Group 5 - CSE D
Payment Wallet
Epics, User stories and tasks

1. User Registration/Authentication(21)

1. I want to register for the application using my phone number, email address and password. (10)

1.1. Create a database in MongoDB using MongoDB Atlas in Europe or US AWS cloud partition.
1.2. Create and deploy a server in Heroku using NodeJS and Express and test it using Postman API.
1.3. Connect server to MongoDB using Mongoose library.
1.4. Create Logo for the application.
1.5. Implement Registration page in Flutter with REST API support for making requests to the server.
1.6. Write an endpoint in the server to receive requests from the app and test it using Postman API.
1.7. Write code to Make POST request over SSL to the server.
1.8. Write code on the server side to send OTP to the phone number entered by the user.
1.9. Write code on application to verify OTP entered by user in the server.
1.10. Write code to send verification link to email address entered by the user.
1.11. Write code to verify if the user clicks on the verification link.
1.12. Write code to encrypt the password before storing it in the database and test database schema using MongoDB compass.
1.13. Write logic to write the user data to the database to create a new account once all verification steps are complete.

2. I want to authenticate using my fingerprint.(2)

2.1. Implement a Login page in Flutter with REST API support for making requests to the server and test it using Postman API .
2.2. Include libraries to Flutter App to access fingerprint biometric built into the device.
2.3. Write code to check if any of the above biometrics are available in the device.
2.4. Write code to ask users if they are interested in biometric authentication.
2.5. Write code to access biometric data from the phone if the user opted yes.
2.6. Write code to verify biometric if registered by user.
2.7. Write code to redirect users to the home page if biometric verified.

3. I want to authenticate using a passcode specific to the app.(3)

3.1. Include libraries to authenticate passcode/pin.
3.2. Write code to ask users if they are interested in passcode/pin authentication.
3.3. Write code to write and access the user's passcode/pin data from the server if the user opted yes.
3.4. Write code to verify passcode/pin.
3.5. Write code to redirect users to the home page if passcode/pin is verified.

4. I want to Update my authentication credentials.(4)

4.1. Write Edit User Information page in Flutter with REST API support for making requests to the server and test it using Postman API.
4.2. Write an endpoint in the server with code to update in MongoDB whenever requested and test the endpoint using Postman API, test the database schema using MongoDB compass.
4.3. Write code to get data from fields in the Edit User Information page.
4.4. Write code to send data to the server from the Edit User Information page for verification and updation.

4.5. Verify if the entered data are valid and current credentials like passcode and email match and update data if they do.

5. I want a Forgot passcode Option.(2)

5.1. In Login Page add a Forgot passcode Button
5.2. Write code to access the server for User's registered mobile number
5.3. Create a plugin for sending OTP through SMS.
5.4. Verify the OTP entered by the user using POST request to the server and test the OTP verification module using Postman API.
5.5. Write code to update the password in the database, test updation in the database using MongoDB Compass.
5.6. Write code to redirect the user to the login page.

2. Transactions and money limits(24)

1. I want to add money to the wallet.(6)

1.1. Create a wallet Balance page in Flutter App.
1.2. Create a Dummy Bank Server for testing the transactions.
1.3. Add module to add money to wallet from Bank Server.
1.4. Create a Dummy Bank Database for maintaining Dummy Bank's User Information
1.5. Create a Database to store all transactions and test saving using MongoDB compass.
1.6. Create a HTTP/Websocket Connection Between our server and Dummy Bank Server and test the connection using Postman API.
1.7. Create a Verification module for checking the Bank Balance and Transaction Limit for specified Bank Account and send approval to flutter app.
1.8. Create a Transaction Status Page in Flutter and test the transaction values using Flutter Driver.

2. I want to receive payments by scanning QR code.(4)

2.1. Create a QR Code Screen In Flutter for Receiving Payments.

2.2. Create a QR code Scanning Module and create a button on the home screen to link the module.
2.3. Create a secure person to person transaction handling module.
2.4. Create a screen to accept the amount to be transferred.
2.5. Create a screen to confirm payment to the recipient.
2.6. Create a screen to show the received payment on the recipient side.
2.7. Add module to add received credits to wallet from Bank Server.

3. I want to do international transactions.(6)

3.1. Setup international credit holding services to support currencies in all countries.
3.2. Add a module to convert transaction amount to the currency type based on the location of the user.
3.3. Add a module asking the user to allow international transactions.
3.4. Add an end-point to accept and convert international transactions and test the end-point using Postman API.
3.5. Add a module to accept payments and make international transactions by extending the transaction module.

4. I want to revert transactions before buffer time ends, triggered by transactions above the threshold.(4)

4.1. Add a feature in the Settings Screen to enable or disable “Hold transactions”.
4.2. Add a feature to toggle between automatic and manual threshold setting.
4.3. Add a module to compute the threshold amount, after which the transaction is held for a specified amount of time.
4.4. Add a screen to display time left until the transaction cannot be reversed.
4.5. Add a module to locally cache the amount that is blocked.
4.6. Add a timer module to execute transactions after the timer ends.

