NUS Orbital 2022 - Milestone 3

Grow
Fit

Proposed Level of Achievement:
Apollo 11

Team Name:
FullSend (Neo Jia Han and Carmen Ang)

Motivation:
We want to be more motivated to exercise, so we decided to think of something which can appeal to more people and especially students. We felt that current exercise applications are too unvaried and boring. Hence, we want to make something more exciting by gamifying exercise.

Aim:
To encourage a healthier lifestyle by the means of a reward system of a game.

User Stories:

For basic features:

1. As a student who is busy with schoolwork and lacks motivation to exercise, I want to be more motivated to exercise than with the current fitness apps.

2. As a student who tends towards games to destress, I want a fitness app that allows me to exercise and earn rewards at the same time.

3. As a student who seldom exercises, I want a fitness app that has a daily goal to encourage myself to exercise more frequently.

4. As a pet lover, I want a fitness app that caters to my preferences in terms of design.

For additional features:

5. As a student who wants to share my progress with my friends, I want a fitness app that has a screenshot and gallery function.

Scope of Project:
The mobile application suitable for Android provides a gamified approach to current fitness apps, with users taking care of their pet through completing daily exercise goals.

Features and Timeline:

Finished by Milestone 1 & 2 (Basic Features):

- Login page
    - User's information is authorized in order to move forward to the home page
- Home page
    - shows the user's virtual pet and buttons that lead to the other screens
- Daily challenges:
    - the amount of steps in a day, the user needs to obtain to earn coins
- Coin shop:
    - where the user is able to buy treats and accessories for the virtual pet
- Treat system:
    - the user feeds the pet treats and the pet can 'grow' after a certain amount of treats are fed

Finished by Milestone 3 (Additional Features):

- Gallery Page
    - Save a snapshot of the pet in the homepage and displays in a horizontal scrollable navigation
- Save to Device Gallery
    - Allow users to save snapshot of pet to device gallery
- Share Functionality
    - Allow users to share snapshot of pet to different platforms
- Push Notifications:
    - As per user feedback, many mentioned that the application did not give them too much motivation to exercise/ walk.
    - Hence, we thought that adding push notifications would remind users to open the app.
- Enhanced design of application
    - Also as per user feedback, the application interface looks dull and simple.
    - Hence, designs of widgets are changed.
    - Added a level feature where a numerical level (calculated based on the number of treats fed to pet) is displayed on the home screen so, users can better quantify their progress

Features to Implement in the Future:

- More toys/ clothes in shop
    - Need to implement navigation tool to navigate through the shop if there are more items, furthermore list to store the state of each toy in the database would be longer
    - Adding clothing items for pets would also have to consider adjusting the widgets with the pet, base on the size of the pet
- User feedback option
    - Users can then submit their feedback/ reports on the app through a form
- Enhance user experience
    - Add music in the background of the application, this may bring a more positive experience towards the user
    - Add animation on pet, this may come in the form of blinking, head or tail movement
    - Allow users to
- Customizability of challenges
    - Allow users to customize the amount of steps they can take and fulfill their own targets
    - Add more challenges in terms of other form of exercise (as per peer feedback)

- Allow users to view other users' progress within the app
    - A feature where users can make friends within the app, and users can 'visit' other people's pets

- Make the application available on IOS
    - As the app is built with Flutter, enabling the app to be available on IOS would require minimal changes hence, very plausible
    - Moreover, allows user cross platforms to share the same experience

Tech Stack:

1. Flutter (Frontend)
2. Firebase (Backend)
3. Google Fit API

How are we different from similar platforms?

- Strava
    - The application has the same boring interface as other fitness applications, with the exception of its focus towards social workouts, thus it is not engaging enough for users who do not have the motivation to exercise in the first place.

- Nike Training Club
    - The application is catered more towards users who were already motivated to exercise and want workout plans to help them be more focused, thus it is not suited for beginners who desire the motivation to start exercising in the first place.

