

BM Graduation

Project Idea: Money Transfer Application

Concept:

This project proposes a mobile application for secure and convenient money transfers, incorporating features similar to online banking and mobile money apps.

Features:

User Accounts and Profiles: Users can create profiles, set up online accounts, and link them to existing bank accounts.

Internal and External Transfers: Users can transfer funds between their own accounts (internal) or to other users within the app (internal) or to external accounts

Account Management: Users can view account balances, transaction history, and manage personal information.

Sub-Accounts: Users can create sub-accounts for budgeting purposes, categorizing debits and credits for better financial management.

Benefits:

Users can transfer money anytime, anywhere using their mobile devices.

Team Division:

To efficiently develop the Money Transfer app, we can divide the work among four teams as follows:

1. Backend Team:

- Focuses on server-side development.
- Handles user authentication, account management, transaction processing, and API development.

2. Frontend Team:

- Designs the user interface and user experience for the application.
- Ensures a user-friendly experience for creating accounts, logging in, and managing finances.

3. Mobile Developer Team:

- Develops native mobile applications for both iOS and Android platforms.

- Ensures functionality aligns with user stories and integrates seamlessly with the backend.

Each team should collaborate closely, communicate effectively, and follow agile development practices Regular sync-ups and coordination among the teams are essential to ensure a cohesive and unified money transfer app experience across different platforms.

Here are user stories for each feature and for each team separately.

**** User Stories for Backend Team:****

1. *User Account Management:*

- *Account Creation:*

- Endpoint to create a new user account with name, email, and password.
- Validate and store user information securely, including password hashing.

- *Account Login:*

- Endpoint to authenticate users using email and password.
- Generate and return authentication tokens upon successful login.

- *Account Logout:*

- Endpoint to invalidate authentication tokens upon user logout.
- Mechanism to handle token expiration or invalidation after 30 minutes of inactivity.

2. *Session Management:*

- Implement session tracking to monitor user activity.
- Automatically invalidate sessions after 30 minutes of inactivity.

3. *Account Balance:*

- Endpoint to fetch the current balance of the user's account.
- Secure access control to ensure only authenticated users can access their balance.

4. *Transaction History:*

- Endpoint to retrieve the transaction history for a user.
- Support pagination or date-based filtering for efficient data retrieval.

5. *Money Transfer:*

- Endpoint to initiate a money transfer to another user.
- Validate recipient details and ensure sufficient funds before processing.
- Record the transaction details and update account balances accordingly.

6. *Favorite Recipients:*

- (Optional) Endpoint to save and manage favorite recipients for a user.
- Allow retrieval of favorite recipients for quick access during money transfers.

Specific Backend Endpoints

1. *User Account Management:*

- POST /api/register - Create a new user account.
- POST /api/login - Authenticate a user and return an auth token.
- POST /api/logout - Invalidate the user's auth token.

2. *Session Management:*

- Middleware to track user activity and manage session timeouts.

3. *Account Balance:*

- GET /api/balance - Get the current account balance for the authenticated user.

4. *Transaction History:*

- GET /api/transactions - Get the transaction history for the authenticated user.

5. *Money Transfer:*

- POST /api/transfer - Transfer money to another user.

6. *Favorite Recipients (Optional):*

- POST /api/favorites - Add a favorite recipient.
- GET /api/favorites - Get the list of favorite recipients.
- DELETE /api/favorites/{id} - Remove a favorite recipient.

Additional Backend Considerations

1. *Security:*

- Implement secure authentication and authorization mechanisms.
- Use HTTPS to encrypt data in transit.
- Ensure proper validation and sanitization of all input data to prevent SQL injection and other security threats.

2. *Data Integrity:*

- Ensure atomicity of money transfer operations to prevent inconsistencies.
- Implement proper error handling and logging.

3. *Scalability:*

- Design the backend to handle a large number of concurrent users.
- Optimize database queries and consider using caching mechanisms for frequently accessed data.

4. *Testing:*

- Write unit tests and integration tests for all endpoints and functionalities.

- Perform security and performance testing to ensure robustness.

5. *Deployment:*

- Packing spring boot application into Jar file
- Create and run docker container along with database container in a docker compose file
- Deploy the image to any cloud providers (Azure container instance , AWS ,)
- Enables CORS support for all endpoints (Use crossOrigin annotation)
- .

