Project Report

Project Name: Expensify

Demo:

https://youtu.be/8O3v5gYvuHg

Introduction

Any employee in their professional career goes on a business trip whether it is meant for meeting a client, for training purposes or to attend a conference. For such business trips, almost all the companies provide their employees 100% compensation for food, airfare, hotel etc. Keeping track of all the expenses during the trip could be a hectic job for so many people. For each expense, you must save all the receipts so you can claim your money at the end of the trip which could be a lot of work. Expensify, an expense management app, will take care of everything and provide the solution to save all the expenses at one place. Expensify will have two types of users, a regular employee, and a manager/supervisor. Any employee, whether a regular one or a supervisor, can go on a business trip so both types of users can file their expenses using the app. Once a user files their expenses, the report directly goes to the employee's supervisor, whoever he/she is. Then using the in-app notifications, the supervisor will be notified about the new expense report which he can check on the app. Once a supervisor sees the notification, he can click on it and can view the expense. After reviewing the expense report, a supervisor can approve or reject the expense. Once the decision is made the user will be notified through in-app notification. Following suit once the report is there, a user can see the status if it was approved or not.

How I Got This Idea?

At the start of the semester when I informed told that there is going to be a final project. I talked all of my acquaintances about the app ideas because I wanted to build something which has practical implications. The idea for the Expensify app came from a problem my manger was having with managing per diem and travel expenses with some of them. My manager used and app called Concur which is basically an online service to keep track of the work-related expenses, but in order to get an account with Concur, you have to pay thousands of dollars every year. So, I reached out to my old managed and they gave me the idea to make an app like Concur which can at least let the employees file their expenses systematically. However, Concur has a lot more features and is very professional in a way that's why they charge a lot of money. Expensify on the other hand, is also keeping professional and all the business rules in mind, but tries to at least provide the basic functionality of filing expenses at an affordable rate. After talking to my manager, I discussed this idea he thought it could prove to be quite the handy tool for managing travel expenses. Then I went to work on this application and the biggest advantage was that I had was a live client who could test our application as I move forward with the development.

The Process

As soon as I finalized my thoughts on the app idea, I started to plan how I am going to develop this app. I had a couple meetings initially about with the professor on how I am going to approach this project. In the first two meetings, I planned out the entire project and how I are going to work in different phases. I also

discussed and decided how much time I needed for each phase, and I would say that was one of the toughest tasks I had to do because all of us had no prior experience about developing an app in a group. I started off with the requirements gathering phase. I interviewed one of our head TA who is a grad student and doing research in HCI, I asked him what functionalities he wants to see in the app and how he wants the app to be. I also talked to another professor who recently went on a business trip used Concur to file their expenses. I wanted to know what issues he went through while using concur and how I could implement Expensify to cater their needs. After I had a zoom meeting with our head TA and my other professor, I had all the requirements I needed to build this application. After that, I set up milestones and goals for each task and assign the number of days to each milestone. This gave me a better understanding of how much time I all need to spend on each task. While I was working on the tasks, I had daily and weekly meetings to discuss the progress of the project with my professor for the course. I did not have meetings every day, but I mostly met when the need arose to discuss an idea or something. After the completion of each bigger milestone, I talked to the professor and told him the progress of the project to get feedback from him. The process worked me, as I had to go back and forth to make changes in the app each time after I met with the professor, but it worked in the end.

Technologies I Used

Choosing the appropriate technology was not an easy task for us. I had to research what technology would suit our needs and prove to be most efficient. I had to spend a lot of time researching and consulting different people about this. For the implementation part, obviously, I used the Swift language. I used SwiftUI to create and design the interface and at first utilized a local database when there were no users but quickly discovered that this wasn't a good approach. This was the reason I

decided to use firebase for the database and core data for logged in users only. I also provided ideas to each other as to what the front end could potentially look like to ensure the most user-friendly experience possible. Next, I set up the core database and its entities and relationships, ensuring that it would allow for ease into the implementation of our code. I utilized Firebase, Firestore for back-end database and user authentication, mostly APIs, and a URL for uploading receipts. In regards to the receipt view I chose to present it that way so that the user can only see their expenses and no one else but the manager can see everyone's expenses and approve or reject it based on what their budget holds. This change in scope was crucial for our code to work in the best possible way.