Development Plan:

Timeline	**Event**
3rd week of May	Finalized pitch for Orbital Lift-off
4th week of May	Created Mockup
5th week of May	Pick up necessary technologies - Flutter, Firebase, Google Fit API
1st week of June	Start building database for user information, testing and debugging
2nd week of June	Start building frontend (Sprites, Reward system)
3rd week of June	Continue with frontend and backend (Treat interaction, growth, login, read and writes)
1st week of July	Testing and debugging
2nd week of July	Implement additional capabilities – (Screenshot, sharing, gallery, push notifications)
3rd week of July	Testing and debugging

System Testing:
Unit and Integration Testing (for features implemented this Milestone):

Feature	Testing
Screenshot widget	Does it screenshot the widgets wrapped in the Screenshot widget, (pet, feedbowl and level) - **True**
Level Mechanism	Does it increase whenever 10 ten more treats are fed to the pet - **True** Does not go to a level below 0 - **True** Does not round up to the closest 10 number of treats (ie 98 does not mean level 10) - **True**
Pageview Widget (Image Gallery)	Does it display the image taken by the screenshot function in the homepage -

	True Does it display more than one image - **True** Does it navigate horizontally - **True** Does it display the images in order of the images taken - **True**
Save to Gallery Function	Does it save an image to the gallery when first clicked - **True** Does it save the correct image (one that is in the current view of the gallery) - **True**
Share Function	Does it pop up the share navigation - **True** When shared to a different platform, the image corresponds to the one in the gallery view - **True**
Gallery Page as a whole	When gallery icon in Home/ Shop page is clicked does it go to the Gallery Page - **True** When Home/ Shop page is clicked it goes to the Home/ Shop page - **True** Health Button shows correct number of steps in Gallery Page - **True** Able to claim reward, claimed reward can only be claimed once a day - **True** Treat count and coin count syncs to the data in Firebase and in Home / Shop page - **True**
Push Notifications	Does it send a reminder to the user after more than a day of inactivity - **True** Does the notification lead to the application when clicked - **True** Does the notification lead to the Home page (when signed in) - **True** Does the notification lead to the Sign In page (when not signed in) - **True**

User Testing:
We asked our friends and family to try out our prototype to gauge the level of understanding of our app towards different kinds of users. Afterwards, we asked them to fill a google form. Here are their responses:

Summary of Responses:

User	Proficiency	Questions	Response (for MS2)	Response (for MS3)
Family member	Novice	Is the system easy to use? Is the system intuitive? How usable is the current implementation/ do the functions work as intended? Do you see yourself using the current implementation on a daily basis/ Does it motivate you to exercise? How much does the current implementation differ from other exercise applications in the market?	Yes, Yes, 5, Yes, 5 Remarks: - App is easy to use	Yes, Yes, 5, Yes 5 Remarks: - App is now more clean
Family member	Novice	Is the system easy to use? **Yes** Is the system intuitive? **Yes** How usable is the current implementation/ do the functions work as intended? **4** Do you see yourself using the current implementation on a daily basis/ Does it motivate you to exercise? **No** How much does the current implementation differ from other exercise applications in the market? **4**	Yes Yes 4 No 4 Remarks: - App can be a bit slow sometimes but a unique take on exercise apps	Yes Yes 5 Yes 4 Remarks: - App is now less buggy
Friend	Novice	Is the system easy to use? Is the system intuitive? How usable is the current implementation/ do the functions work as intended? Do you see yourself using the current implementation on a daily basis/ Does it motivate	Yes Yes 5 Yes	Yes Yes 5 Yes

		you to exercise? How much does the current implementation differ from other exercise applications in the market?	4 - App is cute and creative	4 - The new designs make the app look cleaner, the gallery function is a good addition
Friend	Expert	Is the system easy to use? Is the system intuitive? How usable is the current implementation/ do the functions work as intended? Do you see yourself using the current implementation on a daily basis/ Does it motivate you to exercise? How much does the current implementation differ from other exercise applications in the market?	Yes Yes 4 No 3 - Current implementation is quite simple - Text can be a bit laggy but functions fine	Yes Yes 4 Yes 3 - The addition of design and push notifications increase the chance of using the app
Friend	Expert	Is the system easy to use? Is the system intuitive? How usable is the current implementation/ do the functions work as intended? Do you see yourself using the current implementation on a daily basis/ Does it motivate you to exercise? How much does the current implementation differ from other exercise applications in the market?	Yes Yes 4 No 5 -Doesn’t really create much motivation to exercise - App does not have a lot of features	Yes Yes 4 Yes 5 - The new features work correctly and make the app more engaging. However, there can still be further expansion for the app to make it more engaging in the long term.

