Risheet Peri – 300594213 – SWEN303 –Assignment 2

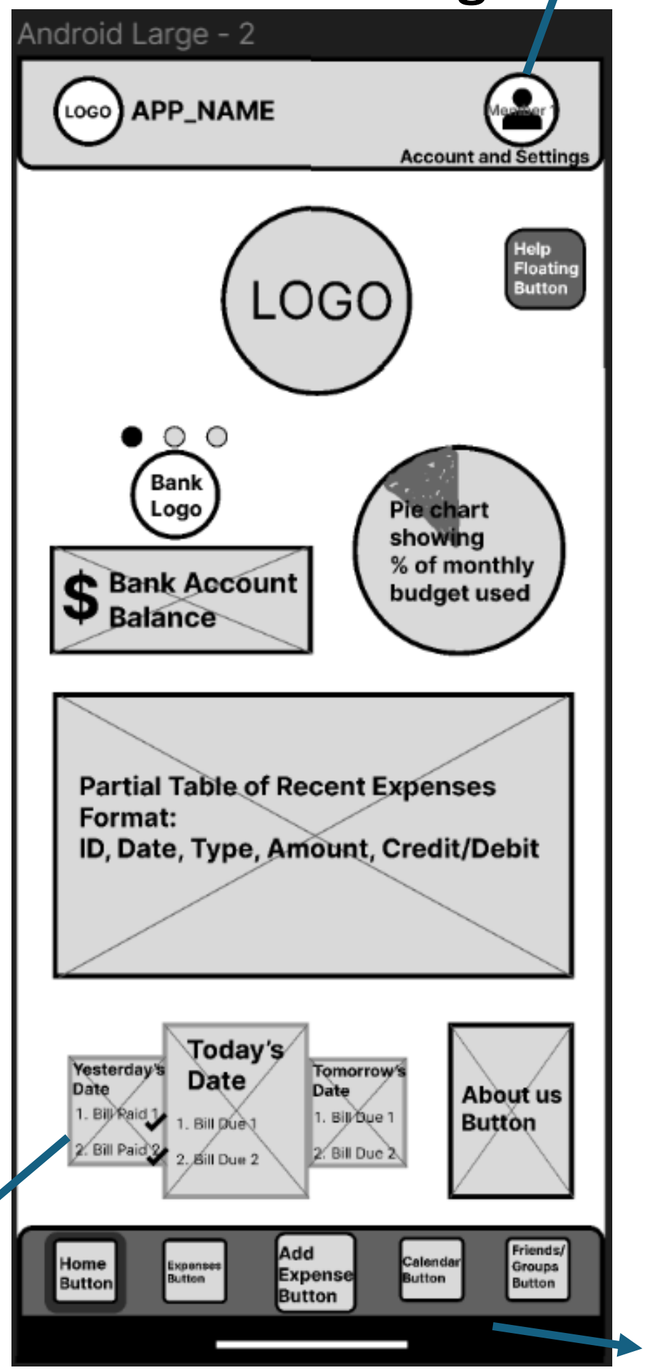
1. Wireframes

User Task 1: Add a new payment detail within a group and make the payment

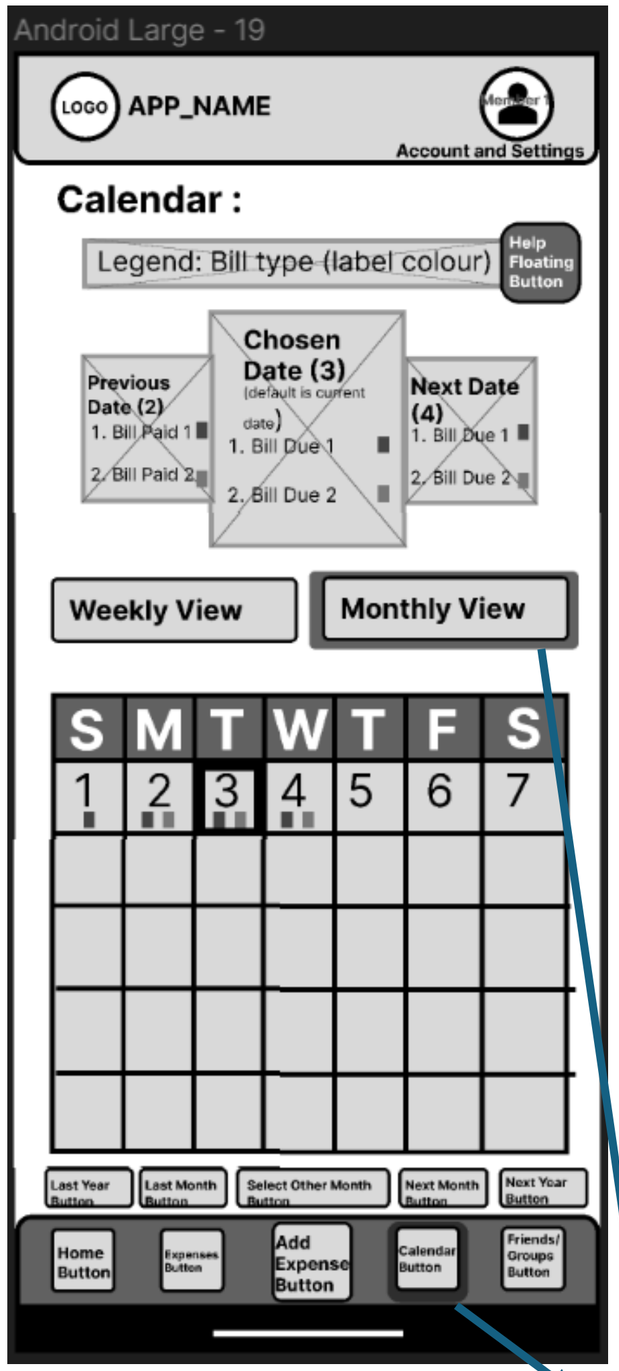
Personas:

- Alex Singh who has a very busy schedule and wants to get jobs done fast. They have less time due to their software job, them attending many technology events and participating in outdoor activities
- John Baker, a fellow flatmate, who is cautious and takes their time looking at finer details.

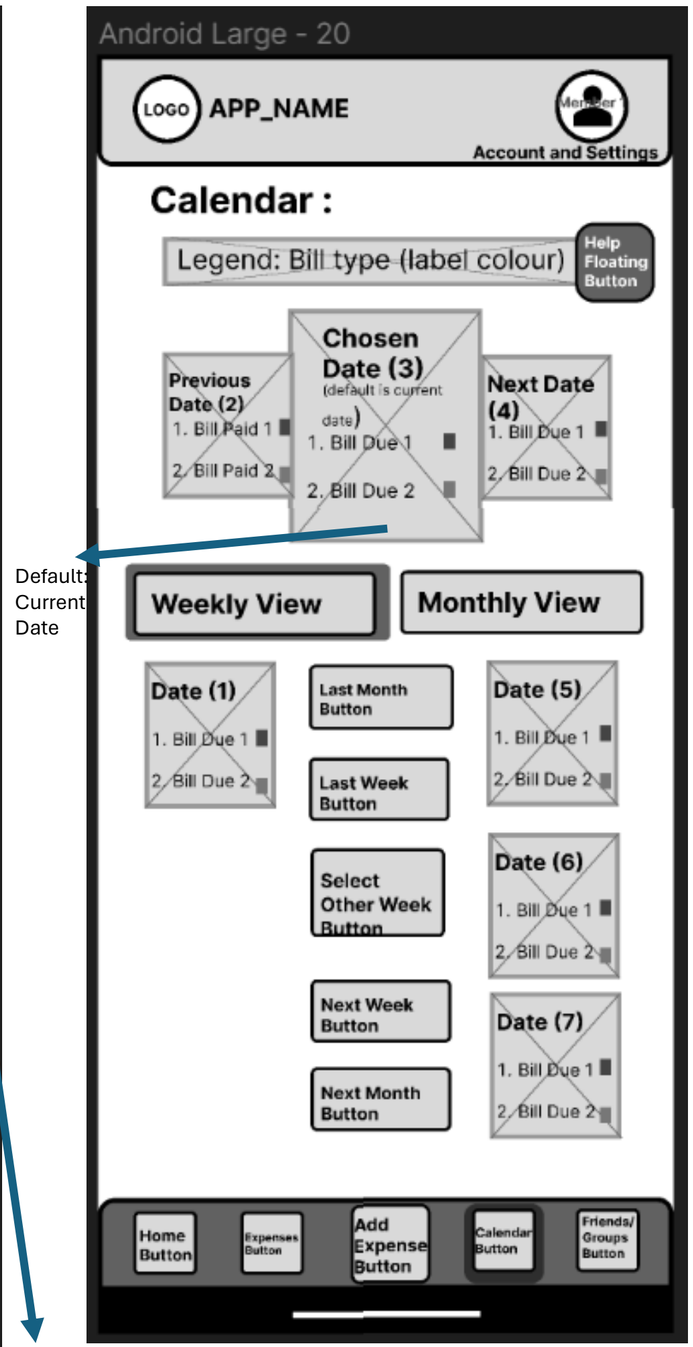
1. Home Page

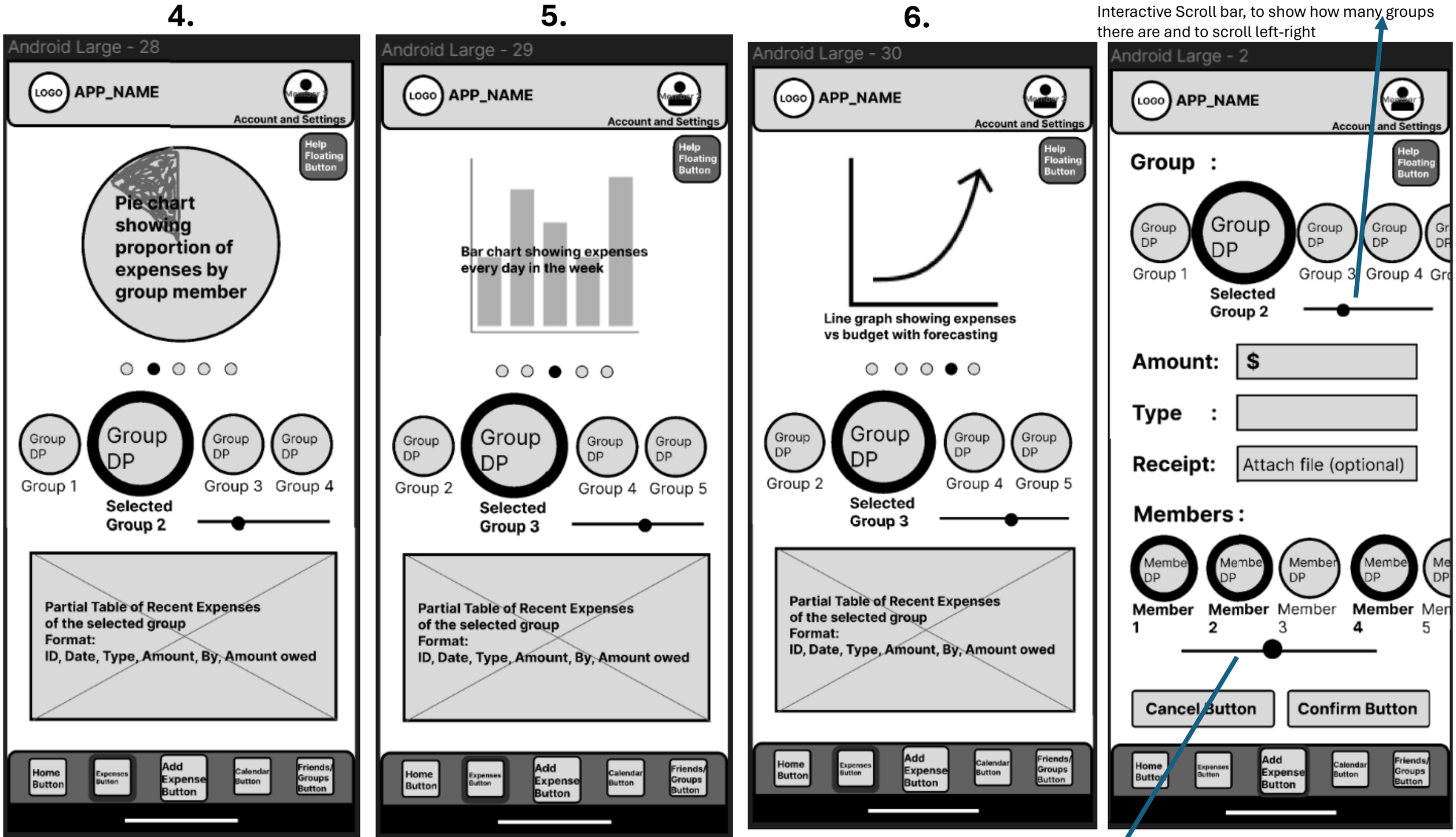


2. Calendar Page (Monthly View)



3. Calendar Page (Weekly View)

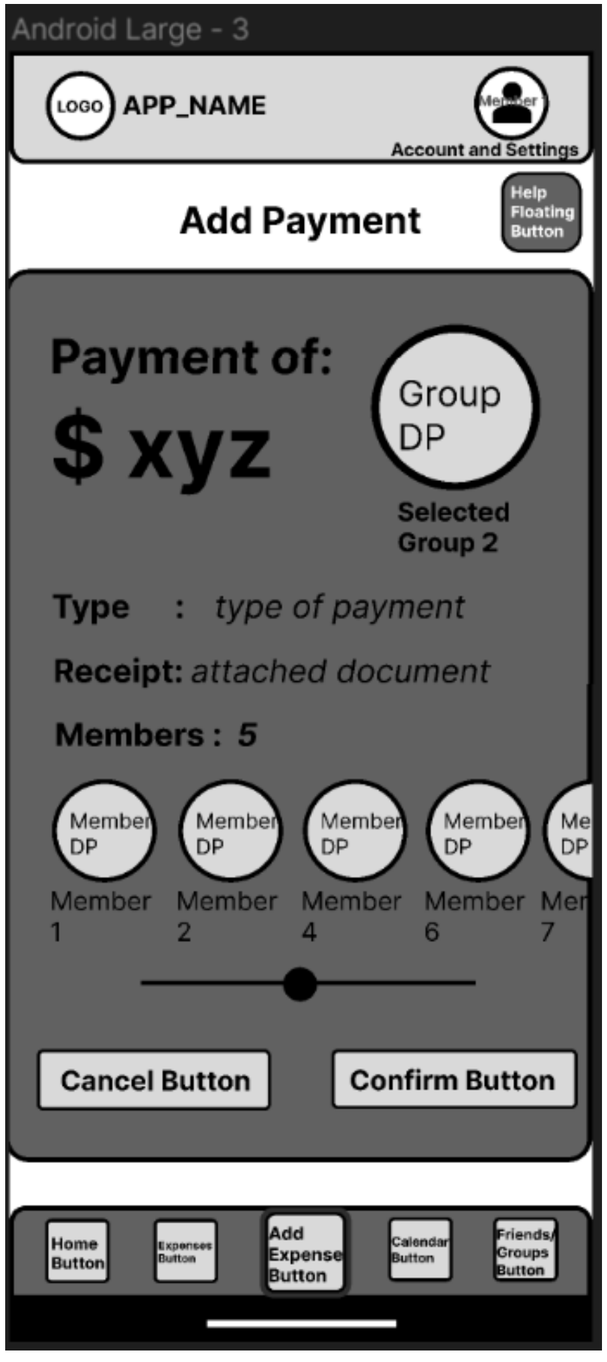




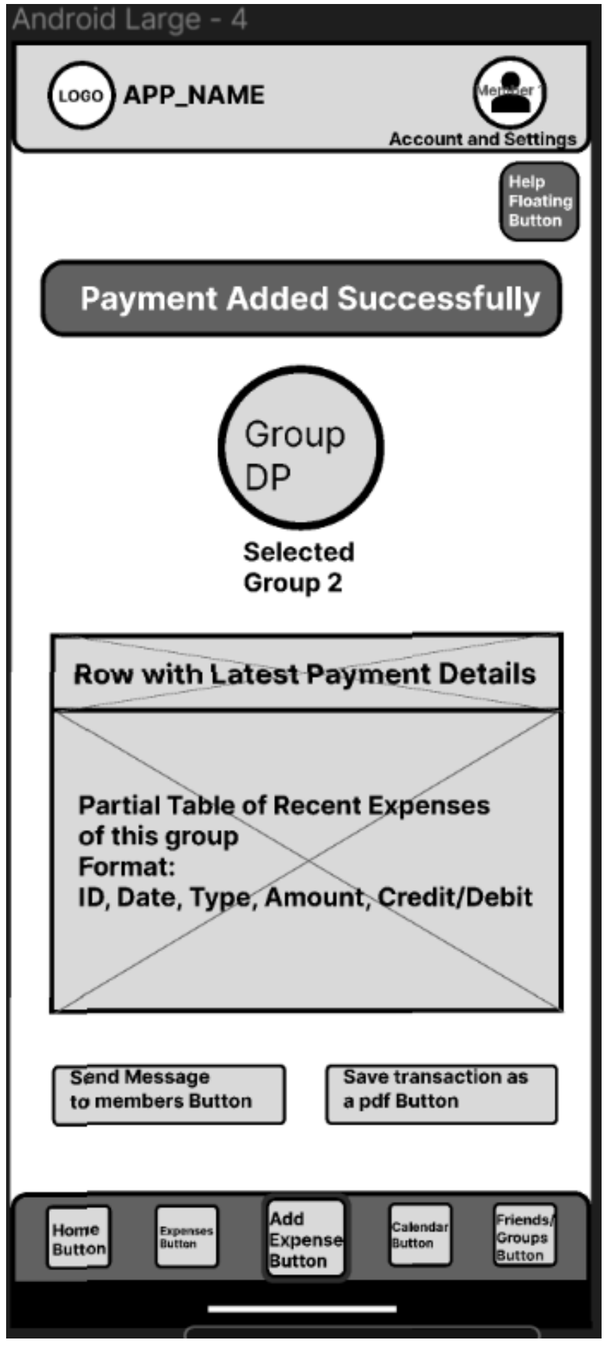
Expense Table and Graphs Page (of different groups and graphs)

7. Add Expense Page

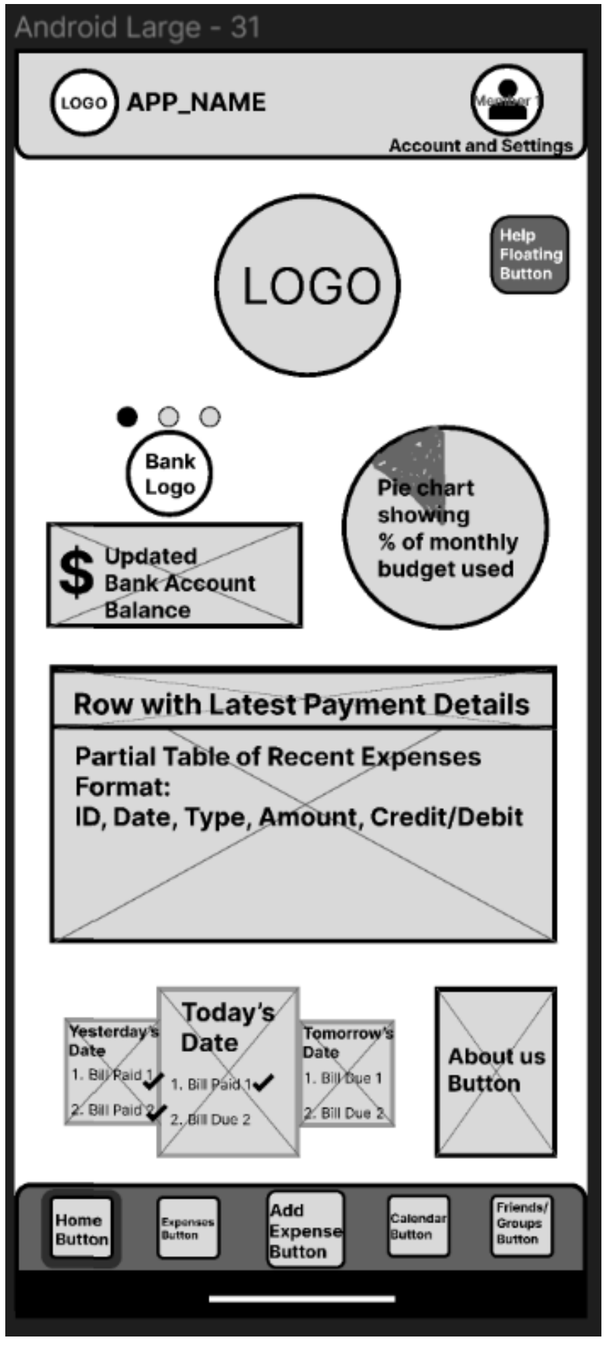
Interactive Scroll bar, to show how many members there are and to scroll left-right