4.7. Add a page to cancel the cached transaction before the timer ends, and a button to skip the timer if the user is sure about the transaction.

5. I want to send money to others(4)

5.1. Create a Send Money page in Flutter where one can scan QR code/search phone numbers/search names to find the recipient.
5.2. Create a page for specifying the amount to send, in Flutter.
5.3. Create an option to choose from the linked bank accounts of the user.
5.4. Create a loading screen
5.5. Create a Payment Status page and display the status of the transaction.

3. Notification(13)

1. I want to get notified when money is wired into my wallet.(3)

1.1. Create an always on function in the server which periodically checks change in User Wallet Balance.
1.2. Create a Function in the server which looks for User's Token to send notification messages to the user.
1.3. Create a function to retrieve the most recently received payment to the user to append the details in the Notification.
1.4. Redirect the user to Wallet Balance Page on clicking the notification.
1.5. Create a function to retrieve the payment information on querying the server(from the app).

2. I want to get notified whether my payment was successful or not(2)

2.1. Create a Local Notification Module in Flutter.
2.2. Create a Server function which monitors ongoing transactions.

2.3. Create a Server function to send the Confirmation Message via HTTP.
2.4. Create a Server function to check the wallet balance once a transaction is successful and append the details in notification.
2.5. Create a function to retrieve the most recently sent transaction from the user to append the details in the Notification.

3. I want to get notified when the money in my wallet lowers beyond a set threshold(2)

3.1. Add an Option in settings to turn on or off this functionality.
3.2. Add an option in setting to set the threshold value.
3.3. Create a Function to compare threshold value and wallet balance and trigger the notification.
3.4. Extend the Local Notification Module to send the notification to User.
3.5. Add an option to automatically update the wallet balance to threshold in the notification message from the default bank account.
3.6. Redirect from notification to 'Wallet Balance' page.

4. I want to get notified when I don't have sufficient money for the transaction(2)

4.1. Create a server function which checks the wallet balance of the user.
4.2. Write code in the app to trigger this function when a transaction is initiated.
4.3. If not enough wallet balance, redirect to 'Wallet bank page'
4.4. Show an error message stating that enough wallet balance is not there.
4.5. Add money needed to complete transactions in the 'how much money to add' field.

5. I want to get chat notifications (4)

5.1. Extend the Function to retrieve user token to get token on receiving a message from contacts.

5.2. Create a function to retrieve message sender information.
5.3. Create a Notification Message by appending Message Sender Information.
5.4. Create a functionality to redirect the user to the respective chat on clicking the notification.
5.5. Add a functionality to directly send messages to the sender from notification.

4. Chat(25)

1. I want to chat with payment receivers. (6)

1.1. Add a page to add new payment receivers to friends list.
1.2. Add a page to display all payment receivers in the friends list and test the ListView Builder using Flutter Driver.
1.3. Add a page to send and receive messages and media content with the receivers.
1.4. Add a Socket based end-point in the web server to send and receive chat messages realtime and test the end-point using Postman API.
1.5. Add Socket implementation to send or receive data from the server to allow chatting.
1.6. Add a local database to store received chats to allow offline capabilities and test the schema and storage using MongoDB Compass.

2. I want to chat with people in my contacts.(3)

2.1. Add a module to request permission to access Contacts.
2.2. Add a module to scan and get a list of all Contact Members who have the application.
2.3. Add a module to add all members to the online profile so that the members are still visible to the application even if contact information is not available in the future.
2.4. Add a module to receive the list and display it in the app.
2.5. Extend the Chat Module to allow chats with the people from contacts.

3. I want to chat with technical support.(10)

3.1. Add a page in application for people to access when they have technical queries.
3.2. Add a module to redirect users to technical support email for more formal communication.
3.3. Add a rule based chat bot in the backend to allow the user to get immediate answers.
3.4. Add an option to view availability of technical support members.
3.5. Add an option to redirect the queries to an available person for more sophisticated solutions.
3.6. Extend the Chat Module for chat based communication.

4. I want to join a chat on scanning QR code.(3)

4.1. Extend the QR code scanning module to scan QR codes for Chat ID.
4.2. Extend the QR code screen to allow the user to use the camera to scan for QR codes.
4.3. Add an end-point to generate a unique QR code to allow joining the user and test the end-point using Postman API.
4.4. Add a REST API module to send the scanned QR code to validate it and establish connection between the users.
4.5. Add a module to add a new user to chat list when QR code successfully scanned.

5. I want to join a chat through username.(3)

5.1. Add an end-point to query the server for a list of all matching users with the query string who are related to the current user.
5.2. Add a search bar to allow the user to search for users.
5.3. Add a module for real time querying using sockets from the Flutter Application.
5.4. Add a flutter module to highlight the queried text in results of the users found.
5.5. Add a module to display “No users found” if none are found from the query.