Generally, the test users see an improvement from Milestone 2. Users believe that now the application is more interactive with the addition of new features. However, they would also like to see more refinements and creativity within the application.

Issues/ Bugs Encountered when Building the Application:

Login Mechanism

- Initially, we relied on Flutterfire for user authentication through Firebase Authentication and creating user data in Cloud Firestore.
- However, FlutterFire only provided its own default screens with limited customisation, and we wanted something that was more in line with our theme.
    - Hence, we decided to go back to using Firebase Authentication for CRUD of user data, even though it was more tedious and had more chances of bugs.

Google Fit API

- We used Flutter's health 4.0.0 package to read data from the Google Fit API. However, a null value occurs when trying to read the STEPS variable on the Android emulator.
    - Solved it by downloading Google Fit app on the emulator and having an account with actual steps data

Firebase Database

- We had some issues with using Cloud Firestore at first, especially with the CRUD of user data, since Cloud Firestore field value queries require reading of documents which are provided in async and not sequentially.
    - We solved the issue through making sure that our reads were done after the document was confirmed to be queried into our application.

Pet Resize

- When figuring out how to resize the pet depending on the number of treats fed, we decided to tweak the padding. However, tweaking the padding of a single widget in Flutter affects the size of the other widgets.
    - Solved this with using the Stack class which is a widget that allows the inner widgets to be stacked on top of each other. So that changing the padding of the pet widget does not affect anything else.

User Information

- When passing the user data through the different screens, we initially went with the approach of taking only what was needed and updating the user accordingly. However, we realised that this would be very troublesome and prone to bugs since any updates would require a lot of changes to be made.
    - Hence, we decided to only pass the user around as a whole, and read/update as required on each screen.

Gallery

- When deciding on how to approach the showing of images in the gallery, we initially proceeded with a boolean value to do the checking of whether a screenshot was cached and then displaying the screenshot accordingly. However, this limited us to only displaying one image only.
    - Hence, we decided to use the PageView widget instead to display the cached images from the application directory through a List<File>, ensuring that the gallery contained all the screenshots that were taken.

Minor bugs fixed:

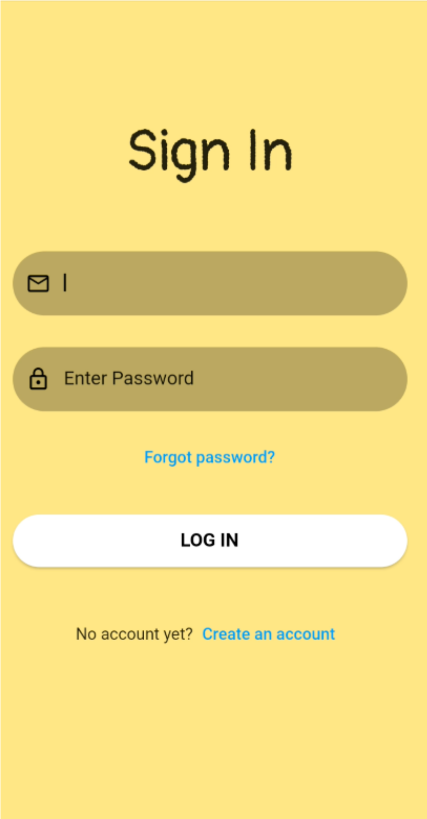
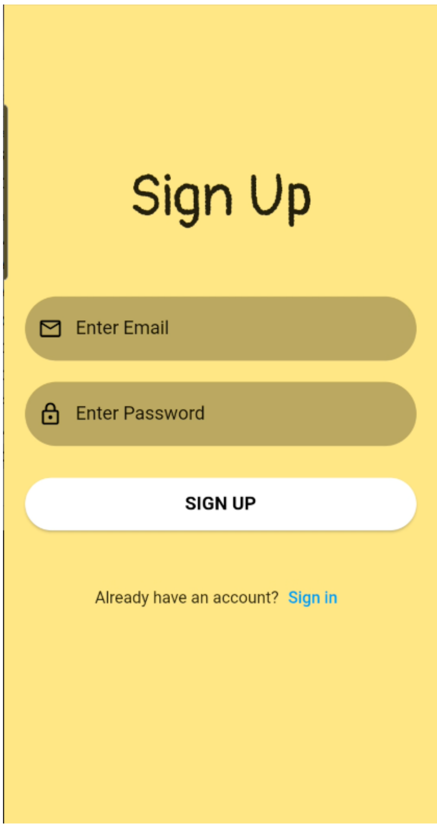
Sign Out

