CS683 Project Assignment

Boston Intelligent Transportation Application

YingChen Liu

1. Overview

I am currently trying to develop an application for the Boston Transportation System. The specific reasons why I wanted to develop this software are:

1. Not all places can buy transportation card immediately
2. Cards are easily lost or forgotten.
3. You need to go to a subway station or a specific supermarket to reload.

Once I develop the application, passengers will be able to use the e-card. They can recharge their cards on their cell phones and don't have to worry about forgetting their cards or losing them. Passengers will also be able to check their recharge history, and their ride history at any time.

This software will benefit all those who visit or live in the city and use public transportation.

2. Related Work

My project was inspired by the Presto Card and Alipay's travel features.

Presto Card:

The Presto Card is a smart transit card that is widely used in Ontario, Canada. It is designed to provide commuters with a more convenient and flexible way to travel. Presto card features are:

1. Versatility: The Presto card can be used to pay for many different modes of transportation.

2. Auto Reload: Passengers can choose to enable the auto reload feature, which automatically reloads the card when the card balance falls below a set threshold.

3. Online management: Manage their card through the Presto website or mobile app, including reloading, viewing transaction history, setting up auto-reload options, and more.

4. Security: Presto cards have security measures in place, including PIN codes and reports of lost or stolen cards. If the card is lost or stolen, users can immediately freeze the card to prevent misuse by others.

Alipay's travel features:

Alipay's (Alipay) public transportation feature has the following features:

1.Public Transportation Card Binding: Users can bind public transportation cards (e.g. metro cards, bus cards) from various cities to their Alipay account. This allows them to use

Alipay to make subway and bus fare payments without the need for an actual physical card.

2. E-tickets and Ride Codes: Alipay offers an e-ticket service that allows users to purchase e-tickets in Alipay and show a QR code on their phone to enter a station or board a bus when needed. This eliminates the need to wait in line to purchase a ticket and improves the convenience of riding the bus.

Similarities and Differences

I'm going to blend the advantages of the Presto Card that I listed with Alipay's e-transit card. I removed the requirement for a physical card for the Presto Card and used the Alipay account feature instead.

3. Requirement Analysis and Testing

Title	Register Account*(Essential)*
Description	Users can register for an account to use the app based on their information
Mockups	1.Information entry boxes (first name, last name, e-mail, password, confirmation password) 2.Confirmation box for agreeing to the terms of use 3.Register Button
Acceptance tests	1.The information entered by the user meets the requirements: 1. First name and last name are letters of length greater than or equal to 2 2. Valid Email address 3. Password that meets the requirements 4. Password entered twice is the same 2.Email not registered After completing the first two requirements, the user can successfully register an account and log in to the Application.
Test Results	Currently successfully passed
Status	Iteration1: Completed UI and main function, have few bug Iteration2: Completed Iteration3: Completed

Title	Login Account *(Essential)*
Description	The user logs in to the main application by entering the registered account and password.
Mockups	1.Receiver of information about account numbers and passwords 2.Login Button
Acceptance tests	1.Log in to your registered account 2.Unregistered and incorrect password will be detected as an exception.
Test Results	Meets the expectation that the user can log in successfully, and provides the correct information to the user in the case of a failed login.
Status	Iteration1: Implement UI and main function, have few bug Iteration2: Completed Iteration3: Completed

Title	Paying the fare Function*(Essential)*
Description	Display a QR code for payment, or use a third-party or mobile phone's native Tap method.
Mockups	1.A QR code 2.A Tap Button
Acceptance tests	1.Clicking the Tap button pays the fare and records the payment. 2.If the user's account balance is insufficient and without a Pass, a Payment Failure Screen will be displayed.
Test Results	Pass
Status	Iteration1: Not start Iteration2: Completed Iteration3: Completed

Title	Transportation Card Function*(Essential)*
Description	Show information about the transit card that corresponds to the account: Account Number, Balance, Pass status
Mockups	1.Public Transportation Card 2.Account Number 3.Account Balance 4.Account Day Pass Expiration Date
Acceptance tests	Display transportation card information, update Balance or Day Pass date after reloading or spending.
Test Results	Pass
Status	Iteration1: Not start Iteration2: Completed Iteration3: Completed

Title	Recharging Function*(Essential)*
Description	Users can choose to recharge for a fixed amount or purchase the Day Pass
Mockups	Conversion bar for Money Button Group and Pass Button Group Recharging money Button Group(5$, 10$, 20$, 50$, 100$, 200$) Day Pass Button Group (1-Day Pass, 7-Days Pass, 31-Days Pass)
Acceptance tests	Click on the button and jump to the payment screen
Test Results	Pass
Status	Iteration1: Not start Iteration2: Completed Iteration3: Completed

Title	Payment Function*(Essential)*
Description	Enter the customer's bank card information and pay the fee.
Mockups	1.Total amount to be paid 2.Input or selection fields for card number, cardholder name, card expiration date, card CVV 3.Payment Button
Acceptance tests	1.Bank card expiration date must be later in the month 2.The rest of the information must conform to the formatting requirements If the first two conditions are met, the payment will be considered successful, and the screen will jump to the Payment Success Screen, record the recharge information, and update the account information.
Test Results	Pass
Status	Iteration1: Not start Iteration2: General Completed Iteration3: Completed

Title	Account Function*(Essential)*
Description	Showing personal information and some features: My Profile, Ride history, Charge history, Logout
Mockups	1.User's image, name, e-mail. 2.My Profile Function 3.Ride History Function 4.Recharge History Function 5.Sign Out Button
Acceptance tests	Jump to the correct Screen and log out of the customer account.
Test Results	Pass
Status	Iteration1: Not start

	Iteration2: General Completed Iteration3: Completed

Title	Profile Function*(Essential)*
Description	Change Name or Change Password
Mockups	1.Change Name Function 2.Change Password Function
Acceptance tests	Successfully changed the first and last name and password of the account
Test Results	Pass
Status	Iteration1: Not start Iteration2: Completed Iteration3: Completed

Title	History Function*(Essential)*
Description	Displaying ride records and recharge records
Mockups	1.Ride Records Function 2.Recharge Records Function
Acceptance tests	All records can be displayed correctly 1.Ride Records Function: Display the time of the ride, the payment action, and the remaining balance. 2.Recharge Records Function: Display purchased products, time of recharge, new balance
Test Results	Pass
Status	Iteration1: Not start Iteration2: Completed Iteration3: Completed

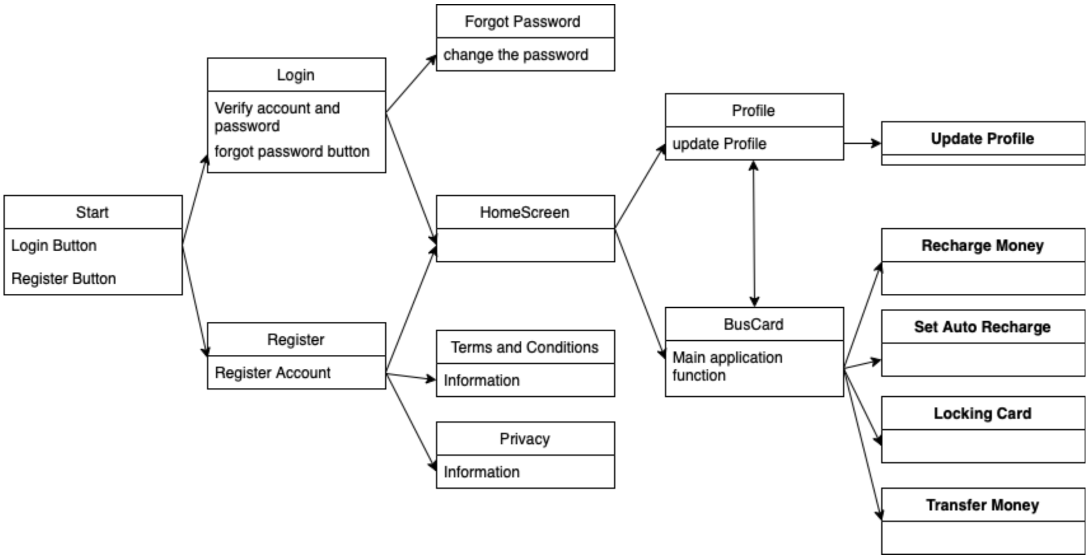
4. Design and Implementation

Overview:

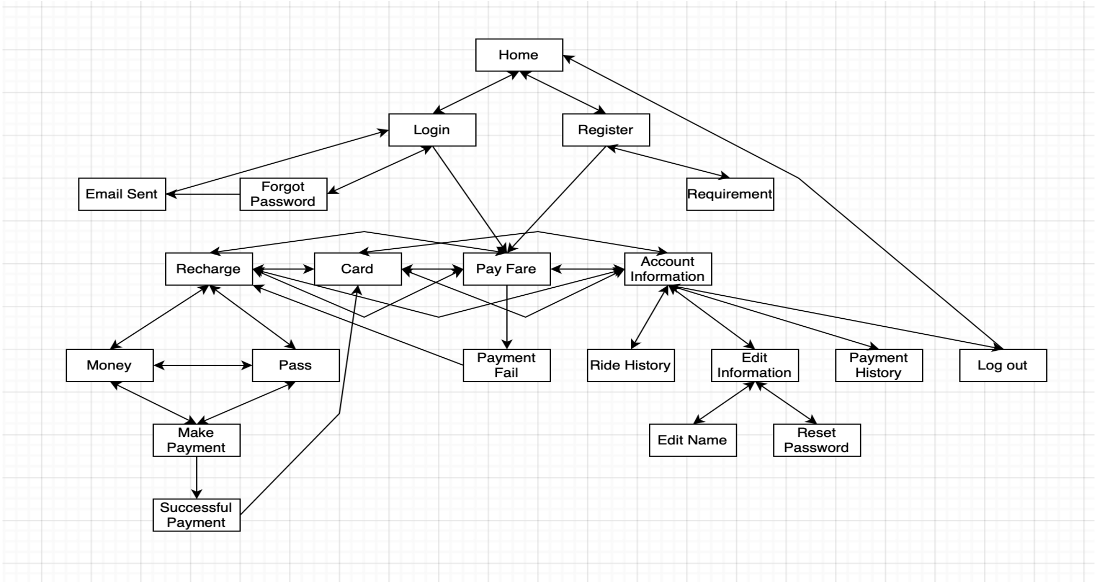
These features are Automatic Reloading, Recharging, Lost, E-card, Account Management and Personal Information. A brief description of each feature can be found in Requirement Analysis and Testing.(Iteration 0)

Basic architecture:

(Iteration 1)

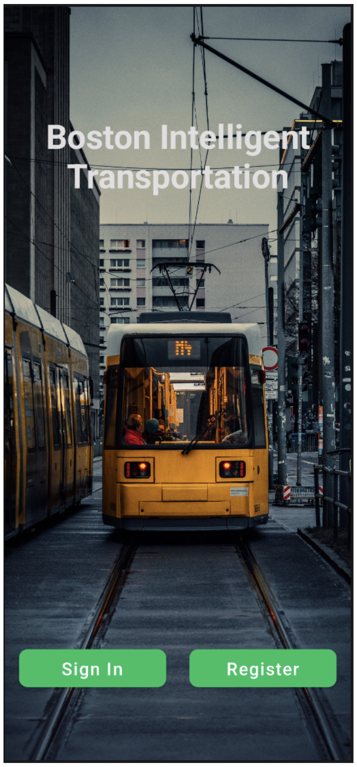


(Iteration 2)

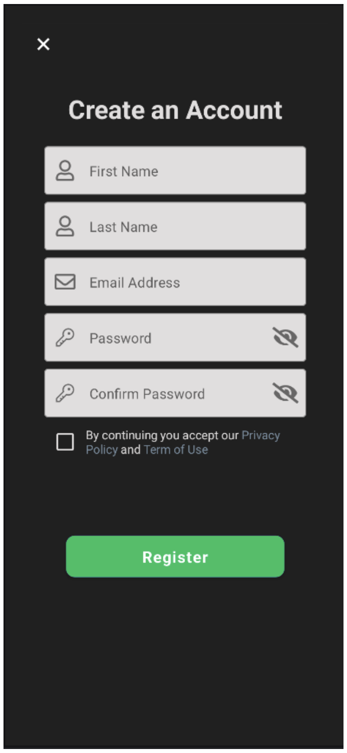


UI design and implementation:

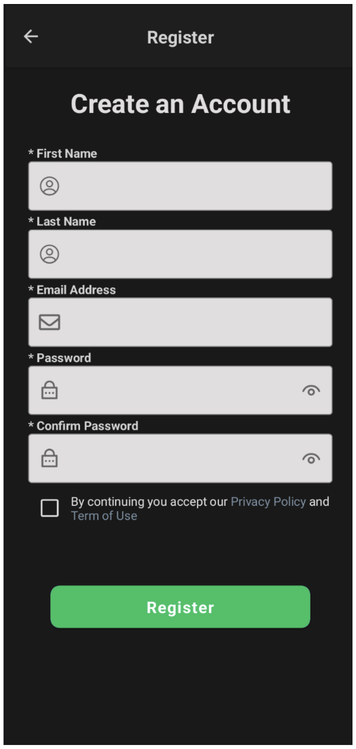
Home Screen:(Iteration 1):



Register Screen:(Iteration 1 and Iteration 2):

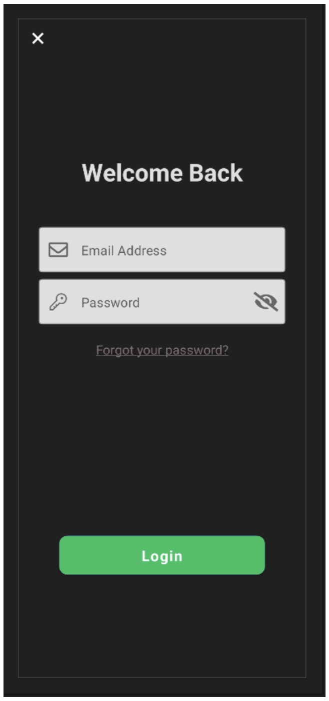


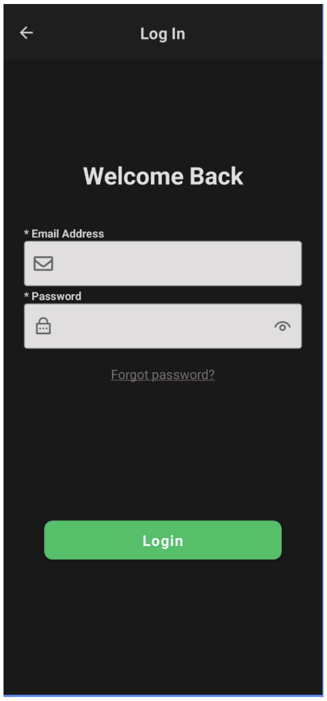
□ By continuing you accept our Privacy Policy and Term of Use



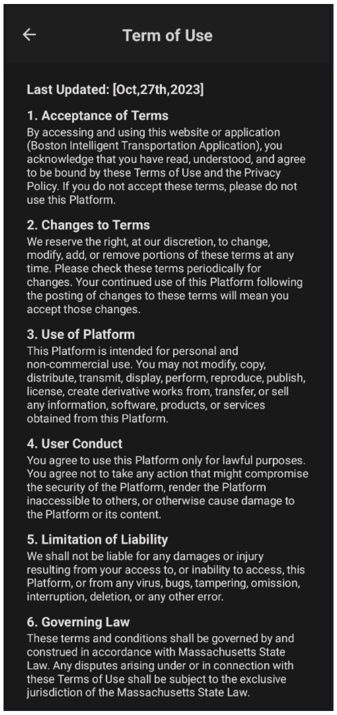
□ By continuing you accept our Privacy Policy and Term of Use

Login Screen:(Iteration 1 and Iteration 2):





Terms and Conditions Screen && Privacy Policy Screen(Iteration 1):



Term of Use

Last Updated: [Oct,27th,2023]

1. Acceptance of Terms
By accessing and using this website or application (Boston Intelligent Transportation Application), you acknowledge that you have read, understood, and agree to be bound by these Terms of Use and the Privacy Policy. If you do not accept these terms, please do not use this Platform.

