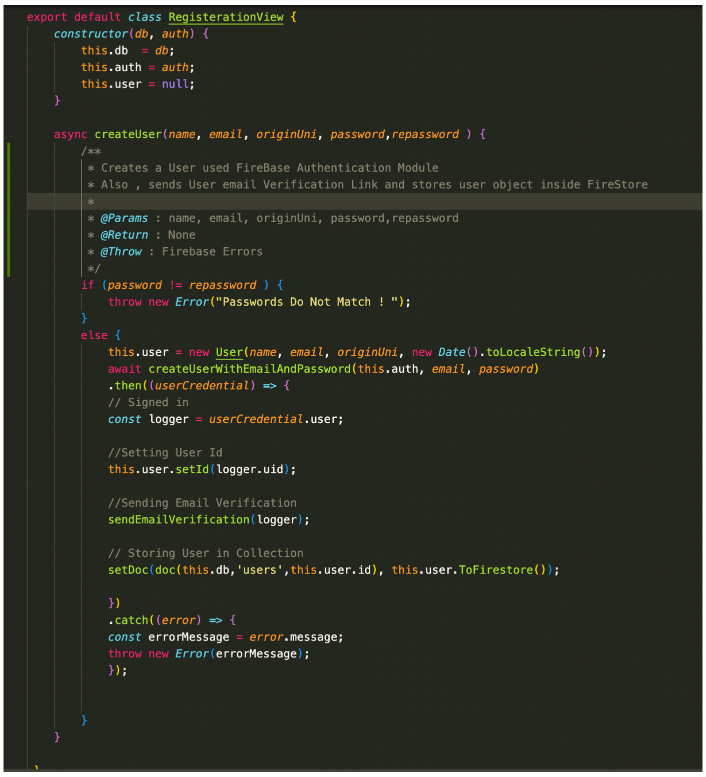
5. **views**
    a. This folder stores the backend functionality of the app.
    b. Views deal with transferring of the data between the app and the firestore database or the cloud storage.

c. This is required as it separates backend and frontend functionality, thus making it easier to debug later.



views
AuctionView.js M
ForgotPasswordView.js M
LoginView.js M
RegisterationView.js M

View Folder Implementation



```
export default class RegisterationView {
    constructor(db, auth) {
        this.db  = db;
        this.auth = auth;
        this.user = null;
    }

    async createUser(name, email, originUni, password,repassword ) {
        /**
         * Creates a User used FireBase Authentication Module
         * Also , sends User email Verification Link and stores user object inside FireStore
         *
         * @Params : name, email, originUni, password,repassword
         * @Return : None
         * @Throw : Firebase Errors
         */
        if (password != repassword ) {
            throw new Error("Passwords Do Not Match ! ");
        }
        else {
            this.user = new User(name, email, originUni, new Date().toLocaleString());
            await createUserWithEmailAndPassword(this.auth, email, password)
            .then((userCredential) => {
            // Signed in
            const logger = userCredential.user;

            //Setting User Id
            this.user.setId(logger.uid);

            //Sending Email Verification
            sendEmailVerification(logger);

            // Storing User in Collection
            setDoc(doc(this.db,'users',this.user.id), this.user.ToFirestore());

            })
            .catch((error) => {
            const errorMessage = error.message;
            throw new Error(errorMessage);
            });


        }
    }
```

RegistrationView Class

For Example, RegistrationView class stores **createUser** function that creates a user reference using firebase authentication module and stores user information inside the firestore database. Mostly all functions inside views are asynchronous, as they involve making requests to the firebase server.

Response to Milestone Evaluations

As a team, we valued feedback and reviews given to us as it helps us better understand user's needs and improve their experience. By analyzing and implementing user's feedback, it could help us understand overall users satisfaction, as the evaluation can provide us regarding what our users truly want.

We would like to thank all reviewers for your responses. Moreover, we also wish to highlight that as Orbital is only run for a short period of three months, there may be time limitations to implement so many features. However, we will do our best to take your suggestion into consideration, and please be assured that your feedback has been accounted for.

As part of the Milestone evaluation, we have compiled a list of items that we would like to address below:

Responses to Milestone 1 Evaluation

Feedback: While the use cases diagram where included in the README, the link to it on Lucidchart was not working as no public access was provided.

Response: Thanks so much for highlighting this issue! We realised that the link we provided needed users to sign up for an account before they can access the Lucidchart diagrams. We have solved the issue and you should be able to access the links now without signing up.

Feedback: Poster provided is very informative, but appears slightly wordy and improvements may have to be made towards readability.

Response: Thanks so much for your input! We will try our best to cut down on the number words in our poster as we finalise our technical implementation. We will probably update our final poster with better readability at the end of Milestone 3 when everything is ready to go!

Feedback: It would be better if a chance was given to test out a technical proof of concept that was actually built with the tech stack mentioned in the README.

Response: Thanks so much for your feedback! We have included our technical proof of concept using Expo (React-Native) in our GitHub README for you to test our application! We are truly apologetic for not specifically including a proper technical concept for our Milestone 1 Submission as we are deeply focused on planning the technical software design and architecture, as well as UI/UX design before embarking on the development of our application. Even so, we believed that you should have tested on Figma on the overall feel of how our app prototype will run in production!