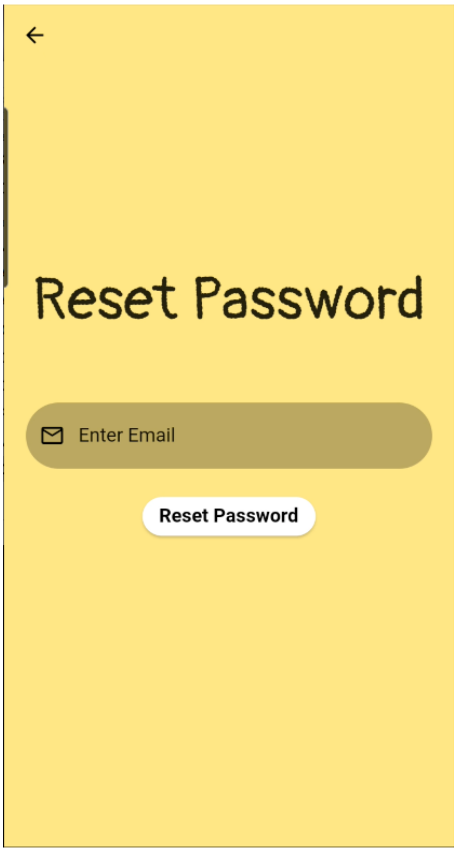
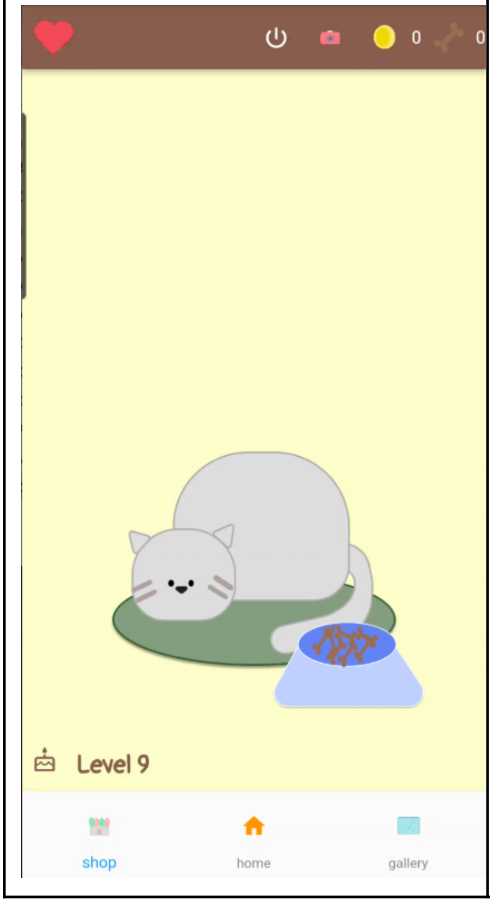
- During our testing, we realised that the Sign Out button was not redirecting the user to the Sign In Page after the user was successfully signed out.
    - This was fixed by ensuring that the page navigation occurs after the Firebase Auth stream listener receives the sign out notice.