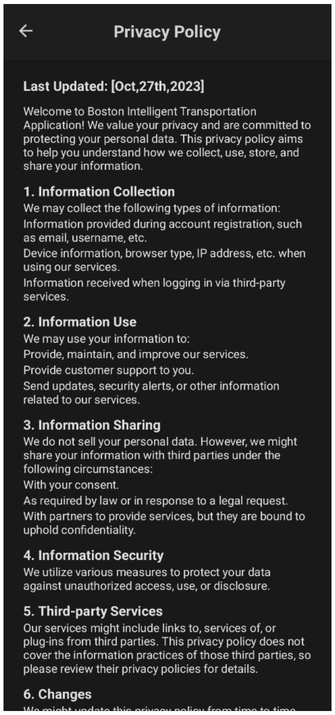
2. Changes to Terms
We reserve the right, at our discretion, to change, modify, add, or remove portions of these terms at any time. Please check these terms periodically for changes. Your continued use of this Platform following the posting of changes to these terms will mean you accept those changes.

3. Use of Platform
This Platform is intended for personal and non-commercial use. You may not modify, copy, distribute, transmit, display, perform, reproduce, publish, license, create derivative works from, transfer, or sell any information, software, products, or services obtained from this Platform.

4. User Conduct
You agree to use this Platform only for lawful purposes. You agree not to take any action that might compromise the security of the Platform, render the Platform inaccessible to others, or otherwise cause damage to the Platform or its content.

5. Limitation of Liability
We shall not be liable for any damages or injury resulting from your access to, or inability to access, this Platform, or from any virus, bugs, tampering, omission, interruption, deletion, or any other error.

6. Governing Law
These terms and conditions shall be governed by and construed in accordance with Massachusetts State Law. Any disputes arising under or in connection with these Terms of Use shall be subject to the exclusive jurisdiction of the Massachusetts State Law.



Privacy Policy

Last Updated: [Oct,27th,2023]

Welcome to Boston Intelligent Transportation Application! We value your privacy and are committed to protecting your personal data. This privacy policy aims to help you understand how we collect, use, store, and share your information.

1. Information Collection
We may collect the following types of information:
Information provided during account registration, such as email, username, etc.
Device information, browser type, IP address, etc. when using our services.
Information received when logging in via third-party services.

2. Information Use
We may use your information to:
Provide, maintain, and improve our services.
Provide customer support to you.
Send updates, security alerts, or other information related to our services.

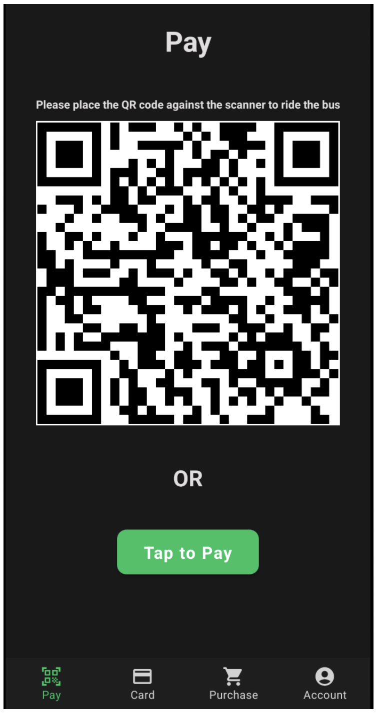
3. Information Sharing
We do not sell your personal data. However, we might share your information with third parties under the following circumstances:
With your consent.
As required by law or in response to a legal request.
With partners to provide services, but they are bound to uphold confidentiality.

4. Information Security
We utilize various measures to protect your data against unauthorized access, use, or disclosure.

5. Third-party Services
Our services might include links to, services of, or plug-ins from third parties. This privacy policy does not cover the information practices of those third parties, so please review their privacy policies for details.

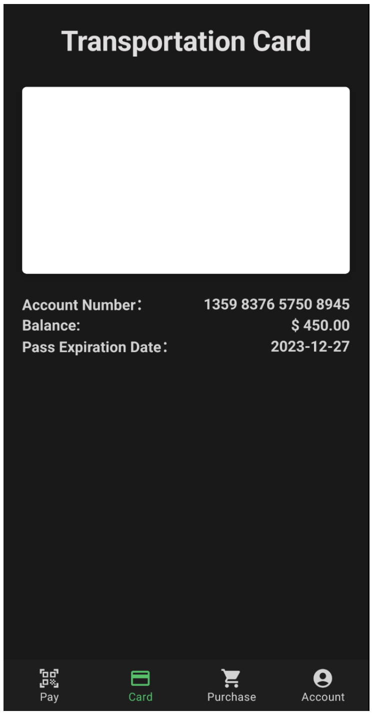
6. Changes
We might update this privacy policy from time to time

Pay the fare Screen(Iteration 2):

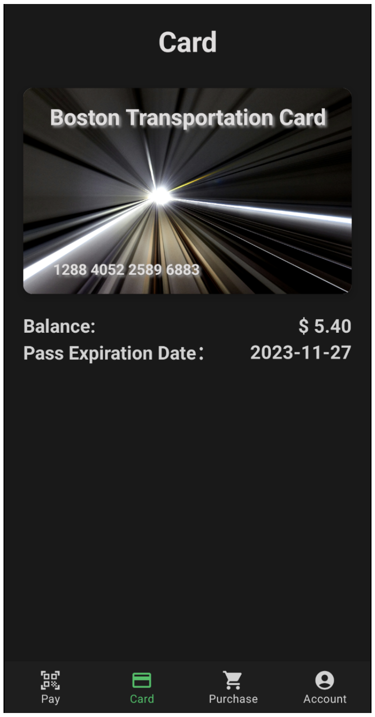


Please place the QR code against the scanner to ride the bus

Transportation Card Screen(Iteration 2 and Iteration 3):

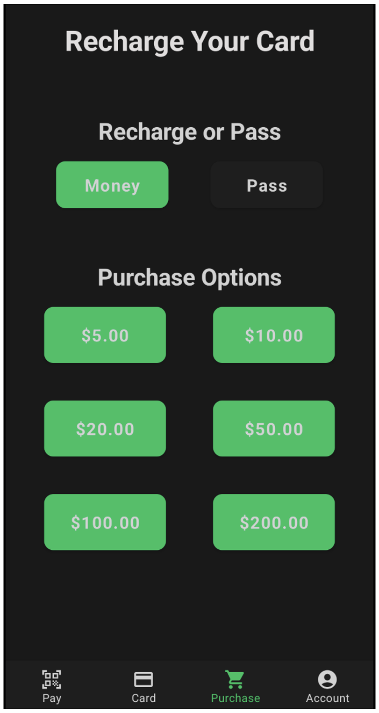


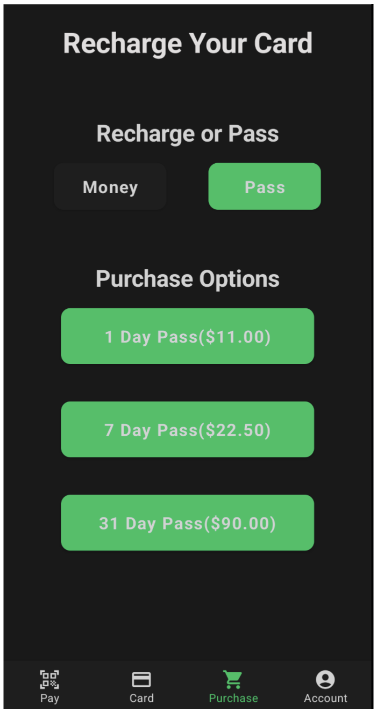
Account Number： 1359 8376 5750 8945
Balance: $ 450.00
Pass Expiration Date： 2023-12-27



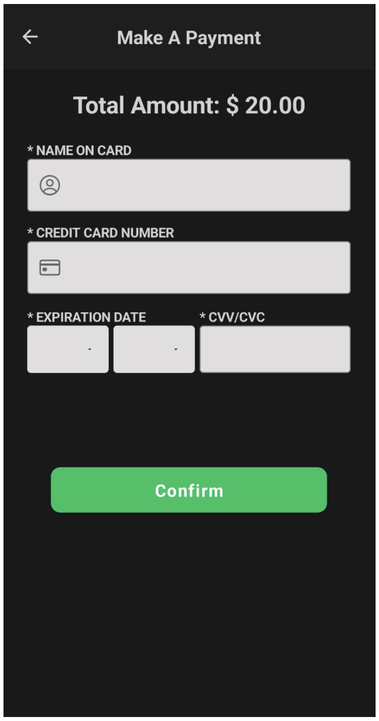
Balance: $ 5.40
Pass Expiration Date： 2023-11-27