8. Add Expense Review Page



9. Add Expense Confirmation Page



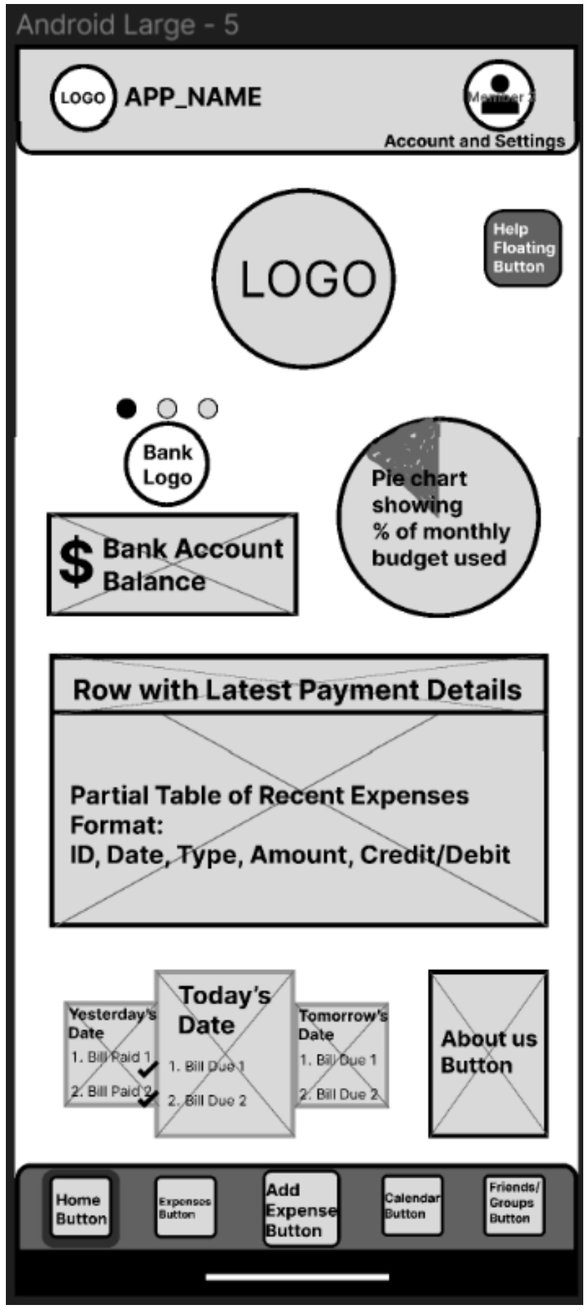
10. Home Page

User Task 2: Edit a payment's details in which Alex Singh (flatmate) added John Baker as a member by accident, since they were in hurry to add the payment due to their busy schedule.

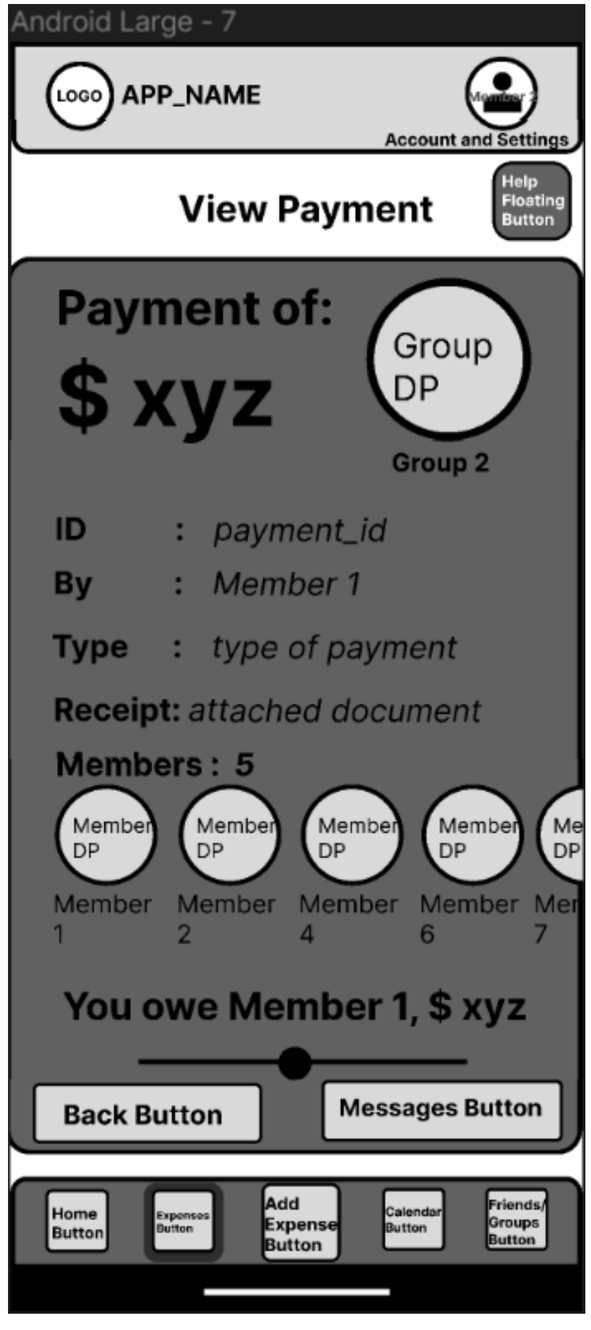
Personas:

- Alex Singh who has a very busy schedule and wants to get jobs done fast. They have less time due to their software job, them attending many technology events and participating in outdoor activities
- John Baker, a fellow flatmate, who is cautious and takes their time looking at finer details. They realize that they have been added as a payee by mistake since they were out of town during the time and did not need to pay for the bill. The mistake could be because Alex was in a hurry due to their busy schedule. John contacts Alex to remind him to change the payment details. Alex changes these details.

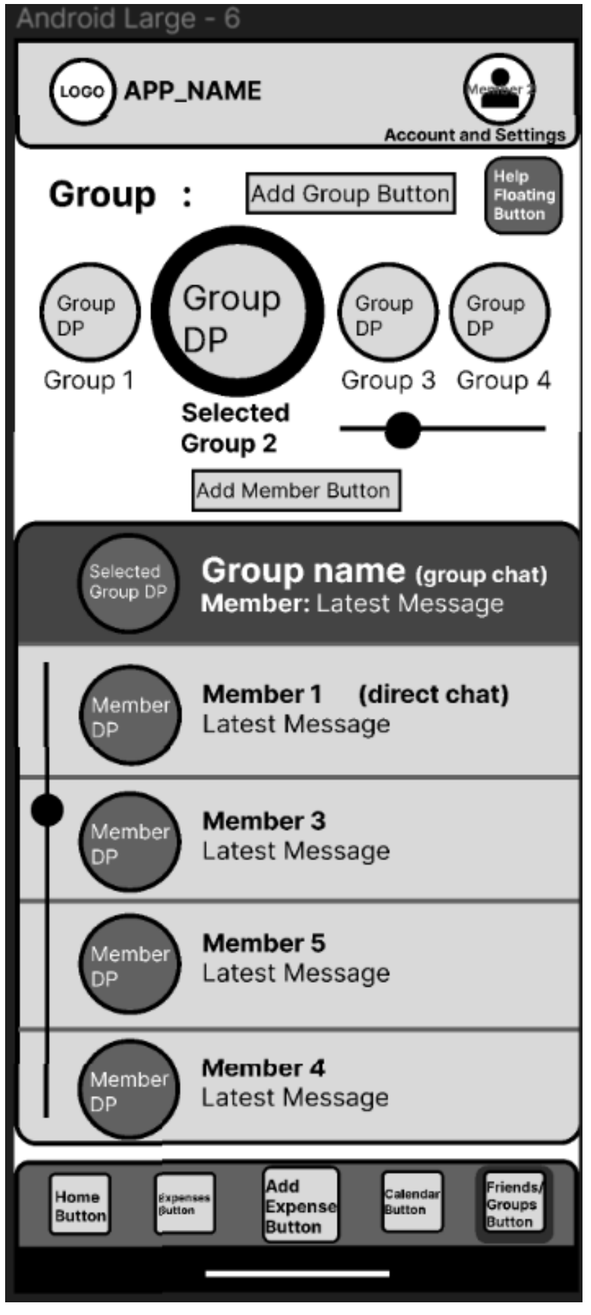
Part 1: John Baker's Perspective:



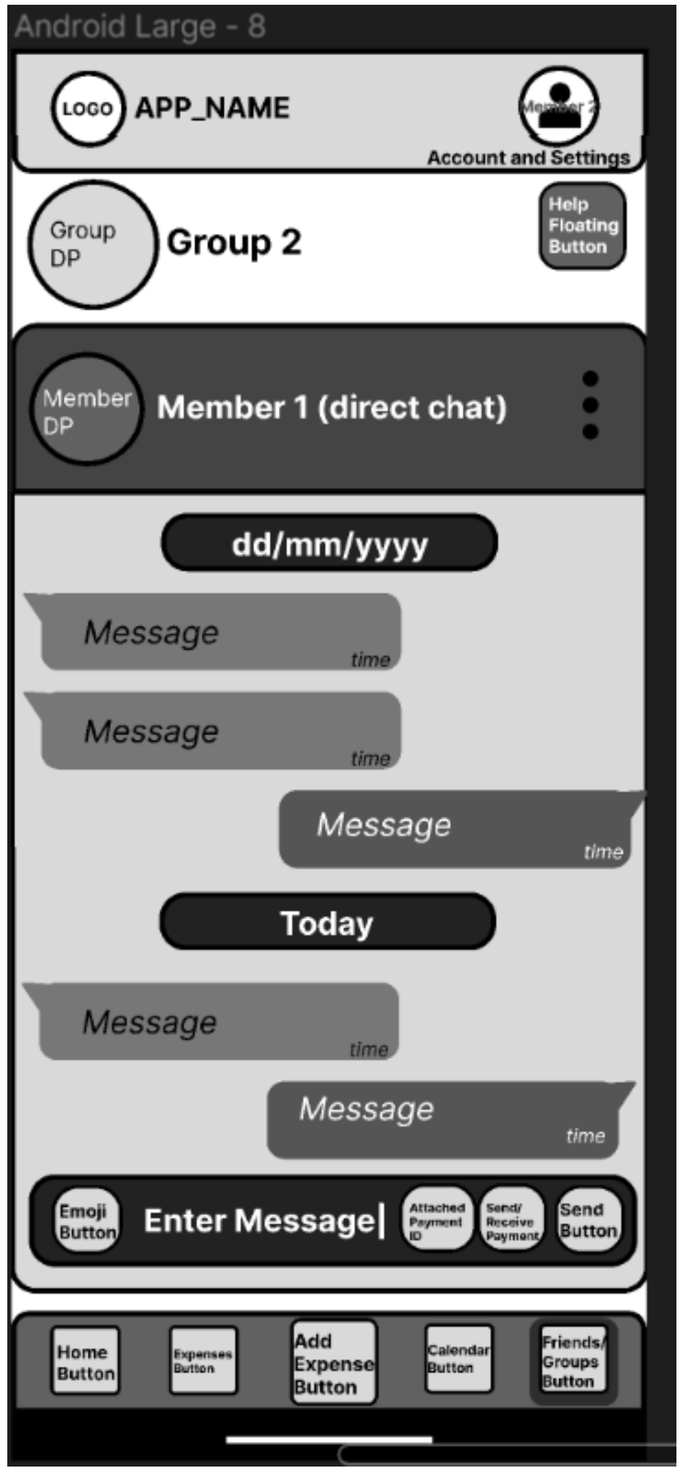
11. Home Page



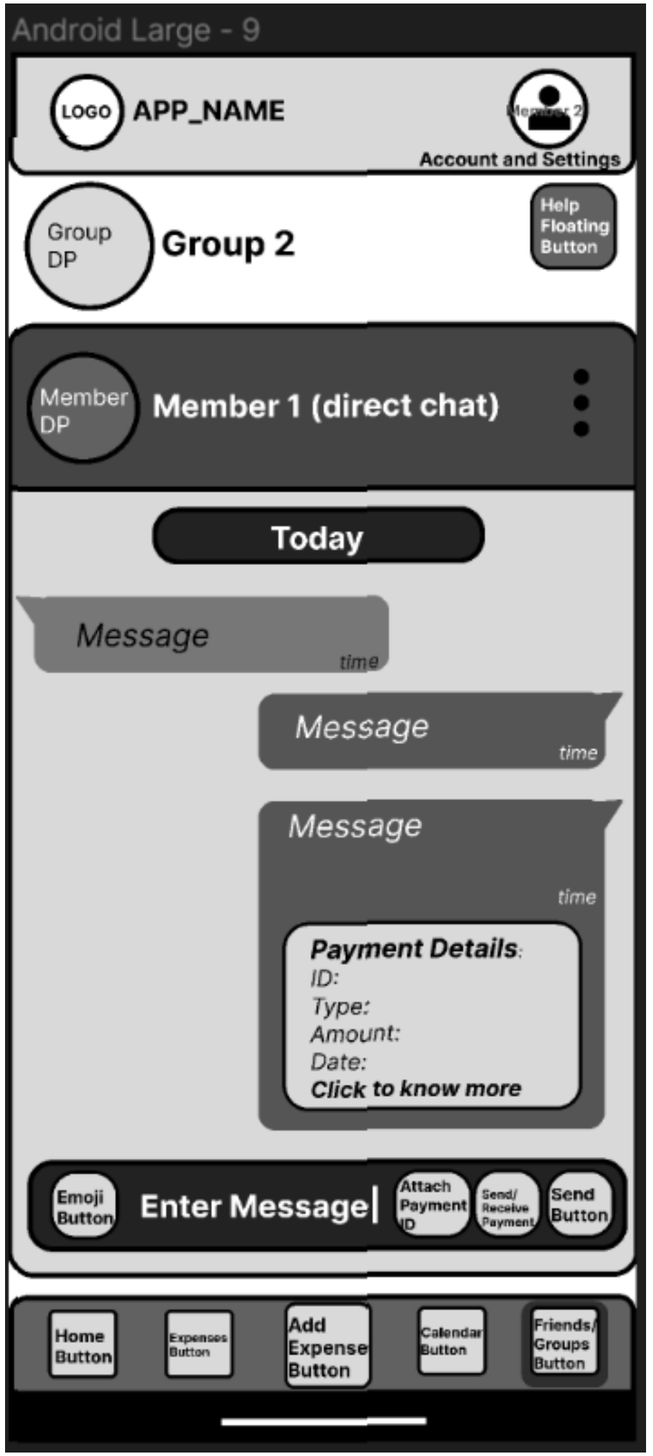
12. View Expense Page



13. Friends/Groups (Messages) Page

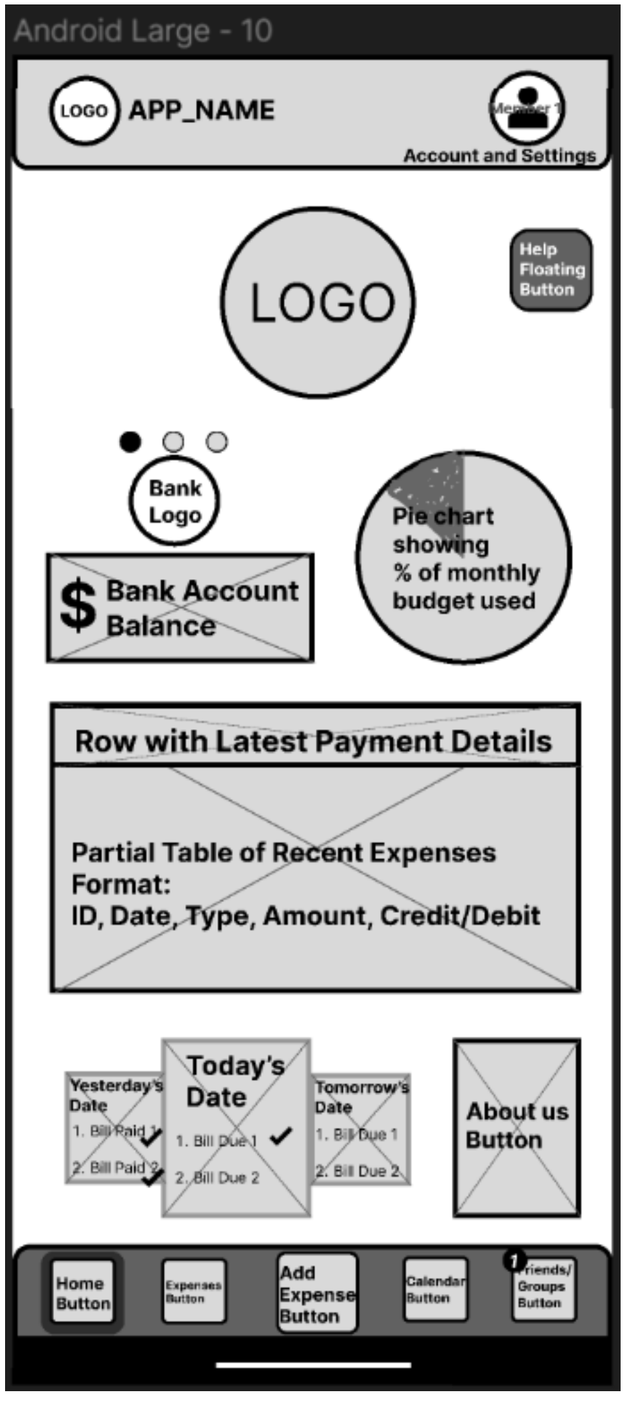


14. Individual Chat (Messages) Page

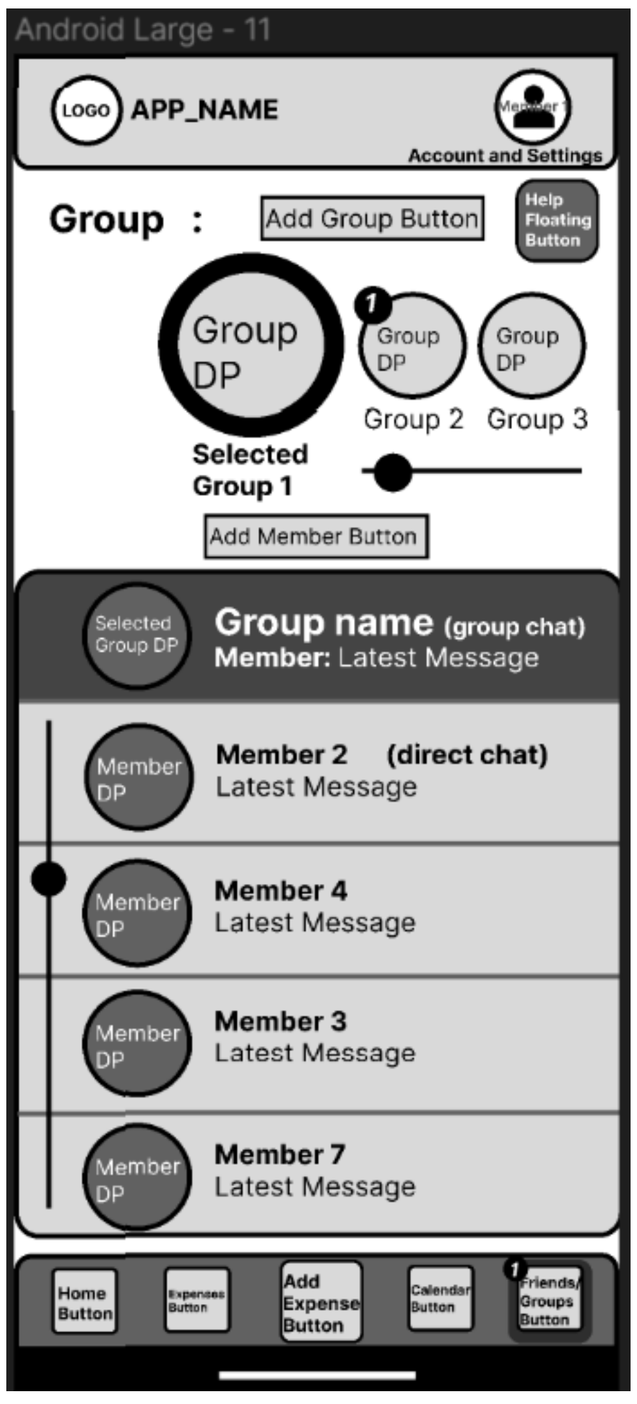


15. Individual Chat (Messages) Page

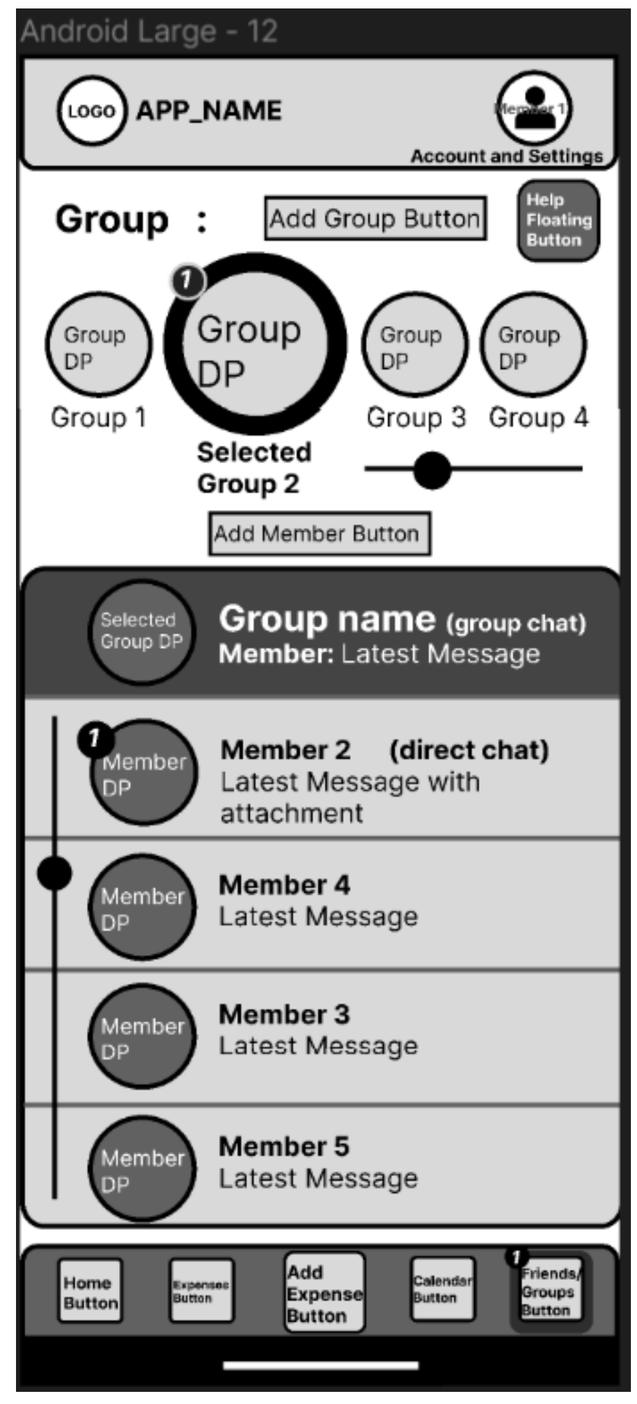
Part 2: Alex Singh's Perspective:



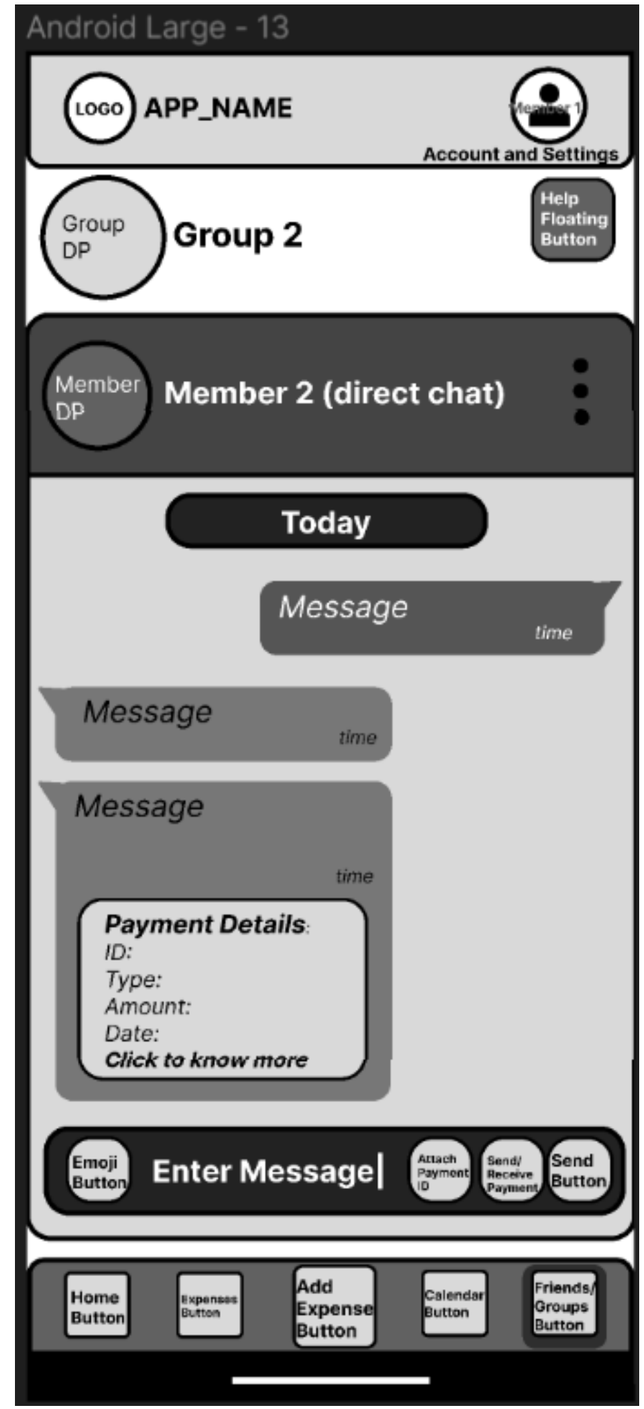
16. Home Page



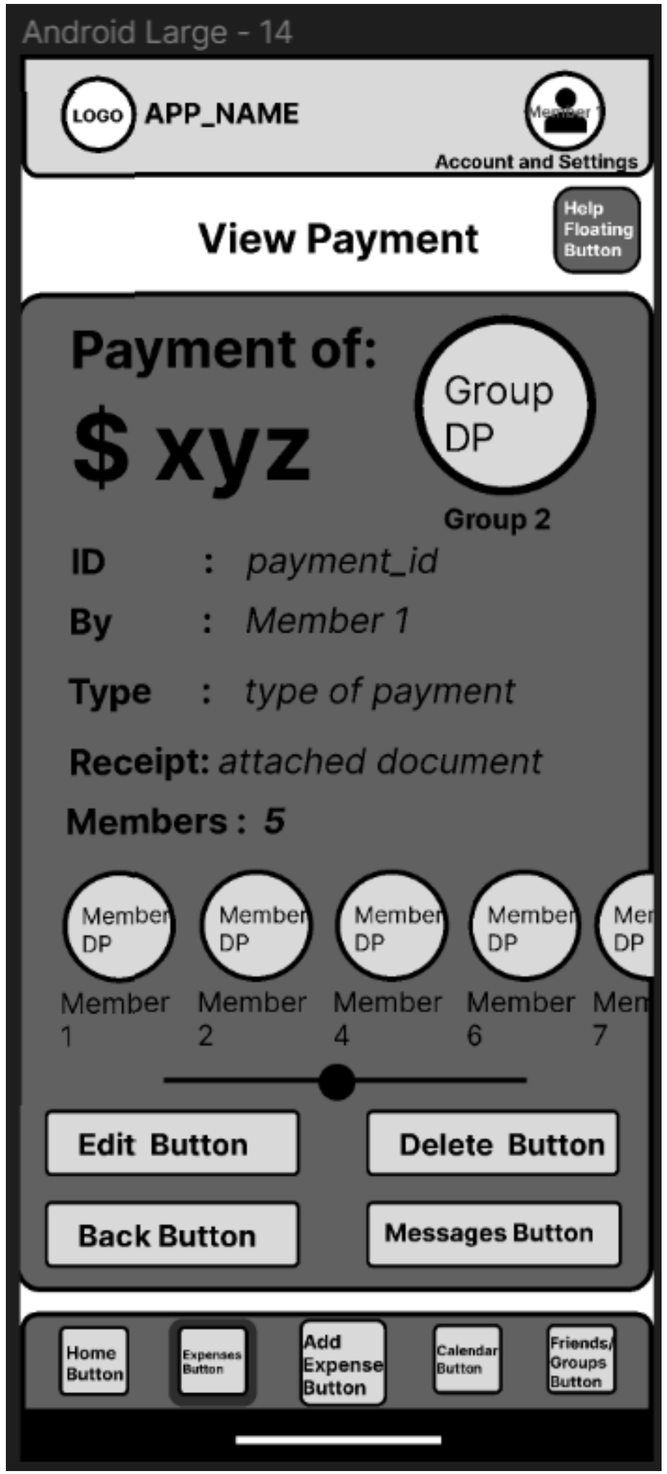
17. Friends/Groups (Messages) Page



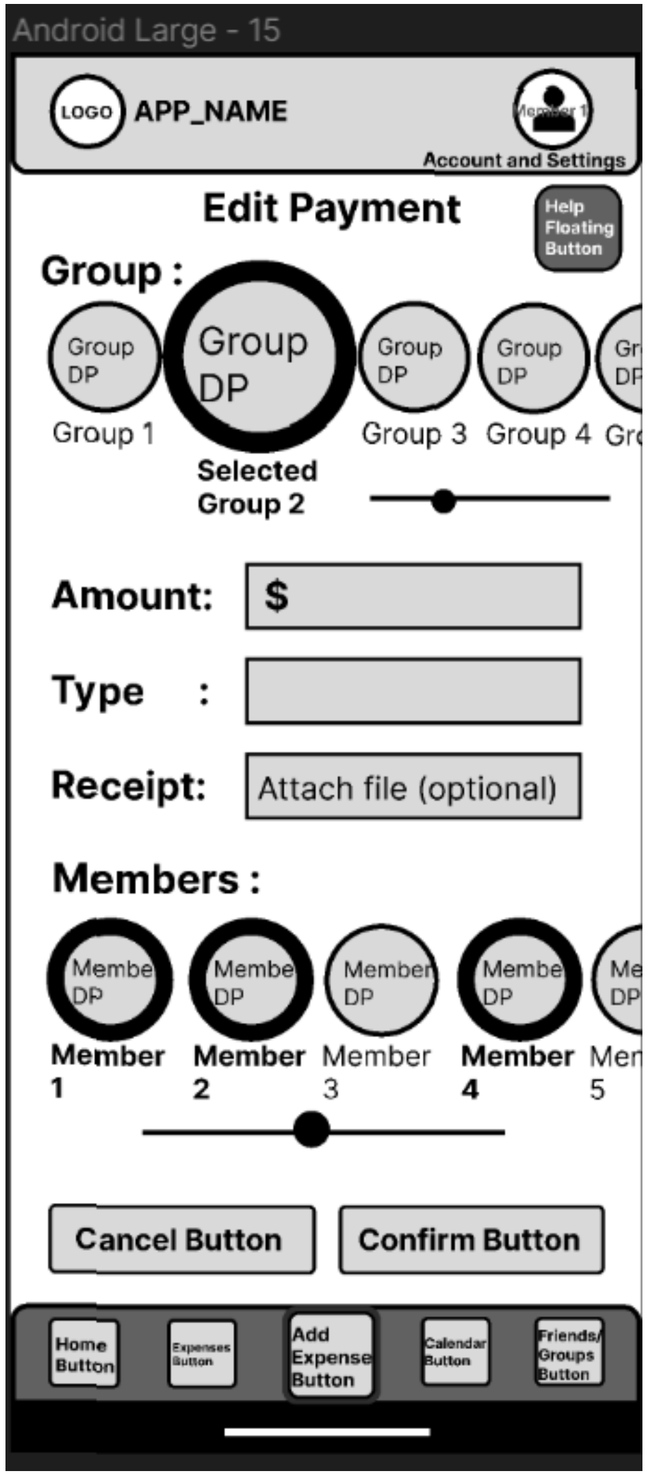
18. Friends/Groups (Messages) Page



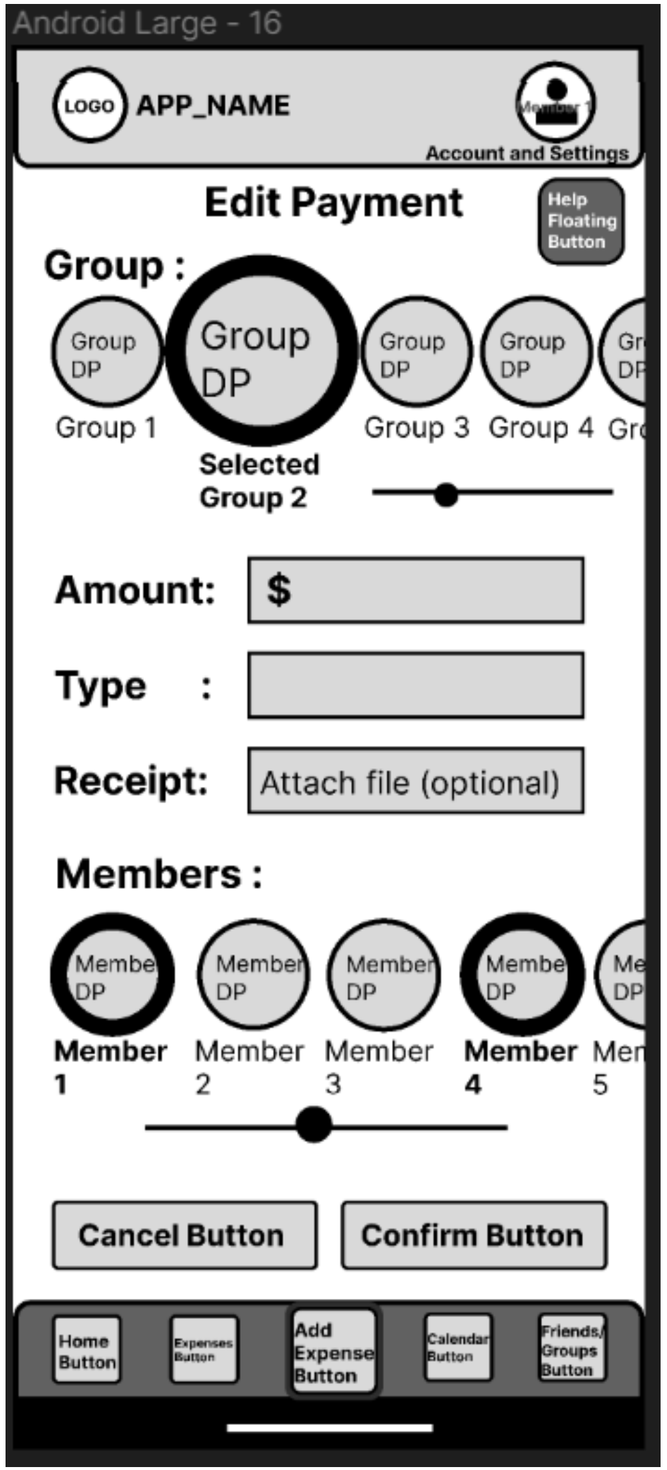
19. Individual Chat (Messages) Page



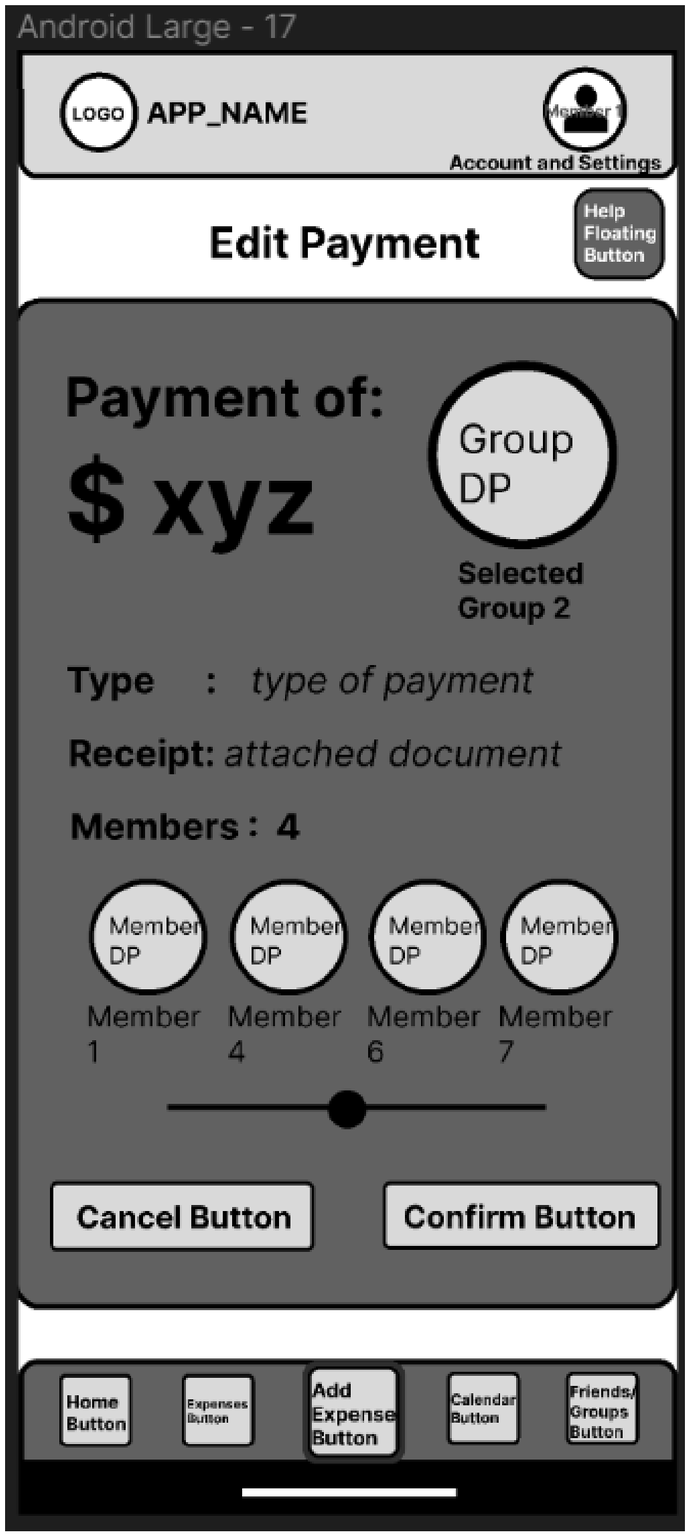
20. View Payment Page



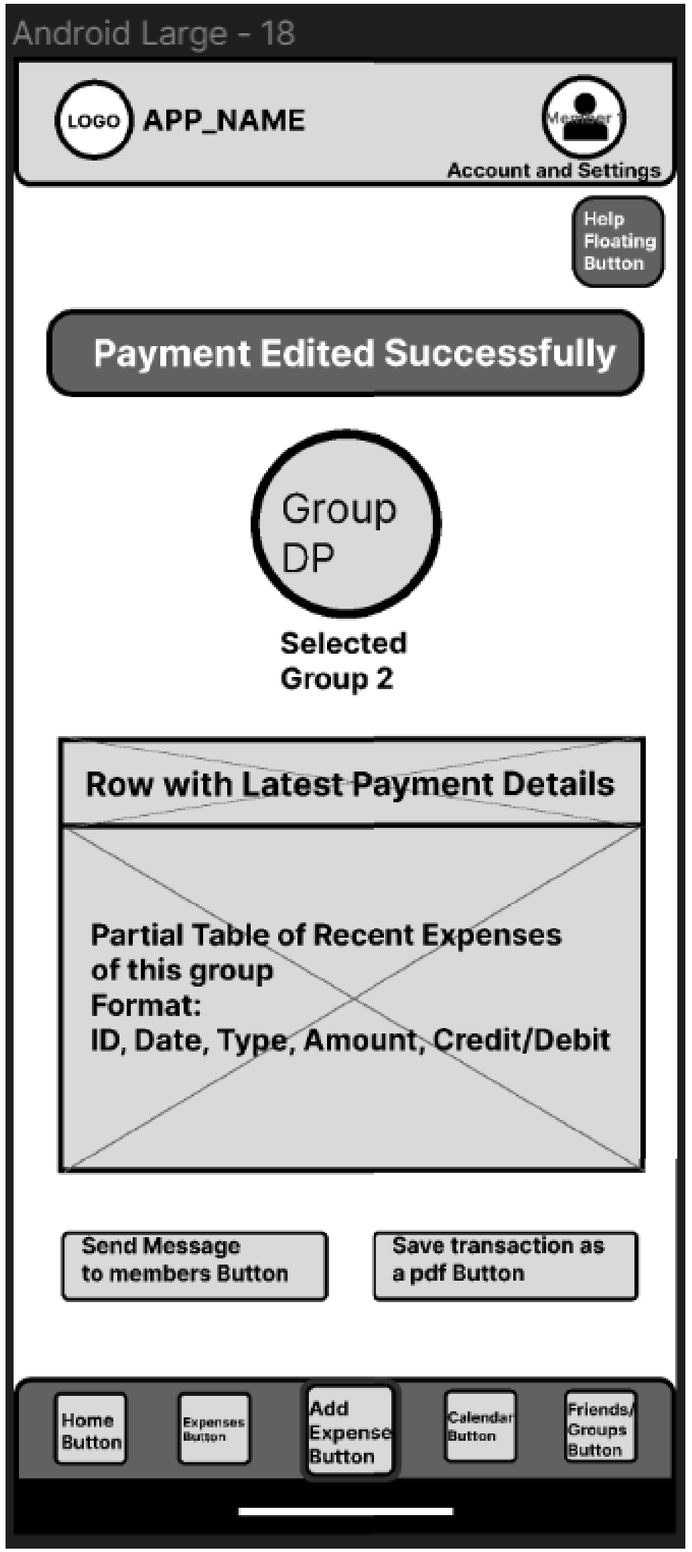
21. Edit Payment Page



22. Edit Payment Page



23. Edit Payment Review Page

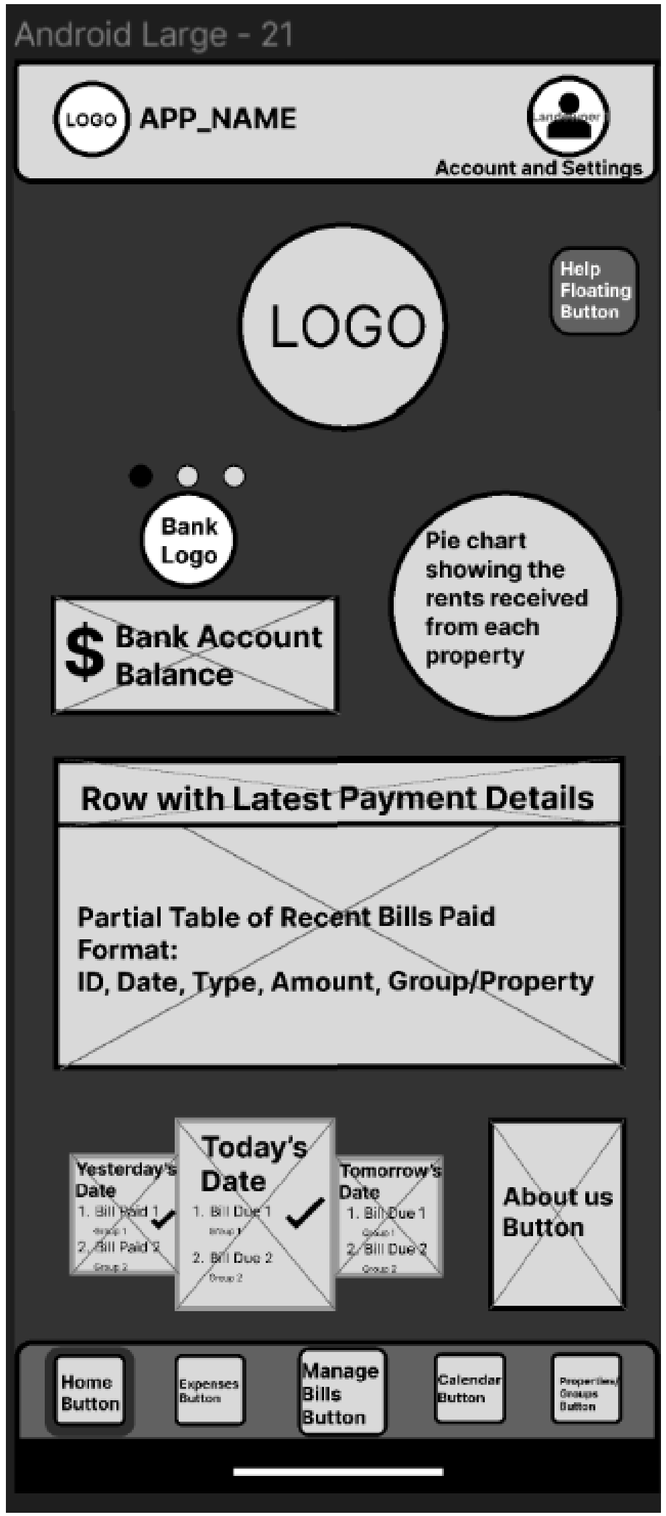


24. Edit Payment Confirmation Page

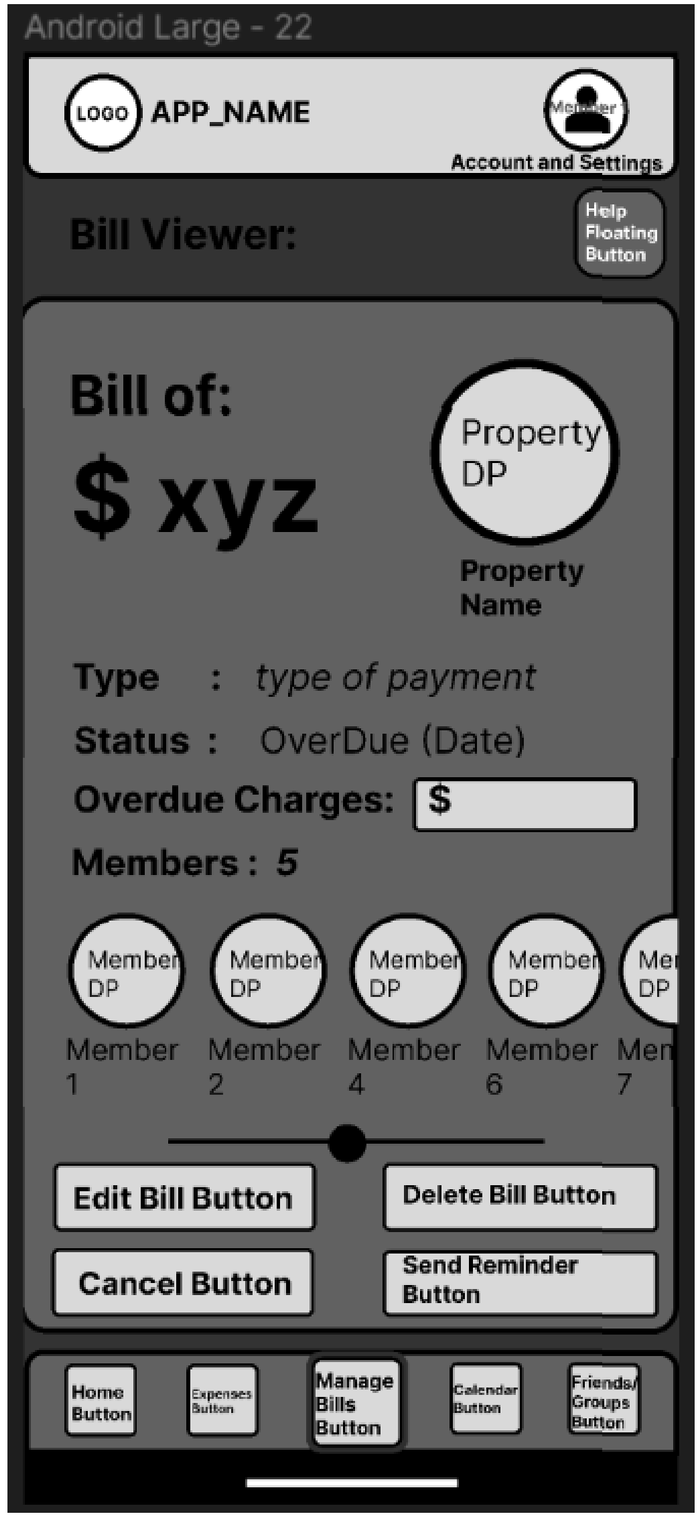
User Task 3:

Remind a group of tenants to pay an overdue bill and add a late penalty to it.

Persona: Trevor Smith, a flat owner, who expects their tenants to pay bills like rent regularly. They always like tenants to be prompt in paying their bills and is strict on penalties.

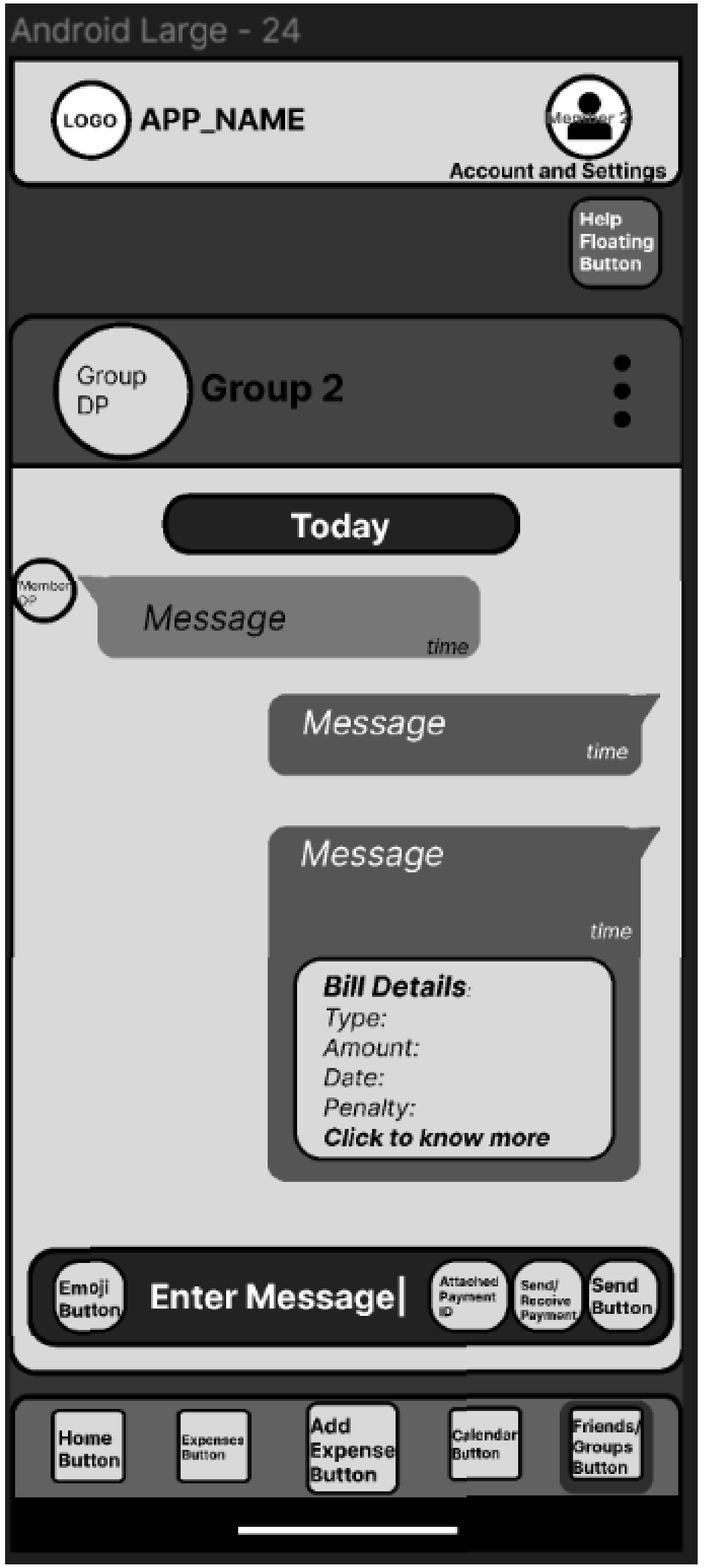


25. Home Page

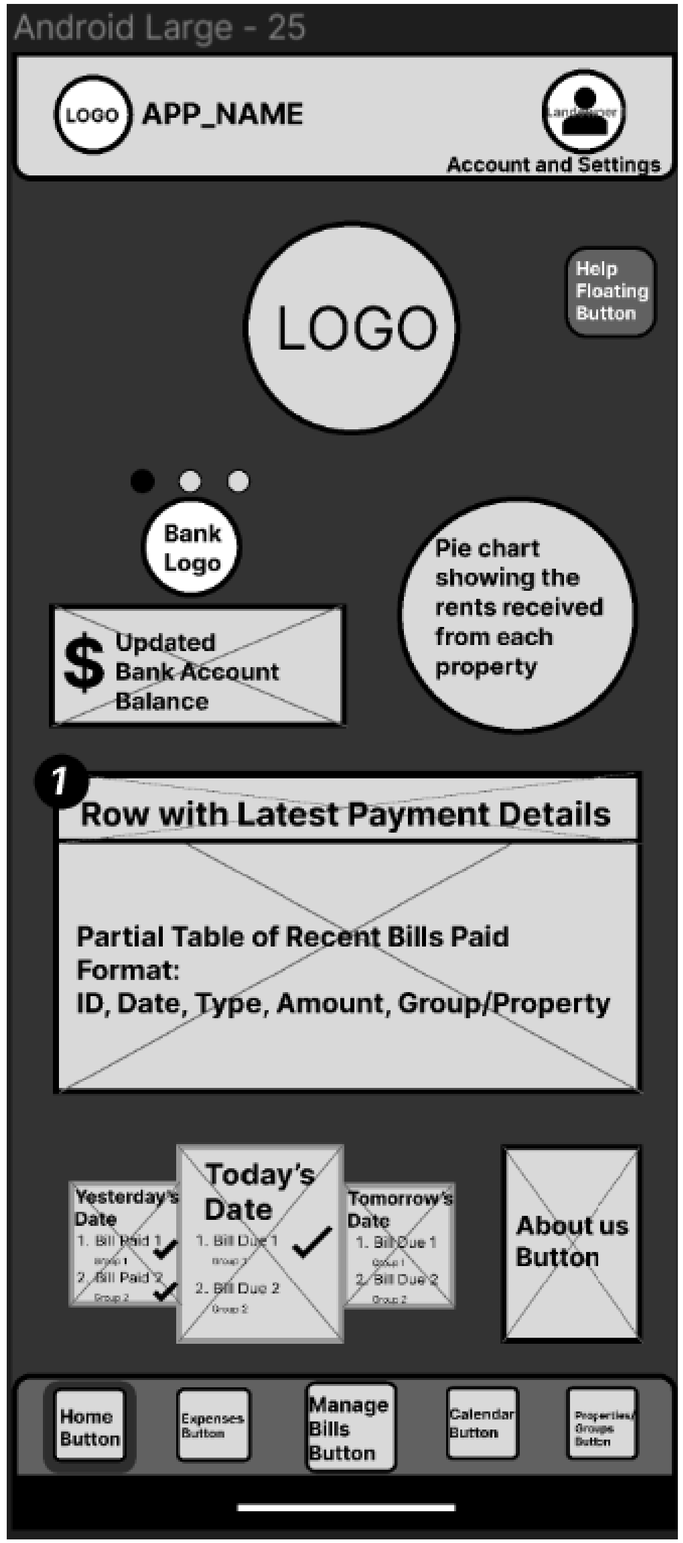


26. View Bill Page

27. Group Chat/Messages Page



28. Group Chat/Messages Page



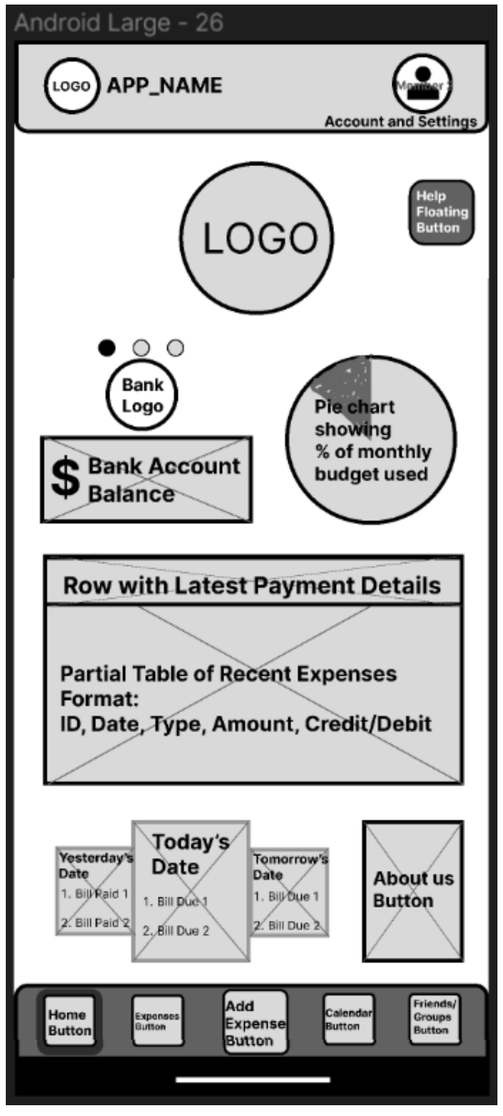
29. Home Page

User Task:

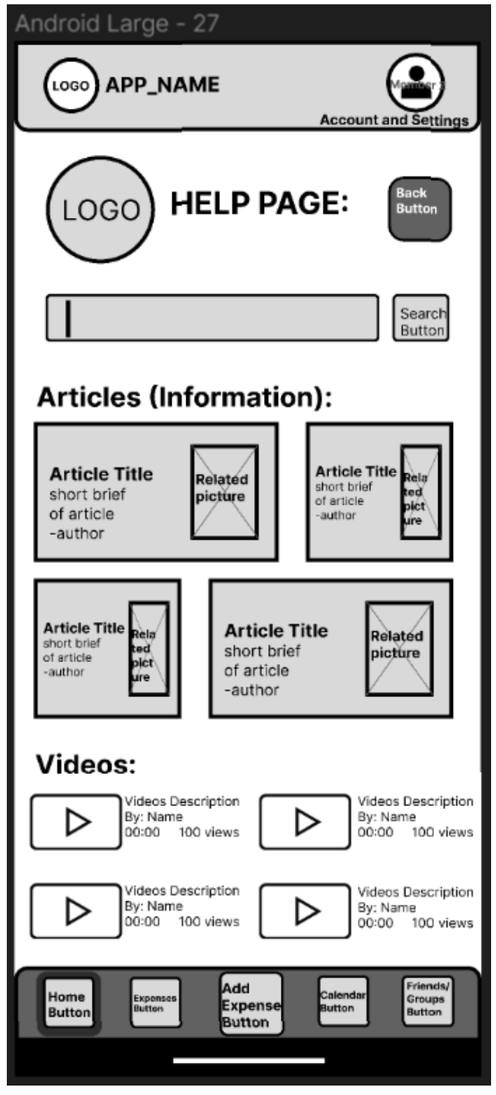
Understand how a particular feature in the app works (get help) to finish a task OR to gain extra knowledge on budgeting. This can be done during any other task like Task1, 2 and 3.