Feedback: There would be a minimum viable product to demonstrate with the bidding system as it provides live updates of prices and no interaction and negotiation is required. However, I feel that the bidding system might be more suitable for expensive items as the range of prices will be larger.

Response: Thanks so much for your suggestion! This is good constructive feedback for us and we really appreciate your response. We did not really take into account for cheaper items (What happens if the product selling price is $5? Would the bidding be useful?). We definitely will try to work around a solution for this (Maybe if less than a certain amount <$10, we would only allow a buyout option rather than bidding). However, as time limit is constraint for Orbital, we could not promise that the feature will be implemented but please be assured that your suggestion have been taken into consideration.

Feedback: Sellers can ask friends to place misleading bids such that genuine buyers will place higher price. If there is a conflict between sellers and buyers, e.g. sellers sold counterfeit goods, who will be in charge of customer service? and is the reporting feature enough?

Response: This is another amazing point for us to take note of! However, for this short span of period for our Orbital Project, we as developers, are only concerned with the app development side for the project. As customer service is an external point of view, we will only focus on external aspects if our app is ready for production, which may not be the main scope of our Orbital Project. Moreover, misleading bids cannot be avoided (For example: In other platforms such as Carousell, sellers could also place misleading bids or counterfeit goods to buyers). However, we totally valued your suggestion and believed that misleading bids/counterfeit goods can be prevented significantly with the right features. The reporting feature (if implemented) will be the first step to reducing such cases. As mentioned earlier, as time limit is constraint for Orbital, we could not promise that the feature will be implemented but please be assured that your suggestions have been taken into consideration.

Feedback: A very slight detail that I noticed is the sorting of notifications/bidding placed. I think it's more intuitive to sort the newest bids/notifications from top to bottom. Otherwise, everything looks great!

Response: Thanks so much for your feedback! We will definitely take your suggestion into consideration!

Feedback: One small problem is that in the poster, the visual graph for bid history was mentioned, but we are not able to find details about it in the project README. There can also be a more detailed explanation for the motivation behind the anonymous price bidding.

Response: Thanks so much for your feedback! The visual graph for bid history is part of the additional features for our Orbital Project (If time allows us to implement). It is not part of our main/core features implementation but we have included a brief description of the functionality at the 'Additional Features Section'. We will definitely try our best to improve on the motivation behind the anonymous price bidding if we have sufficient time.

Feedback: Small request to number your headers so that I can keep track of the sections as I go through the README.

Response: Thanks so much for your suggestion! We definitely will incorporate the headers, as well as indexing for each component and labelling the pages for easier tracking of sections for our next milestone submission.

Responses to Milestone 2 Evaluation

Feedback: The prototype has layout issues smaller phones, good job for pointing that out on GitHub issues.

Response: Thanks so much for highlighting this issue! We also realised this issue prior to submitting our project for Milestone 2. As of now, we have tried our best to adjust the layout to fit the majority types of phones and sizes. However, we could not promise that the layout will fit on very old phone models or very small phone models. We will definitely try our best to make the layout fit appropriately if we have sufficient time in the future.

Feedback: The project poster is still a little bit wordy and difficult to read.

Response: Thanks so much for your suggestion! As of now, we are trying our best to reduce the number of words in the poster but we also do not want to leave out any key information. We believe that the poster design and wording is based on personal preference because we do receive feedback from other groups that has complemented our poster design being detailed and oriented. Moreover, our advisor and mentor also does not provide any critical feedback regarding our poster design during evaluation. Therefore, we decide to follow the overall point of view before making any major changes to the poster design. Nevertheless, we would like to apologise if the poster does not suit your personal preference but please be assured that your suggestions have been taken into consideration by us.

Feedback: Auction Page - category and price: all the fields were working, photo uploaded, however was not able to publish item, tried 3 times and expo go crashed. Not accepted since was not able to add an item.

Response: Thanks so much for your feedback! We are sorry for not noticing this issue before submission. We have tried to debug this and realise that it is an error that involves Firebase Storage. This is because Firebase Storage has a limitation of the file size uploaded and may not work with a big file. Therefore, some of the images uploaded from your phone may crash the app. Fortunately, we manage to work around a solution for this by compressing the image / reducing the image quality prior to sending it to the server database. As of now, it should work on your device and you should be able to publish your item.

Feedback: The experience so far was okay overall, there are just some small details that might bother users in the long run, one example is: clicking on "enter" after typing in search keyword usually enables enter automatically, in this case we should click enter to close the keyboard, then click "search" to search.

Response: Thanks so much for your input! We will definitely take your suggestion into consideration.

Feedback: [Tested on iPhone 11 pro, general feedback on compatibility: not quite compatible on iPhone 11, the top part of the app was cut, esp for welcome slides]

Response: Thanks so much for your feedback! This is a similar issue from another group of testers as mentioned above regarding prototype layout problems with smaller models. As stated above, we will try our best to adjust the layout to fit the majority types of phones and sizes to improve user experience.

