1. Introduction

Document purpose

This document aims to describe the expected behaviour of the application "Mobanko", described below.

System purpose

The system (hereafter also referred to as "Mobanko") aims to function as a mobile banking service.

Definitions and abbreviations

A user that has logged in before last closing the application, or that has inputted his basic credentials and passed the first database check during the log-in procedure, will be called "recorded".

There are default conditions for each flux detailed in section: the user must have a working internet connection during any operation and have the application installed.

2. General description

Short description

Mobanko is a mobile banking application developed to allow users to access basic banking services via their Android mobile devices. The application aims to provide a seamless banking experience, allowing users to manage their finances on the go securely. The following user specification model outlines the necessary features, functionalities, and requirements that will make Mobanko a successful mobile banking application.

Motivation

Since many people find in-person interaction to be cumbersome and scheduling an appointment to find details about the current state of a bank account in-person is difficult, an application that simplifies the necessary procedures can find a growing number of customers. Mobanko aims to be such an application.

Similar products

Since the beginning of the millenium, PayPal has provided a satisfactory experience for personal computers, and more recently, banks have started distributing applications aimed at their clients in particular.

Project risks

As the competition is well established, this application must prove itself to be superior to them in some way.

3. The system

Users

There will be only three categories of individuals:

1. Users, who are capable of performing standard operations. Estimated number: ~10.000. Competency required: reading knowledge, basic level of familiarity with technical financial terms.
2. Providers, who will provide services sold through the platform. Estimated number: ~20. Competency required: familiarity with financial terms.
3. Other individuals who have a bank account but do not directly interact with the system; they are indirectly interacting by receiving and sending money by transfers from the system. Estimated number: ~4 bln. Competency required: various. Estimated frequency of system interaction by any individual is expected to be ~0.03/day on average.

System requirements

The device required to use the system should have the minimum requirements as specified by the operating system Android 10, and be running said system or a more recent version. The device should also have celular signal, and optionally be capable of reading fingerprints for biometric authentication.

Natural requirements

Security Features

Mobanko should have strong security features to ensure safe and secure transactions.

User Interface

The application should have an intuitive and user-friendly interface that allows for easy navigation and transaction processing.

Account Management

Users should be able to open and manage their bank accounts through the Mobanko mobile banking application.

Money Transfer

The application should allow users to transfer money to other bank accounts either within the same bank or other banks.

Account Balances

Users can check and view the balance of their accounts on the mobile app.

Transaction History

The application should allow users to view their transaction history to access details of their financial transactions.

Notifications

Users should receive notifications for every transaction processed through their accounts.

Functional Requirements

Login

Mobanko users should be able to log in using their registered usernames and passwords.

OTP Verification

The application should have a one-time password (OTP) verification process for every transaction made by users.

Biometric Authentication

Mobanko should have biometric authentication features, as fingerprints, to secure user information.

Navigation Menu

Mobanko should have a navigation menu that should be easily accessible to users, providing them with all essential financial services.

Non-functional Requirements

Security

Mobanko should have robust security features to safeguard users' sensitive data.

Responsiveness

The mobile banking application should be responsive to users' commands, allowing for real-time transactions and updates.

Device Compatibility

The application should be compatible with different Android devices in various screen sizes.

Performance

Mobanko should have exceptional performance that guarantees speedy transactions with minimal downtimes.