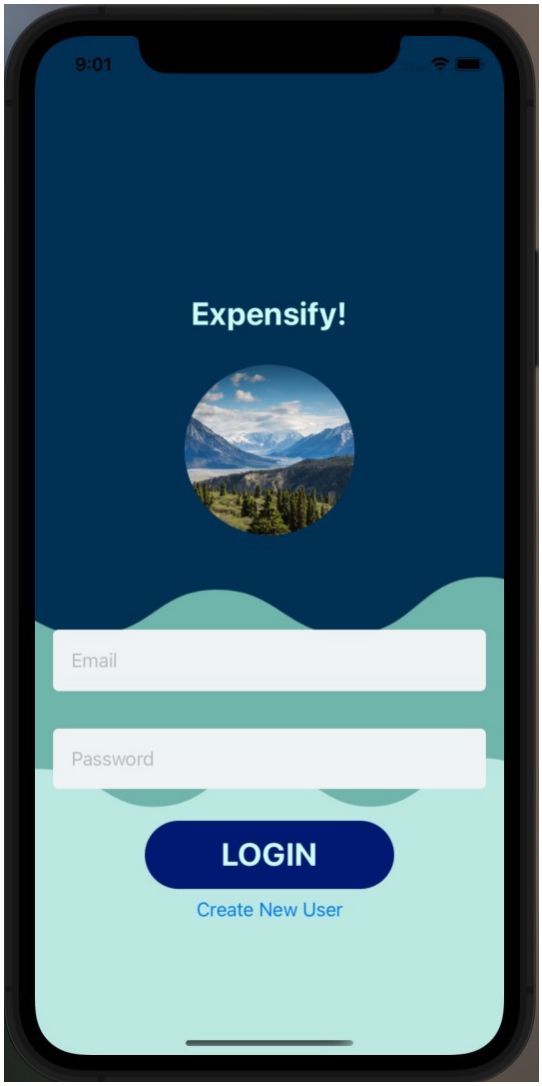
Technical Issues

First off, I had to understand the utilization of Gitlab and how to concurrently work on a project at the same time. Secondly, I had to come up with an interface and design which could be user-friendly and helpful to the users. UI Design is the most important aspect of the application because the UI is a bridge between a user and the system. If the UI is not easy to use, it was the issue of design and needed to figure out what design I wanted to allow for user friendliness and ease of use. Next, I ran into an issue of data connectivity which is what Syed expanded on. Finally, the last issue was more related to what APIs to use and where to look things up to gain an understanding of the problem and solution.

Features/Screens

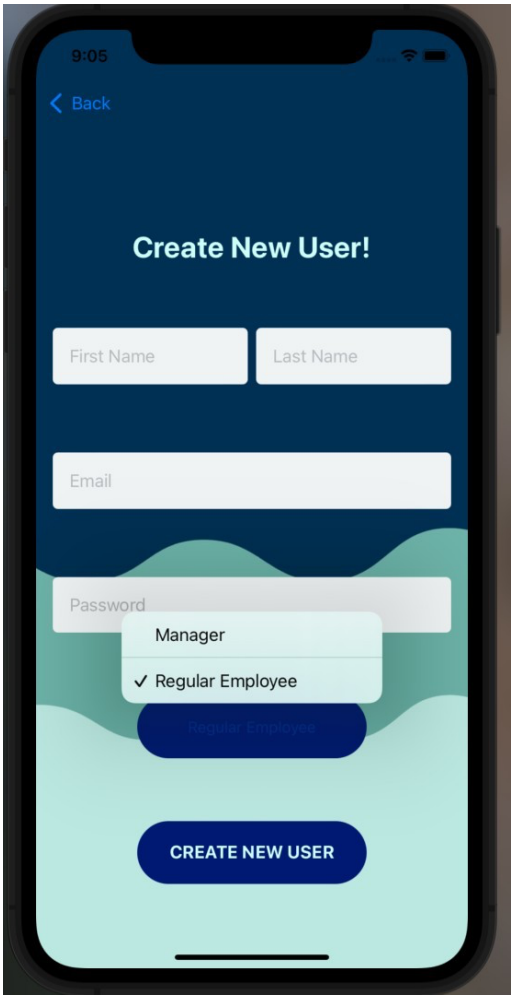
Log in:

Expensify has a login screen which is the first point of interaction between the user and the app. In order to access the app, a user has to log in the app. The app can save data for each user so for that purpose, a log in was required so the data can be saved for each user specifically. The log-in requires a username and the password in order to access the features of the app. The username is basically an email address of the user which can support any email server. So, the system is designed in a way that the username field only accepts the strings which are in the format of an email address. The input validation was put in place to make the app secure so no one can input invalid data to crash the system. The password field also has a specific format, and the user can only input that format for the password. For the password, the user can input at least 6 characters. The password cannot be less than 6 characters. This was also put in place as a safety feature to make the application secure.



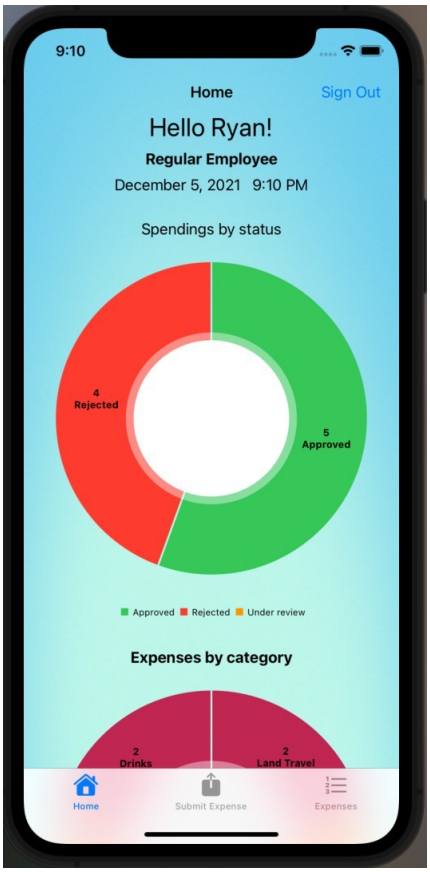
Create New User:

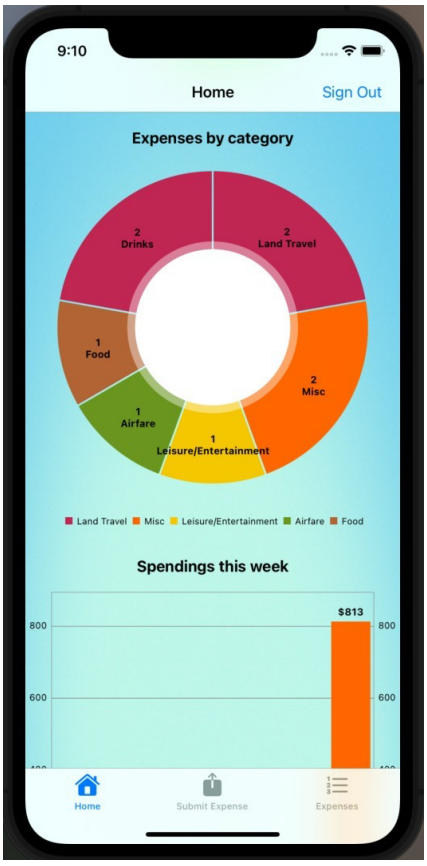
The system also has the ability to create new users. Expensify gives the capability to the user to add himself as a new user on the app. If a user does not have an account, he can create an account and log in to the app in order to take advantage of the app's functionality. While creating the account, a user can create an account as a regular employee or a manager/supervisor. There are two types of users the app currently has, superuser which is a manager or a supervisor and a regular employee. A regular employee has limitations in terms of using the app features whereas, a manager has access to all the features of the app. A manager has the admin level rights. While creating a new user, the user has to follow the guidelines in terms of username and password. The username must be in the form of an email address and a password should have at least 6 characters in it. Once a new user is created, the user can then log in to the app and use the features.