Purchase Screen (Iteration 2):

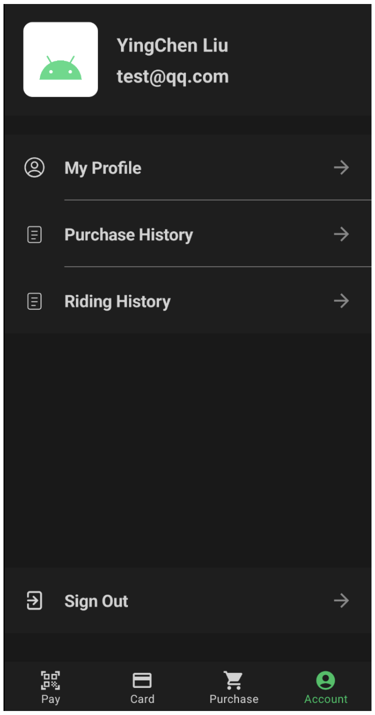




Payment Screen (Iteration 2):

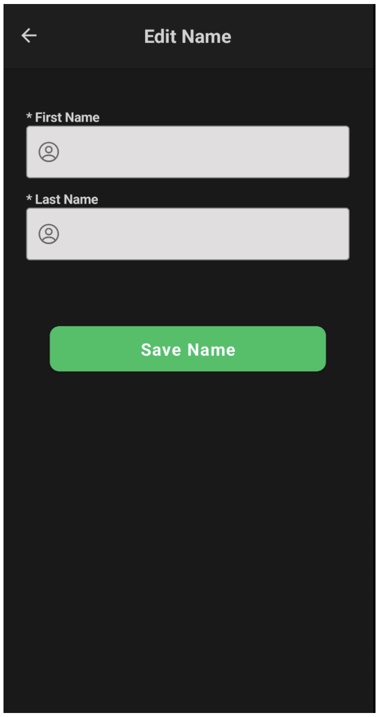


Account Screen (Iteration 2):

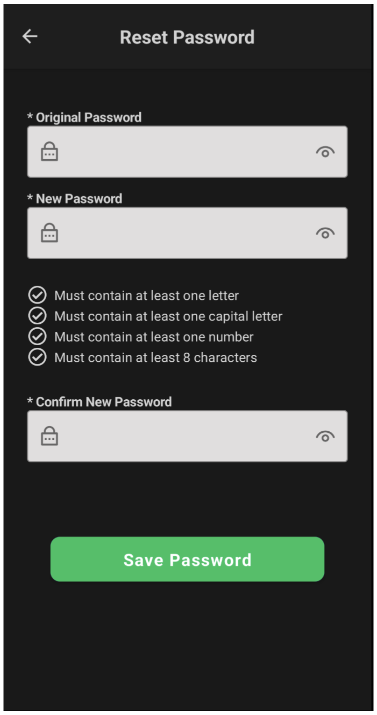


YingChen Liu
test@qq.com

Edit Name Screen (Iteration 2):



Reset Password Screen (Iteration 2):



Reset Password

* Original Password

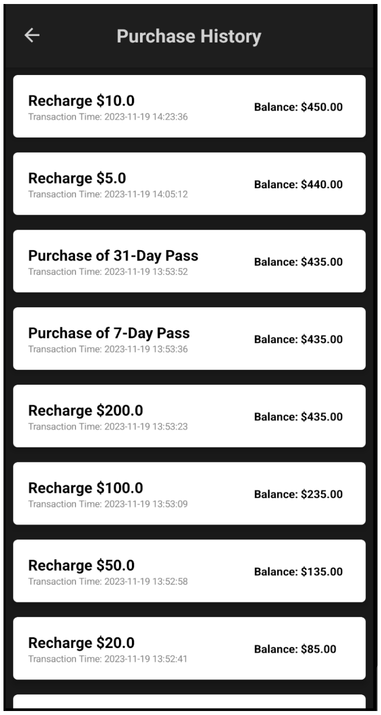
* New Password

✓ Must contain at least one letter
✓ Must contain at least one capital letter
✓ Must contain at least one number
✓ Must contain at least 8 characters

* Confirm New Password

Save Password

Purchase History and Ride History Screen (Iteration 2):



Purchase History

Recharge $10.0
Transaction Time: 2023-11-19 14:23:36
Balance: $450.00

Recharge $5.0
Transaction Time: 2023-11-19 14:05:12
Balance: $440.00

Purchase of 31-Day Pass
Transaction Time: 2023-11-19 13:53:52
Balance: $435.00

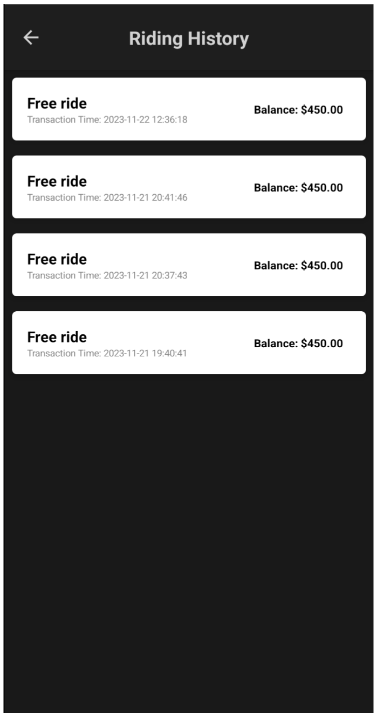
Purchase of 7-Day Pass
Transaction Time: 2023-11-19 13:53:36
Balance: $435.00

Recharge $200.0
Transaction Time: 2023-11-19 13:53:23
Balance: $435.00

Recharge $100.0
Transaction Time: 2023-11-19 13:53:09
Balance: $235.00

Recharge $50.0
Transaction Time: 2023-11-19 13:52:58
Balance: $135.00

Recharge $20.0
Transaction Time: 2023-11-19 13:52:41
Balance: $85.00



Riding History

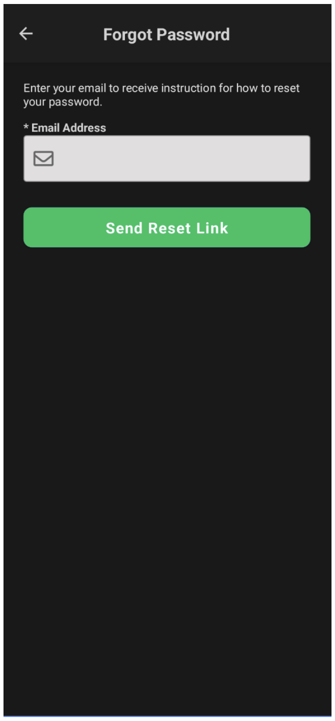
Free ride
Transaction Time: 2023-11-22 12:36:18
Balance: $450.00

Free ride
Transaction Time: 2023-11-21 20:41:46
Balance: $450.00

Free ride
Transaction Time: 2023-11-21 20:37:43
Balance: $450.00

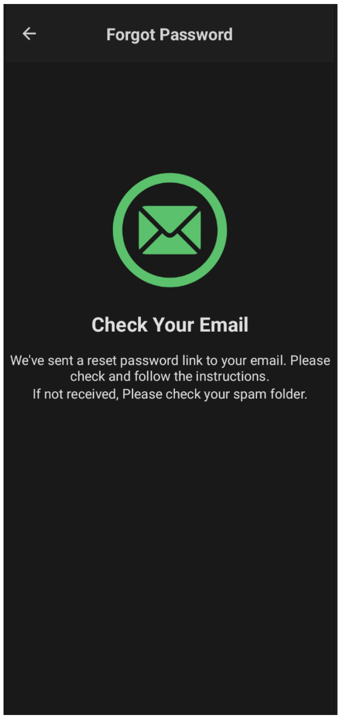
Free ride
Transaction Time: 2023-11-21 19:40:41
Balance: $450.00

Forgot Password Screen (Iteration 2):



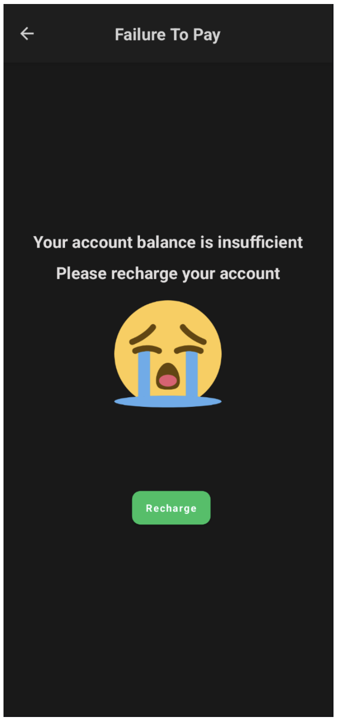
Enter your email to receive instruction for how to reset your password.