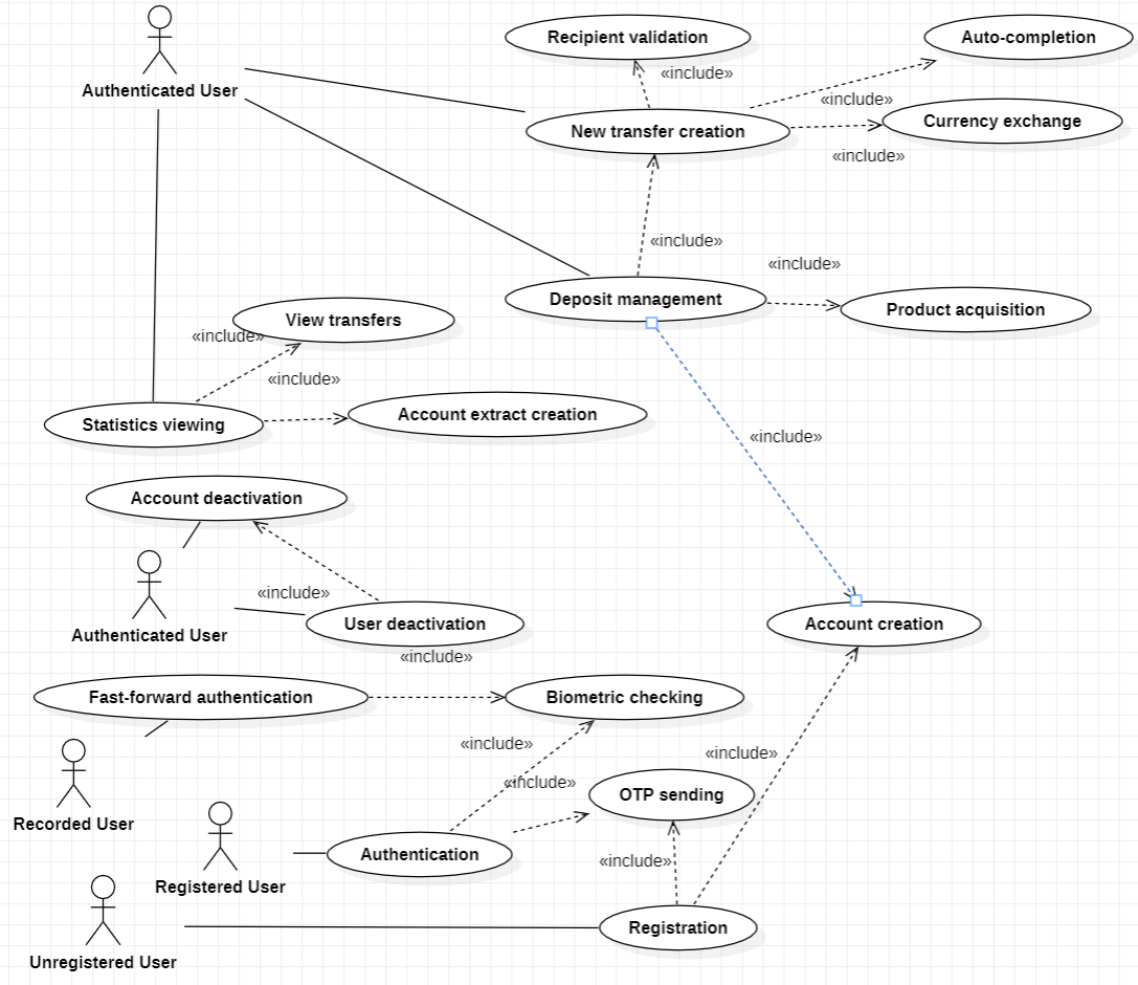
Reliability

The application should be reliable, ensuring that users can access its features and functions regardless of their location or time.

4. System models

General notes

Use-cases are described here.



Registration flux

Main flux

Pre-condition: the user must not be already logged in.