Personas:

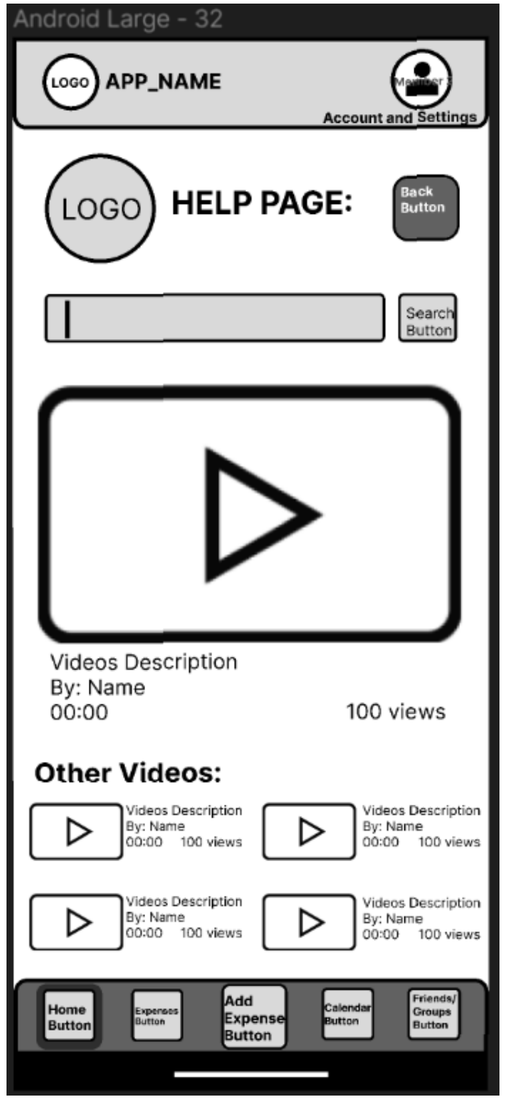
- **John Baker**, a fellow flatmate, who is cautious and takes their time looking at finer details
- **Financial experts** as a stakeholder who would want to ensure that financing information is accurate
- **Technical staff and support** as a stakeholder who would want the customer to use the application properly.



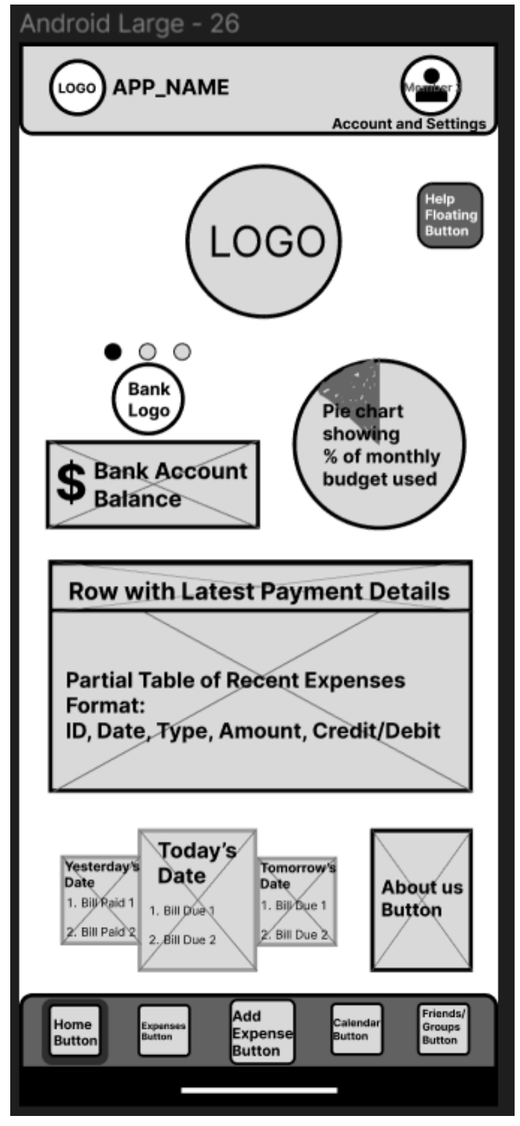
30. Home Page



31. Help Page



32. Help Video Page



33. Home Page

2.User flows

USER FLOW 1

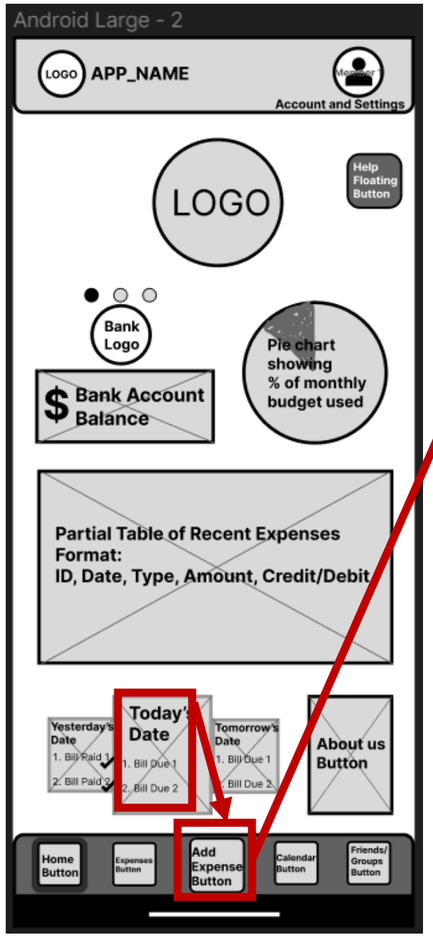
User Goal: Add a new payment detail within a group and make the payment

Persona: Alex Singh who has a very busy schedule and wants to get jobs done fast

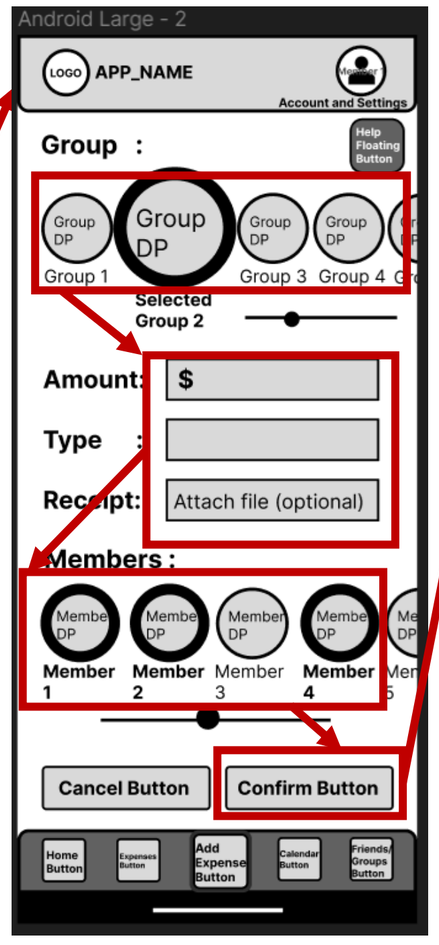
Tasks: 1. Notice that a payment must be done on the calendar widget on the dashboard (but not go to the Calendar tab since Alex does not have much time to check the entire calendar)

2. Make a payment by selecting and confirming details like group, amount, type, receipt & members.

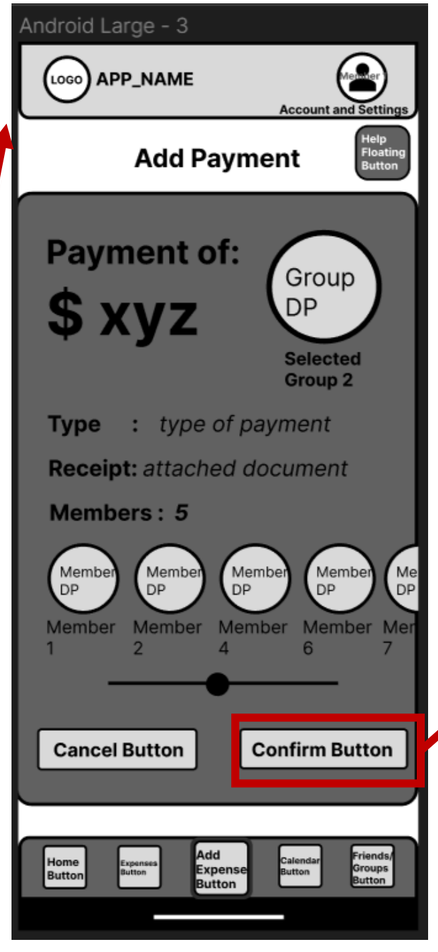
3. Confirm that the payment is done by glancing at their bank balance and expenses table widgets on the dashboard on the "Home Page".



1. HOME PAGE:
Check Today's Due Bills in the Calendar Dashboard and click on the "Add Expense" button to pay the bill



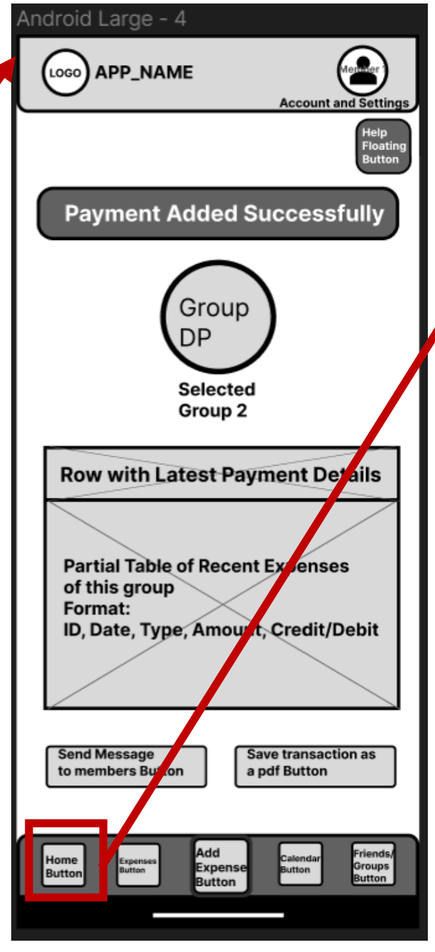
2. ADD EXPENSE PAGE:
Scroll Left-Right to select the group (indicated by thick border), enter the amount and type of bill, attach a receipt and scroll left-right to select the members (indicated by thick border). Click the 'Confirm" button when all details are entered.



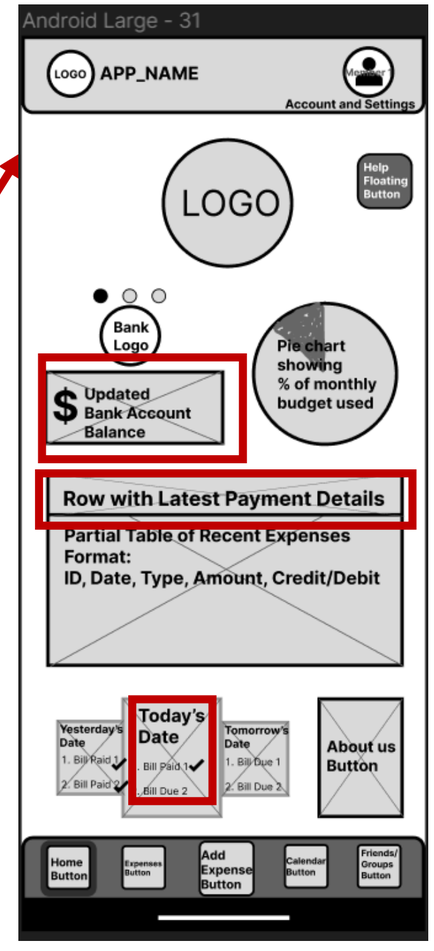
3. ADD EXPENSE REVIEW PAGE:
Review all the previously-entered details (group, amount, type, receipt, members) and click the "Confirm" button again to confirm they are all correct.

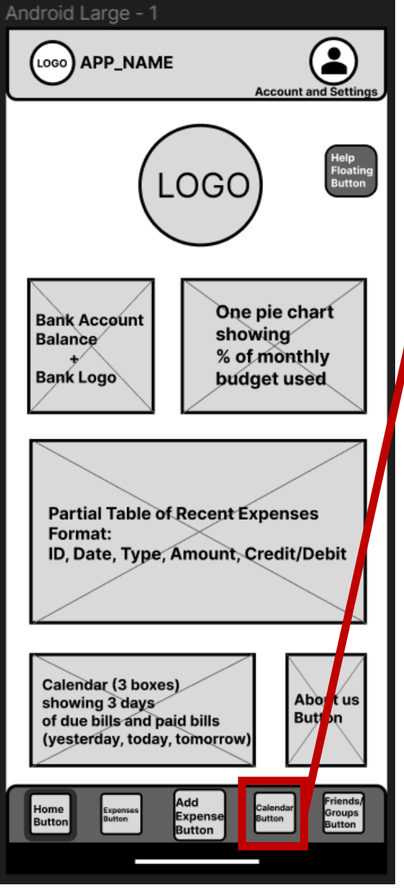
Payment made through API through bank application

4. ADD EXPENSE COFIRMATION PAGE:
The payment details are added to the expenses table and a confirmation message is seen. If clicked on the "Home" Button...



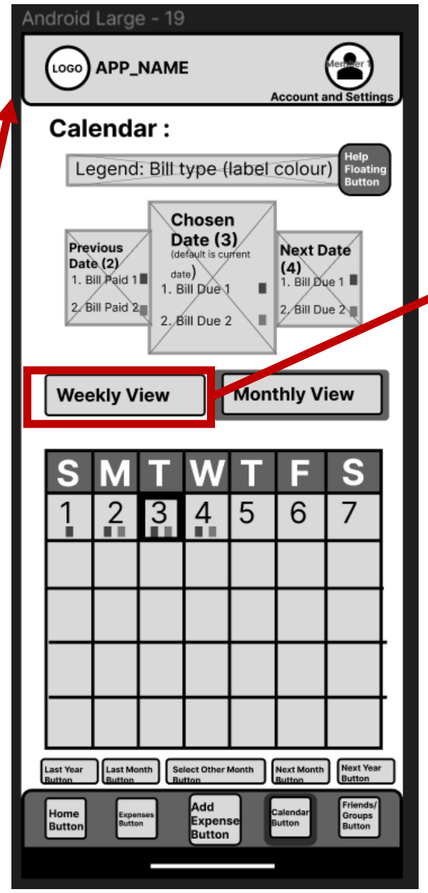
5. HOME PAGE:
If the "Home" button is clicked, and we look at the Calendar Widget on the Dashboard, the bill is paid (indicated by a tick). The bank balance and expense table shown are also updated.





1. HOME PAGE:

Go to the Calendar page to check when any payments are due.

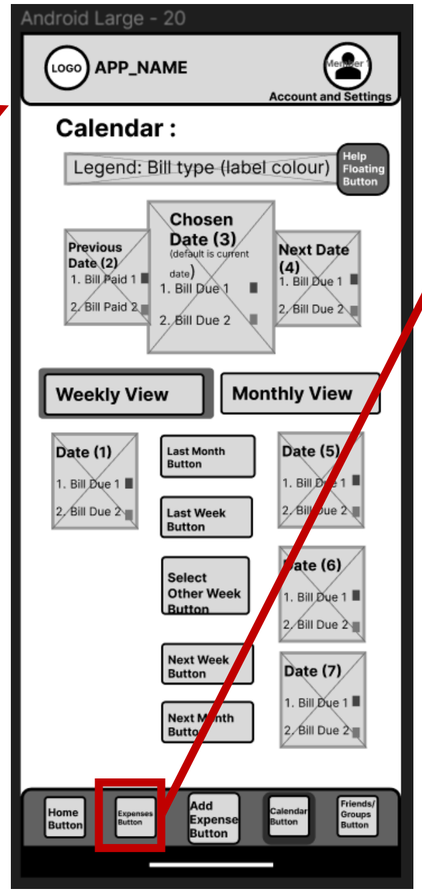


2. CALENDAR PAGE (MONTHLY VIEW):

Check the monthly view of bills due. Click on the ‘Weekly” view to see the bills due in the current week

3. CALENDAR PAGE (WEEKLY VIEW):

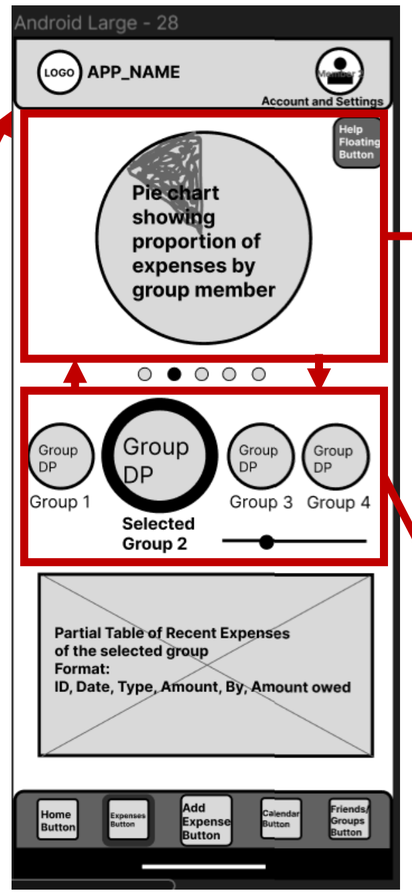
Now that we know the date for the payment, go to the Expenses page to check personal expenses

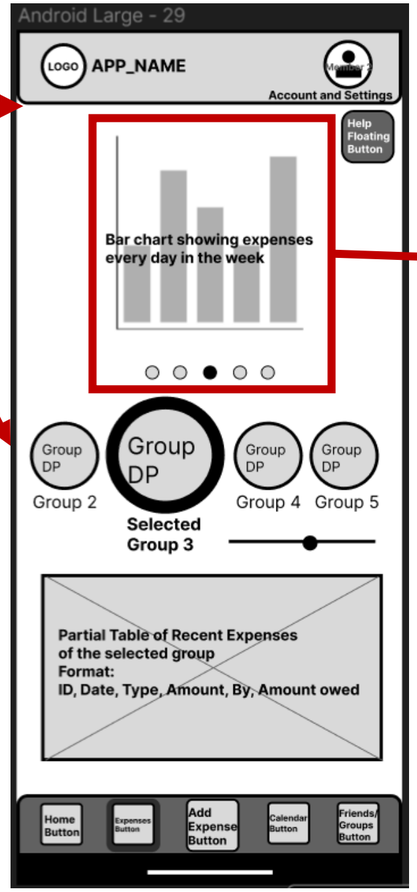


4. EXPENSE TABLES AND GRAPHS PAGE (different views):

Change graphs of a particular group by swiping left-right to select what type of financial information one wants to see.