**** User Stories for Frontend Team: ****

1. As a user, I want to create an account by providing my name, email, password, country, and date of birth so that I can start using the banking application. The password should be at least 6 characters long and include at least one uppercase letter, one lowercase letter, and a special character (e.g., \$%^).
2. As a user, I want to log in using my email and password so that I can access my banking account. All input fields should properly handle validation errors.
3. As a user, I want to log out from my account to ensure my information remains secure.
4. As a user, I should receive an alert dialog after 2 minutes of inactivity (e.g., "We logged you out because you were inactive for 2 minutes - it's to help keep your account secure") that redirects me to the login page (didn't request or send anything from/to the server).
5. As a user, I want to view my current account details, including my balance, so that I can keep track of my finances.
6. As a user, I want to be able to update my profile information, such as my email address, country, DOB or password, so that I can keep my account secure and up to date.
7. As a user, I want to be able to view my transaction history, including details such as transaction date, recipient name, and transaction amount, so that I can review my past transactions.
8. As a user, I want to transfer money to other users by entering only the amount and recipient details. I want to ensure that the recipient's account details are valid to avoid errors and that the transaction does not exceed my current balance.
9. As a user, I want to receive appropriate error messages or notifications if there are any issues during the transaction process or during the registration process, such as (insufficient

funds) or (invalid recipient details) or (invalid email or password), so that I can take appropriate actions or rectify the problem.

10. As a user, I want to save one or more recipients to my favorites so that I can quickly retrieve them for transfers. I also want to be able to delete a recipient from the favorites list if needed. (Optional – If the backend provides the API, use it; if not, use local storage).

****User Stories for IOS Team:****

1. As a user, I want to be able to know about the app through onboarding screens. The user should see those onboarding screens only the first time they use the app.
2. As a user, I want to be able to create an account (profile) with my personal information, including my name, email, and password, so that I can securely access the app and perform transactions. The password should be 6 characters with at least one capital letter, one small letter, and a special character (e.g. \$%^).
3. As a user, I want to be able to complete my profile by entering the required details which are the country and the DOB to be able to use the app.
4. As a user, I want to be able to log in to my account using my credentials, so that I can access my account details and perform transactions.
5. As a user, I want to be able to view my account balance, so that I can keep track of my available funds.
6. As a user, I want to be able to view my transaction history, including details such as transaction date, recipient name, and transaction amount, so that I can review my past transactions.
7. As a user, I want to be able to send money to another user by entering the recipient's account details, including their name and account number, so that I can transfer funds securely.
8. As a user, I want to be able to verify the recipient's account details via an API call, such as their name and account number, before confirming the transaction, to ensure that I'm sending money to the correct person.
9. As a user, I want to receive a confirmation or notification after completing a successful transaction, so that I can have a record of the transaction and ensure it was processed correctly.
10. As a user, I want to receive appropriate error messages or notifications if there are any issues during the transaction process, such as insufficient funds or invalid recipient details, so that I can take appropriate actions or rectify the problem.
11. As a user, I want to be able to update my profile information, such as my email address, country, DOB or password, so that I can keep my account secure and up to date.
12. As a user, if there is an issue in the app, I would like to have two options: (Call the support & Send an email to the

support)

13. As a user, I should receive an alert dialog after two minutes of inactivity (e.g. didn't request or send anything from/to the server) that takes me to login again.
14. As a user, I want to save one or more recipients to my favorites so that I can quickly retrieve them for transfers. I also want to be able to edit and delete a recipient from the favorites list. (Optional)
15. As a user, I want to be able to know if there's a server error or if there is no internet connection via its specified screens in the design. (Optional)

**** User Stories for Android Team: ****

1. As a user, I want to be able to know about the app through onboarding screens. The user should see those onboarding screens only the first time they use the app.
2. As a user, I want to be able to create an account (profile) with my personal information, including my name, email, and password, so that I can securely access the app and perform transactions. The password should be 6 characters with at least one capital letter, one small letter, and a special character (e.g. \$%^).
3. As a user, I want to be able to complete my profile by entering the required details which are the country and the DOB to be able to use the app.
4. As a user, I want to be able to log in to my account using my credentials, so that I can access my account details and perform transactions.
5. As a user, I want to be able to view my account balance, so that I can keep track of my available funds.
6. As a user, I want to be able to view my transaction history, including details such as transaction date, recipient name, and transaction amount, so that I can review my past transactions.
7. As a user, I want to be able to send money to another user by entering the recipient's account details, including their name and account number, so that I can transfer funds securely.
8. As a user, I want to be able to verify the recipient's account details via an API call, such as their name and account number, before confirming the transaction, to ensure that I'm sending money to the correct person.
9. As a user, I want to receive a confirmation or notification after completing a successful transaction, so that I can have a record of the transaction and ensure it was processed correctly.
10. As a user, I want to receive appropriate error messages or notifications if there are any issues during the transaction process, such as insufficient funds or invalid recipient details, so that I can take appropriate actions or rectify the problem.
11. As a user, I want to be able to update my profile information, such as my email address, country, DOB or password, so that I can keep my account secure and up to date.
12. As a user, if there is an issue in the app, I would like to have two options: (Call the support & Send an email to the support)
13. As a user, I should receive an alert dialog after two minutes of inactivity (e.g. didn't request or send anything from/to the server) that takes me to login again.
14. As a user, I want to save one or more recipients to my favorites so that I can quickly retrieve them for transfers. I also want to be able to edit and delete a recipient from the favorites list. (Optional)
15. As a user, I want to be able to know if there's a server error or if there is no internet connection via its specified screens in the design. (Optional)