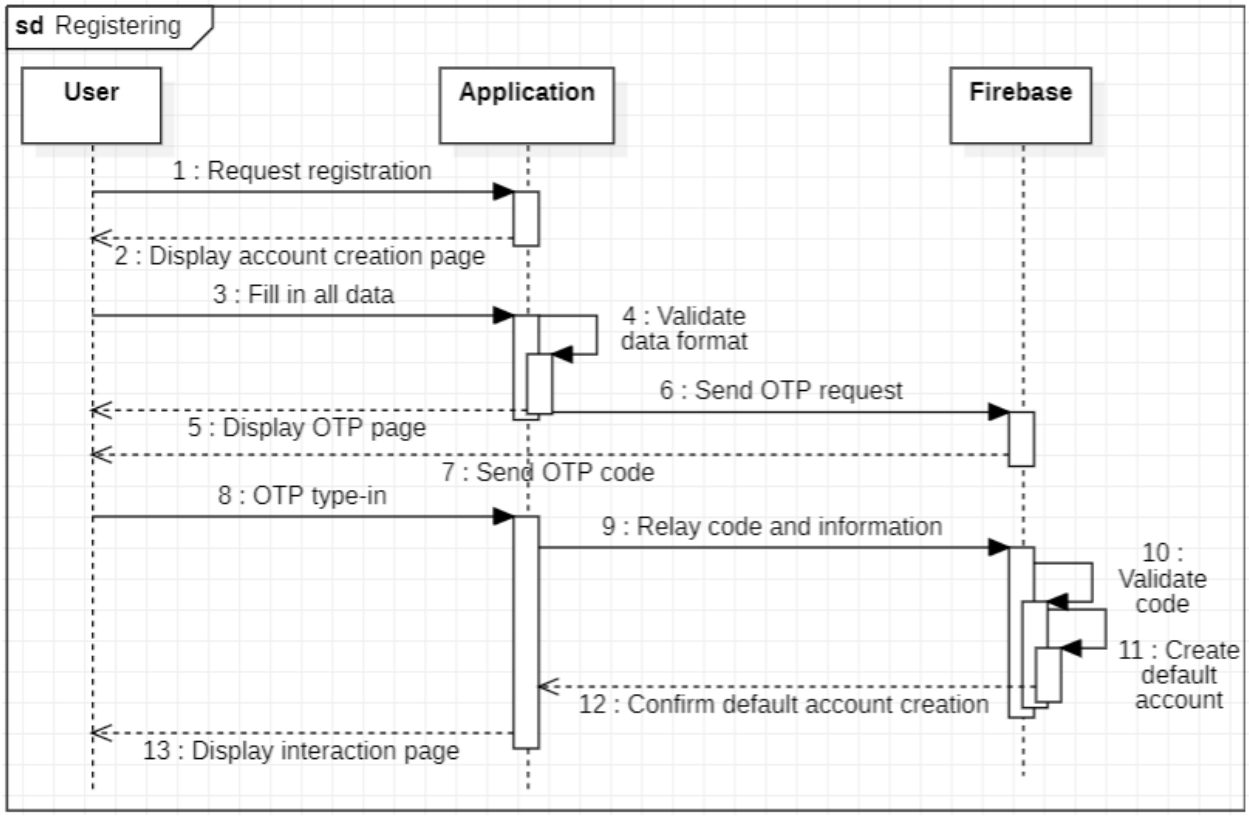
1. The user will press the "register" button.
2. The system will prompt the user for their phone number.
3. The user should fill it in.
4. The system will validate the data format.
5. The system will send the data to the database.
6. The database will update and send a confirmation.
7. Steps 1 - 5 of the "OTP sending" flux will be taken, along with exceptions.
8. The system will prompt the user for other data: email, phone number, password.
9. The user should complete the data.
10. The system will validate the data format and send the data to the database for registering.
11. The database will reply with a confirmation.
12. Steps 1 - 5 of the "Account creation" flux will be taken.
13. The system will display the main page of the new account.

Post-condition: the user will be registered, authenticated and have one account.

Alternatives

4.
    1. If the phone number format is incorrect, return to step 2.
6.
    1. If the database does not confirm, return to step 2.
7.
    1. If any exception happens, follow corresponding steps in "OTP sending" flux.
10.
    1. If the data format is invalid, return to step 8.
11.
    1. If the database does not reply with a confirmation, return to step 8.
12.
    1. If any exception happens, follow correspinding steps in "Account creation" flux.

Diagram



Account creation flux

Main flux

Pre-condition: the user must be authenticated.