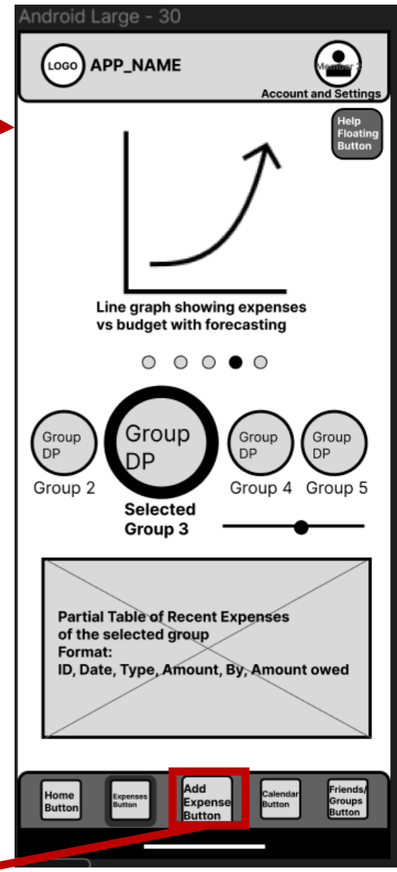
Change the group for which you want to see the recent expenses table and the graphs for by swiping left-right.





5. EXPENSE TABLES AND GRAPHS PAGE:

After reviewing the financial details, add an expense/payment by pressing the button below.



USER FLOW 1

with a different Persona

User Goal: Add a new payment detail within a group and pay for it

Persona: John Baker, a fellow flatmate, who is cautious and takes their time looking at finer details.

Tasks: 1. Go to the Calendar page and check which payment is due.

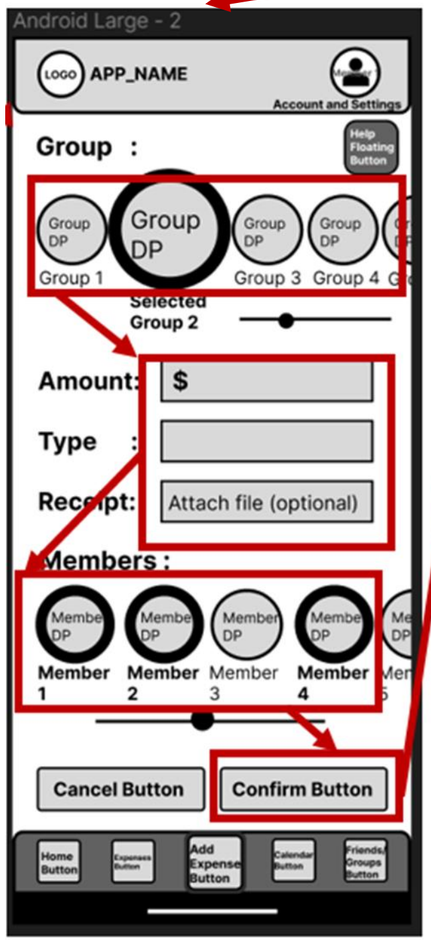
2. Go to the Expenses page to check their own budget and financial details to know if they are eligible to pay.

3. Make a payment by selecting and confirming details like group, amount, type, receipt & members.

4. Confirm that the payment is done by glancing at their bank balance and expenses table widgets on the dashboard on the “Home Page”.

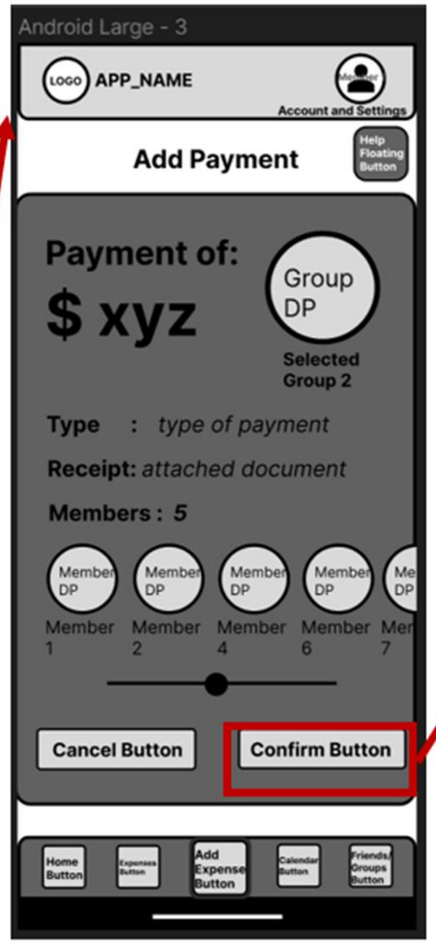
USER FLOW 1

with a different Persona - CONTINUED



6. ADD EXPENSE PAGE:

Scroll Left-Right to select the group (indicated by thick border), enter the amount and type of bill, attach a receipt and scroll left-right to select the members (indicated by thick border). Click the 'Confirm" button when all details are entered.



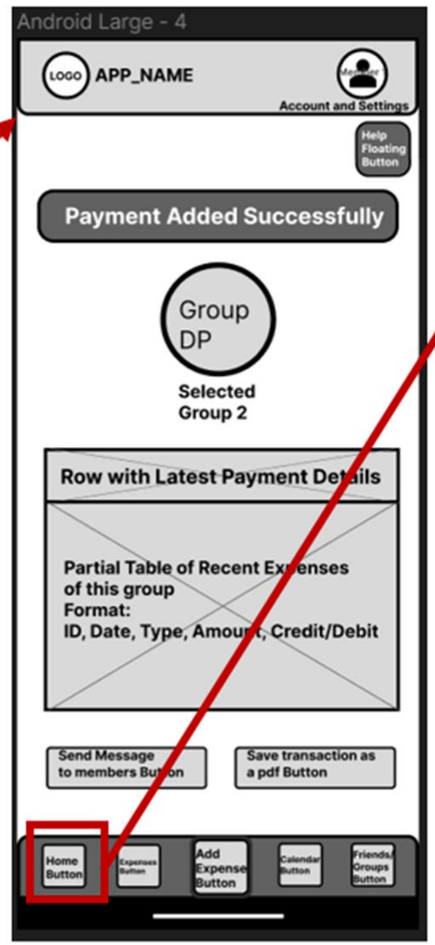
7. ADD EXPENSE REVIEW PAGE:

Review all the previously-entered details (group, amount, type, receipt, members) and click the "Confirm" button again to confirm they are all correct.

Payment made through API through bank application

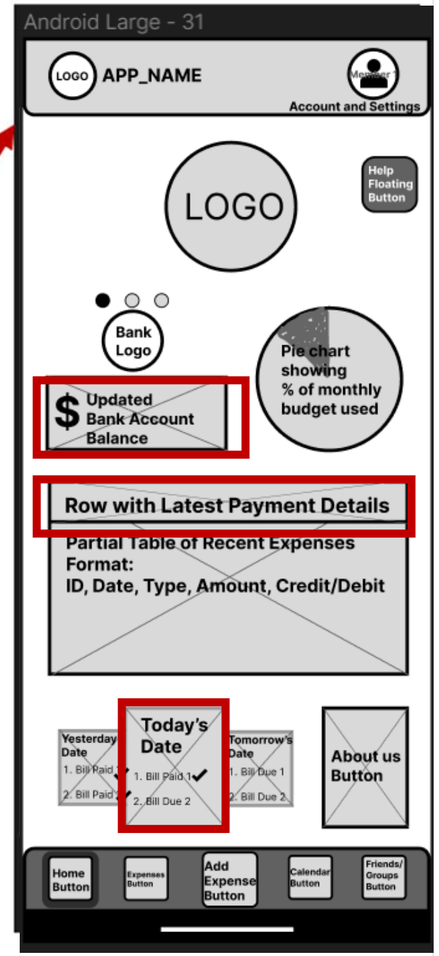
8. ADD EXPENSE CONFIRMATION PAGE:

The payment details are added to the expenses table and a confirmation message is seen. If clicked on the "Home" Button...



9. HOME PAGE:

If the "Home" button is clicked, and we look at the Calendar Widget on the Dashboard, the bill is paid (indicated by a tick). The bank balance and expense table shown are also updated.



USER FLOW 2

Part 1: John Baker's Perspective

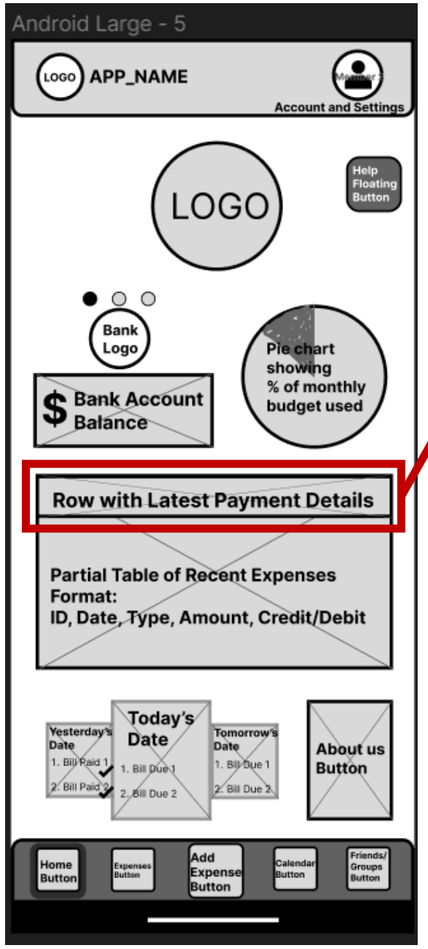
User Goal: Edit a payment's details in which Alex Singh (flatmate) added John Baker as a member by accident, since they were in hurry to add the payment due to their busy schedule.

Persona: John Baker, a fellow flatmate, who is cautious and takes their time looking at finer details.

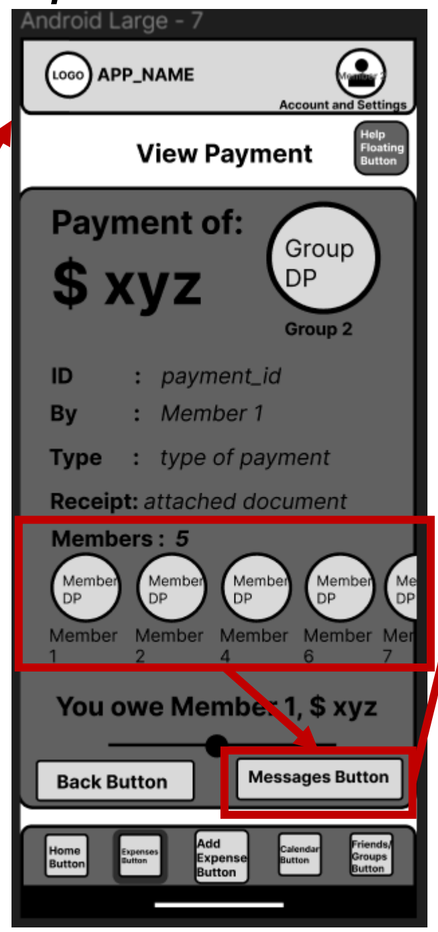
Tasks: 1. Notice that a new payment has been added and review the details of it

2. Notice that they (John Baker) have been added as a member for that payment by mistake

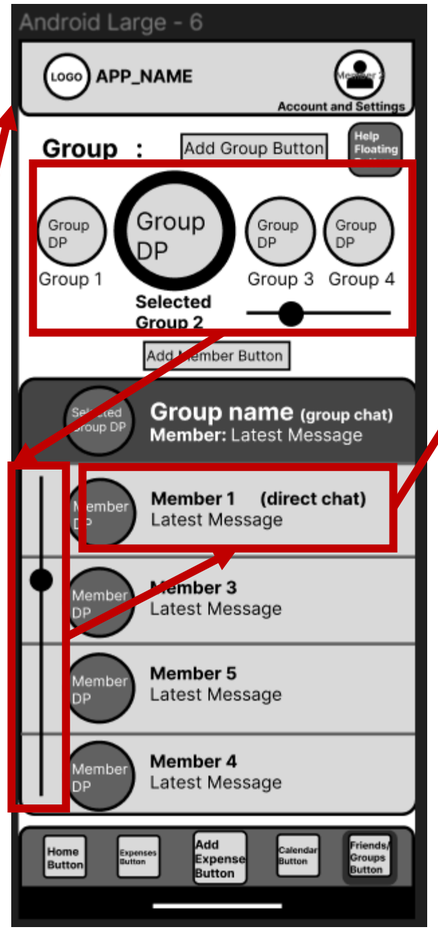
3. Notify and message the person who made the payment (Alex SIngh) with the attached payment so that they can act on it.



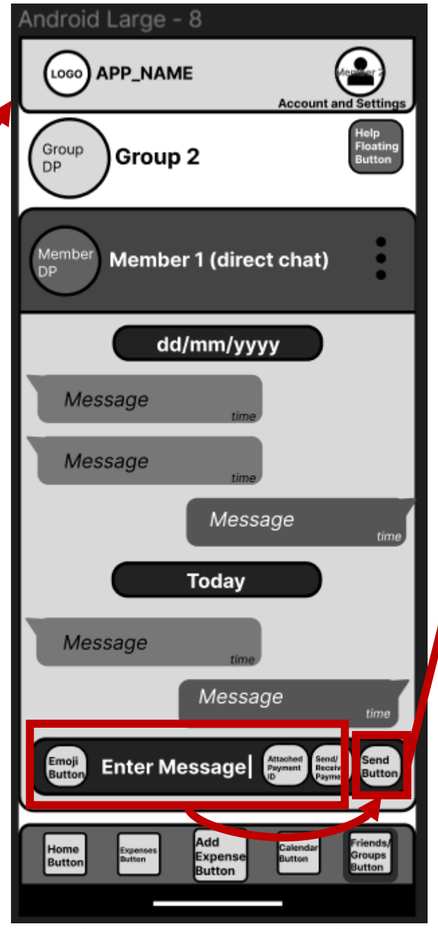
1. HOME PAGE:
See the latest payment details row on the expense table widget on the dashboard on the homepage and press on it.



2. VIEW EXPENSE PAGE:
Sees that they are a member involved in the payment and that they owe Member 1 (Alex Singh) some money (which should not be the case). Press the Messages Button to discuss with Alex about this.

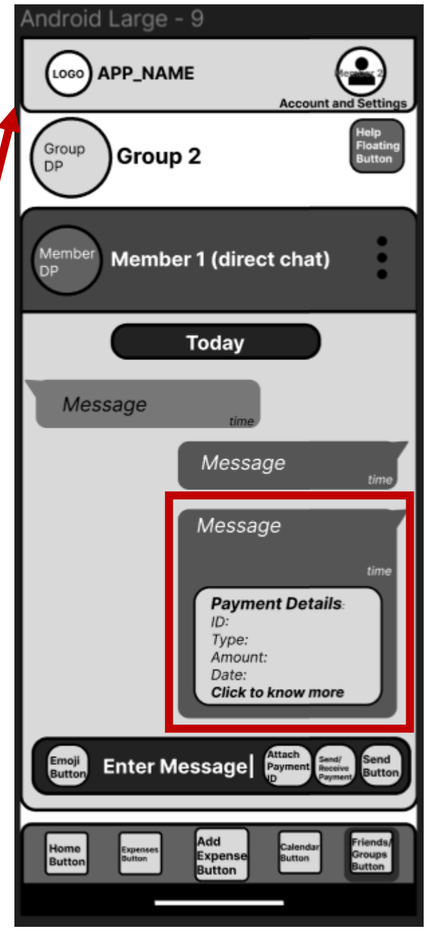


3. FRIENDS/GROUPS (MESSAGES) PAGE:
Is taken to the Messages page with the correct group pre-selected. They scroll up or down and select the chat of the Member they want to message to resolve the issue with.