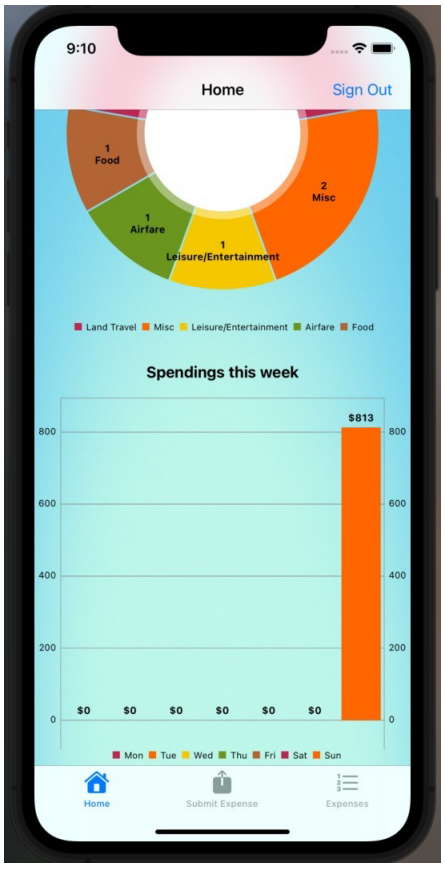


Home - Regular Employee:

There are a bunch of functionalities shown on the home screen of regular employees. the first thing you see on the home screen is a pie chart which shows the number of approved expenses and number of rejected expenses. When you click on any of the parts of the chart, it will display the list of expenses on the screen. The next thing you see on the screen is the expenses by category. This pie chart lists all the expenses by category. As you can see in the attached screenshot, there are some expenses spent on drinks, some on food and so on and so forth. Another thing is that it shows is the spending, as you will see in the third screenshot, the bar chart shows the spending as such. The long orange bar shows $813 spent this week.