1. The user will press the button "New account".
2. The system will prompt the user for the account currency.
3. The user should select the wanted currency.
4. The system will forward the user's data and the currency to the database.
5. The database will send an update confirmation.
6. The user will be able to select an account to view.

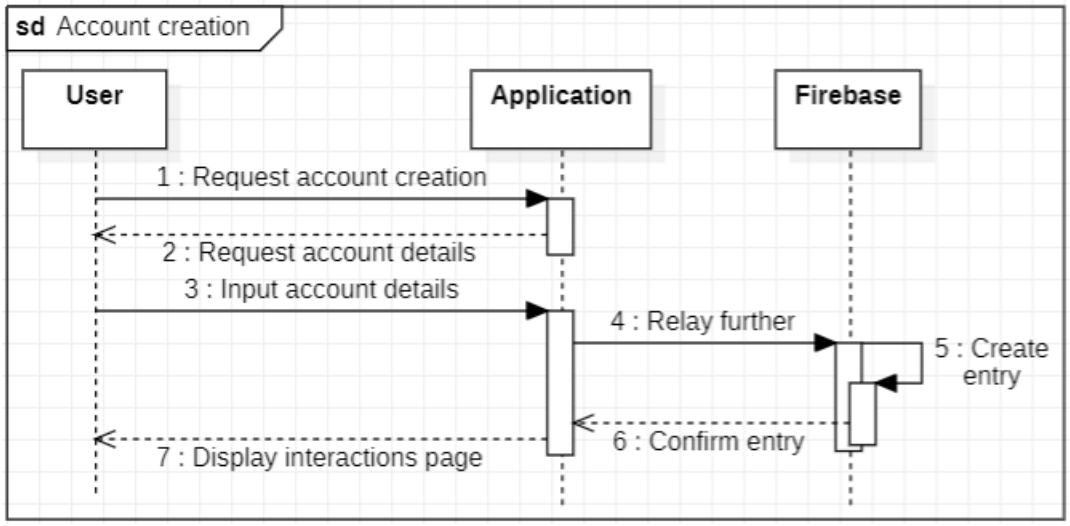
Post-condition: the user will have created a new, empty account.

Alternatives

5.

    1. If the database does not confirm, the user will be returned to step 2.

Diagram



Fast-forward authentication flux

Main flux

Pre-condition: the user must be recorded.

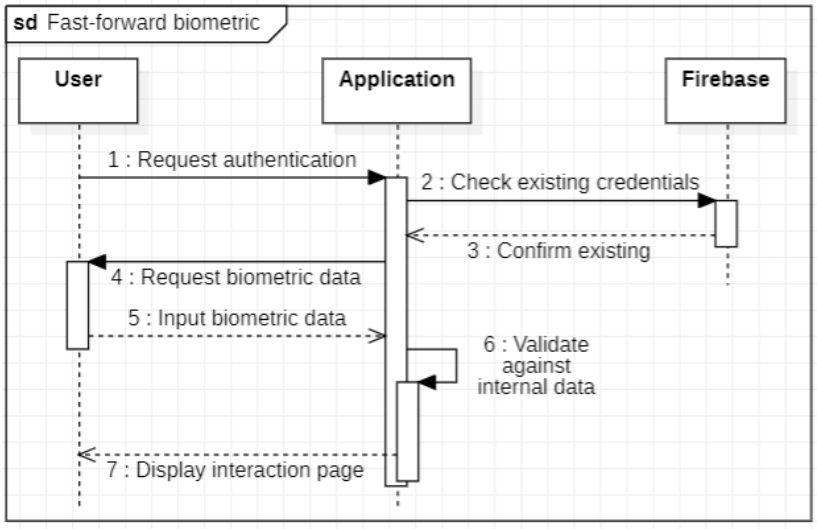
1. The system will send a request to the database about the last logged in user and whether the data is still valid.
2. The database will confirm the credentials that were last used.
3. The system will proceed to the "Biometric authentication" flux.

Post-condition: none.