5. Calculators(13)

1. I want an EMI Calculator Option for Home Loan(3)

1.1. Add a Calculator Screen which has buttons to trigger different calculators.
1.2. Create a Screen for Home Loan Calculation.
1.3. Create a Function to handle the form to receive the parameters.
1.4. Create a Function to calculate estimated Home Loan and test the output using Flutter Driver.
1.5. Create a Function to Redirect the user to Partner Bank For Home Loan (Earning Strategy).

2. I want an Calculator Option for Education Loan(2)

2.1. Create a Screen for Education Loan Calculation.
2.2. Create a Function to calculate Education Loan and test the output using Flutter Driver.
2.3. Create a Function to Calculate estimated Years to completely pay the home loan and test the output using Flutter Driver.
2.4. Create a Function to Redirect the user to Partner Bank For Education Loan (Earning Strategy).
2.5. Add a function to Notify Bank Databases the interest of User to take an education loan and test the database using MongoDB Compass. (Earning Strategy).

3. I want an EMI Calculator Option for Car Loan(2)

3.1. Create a Screen for Car Loan Calculation.
3.2. Create a Function to calculate Car Loan and test the output using Flutter Driver.
3.3. Create a Function to handle the form to receive parameters for calculation.
3.4. Create a Functionality to Redirect user to partner banks for Car Loan Information Earning Strategy).

3.5. Create a Function to notify the partner bank database the interest of the user to apply for a car loan and test the database and schema using MongoDB Compass.

4. I want a Crypto Currency Conversion Calculator(3)

4.1. Add a Function to retrieve current Crypto currency Rates from the internet.
4.2. Add a Function to convert crypto currency to normal currency ,
4.3. Add a Function to convert crypto currency to other crypto currency.
4.4. Redirect users to Partner Crypto Exchanges for buying crypto currencies.(Earning Strategy)
4.5. Redirect Users to a list of Trending Cryptocurrencies.
4.6. Add a Page for Crypto Currency calculator.

5. I want a Currency Converter(3)

5.1. Add a page for currency conversion.
5.2. Add a function for retrieving currency rates from the internet.
5.3. Add a function for converting Currency to another currency.
5.4. Add an option to include conversion charges in the conversion.
5.5. Add a function to convert currency to another currency taking into account the conversion charges.

6. Analytics(17)

1. I want analysis of my yearly expenditures/revenues(2)

1.1. Add a module to monitor and record all expenses made through the app in Flutter Application.
1.2. Add an end-point to receive all recorded yearly expenses in the backend server and test the end-point using Postman API.

1.3. Create a database to maintain all yearly expense data and test the database using MongoDB Atlas.
1.4. Create a backend module to generate analysis over the yearly data.
1.5. Create a Flutter UI to display analysis of the yearly expense data using data visualisation libraries.

2. I want analysis of my monthly expenditures/revenues(6)

2.1. Add a module to monitor and record all expenses made through the app in Flutter Application.
2.2. Add an end-point to receive all recorded monthly expenses in the backend server and test the end-point using Postman API.
2.3. Create a database to maintain all monthly expense data and test the database using MongoDB Atlas.
2.4. Create a backend module to generate analysis over the monthly data.
2.5. Create a Flutter UI to display analysis of the monthly expense data using data visualisation libraries.

3. I want analysis of my weekly expenditures/revenues(2)

3.1. Add a module to monitor and record all expenses made through the app in Flutter Application.
3.2. Add an end-point to receive all recorded weekly expenses in the backend server and test the end-point using Postman API.
3.3. Create a database to maintain all weekly expense data.
3.4. Create a backend module to generate analysis over the weekly data.
3.5. Create a Flutter UI to display analysis of the weekly expense data using data visualisation libraries.

4. I want analysis of my daily expenditures/revenues(2)

4.1. Add a module to monitor and record all expenses made through the app in Flutter Application.
4.2. Add an end-point to receive all recorded daily expenses in the backend server and test the end-point using Postman API.
4.3. Create a database to maintain all daily expense data.

4.4. Create a backend module to generate analysis over the daily data.
4.5. Create a Flutter UI to display analysis of the daily expense data using data visualisation libraries.

5. I want to know my most frequently contacted user accounts (for transactions)(5)

5.1. Add a module to record and send whenever a contact or account is opened for a transaction.
5.2. Add an end-point to receive collected data about all opened accounts and test the end-point using Postman API.
5.3. Create a database to store information about all viewed accounts and test the database using MongoDB Atlas.
5.4. Add a module to find frequently opened accounts with which transactions are made.
5.5. Add a REST API to receive information about all frequently accessed accounts.
5.6. Add a page to display all frequent contacts/accounts.