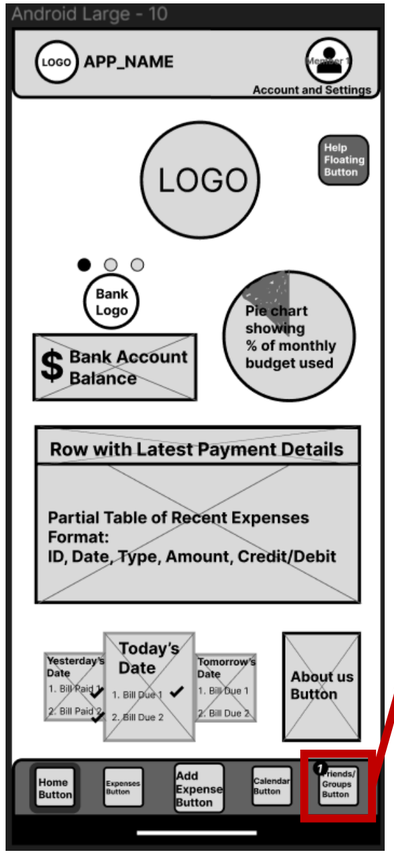
4. INDIVIDUAL CHAT/MESSAGE PAGE:
Type a valid message explaining the situation, with the payment details attached and send the message by pressing the Send Button

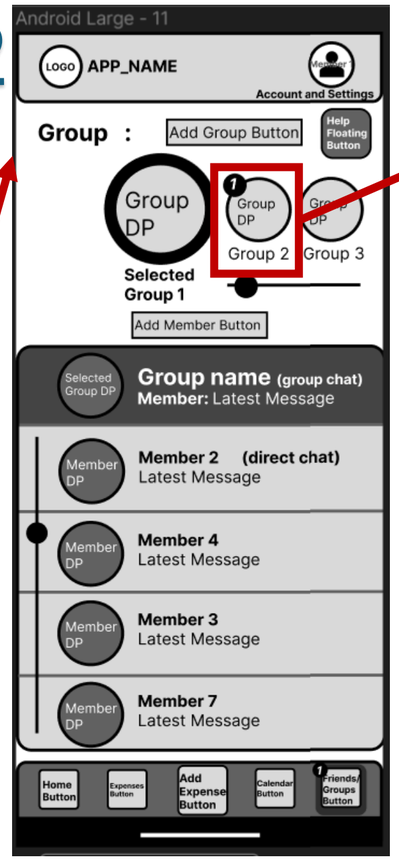
5. INDIVIDUAL CHAT/MESSAGE PAGE:
The Message with the Payment Details attachment is sent and the member (Alex Singh) must read it to edit the payment details and remove John as a member

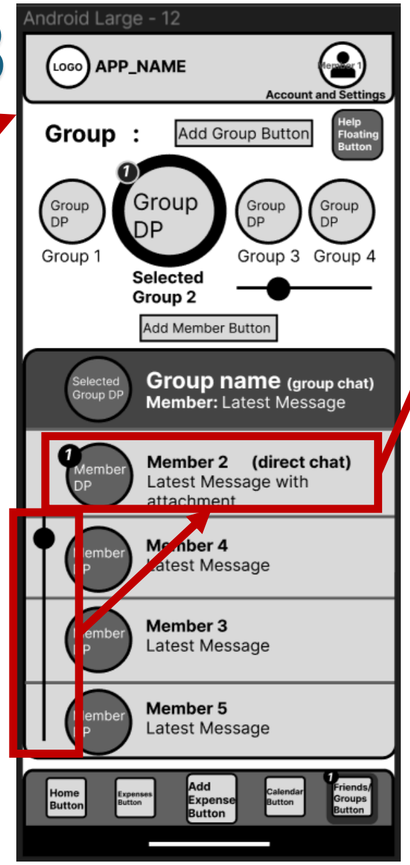


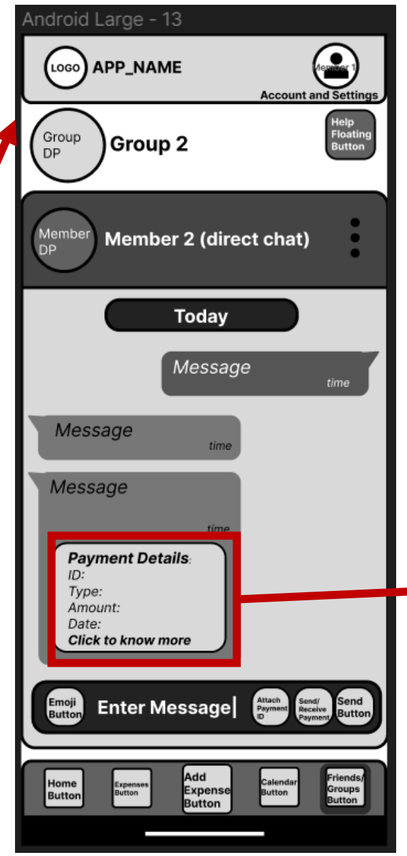
USER FLOW 2

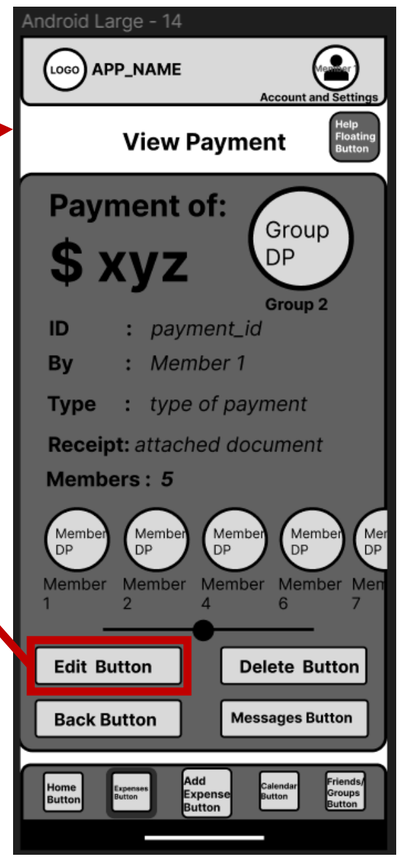
Part 2: Alex Singh's Perspective

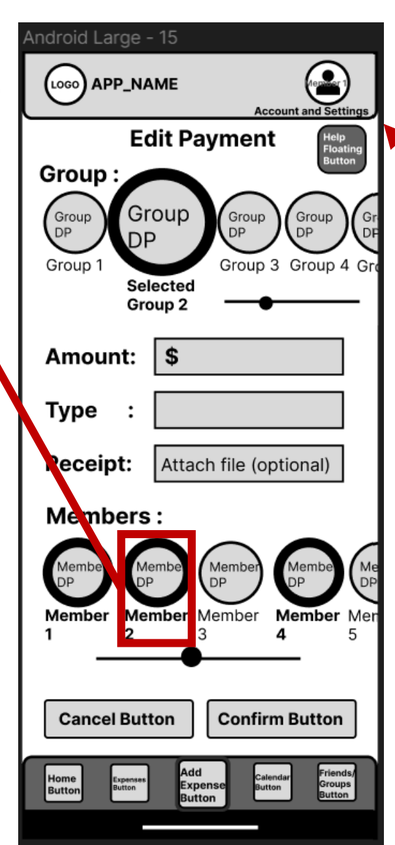


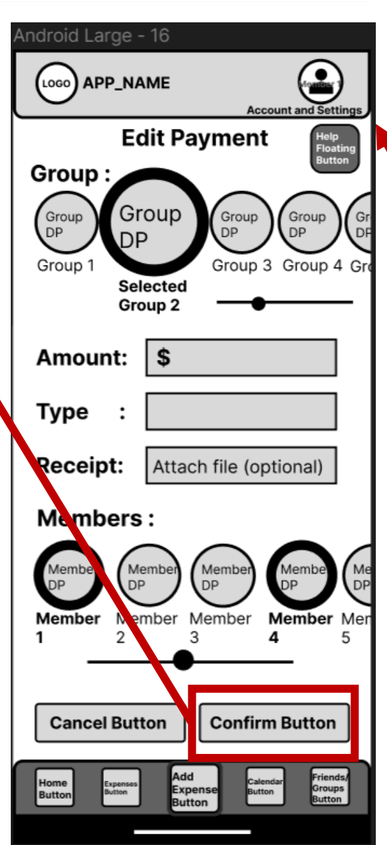


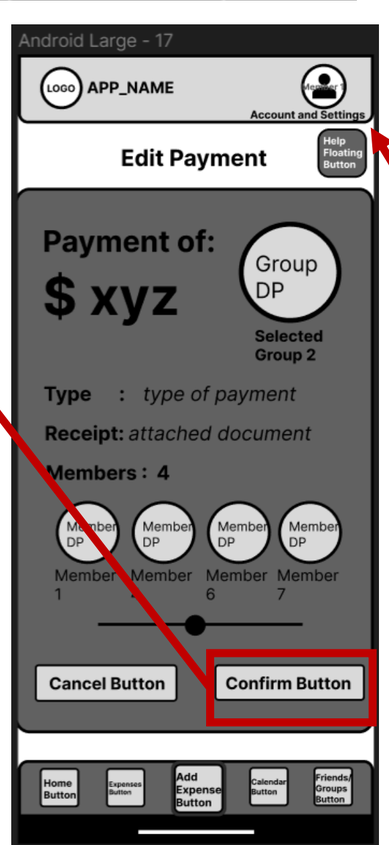


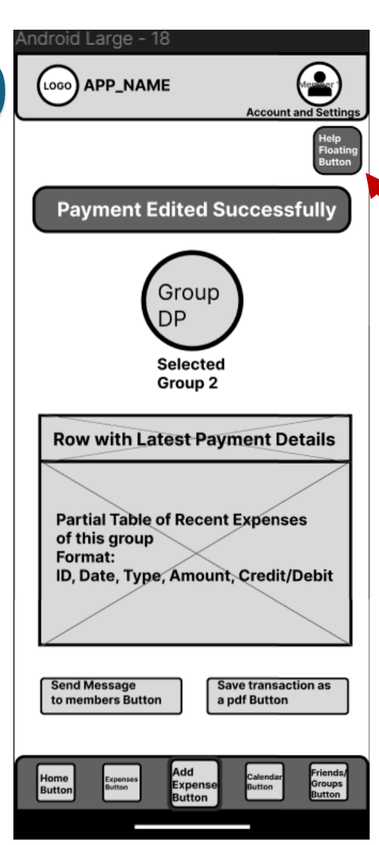












Details on next page based on the wireframe numbers

USER FLOW 2

Part 2: Alex Singh's Perspective – CONTINUED

Persona: Alex Singh who has a very busy schedule and wants to get jobs done fast

1 **HOME PAGE:**
Notice that there is a new message notification bubble on the Friends/Groups Button and Press it

2 **FRIENDS/GROUPS (MESSAGES) PAGE:**
Scroll Left-Right to select the Group whose DP has a notification bubble

3 **FRIENDS/GROUPS (MESSAGES) PAGE:**
Scroll Up-Down to select the Member whose DP has a notification bubble

4 **INDIVIDUAL CHAT/MESSAGES PAGE:**
Notice the latest message sent by them mentioning that you have done an incorrect payment and click on the payment details attachment to view further details

5 **VIEW PAYMENT PAGE:**
Observe the complete payment details and notice that indeed you have added Member 2 (John Baker) by mistake. Click the "Edit Payment" Button to correct the mistake.

6 **EDIT PAYMENT PAGE:**
Unselect Member 2 (John Baker) as a member for the particular payment by pressing on their DP in the Members section of the payment details.

7 **EDIT PAYMENT PAGE:**
Confirm these changes (removed John Baker as a member of the payment) by pressing on the "Confirm" Button

8 **EDIT PAYMENT REVIEW PAGE:**
Review the new details of the payment and confirm them by pressing on the "Confirm Button"

9 **EDIT PAYMENT CONFIRMATION PAGE:**
The payment details are edited in the expenses table and a confirmation message is seen.

USER FLOW 3

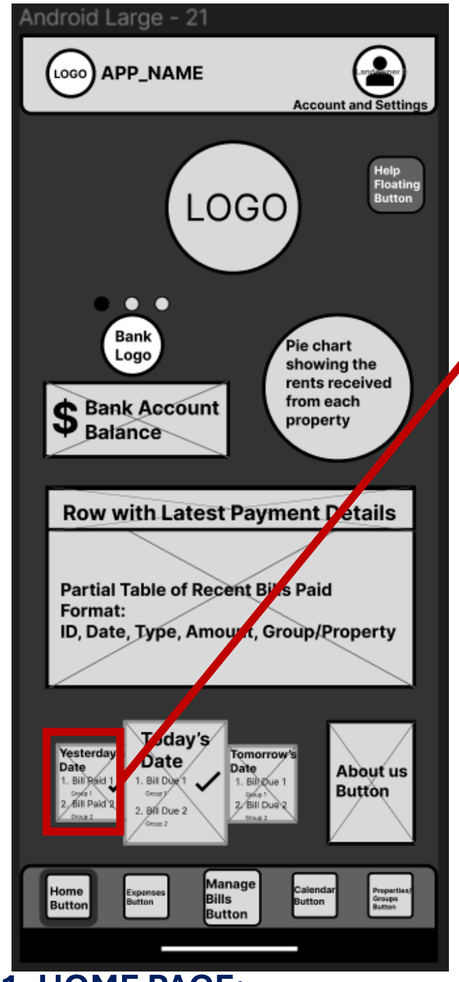
User Goal: Remind a group of tenants to pay an overdue bill and add a late penalty to it.

Persona: Trevor Smith, a flat owner, who expects their tenants to pay bills like rent regularly

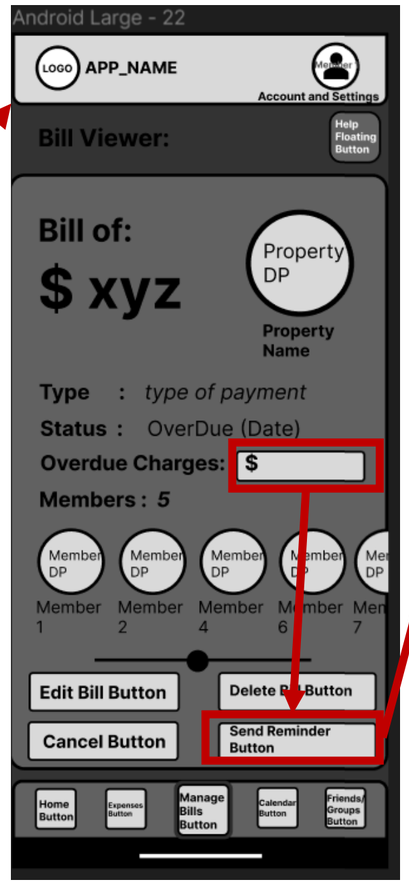
Tasks: 1. Notice that a bill is overdue (due the day before) from a Group on the Calendar Widget on the dashboard, press on it and view the bill details

2. Enter the late penalty amount and send a reminder message to the group with an attachment of the bill

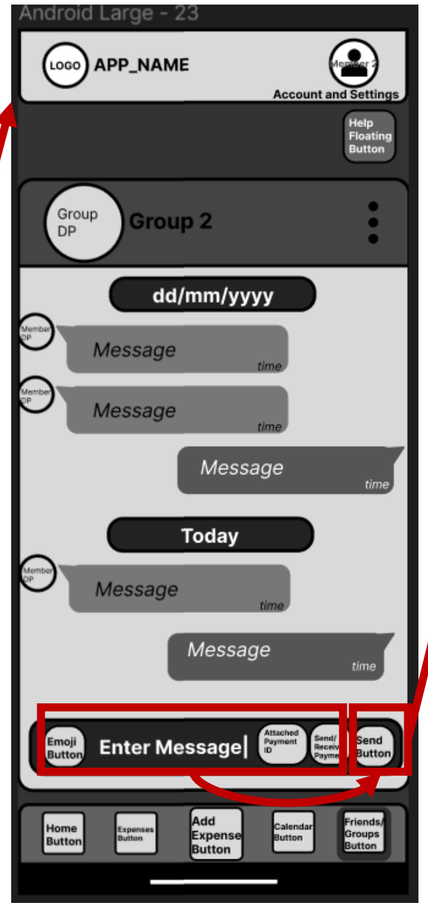
3. When the bill is finally paid, notice that the bill has been paid on the same Calendar Widget on the dashboard



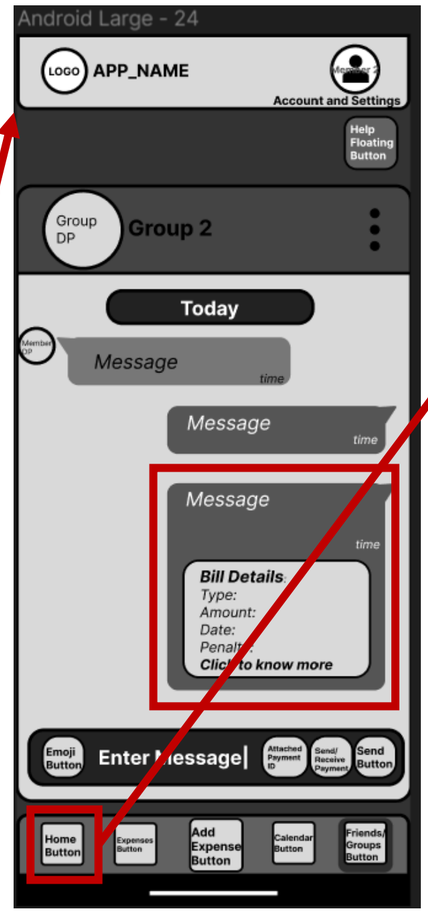
1. HOME PAGE:
See on the Calendar Widget on the dashboard that a particular group has not paid a bill due the day before (indicated by no tick). Click on the bill ID to view more details



2. VIEW BILL PAGE:
View the details of the unpaid bill and enter a late penalty. Press the "Send Reminder" Button to send a message to all the related group members to encourage them to pay the bill along with the penalty.



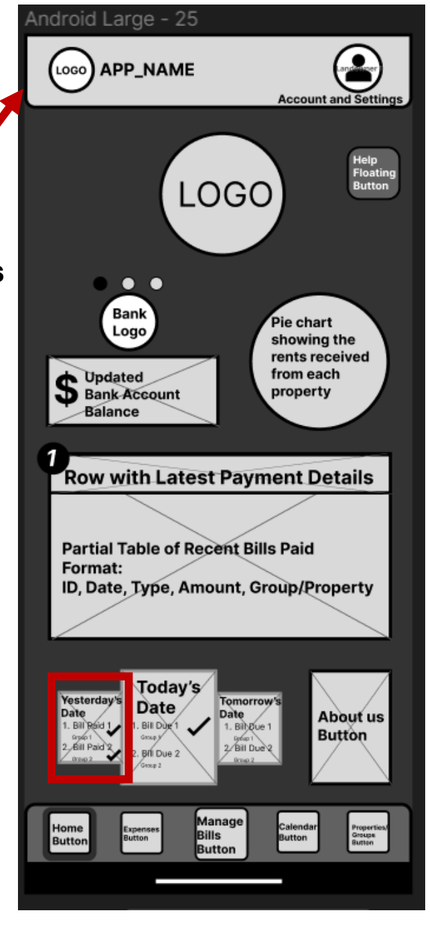
3. GROUP CHAT/MESSAGES PAGE:
Enter the message and attach the bill as a reminder for the group to pay it and send it using the "Send" Button.



4. GROUP CHAT/MESSAGES PAGE:
The message is sent in the group chat so that all the members can get reminded to pay the bill

After Bill is paid

5. HOME PAGE;
When the bill is paid, the calendar widget on the dashboard shows a tick beside the bill indicating that it is paid.



USER FLOW

for “Help” while doing any other task like Tasks 1,2 or 3
(It can be an intermediate task when doing either of the other tasks)

User Goal: Understand how a particular feature in the app works (get help) to finish a task OR to gain extra knowledge on budgeting

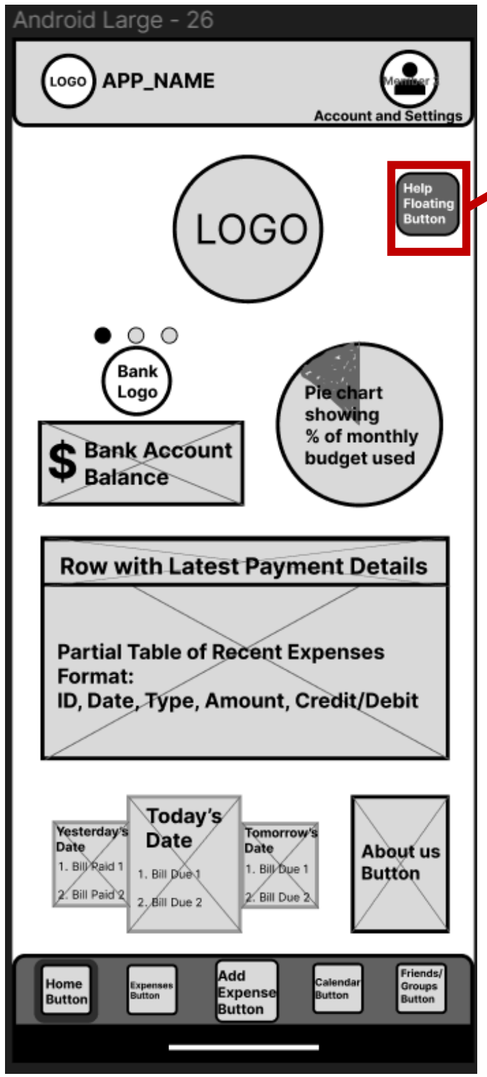
Persona: John Baker, a fellow flatmate, who is cautious and takes their time looking at finer details

Also**, financial experts** as a stakeholder who would want to ensure that financing information is accurate and **technical staff** as a stakeholder who would want the customer to use the application properly.