Alternatives

2.
    1. If the database does not confirm the credentials, enter flux "Account specification".

Diagram, steps 1 – 4



Authentication flux

Main flux

Pre-condition: the user must not be already logged in or recorded, but be registered.

1. When starting the application, the user will be presented with the login screen: an email field, a password field, two "log in" buttons and a "register" button.
2. The user should complete the email and password fields and press one of the "Log in" buttons.
3. The system will check the data formatting and forward it to the external database.
4. The database will validate the data against its internal resources and forward a confirmation to the application.
5. The system will display a screen, depending on the button pressed.

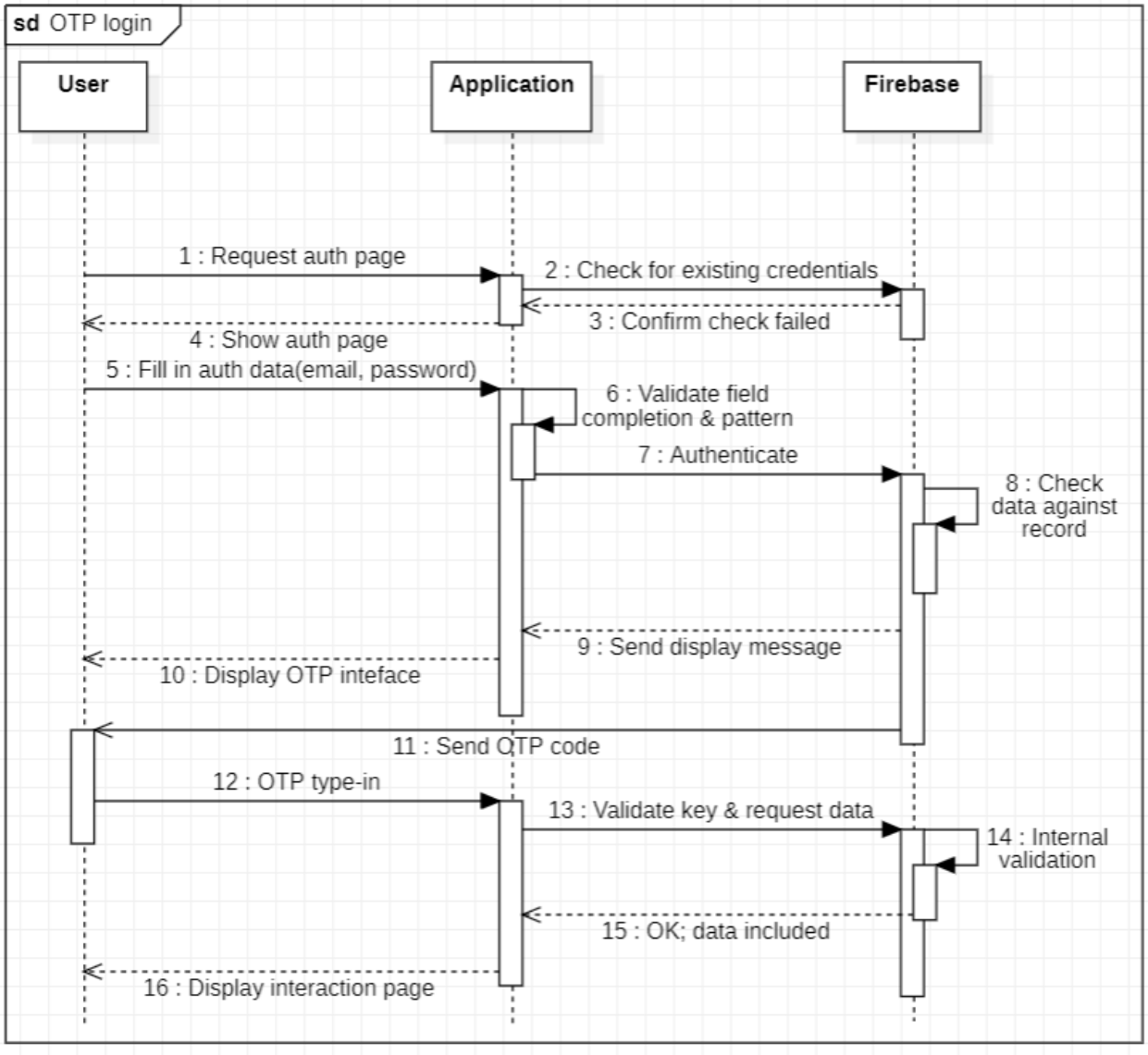
Post-condition: the user is recorded.