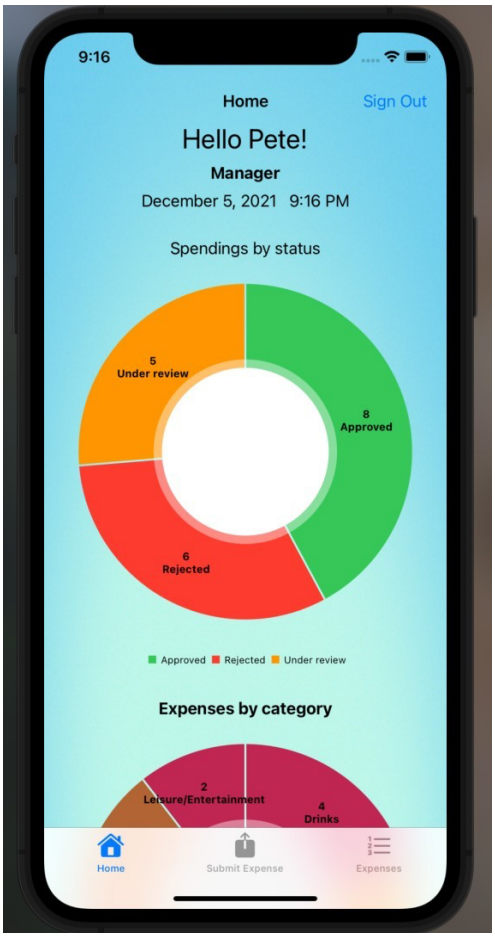


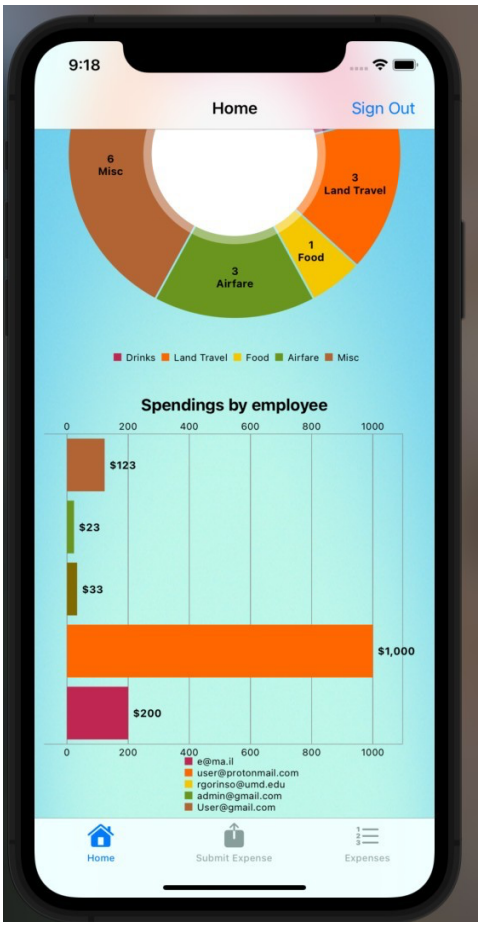




Home - Manager:

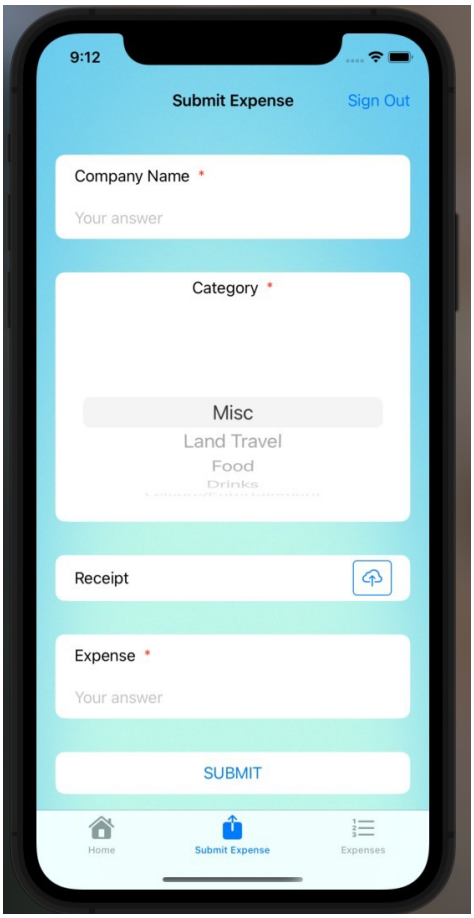
When comparing the functionalities of the home screen for the manager to that of the regular employee, not much is different as the main changes occurred with the user rank, the amount of information outputted in the charts, and finally the last chart. The spending by employee bar chart allow for managers to view all of their employees by email and see who is spending the most of the managers allocated travel budgets. The other charts have the information of all regular employee spending habits but are fairly similar to the regular employee home page.





Submit Expenses:

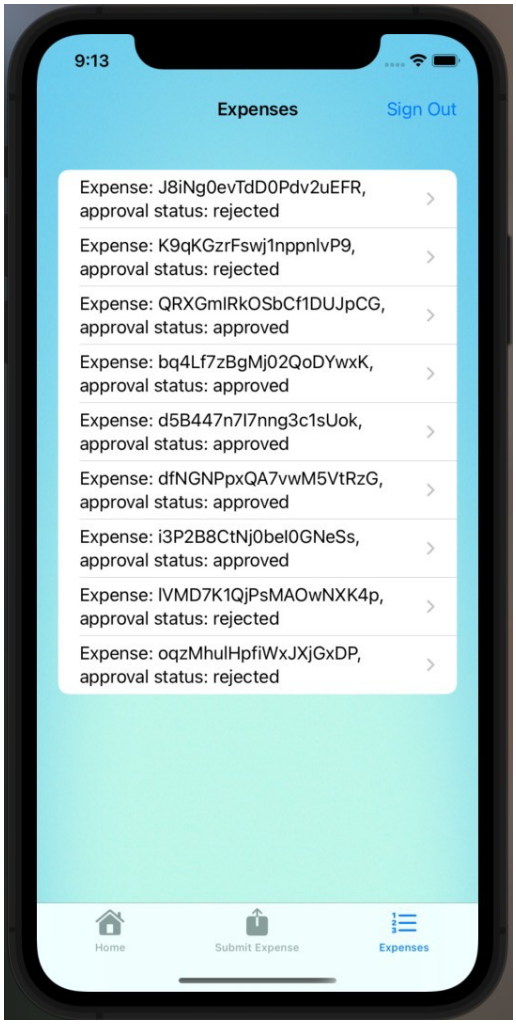
This view is essentially the backbone of the project as it is the view that allows for the employee/manager (if necessary) to submit their personal expenses to potentially get reimbursed. Each of the fields is checked for correct input and to ensure they are not left blank. Once the user clicks submit and the fields are validated, the information is added to the user's expenses to be outputted on the home tab as Ill as the expense tab. Without this tab, there will be no way for the employee to be reimbursed for their expense as there will be no proof of expense.