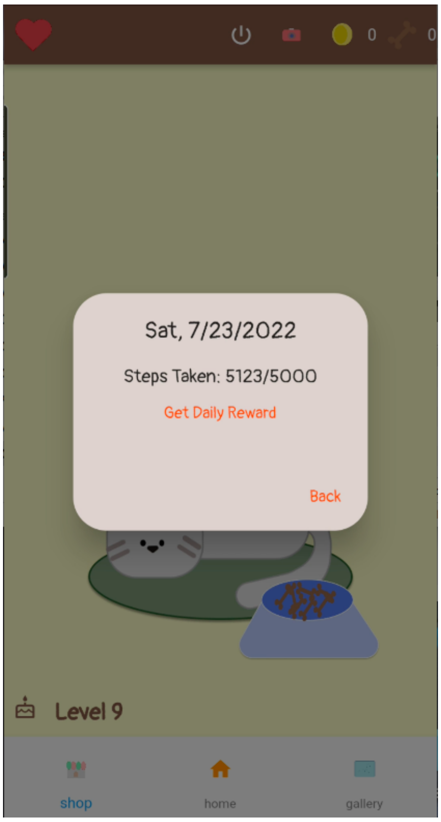
Sign In Pages

- During our testing, we realised that the Sign In Page, Sign Up Page and Forgot Password pages were saving their current state (including information in the email and password fields) when the user navigated away from the page.
    - This was due to them being Stateless Widgets, hence we have changed them to being Stateful Widgets to reset their states upon navigating to them

Application Prototype:

 Sign In I Enter Password Forgot password? LOG IN No account yet? Create an account	Login Page User is first prompted to this page when the app is opened for the first time Consist of textbox for user to input: ● Email ● Password Users can choose to sign in or: - Reset their account's password by clicking on 'Forgot password?' - Make a new account by clicking on 'create an account' Clicking on 'LOG IN' leads to the home page Shows a snackbar if account details do not match in the database
 Sign Up Enter Email Enter Password SIGN UP Already have an account? Sign in	Sign Up Page User is led to this page when they click on 'Create an account' Consist of textbox for user to input: ● Email ● Password Users can choose to sign up or: - Go back to Sign In screen by clicking on 'Sign in' Clicking on 'SIGN UP' leads to the home page Shows a snackbar if account exists

 	<u>Forgot Password Page</u> User is led to this page when they click on ‘Forgot password’ Consist of textbox for user to input: ● Email ● Password Clicking on ‘Reset Password’ sends a reset password email to the User’s email Shows a snackbar telling the user a password email is sent to their mail
 	<u>Home Page</u> User is led to this page when ‘Sign in’ or ‘Sign Up’ is clicked. Displays **coin** count, **treat** count and **level**. With the following 4 clickable options : **Health** button (heart icon) - Pop out widget that shows health data **Sign Out** button (power icon) - Pop out widget to allow user to sign out **Screenshot** button (camera icon) - Creates a snapshot of the pet and displays in Gallery page **Shop** button - Prompts user to the Shop page **Home** button - Prompts user to the Home page **Gallery** button - Prompts user to the Gallery page **Feed** button (treat bowl) - When the user clicks on the button, treat count on the top right decreases by 1



Sat, 7/23/2022

Steps Taken: 5123/5000

Health Pop Up

Pop up shows when the user clicks on the health button. Shows the user's steps for the day over the target (5000)

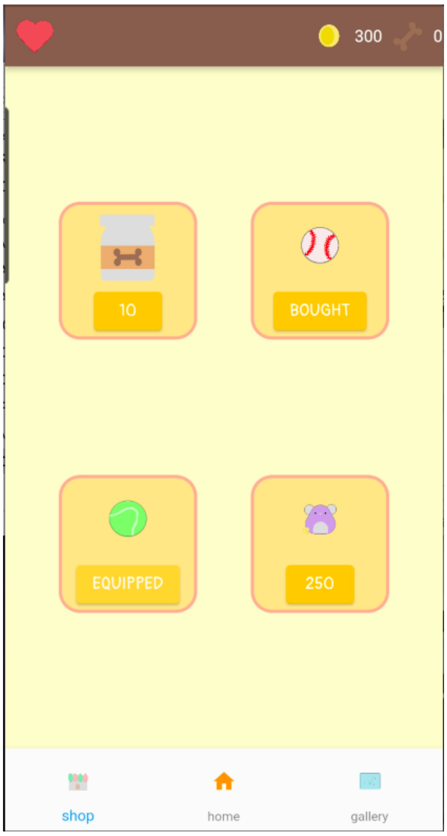
Consist of 2 options:

Get Daily Reward button:

- When user clicks on the button coin count on the top right increase by 50, only can be clicked once a day

Back button

- Returns the user to the home page



Shop Page

User is led to this page when shop button is clicked.