Alternatives

3.
    1. If the data format is incorrect (such as an email address lacking a '@'), the user will be returned to step 2.
4.
    1. If the database determines the data is incorrect:
        1. The database will send an infirmation to the application.
        2. The user will be then returned to step 2.
5.

1. If the button pressed at step 2 is "Log in via OTP", continue with flux "OTP sending".
2. If the button pressed at step 2 is "Log in via fingerprint", continue with flux "Biometric checking".

Diagram, steps 5 – 9



OTP sending flux

Main flux

Pre-condition: the user must be recorded but not logged in and have cell signal available.

1. The database will send an SMS containing the OTP to the user.
2. The system will display the "Input code" screen.

3. The user should input the code received.
4. The application will send the code back to the database.
5. The database will validate the code and send a confirmation message.
6. The user will then be able to select an account to view, or directly enter their account should they only have one.

Post-condition: the user is authenticated and able to perform standard operations.

Alternatives

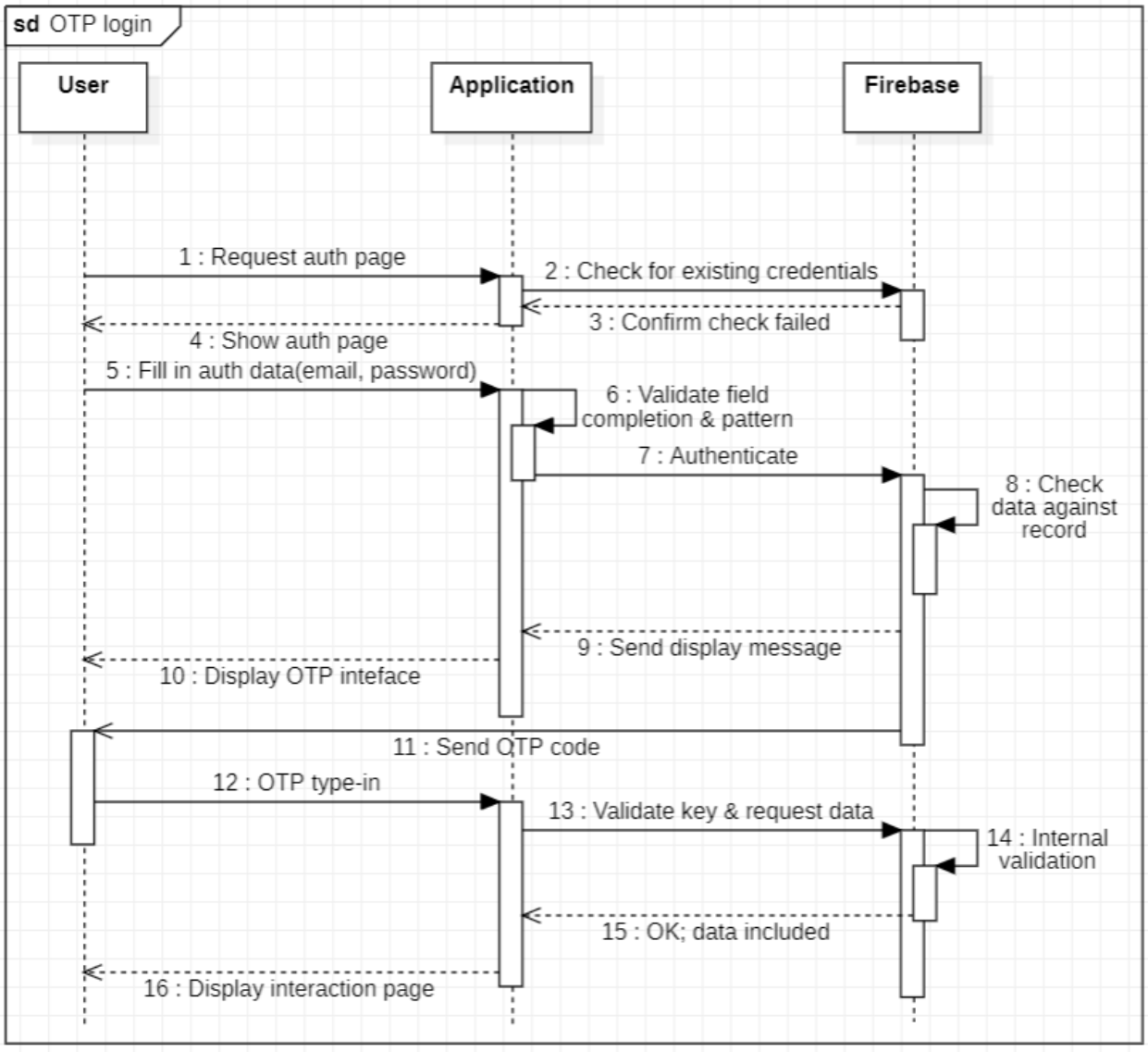
4.