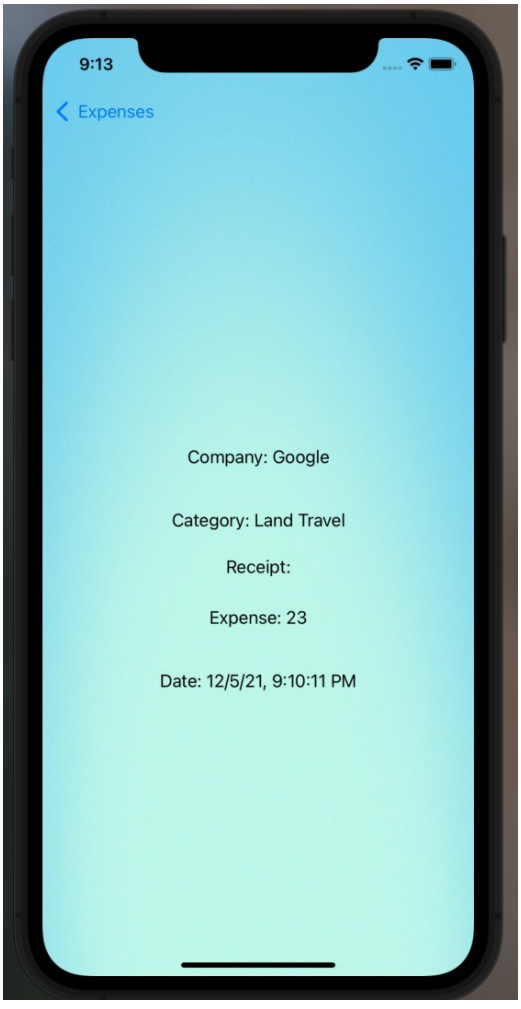


List of Expenses - Regular Employee:

This screen displays the list of expenses filed by a regular employee. As you can see, there are a number of expenses and under the expense name, it displays the status of the expense whether it was approved or rejected. This list will display in great detail the expenses filed during an employee's career.

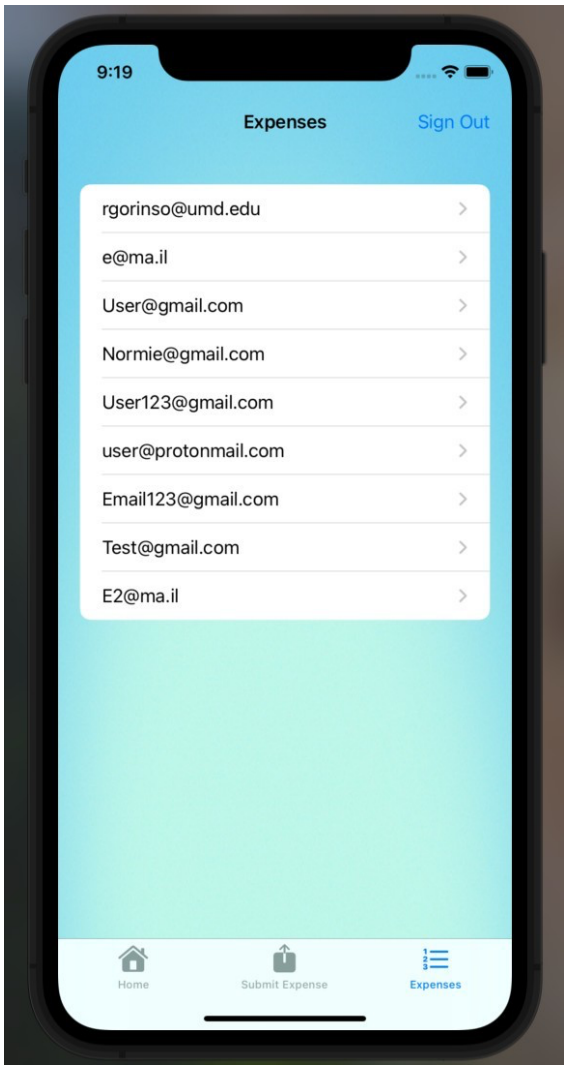
When a user clicks on any expense, it will display the entire summary of the expense and a user can see what he filed. If a user wants to see what he filed years ago, he should be able to see it in this list.