Others Screen (Iteration 2):



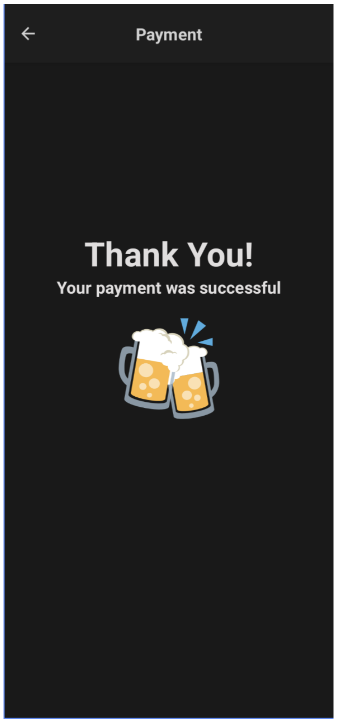
Check Your Email

We've sent a reset password link to your email. Please check and follow the instructions.
If not received, Please check your spam folder.



Your account balance is insufficient

Please recharge your account



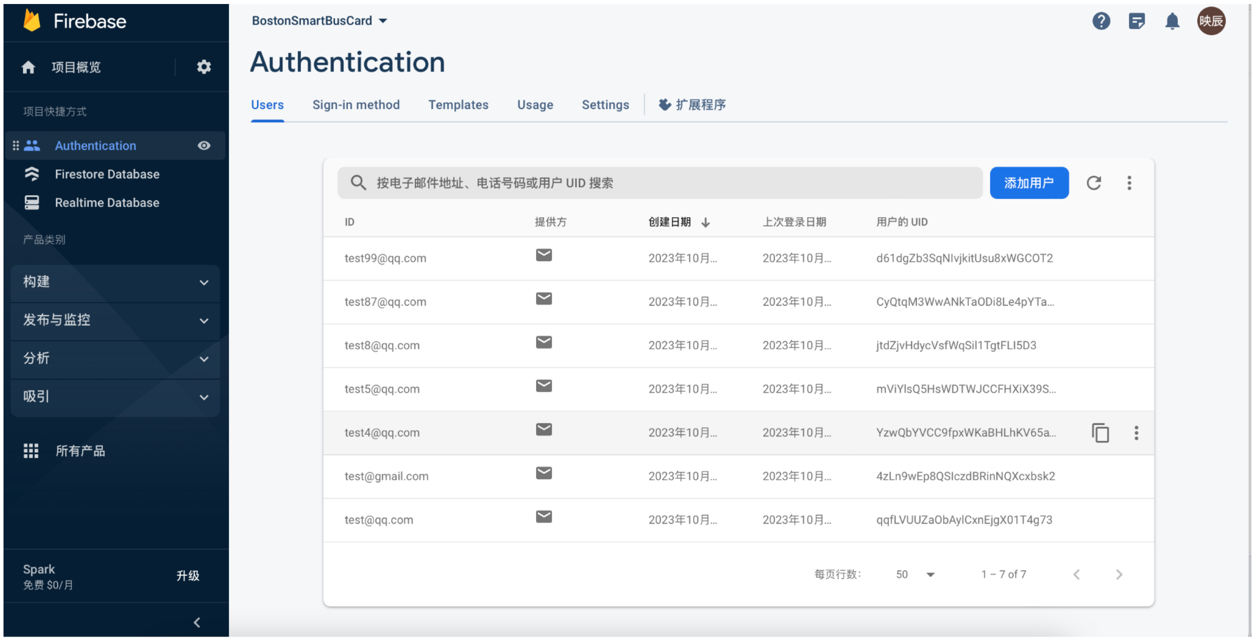
Thank You!

Your payment was successful

Data Design and implementation:

(Iteration 1)

I'm using firebase as my database, and I can register and login in the software correctly already.



Authentication

按电子邮件地址、电话号码或用户 UID 搜索

ID	提供方	创建日期 ↓	上次登录日期	用户的 UID
test99@qq.com	✉	2023年10月...	2023年10月...	d61dgZb3SqNIvjkitUsu8xWGCOT2
test87@qq.com	✉	2023年10月...	2023年10月...	CyQtqM3WwANkTaODi8Le4pYTa...
test8@qq.com	✉	2023年10月...	2023年10月...	jtdZjvHdycVsfWqSil1TgtFLI5D3
test5@qq.com	✉	2023年10月...	2023年10月...	mViYlsQ5HsWDTWJCCFHXiX39S...
test4@qq.com	✉	2023年10月...	2023年10月...	YzwQbYVCC9fpxWKaBHLhKV65a...
test@gmail.com	✉	2023年10月...	2023年10月...	4zLn9wEp8QSlczdBRinNQXcxbsk2
test@qq.com	✉	2023年10月...	2023年10月...	qqfLVUUZaObAylCxnEjgX01T4g73

每页行数： 50 1 – 7 of 7

(Iteration 2)

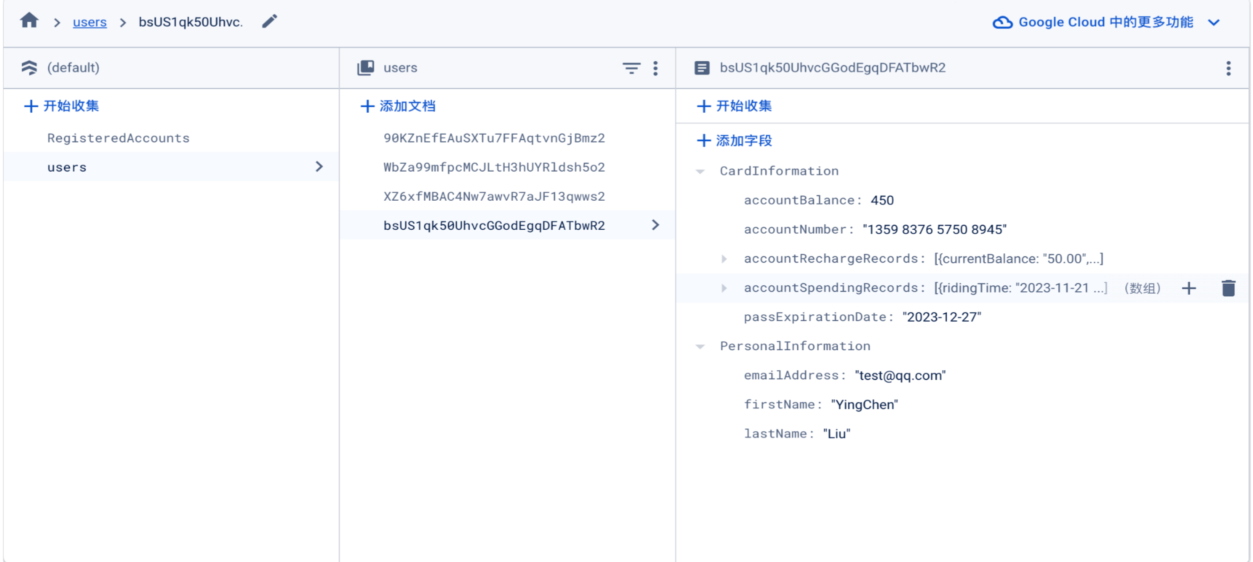
I use FireBase to store all the relevant information for each account.

I use one library to record all valid transit card account numbers, and another to store all user information.

Each user will be given a randomized bus card account number, which is a 16-digit number, once it has been created.

Assuming 100 million users are already registered, the probability of a conflict between the randomly generated numbers is only 0.00001. If the randomly generated number is already registered, the system will regenerate a new number until there are no duplicates, and then assign it to the new user.