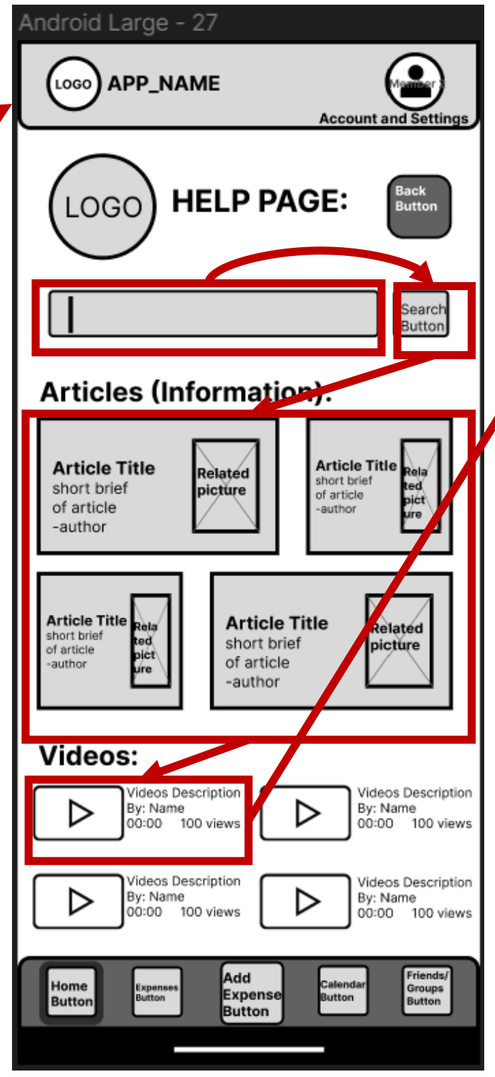
Tasks: 1. Press the “Help” floating button from any page

2. Search for the required topic and look at related articles or videos to gain information/help on that topic

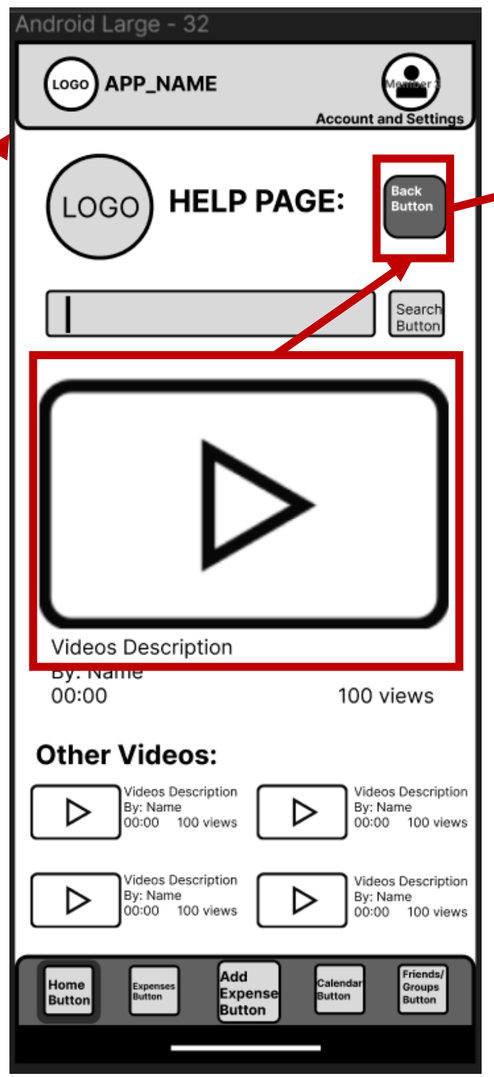
3. Once the help is done, easily go back to the page/task they were on/trying to do.



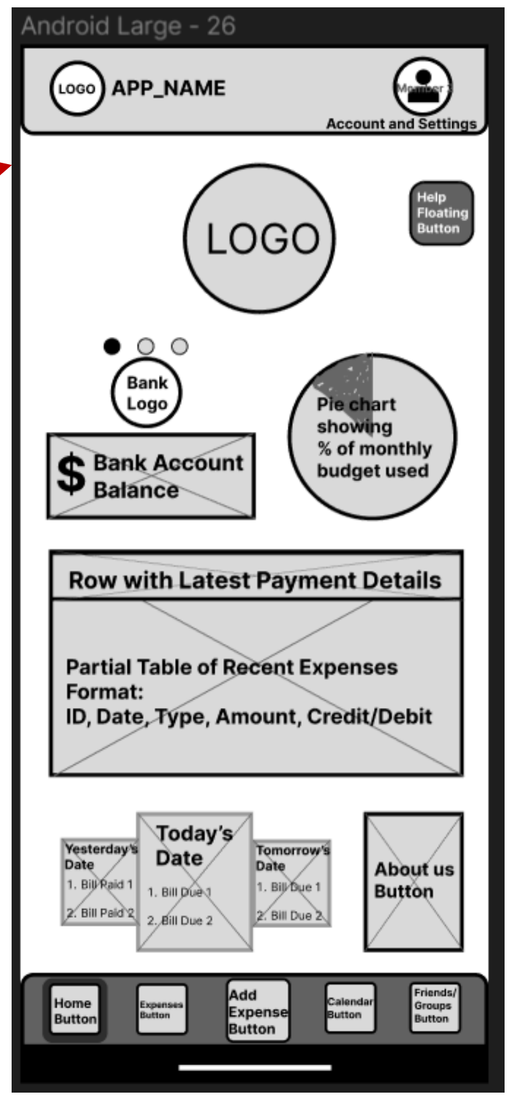
1. HOME PAGE:
Click on the Help Floating Widget to receive advice on finance or help about using the application



2. HELP PAGE:
Search for information on a topic by entering text in the textbox and pressing the search button. Receive information in terms of articles or videos. Click on a video to get interactive information.



3. HELP VIDEO PAGE:
Watch a video by a professional financial advisor on budgeting or by a technical staff member to do tasks in the app. Click on the “Back” button to resume the task on the page you were initially.



4. HOME PAGE:
Back to the Home Page or the page you were on so that you can continue the task after learning new information.

SWEN303 Assignment 2 – Risheet Peri 300594213

3. UX guidelines

FRAME: The wireframes use "Android Big" frames (360 x 800 pixels) and have the gesture "line" at the bottom of the wireframes (used in Android UI design), since the Alex Singh persona is from NZ [1] and most people in NZ (53% of market share) use an Android phone [2]. It is the "android big" version, since they would use a latest phone model with a bigger screen since they are a software engineer [1]. The wireframes are designed for a mobile application since Alex Singh enjoys outdoor activities [1] and accessing the application with a portable mobile phone makes more sense. Moreover, there is no design element close to the edges or corners of each wireframe accounting to variation in mobile screen sizes.

NAVIGATION BAR: In all the wireframes, the application has a Navigation Bar with the Home Button, Expenses Button, Add Expense Button, Calendar Button and Friends/Groups Messages button. This makes it easy to access any of the sections with just one press of a button from any other page in the application. This allows for flexible User Flows according to different Personas following the UI Rule of Thumb/Heuristic #7 Flexibility and Efficiency [3]. The user can check their calendar, messages, or expenses at any time before making the payment, especially for someone like the John Baker Persona who is extremely cautious and takes their time looking at minor details before doing a task.

All the navigation bar buttons are spaced apart evenly signifying that each one of them is a separate task suggested by the Visual Design Principle of Proximity [3] and are the same size signifying an equal importance by the Principle of Scale [3], except for the middle Add Expense Button, which is slightly larger to show more importance, and make it more convenient to press when trying to make a quick payment following UI Heuristic #7 Efficiency [3]. This is also useful for a persona like Alex Singh who has less time to manage their expenses due to their busy schedule, since the button is easily noticeable and the user flow to make an easy payment [User Flow 1: Alex Singh Persona] only has a few steps. The currently selected button/section is indicated with a dark box behind the button. The navigation bar follows the UI Heuristic #4 of Consistency and Standards [3] since it is at the bottom of the screen like other newer Android apps. This is because the user can press all these buttons with less effort using their thumbs on mobiles with larger screens. In the actual product these buttons could have Metaphors [4] as symbols like a house for the "Home Page" button. If the icons used are Metaphors, it would be useful since the user might already have previous experience with similar ideas.

Similarly, one can access their account and other settings on the top-right of any wireframe. This makes it convenient for any user to manage their budgets, bank accounts, settings, and user accounts while on any page in the application.

HOME PAGE: The application, when opened, defaults to the Home Page. The Home Page [Wireframe 1] contains a dashboard of relevant widgets such as a recent Expense Table, mini-Calendar, Bank Balance, and an Expense Pie Chart. This is useful to the Alex Singh Persona, since they can easily click on the app and view the most essential information. The Expense table is in the Golden Rectangle which is pleasing to the user since it follows the Principle of Scale [3]. In the Calendar widget on the Home Page or the Calendar Page, the rectangle with a list of bills due on 'Today's Date' is larger than the rectangles with the bills overdue/paid or due the next day, since todays' bills have the most priority to be paid immediately, justified by the Principle of Scale [3] by which larger objects tend to be more important. The bills which are paid have a 'tick' beside them which is an instantly recognisable symbol. All the widgets are in an equally distributed asymmetric layout on the Home Page following the Principle of Balance [3], however only the most valuable information is showed on the Home Page to avoid the pitfall of overcrowding in one page [5]. Pressing on each Widget would take the user to another page: Expense Table to the Expense page, Bank Balance to the account settings to manage the bank accounts, Budget Pie Chart to the account settings to manage budgets, and the Calendar widget to the Calendar page.

The Home Page also has the logo of the application in the centre integrating the application's brand identity into its aesthetic (UI Heuristic #8 Minimalist and Aesthetic Design). The logo is in the top-left

corner of every wireframe and pressing on it would lead back to the Home Page, which along with the button on the Navigation Bar are two ways to do the same task- allowing for the UI Heuristic #7 Flexibility and Efficiency [3].

CALENDER PAGE: The Calendar pages [Wireframes 2, 3] have two views- Weekly or Monthly to allow the user to have a broad view of due bills or a more focused view. There is a legend to help determine the type of bills like rent, groceries, or entertainment by their colour. There are buttons to navigate between the various years, months, and weeks.

EXPENSES PAGE: One can scroll left & right to select the group for which they want to see a recent Expenses Table and graphs. The graph takes up the upper half of the screen and the table takes up the lower half of the screen to avoid overcrowding. The selected group is larger and has a thick border (Principle of Scale) and there is a scroll bar to show how many groups there are and how far one has scrolled. There are five graphs per group and five circles below each graph with one being a different colour, indicating which graph is being viewed currently.

USER INPUT: There are some pages like the Add Payment page and Edit Payment Page that require interactive user input [Wireframe 7, 21, 22]. The user must scroll left or right to select the group (the selected group is indicated with a larger DP and an outline showing its importance using the Principle of Scale [3]), scroll left or right to select the members (the selected ones clearly indicated with a thick outline), enter the amount, and type details and optionally attach a soft copy of the receipt from their local device. Since there may be many groups and many members within the same group, it is not possible to show all their DPs and names on the screen at the same time. So, only a few are shown on the screen at the same time, and the user must swipe left or right to view more. The user can get an idea of how far they scrolled left or right (how many groups/members are there) by the horizontal scroll bar and the position of the 'circle' on the 'bar.' This gives the user a feeling of UI Heuristic #1 Visibility of System Status [3], since the user can understand how much they have scrolled and an idea of how many groups/members are there, just by looking at the position of the circle. This design element is commonly used in Android apps and so it maintains previous standards and consistency (UI Heuristic #4 [3]).

REVIEW & CONFIRMATION PAGES: When adding [User Flow 1] or editing a payment [User Flow 2], there is a review page for the user to ensure that the details they entered are correct [Wireframes 8, 23]. If they are correct, they can continue the User Flow by pressing the "Confirm" Button; if they must make a change they can go back to the previous screens [Wireframes 7, 22 respectively] by pressing the "Cancel" Button (or the "Back" Button just to go back in Wireframes 12, 20). This adheres to the UI Heuristics #3 User Control & Freedom, and #5 Error Prevention [3] since the user is allowed to conveniently correct their mistakes by allowing them to review payment details before confirming these changes. This is especially important since these are financial payments, and an incorrect payment can be difficult to rectify. Moreover, the "Confirm" Buttons on the Make Payment Page [Wireframe 7] and its consequent Make Payment Review Page (pop-up) [Wireframe 8] (and similarly on the Edit Payment Page [Wireframe 22] and the Edit Payment Review Page [Wireframe 23]), are on slightly distinct positions on the screen. This is to avoid the user accidently pressing the "Confirm" button twice in a row without carefully reviewing the payment details like an unconscious slip [3].

Moreover, the user gets a confirmation message after the payment is added or edited [Wireframes 9, 24 respectively] with the updated Recent Expenses table. This helps the user feel in control of the system (UI Heuristic #1 Visibility of System Status) and ensures that there is no doubt regarding whether the payment was actually added or edited or not. If there was no confirmation message, the user may be confused or stressed about a payment. They may try to add the payment again, which would lead to duplicate payment additions and unnecessary loss of money.

MESSAGES PAGE: In the Messages Page [Wireframe 13], the user scrolls left and right, with help from the scroll bar to choose the correct group and then scrolls up and down with the help of another scroll bar to choose the correct member to chat with. One can also add new groups or new members to a group. These buttons are beside the list of Groups and list of Members respectively following the Principle of

Proximity [3]. The first 'chat' is always the central group chat, which all members are a part of and can send messages in. The remaining ones are in the order of newest message first like many other applications like WhatsApp or Instagram. An individual chat [Wireframes 14, 15] contains messages and the date and time they were sent. A group chat [Wireframe 27] shows the messages, time they were sent along with the DP of the member who sent the message. One can attach a payment detail with each message to refer to an earlier payment using the Attach Button. The whole application uses Display Pictures (DPs) for groups and members along with their names, to make it more accessible to those with difficulty reading and to make the application less bland and more colourful.

HELP FLOATING BUTTON: There is a floating "Help" button which the user can move around by holding on it and dragging it. It is also accessible on all wireframes so that the user can conveniently get help whenever possible, adhering to UI Heuristic #10 Help & Documentation [3]. The user searches for the appropriate information by typing in some text and pressing on the Search button. The results are the relevant articles or videos. They can then read these articles or watch these videos [Wireframe 32] on the relevant topic [User Flow: Help], which are posted by financial experts or the technical maintenance staff, depending on whether the help is for financial advice or how to do a task on the app properly. The user can go back to the task they were working on by pressing the "Back" button. The videos and articles on the Help Page follow the Information Architecture principle of disclosure [5], since the user can have an idea of what to expect from the information by looking at the video's thumbnails, duration, views, presenter or looking at the article briefs, cover image or author. This allows the user to judge whether the video/article is useful or not without wasting their time actually going through the article/video. The layout is a balance of text and images, which is pleasing to the user.

TENANTS VS PROPERTY OWNERS: I designed my app to be used by both tenants and property owners. The tenant application [Wireframe 1] and the property owner application [Wireframe 25] have a consistent layout. However, the main difference is the background colour; this is to visually differentiate between which version of the app someone is using, especially if they are both a tenant and property owner of different properties. They could download just one app and switch their accounts appropriately using the top right Account and Settings Button (the common place for this button to be located, and so using UI Heuristic #4 Consistency & Standards [3]).

INFORMATION ARCHITECTURE: The IA for my application has a matrix organisation [5] since users can navigate to many different pages from a single page and change their user flow easily. This makes the application very flexible and quick to navigate to different pieces of information. This is due to the Navigation Bar being always accessible and adhering to the IA Principle of Choice [5], giving the user meaningful choices.

USER FLOW 1: There are multiple ways to add a payment, either the quick way like Alex Singh or a more informative way like John Baker. However, once the Add Payment Button is presses, it is a more streamlined linear process which adds to its speed and efficiency.

USER FLOW 2: This user flow is a combination of two smaller user flows, each one from a different account. This is indicated with "Member 1", or "Member 2" mentioned on the Account & Settings Button in the Top-Right of the wireframes. This user flow is very dependent on the notification on the Friends/Groups Messages Button and the user noticing it. However, notifications of a similar design (circle with a number on the top-left of a button) are quite common and follow the UI Heuristic #4 of Consistency and Standards.

USER FLOW 3: This user flow is from the perspective of the property owner and is a small user flow to save time, since an owner like Trevor Smith can own many flats and may have to message many different groups everyday to ensure their bills are being paid.

USER FLOW (HELP): This user flow technically has many start and end points, since the Help Floating Button is available on any other page in the app. It is also a short user flow signifying how easy it is to get help when using this application. This ease is something that the technical staff and financial advisors

would want, since they could increase the application's user base a lot and get users to do more on the application.

WIREFRAME WEAKNESSES: Some wireframes like Wireframe 21 may be a bit overcrowded [5], however this can be fixed by letting the user scroll down to enter all the details. This way more space can be used for the form, and more input questions can be added. Another disadvantage is that these wireframes are not interactive i.e., they do not transition from one to another upon pressing a specific button. This would enhance the understanding of the user flows.

WIREFRAMING PROCESS: I used an iterative process [4] to produce the wireframes, from low-fi to a high-fi model. For instance, my final Home Page [Wireframe 1] improved upon my initial wireframe for it [6].

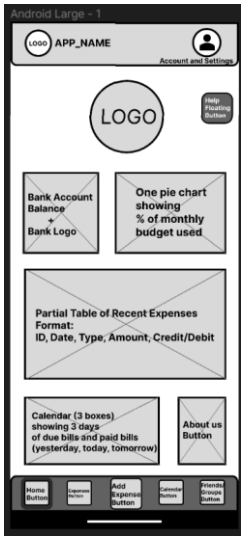
References

[1] Dr J. Ferreira. SWEN303 Assignment 1 Part 2. https://ecs.wgtn.ac.nz/Courses/SWEN303_2024T1/Assignment1#ParT2 (accessed April. 20, 2024).

[2] C. Hughes. "Yearly market share of mobile operating systems in New Zealand 2012-2023" statista.com. https://www.statista.com/statistics/1322277/new-zealand-mobile-operating-systems-market-share/#:~:text=In%202023%2C%20Android%20was%20the,over%20the%20past%20few%20years. (accessed April. 20, 2024).

[3] Dr J. Ferreira. SWEN 303. Class Lecture, Topic: "User Experience Guidelines", School of Engineering and Computer Sciences, Victoria University of Wellington – Te Herenga Waka, Wellington, NZ, Mar. 28 2024, Available: https://nuku.wgtn.ac.nz/courses/17706/files/2576728?wrap=1 [Accessed: April. 21 2024].

[4] Dr J. Ferreira. SWEN 303. Class Lecture, Topic: "Prototyping", School of Engineering and Computer Sciences, Victoria University of Wellington – Te Herenga Waka, Wellington, NZ, Mar. 21 2024, Available: https://nuku.wgtn.ac.nz/courses/17706/files/2547031?wrap=1 [Accessed: April. 21 2024].

[5] Dr J. Ferreira. SWEN 303. Class Lecture, Topic: "Information Architecture", School of Engineering and Computer Sciences, Victoria University of Wellington – Te Herenga Waka, Wellington, NZ, Mar. 26 2024, Available: https://nuku.wgtn.ac.nz/courses/17706/files/2565038?wrap=1 [Accessed: April. 21 2024].

[6]