Displays **coin** count, **treat** count, **treat** and **toys** for sale.

Consist of 8 clickable options:

Health button (heart icon)

- Pop out widget that shows health data

Item button (yellow buttons)

- Treat can be bought repeatedly
- Toys can be bought once
    - When item is bought, price changes to 'BOUGHT'
- Toys can be equipped
    - When 'BOUGHT' button is clicked, button will turn to 'EQUIPPED'
- Toys can be unequipped
    - When 'EQUIPPED' button is clicked, button will turn to 'BOUGHT'

Shop button

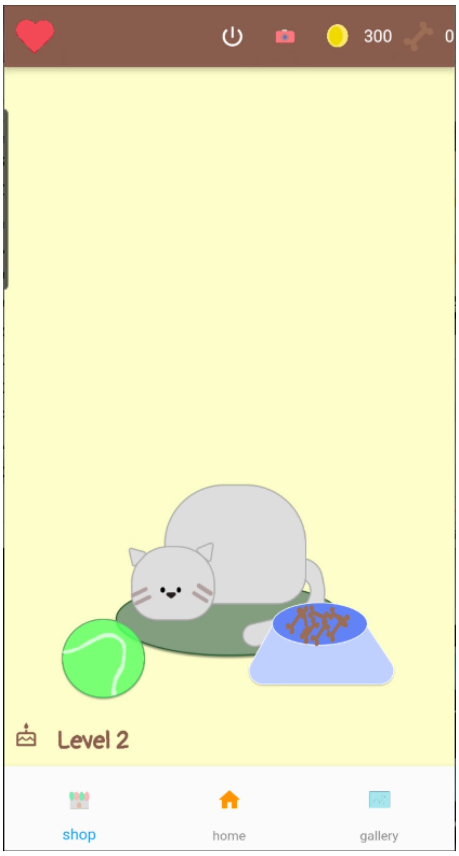
- Prompts user to the Shop page

Home button

- Prompts user to the Home page

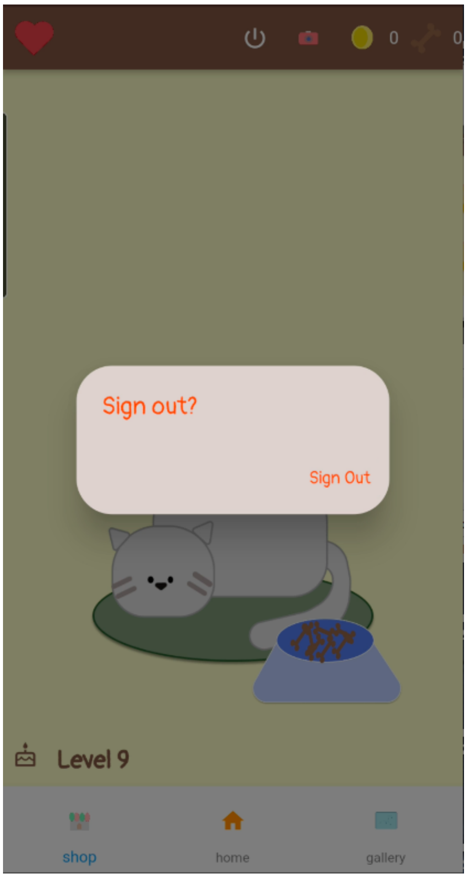