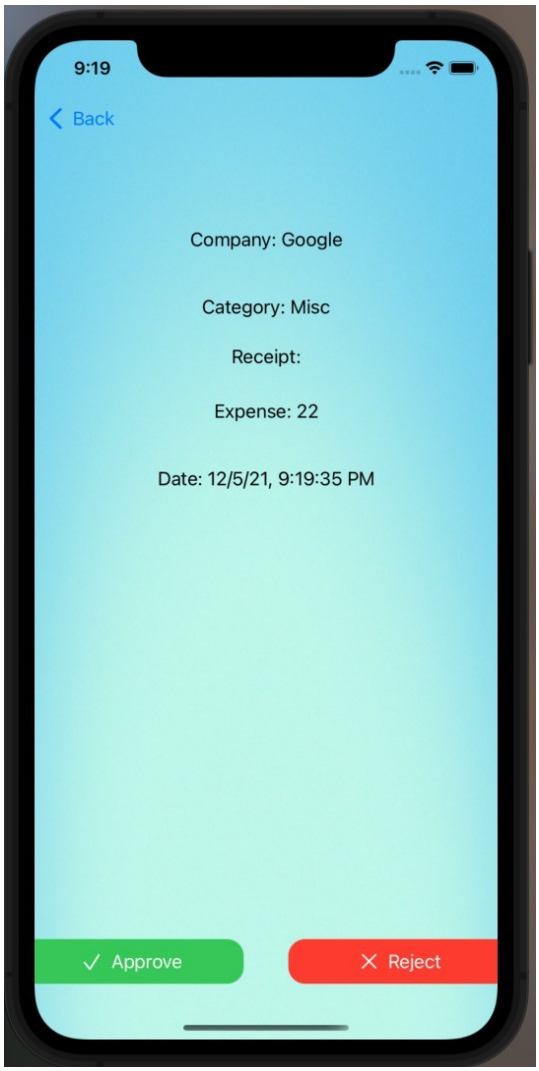




Expense - Manager:

This screen shows the list of expenses filed by different employees. This list is different from a regular employee's screen. This list shows the expenses filed by different employees. All the expenses are going to show in here. When a user clicks on any email, it will display the expense. When the expense opens up, there are two buttons shown at the bottom, accept or reject. The manager can review the expense and make the decision based on their judgement. After taking the decision, the expense will be shown under the respective user's list as rejected or accepted.





Future Goals

Following the completion of the minimal goals, our hope is to be able to accomplish some of the following goals below. I'm looking to dive into the more advanced application topics such as utilizing keywords in a name to find the best match for the spending category. I am hoping to use picture recognition to allow for the user to take a picture of the receipt and have all of the information pulled and directly inserted into the required text boxes. Finally, I hope to use bank APIs to have the expenses directly pulled following transactions and inserted into the fields, but the biggest issue with this may be the security risks posed from this.

- Use keywords in business name to derive whether or not it is a meal, travel, leisure, etc.
- Set up the capability to take pictures of receipts and derive the information directly from said receipts
- Attempt to utilize bank APIs and fulfill required security requirements

Conclusion

At the end, I will say it was fun working on this project and I learnt a lot of good things about working in a team. These lessons are definitely going to help us in our future jobs. I hope Expensify will also benefit different employees and they can take advantage of it while they are on any business trip.