Feedback: Managed to search for items. Maybe the listed items can have more legitimate names rather than random characters.

Response: Thanks so much for the response! We will definitely take your suggestion into consideration.

Feedback: The team can test whether the bidding will actually work with multiple users bidding for the same item.

Response: Thanks so much for your suggestion! We definitely will try to test and work around with multiple phones and devices (including different IOS and Android models) to check for any crashes/bugs prior to the submission.

Feedback: Home page. Search is currently case-sensitive with no fuzziness. Unable to reset search. Filter does not keep track of state (when you tick and apply, and return to the filter, it resets).

Response: Thanks so much for the feedback! We will definitely take your suggestion into consideration for fuzzy searching in the future if we have sufficient time. For search reset, users need to leave the search field empty and Pull to Refresh the Homepage to go back to default settings. We understand that this may not be convenient or well known to users and could hinder user experience. Also, the app not being able to keep track of state when filter resets is another issue that may decrease user experience. As of now, we have identified this as a minor enhancement issue and we are mainly focusing on major bugs such as app crashes, but please be assured that your suggestions have been taken into consideration.

Feedback: When listing the auction, apps crashes (even though I see that this is one of the tests being conducted).

Response: Thanks so much for highlighting this issue! We are sorry for missing out this test case prior to our submission for Milestone 2. This is a similar issue from another group of testers as mentioned above regarding Auction, where publishing items causes the app to crash. As stated above, we manage to work around a solution for this by compressing the image / reducing the image quality prior to sending it to the server database. As of now, it should work on your device and you should be able to publish your item.

Reflection on our Orbital Project

CP2106 (Orbital) was an excellent experience for us as a team, and we wish to summarize everything we learned during our summer holidays. In writing this reflection, we hope to learn and set goals for our project as student developers. By supporting a growth mindset, this will encourage students like us to improve and learn from their mistakes. Furthermore, this will be of benefit to our readers as much as it has to us as developers.

Time Management

As a team, we realise the importance of time management, especially in group project settings. At the start, we do not focus much on the deadlines as both of us have some experience in building applications. However, as the milestone 1 deadline is coming nearer, we realise that we have focused too much time on the planning phase and have not even set-up our development environment. As a result, we only started our app development at the end of milestone 1. Moreover, there were certain time periods where we were rushing the features due to poor time management. Consequently, testers during evaluation return feedback for buggy features in our app due to our lack of testing. Therefore, we feel that having good time management skills is extremely beneficial in group project settings because it helps to increase our focus and improves productivity in the team.

Industry Practices vs Student Developers

Before the start of Orbital, as student developers, we were too focused on developing the app without any concrete planning. We have experience developing apps, but not in the way that apps are developed in the real world. Moreover, we do not have any experience in software engineering. When we had a meeting with our advisor and mentor, we were shocked at how much planning needed to be done before any development phase. We were very pressured by the amount of workload and were not even sure if we were up to the task. There were certain periods of time where we felt that we should withdraw or drop down from our proposed Artemis level. Fortunately, with the proper advice and guidance of our advisors and mentor, we managed to complete all the app functionality that we planned in our development phase. Now, nearing the end of milestone 3, we are glad that we participated in Orbital because we have learned so many new skills related to software engineering, understand how industry practices function, and have learned new technologies.

Viewing things from a broader perspective

As student developers, we were too engrossed with developing the app and making the features work, but we did not take into account users' needs when developing the app. As a result, the app functionalities may look cool and workable on our end as developers, but may not fulfill the overall needs of the users in real life. Consequently, we have to modify our app on numerous occasions because it doesn't fulfill the needs of the user. Therefore, through Orbital, we manage to learn the importance of understanding how our solution could fit into the user's needs instead of focusing on the developer's point of view.

References

Code Base:
https://github.com/anshumaantgi/StuBid

Figma Prototype Demonstration (Proof-of-Concept):
https://www.figma.com/proto/ycRVCcj6LPQZd1haL2gPvv/StuBid?node-id=1%3A6&scaling=scale-down&page-id=0%3A1&starting-point-node-id=1%3A6

Technical Proof of Concept:

1) Download 'Expo Go' on App Store
2) Login with these credentials:
Email: stubiduser2122@gmail.com
Password: testuser2122
3) Go to https://expo.dev/@brannn99/StuBid
4) Scan/Tap the QR code to start testing our app.

OR

Refer to our Github README for a more detailed explanation on how to test/use our app.

Firebase Implementation:

For users who wish to view our implementation of Firebase (Backend) database, users can login with the credentials listed below at **(**https://firebase.google.com/**)**:

Test Account:
Email: stubiduser2122@gmail.com
Password: testuser2122

Technical Implementation:
StuBid Technical Implementation

Project Log:
Team 5005: StuBid (Project Log)

Technologies utilised in this project:

Technology	**Purpose**
React Native	Front-end framework
Firebase	Backend platform (Utilising Cloud Storage, Authentication and Cloud Firestore database)
Figma	UI/UX Wireframe and Prototype
Github/Git	Version control and managing git repositories
Jest	Testing framework