Gallery button

- Prompts user to the Gallery page



Home Page with Item Equipped

Home page display when user equips a toy, only one toy can be equipped at a time

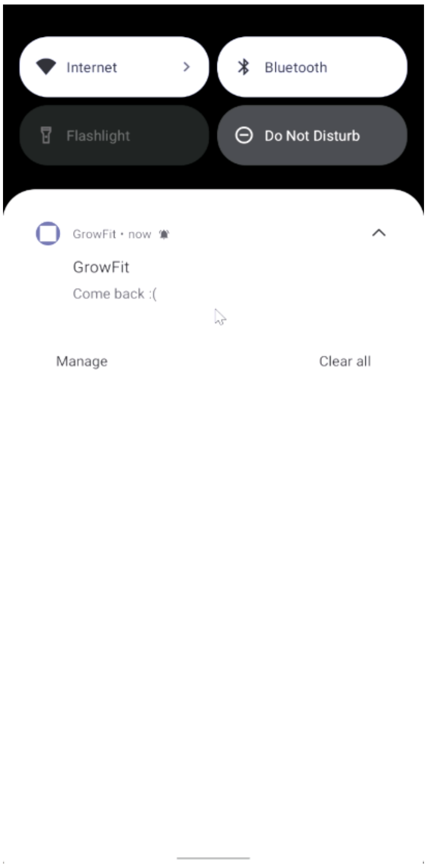


Sign Out Pop Up

Pop up shows when the user clicks on the sign out button.

Consist of one option:

- **Sign out** button, leads to Sign In page on click, logs the user out and deletes their current token.



GrowFit • now

GrowFit

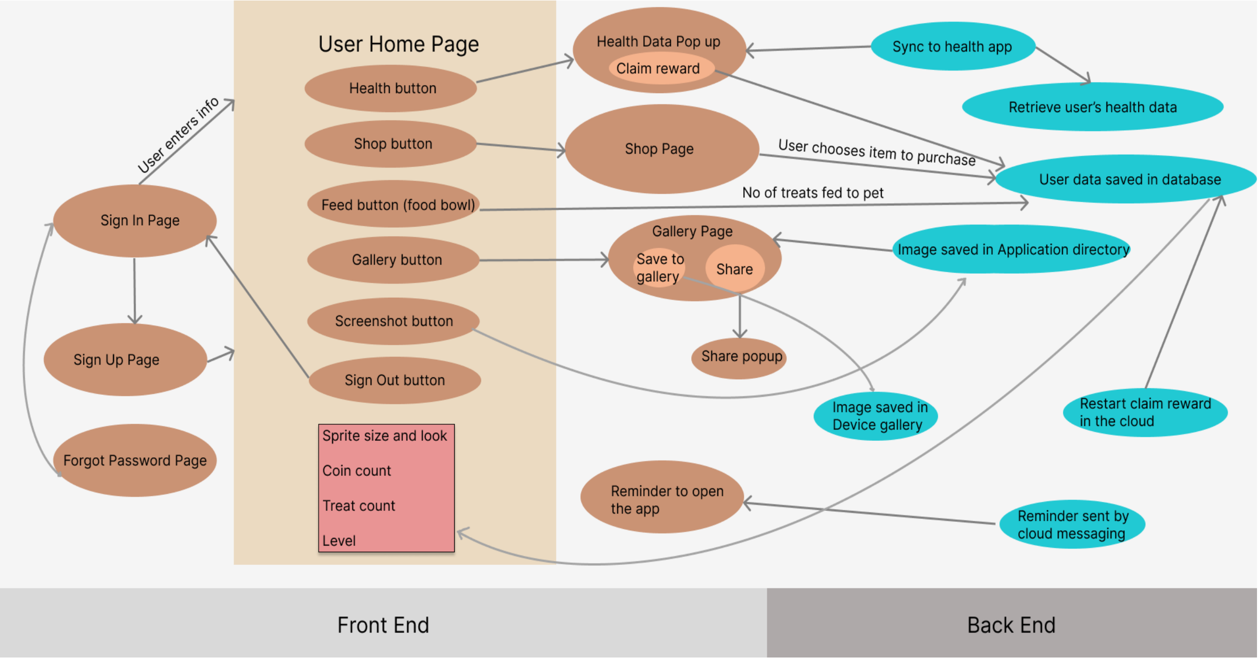
Come back :(

Push Notifications

Application sends a reminder to the user if they have been inactive from the app for a day. A click on the reminder leads the user to the app.