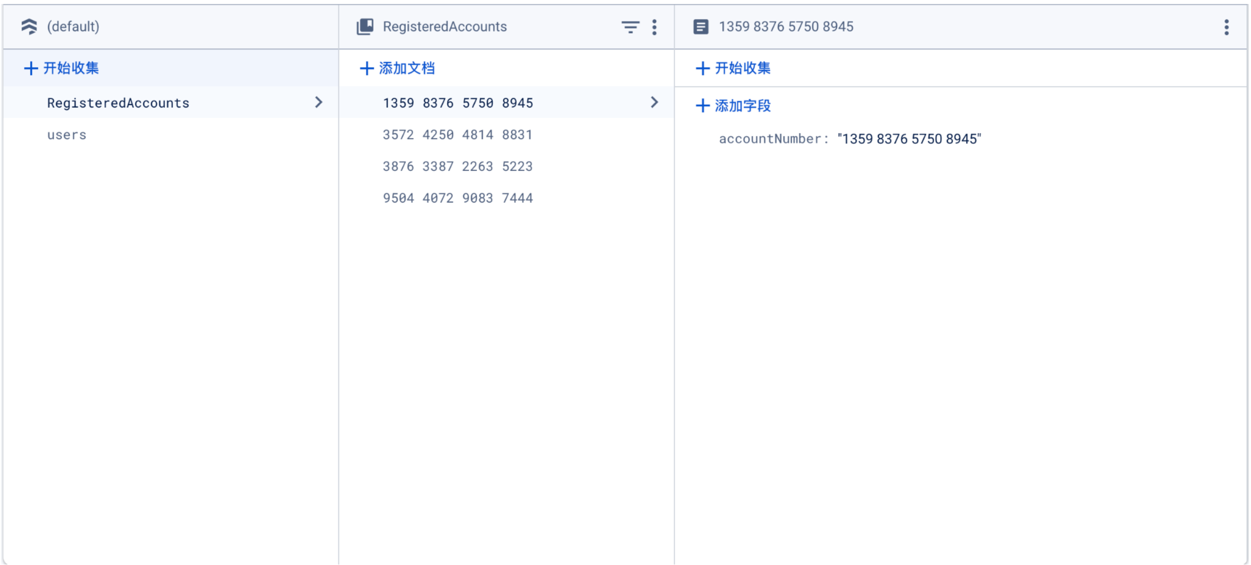
The update of each account information is strictly controlled, and all operations can only take effect if they are identical to the unique id of the account, to avoid the information in the database being changed by other people.



RegisteredAccounts
users

90KZnEfEAuSXTu7FFAqtvnGjBmz2
WbZa99mfpcMCJLtH3hUYRldsh5o2
XZ6xfMBAC4Nw7awvR7aJF13qwws2
bsUS1qk50UhvcGGodEgqDFATbwR2

```
CardInformation
    accountBalance: 450
    accountNumber: "1359 8376 5750 8945"
    accountRechargeRecords: [{currentBalance: "50.00",...]
    accountSpendingRecords: [{ridingTime: "2023-11-21 ...]  (数组)
    passExpirationDate: "2023-12-27"
PersonalInformation
    emailAddress: "test@qq.com"
    firstName: "YingChen"
    lastName: "Liu"
```



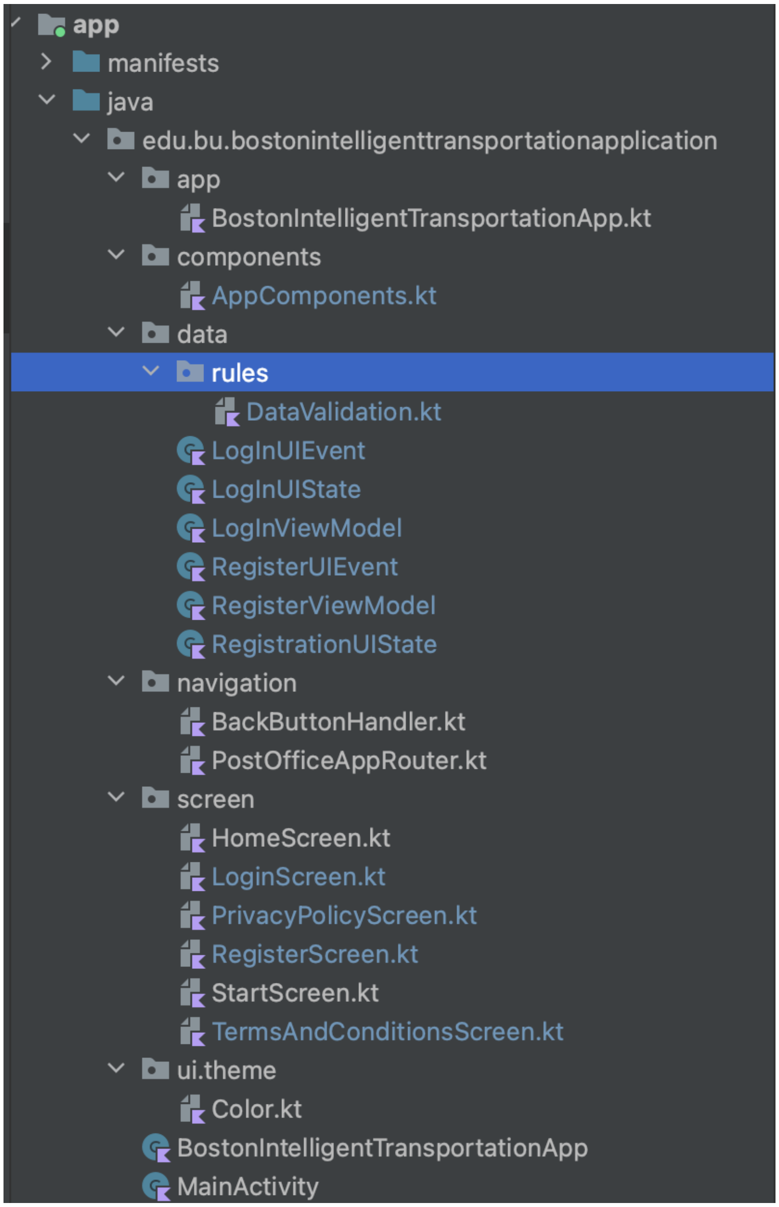
RegisteredAccounts
users

1359 8376 5750 8945
3572 4250 4814 8831
3876 3387 2263 5223
9504 4072 9083 7444

```
accountNumber: "1359 8376 5750 8945"
```

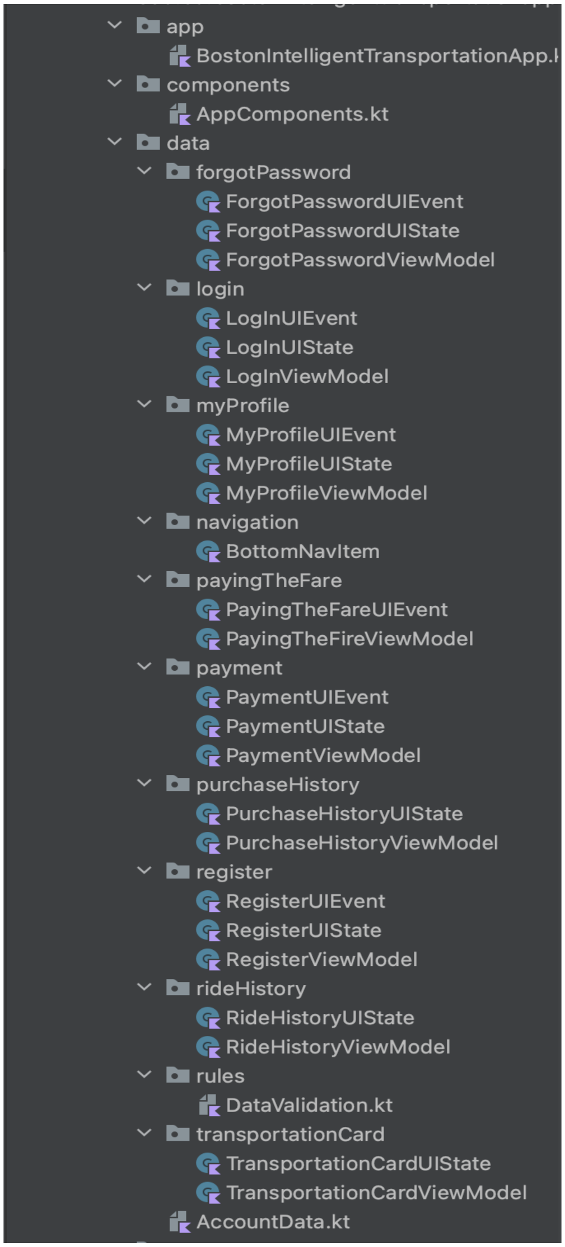
5. Project Structure

Iteration1:

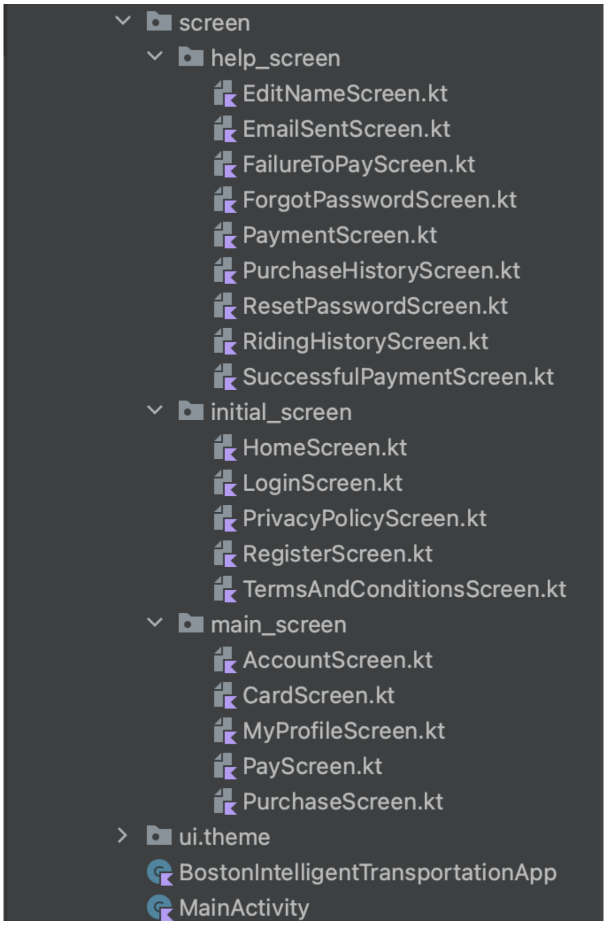


```
app
  manifests
  java
    edu.bu.bostonintelligenttransportationapplication
      app
        BostonIntelligentTransportationApp.kt
      components
        AppComponents.kt
      data
        rules
          DataValidation.kt
        LogInUIEvent
        LogInUIState
        LogInViewModel
        RegisterUIEvent
        RegisterViewModel
        RegistrationUIState
      navigation
        BackButtonHandler.kt
        PostOfficeAppRouter.kt
      screen
        HomeScreen.kt
        LoginScreen.kt
        PrivacyPolicyScreen.kt
        RegisterScreen.kt
        StartScreen.kt
        TermsAndConditionsScreen.kt
      ui.theme
        Color.kt
      BostonIntelligentTransportationApp
      MainActivity
```

Iteration2:

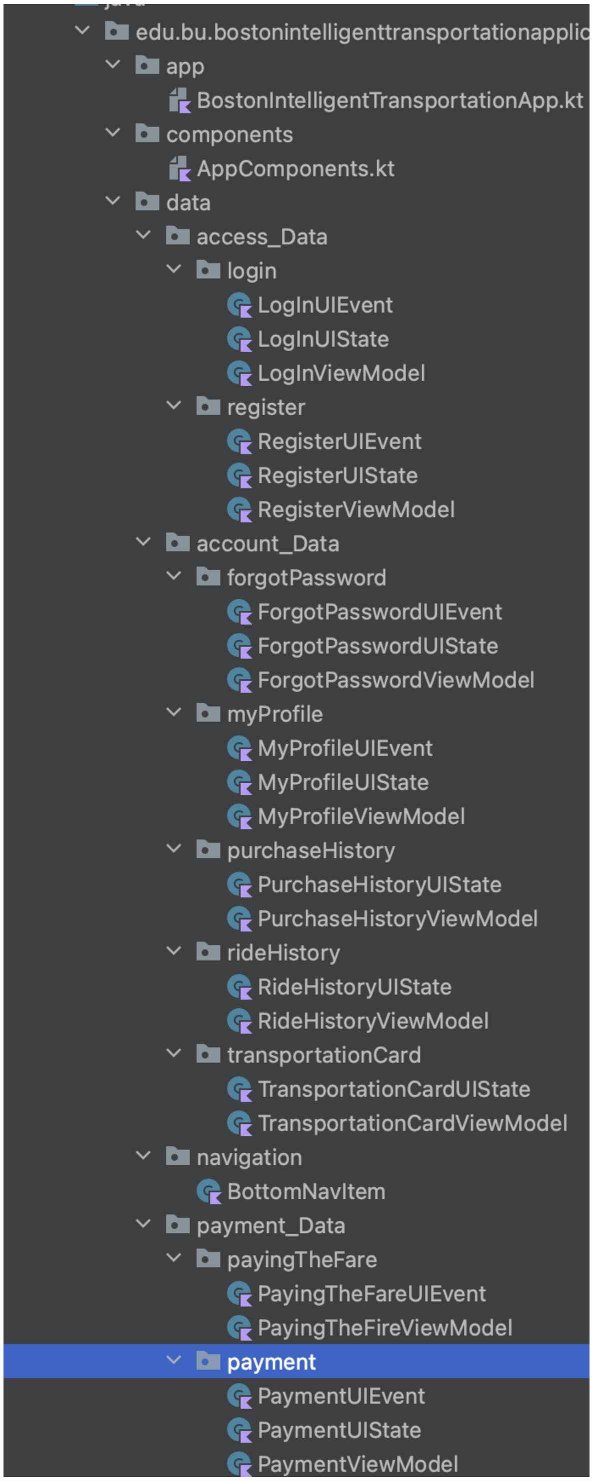


```
app
    BostonIntelligentTransportationApp.k
components
    AppComponents.kt
data
    forgotPassword
        ForgotPasswordUIEvent
        ForgotPasswordUIState
        ForgotPasswordViewModel
    login
        LogInUIEvent
        LogInUIState
        LogInViewModel
    myProfile
        MyProfileUIEvent
        MyProfileUIState
        MyProfileViewModel
    navigation
        BottomNavItem
    payingTheFare
        PayingTheFareUIEvent
        PayingTheFireViewModel
    payment
        PaymentUIEvent
        PaymentUIState
        PaymentViewModel
    purchaseHistory
        PurchaseHistoryUIState
        PurchaseHistoryViewModel
    register
        RegisterUIEvent
        RegisterUIState
        RegisterViewModel
    rideHistory
        RideHistoryUIState
        RideHistoryViewModel
    rules
        DataValidation.kt
    transportationCard
        TransportationCardUIState
        TransportationCardViewModel
    AccountData.kt
```