Technical Proof of Concept:

Project flow



The mobile app consists of these main components:

When the user first downloads the app, the **sign in** page is shown. The user will then move to the **sign up** page, where they will sign up for an account using their email and password, which is handled by the signInWithEmailAndPassword function from FirebaseAuth. The account will then be stored into Cloud Firestore as field values under a document with a unique ID. Once the user has successfully signed up, they will be moved onto the **home** page.

The **home** page currently showcases the virtual pet and the three working buttons:

The **health button** (heart symbol) visually shows an AlertDialog widget with the health data (in the form of a text widget) of the user and daily goals.
The Google Fit API and Flutter's health 4.0.0 package is integrated for this function where the user's health data for the day (midnight to current time) is synced to the Google Fit application in the device whenever the button is clicked.

The application sets a goal (5000 steps) to reach by the end of the day, which is currently the number of steps. After a goal is surpassed, the user is credited with coins, and the updated amount of coins as well as a boolean on whether they have claimed the reward for the day is stored inside Cloud Firestore under their unique ID. The boolean on whether they have **claimed the reward** is restarted everyday at midnight by using Firebase cloud functions.

The **sign out** button when clicked pops out an AlertDialog widget prompting the user if they want to sign out which then notifies the FirebaseAuth stream listener of the sign out, and leads the user back to the **Sign in** page.

The **screenshot** button (camera icon) takes a snapshot of the pet, food bowl, and level text when clicked. By utilizing the screenshot 1.2.3 flutter package, the widgets wrapped inside a Screenshot widget appear in the snapshot. Image is then stored in the Application Directory.

The **feed button** (food bowl) reduces the treat count at the top right and stores this updated data in the database. The amount of times the feed button is pressed is also stored in the database. The size of the sprite changes depending on the amounts of treats fed which is done by adjusting the padding around the image of the pet.

The **shop button** leads to the shop screen.

- The shop screen visually displays the items that can be purchased by the user with coins.
- When buying treats/toys, the data is written into Cloud Firestore and the coin value/treat value/toy information is updated accordingly onto the user's screen.
- Data on whether a toy is equipped or bought is saved in Cloud Firestore, which then updates the visual of the pet in the home display accordingly when going back to the home screen.

The **gallery button** leads to the gallery screen.

- The gallery screen visually displays the snapshot of the homepage taken by the user
- The user can take more than one snapshot and the images stored in a list are then displayed in a PageView widget where users can slide through the images horizontally
- The **save to gallery** button utilizes the image_gallery_saver 1.7.1 flutter package so, the current snapshot displayed in the PageView can be saved to the user's device gallery
- The **share to** button utilizes the share_plus 4.0.10 flutter package, which allows the user to share the image to other platforms

Whenever the user **switches** between screens the data is read from the firebase database and displays the screen depending on the data.

Push notifications:

Firebase Cloud Messaging is used to implement this where if the users are inactive for more than a day, a push notification will be sent to the user's device at 12pm. Upon clicking on the push notification, the onBackgroundMessage function is executed and runs the main() function of the application, bringing users to either the Sign In page or the Home page, depending on whether the FirebaseAuth stream has a snapshot of the user data.

Acceptance Testing:
A walkthrough of our application can be viewed from this link:
https://drive.google.com/file/d/1Kan9IISzx2dZ1kImjrCW7vYZTXSe2Qs7/view?usp=sharing

Try out our app by downloading the APK through this link:
https://drive.google.com/file/d/1Q-ChWpUZRnA-5oWhodO2IzGFqqFkhItm/view?usp=sharing

Note: if you want the Google Fit sync to work, please fill in your email in this sheet and then download the APK:
https://docs.google.com/spreadsheets/d/1snM_AWVEjS9NCSAXITXCTZSU43U748BCrqOkW29adJU/edit?usp=sharing
OR contact us through telegram, @cmang12.
This is because our application does not have Google’s verification as we have not publicized it, so it is still under testing.

For push notifications, the check is scheduled at 12pm, for users who have not accessed the application for **more than** 1 day.