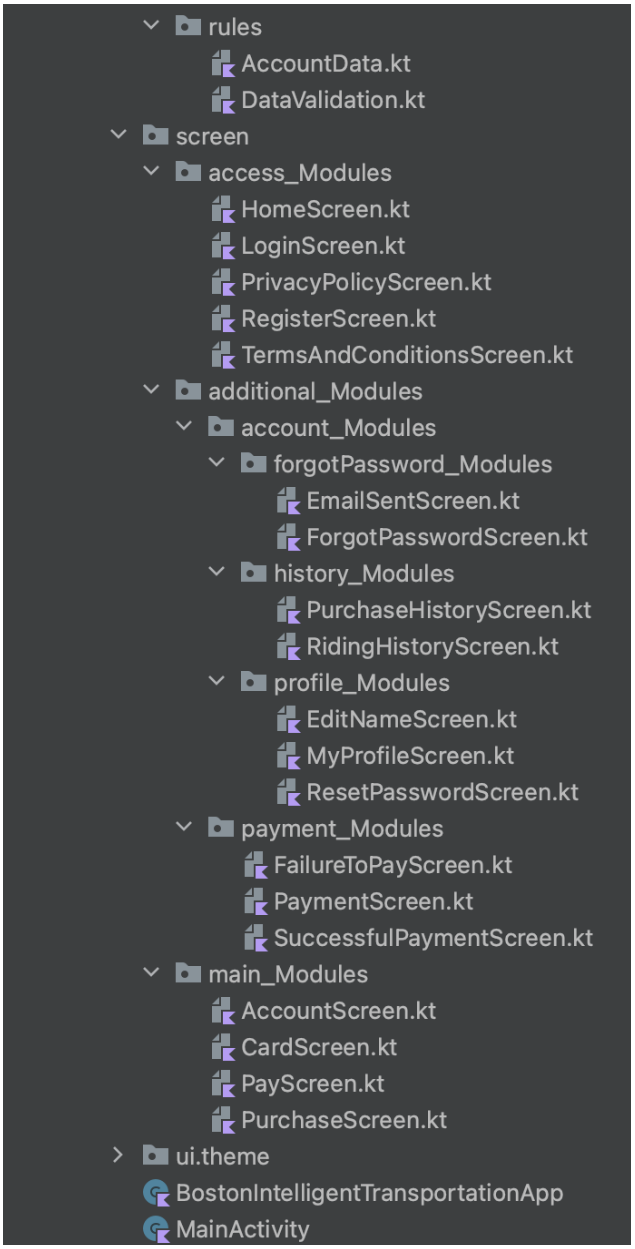


```
screen
    help_screen
        EditNameScreen.kt
        EmailSentScreen.kt
        FailureToPayScreen.kt
        ForgotPasswordScreen.kt
        PaymentScreen.kt
        PurchaseHistoryScreen.kt
        ResetPasswordScreen.kt
        RidingHistoryScreen.kt
        SuccessfulPaymentScreen.kt
    initial_screen
        HomeScreen.kt
        LoginScreen.kt
        PrivacyPolicyScreen.kt
        RegisterScreen.kt
        TermsAndConditionsScreen.kt
    main_screen
        AccountScreen.kt
        CardScreen.kt
        MyProfileScreen.kt
        PayScreen.kt
        PurchaseScreen.kt
ui.theme
BostonIntelligentTransportationApp
MainActivity
```

Iteration3:



```
java
  edu.bu.bostonintelligenttransportationapplic
    app
      BostonIntelligentTransportationApp.kt
    components
      AppComponents.kt
    data
      access_Data
        login
          LogInUIEvent
          LogInUIState
          LogInViewModel
        register
          RegisterUIEvent
          RegisterUIState
          RegisterViewModel
      account_Data
        forgotPassword
          ForgotPasswordUIEvent
          ForgotPasswordUIState
          ForgotPasswordViewModel
        myProfile
          MyProfileUIEvent
          MyProfileUIState
          MyProfileViewModel
        purchaseHistory
          PurchaseHistoryUIState
          PurchaseHistoryViewModel
        rideHistory
          RideHistoryUIState
          RideHistoryViewModel
        transportationCard
          TransportationCardUIState
          TransportationCardViewModel
      navigation
        BottomNavItem
      payment_Data
        payingTheFare
          PayingTheFareUIEvent
          PayingTheFireViewModel
        payment
          PaymentUIEvent
          PaymentUIState
          PaymentViewModel
```



```
    rules
      AccountData.kt
      DataValidation.kt
  screen
    access_Modules
      HomeScreen.kt
      LoginScreen.kt
      PrivacyPolicyScreen.kt
      RegisterScreen.kt
      TermsAndConditionsScreen.kt
    additional_Modules
      account_Modules
        forgotPassword_Modules
          EmailSentScreen.kt
          ForgotPasswordScreen.kt
        history_Modules
          PurchaseHistoryScreen.kt
          RidingHistoryScreen.kt
        profile_Modules
          EditNameScreen.kt
          MyProfileScreen.kt
          ResetPasswordScreen.kt
      payment_Modules
        FailureToPayScreen.kt
        PaymentScreen.kt
        SuccessfulPaymentScreen.kt
    main_Modules
      AccountScreen.kt
      CardScreen.kt
      PayScreen.kt
      PurchaseScreen.kt
  ui.theme
  BostonIntelligentTransportationApp
  MainActivity
```

6. Timeline

Iteration	Application Requirements (Essential/Desirable/Optional)	Android Components and Features to be used
1	RegisterAccount*(*Essential*)* Login Account*(*Essential*)*	Activate, SQLite
2	*Account Manage(*Essential*)* *Payment of fares(*Essential*)* *Recharge(*Essential*)* *Historical record(*Essential*)* *Transportation card*(Essential)	Activate, Notification, SQLite
3	*Update UI(*Essential*)*	

Iteration1:

In the current iteration I have created HomeScreen, LoginScreen, PrivacyPolicyScreen, Register Screen, StartScreen and TermsAndconditionsScreen. Except for HomeScreen which is just starting, all the other Screens have completed most of their logic and UI.

The HomeScreen can go to either the Register Screen or the Login Screen depending on the user's choice. It is also possible to return from these two screens

The Register Screen allows the user to successfully register an account according to the conditions which I set up, and if the conditions are not met, a red alert will be given (only one alert will be given at one time, the order of priority is: FirstName > LastName > Email > Password > Confirm Password > Agree Term and Policy > EmailAddress is Used > Other Exception).

Login Screen allows the user to log in the account that has been registered, currently can only detect whether the email and password meet the requirements, not yet able to distinguish the email is not registered, the password is incorrect and other Exception, for this case, all the prompt information returned is (email or password is not correct)

The PrivacyPolicyScreen and TermsAndconditionsScreen already describe the requirements of the application (generic information), which the user can scroll through.

For the database, the FireBase platform was chosen. The information is entered into the database when the user registers successfully, and is verified when logging in is required.

Iteration2:

During this phase, I completed most of the main functions of the project. The program now has the ability to pay fares, view transit card information, recharge, record history, and modify personal information.

Public Transportation Card System:
Transit cards have a unique account number, so it is not possible to have two accounts using the same transit card. The design of the algorithm is to generate a random account, then check if it exists in the database, if not then assign it to the current user, if it exists then regenerate it until it does not appear in the database. This method is more secure than the incremental method.

Payment System:
The expectation of this part is to create two different payment options, the first one is QR code scanning, which means pointing the QR code at the scanner, and then the payment will be made. However, because of the complexity of this function, only a random QR code will be displayed, and the successful payment will be shown after scanning the code. The second type is Tap Pay, which is similar to Apple Pay. My design expectation is to use a third-party API, but I haven't designed the connection yet. Currently, when you click the tap button, the fare is deducted directly from your account balance.

For payments, if the balance is not enough and no Pass has been purchased, then the payment will be shown as failed. If there is a Pass, no fee will be deducted (it will be shown as a free ride), if there is no Pass, a fee will be deducted (it will be shown as a deduction of ? $)

When a ride is successfully charged, it will be recorded in the history and can be viewed in the History screen.

Recharge System:

The recharge system only allows the customer to recharge by clicking on the given selection. The options available are Amount and Pass. After clicking on the button the customer will be transferred to the payment page, where after entering the valid card details, the account will be topped up immediately. After a successful recharge, the information is also recorded and can be viewed in the history screen. For the bank card system, which is currently a fake system, as long as it meets the format requirements of each one, it will be considered as a successful payment.

Iteration3:

In this last Iteration, I reorganized the structure of the code.

I divided each Screen that interacts with data into a UI Layer and a DataLayer. All the Data Layer code is put into the data folder (there are more specific categories later) and all the UI Layer code is put into the screen folder (same as the data folder).

Added clean comments to all the fun, checked the overall code, removed unimportant logs, and added logs where errors or exceptions might occur.

Extensive testing of all features.

7. Future Work

I'm trying to add Wallet functionality (to store the user's credit card). In this new great feature: first, the user can save the credit card that will be used for payment. Instead of re-entering all the credit card information, the user will be able to use the stored card when making a payment. The public transportation card will support automatic reloading.

For this new feature I need to do the following.
1. improve the database
2. Modify Payment Screen, add new UI Screen and Data files (ViewModel, etc).
3. Ensure the security of user information (the most important and difficult).
4. Some other unknown issues

8. Project Link

https://github.com/CS683/project-yingchenliu

9. References

1. Photo by George Frewat from Pexels:
https://www.pexels.com/photo/two-cable-cars-passing-each-other-on-the-street-18719908

2.Photo by Mathew Schwartz on Unsplash from
https://unsplash.com/photos/sunrays-wallpaper-P-WWHRF7qe0