1. If the code that was inputted is incorrect:

    1. The database will send an infirmation to the application.
    2. The database will send a new SMS, with a new code.
    3. The user will then be returned to step 1.

    Note: this redirection can take place at most 2 times.

Diagram, steps 12 – 16



Biometric checking flux

Main flux

Pre-condition: the user must be recorded but not already logged in and have a phone capable of reading fingerprints.

1. The system will display the "Input fingerprint" screen.
2. The user should input their fingerprint. (Methods to do so vary across devices.)
3. The application will perform internal checks to ensure the same person that unlocked the phone is entering the application.

4. The user will be able to select an account to view, or directly enter their account should they only have one.

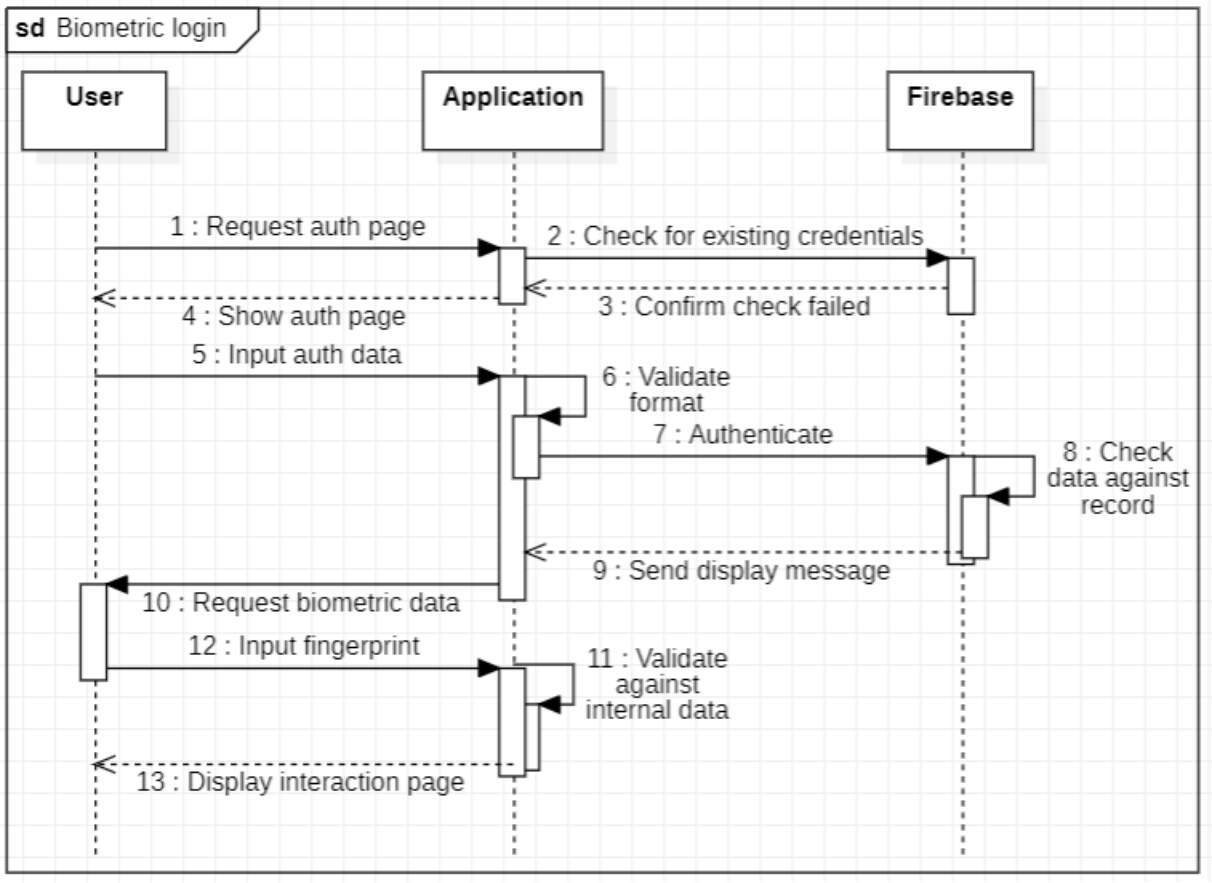
Post-condition: the user is authenticated and able to perform standard operations.

Alternatives

3.

1. If the application does not recognize the fingerprint, return to step 2. (This can happen 4 times at most.)

Diagram (steps 4 - 7)



Statistics flux

Main flux

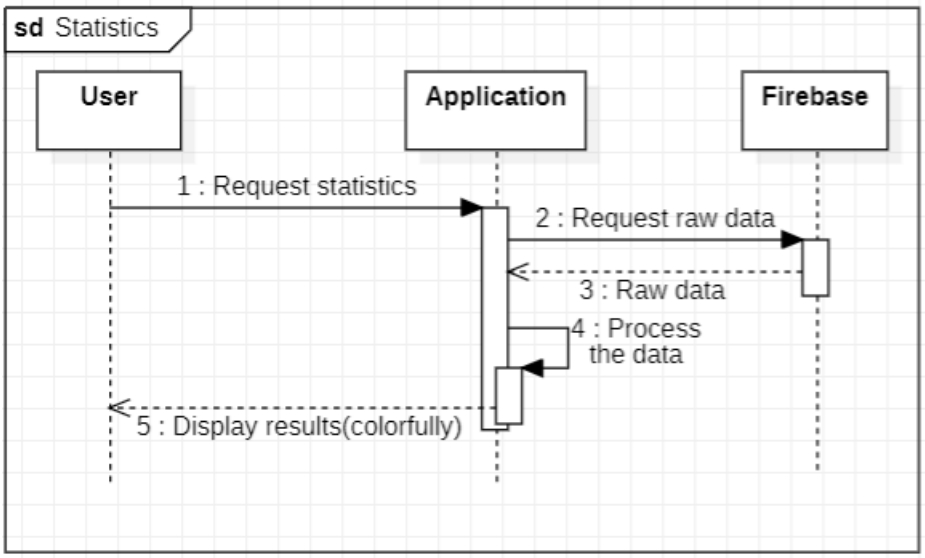
Pre-condition: the user must be capable of standard operations.

1. The user will press the "Statistics" button.
2. The system will request the recent transactions from the database.
3. The database will reply with the data.
4. The system will process the data and display it. Post-condition: the user will see the desired result.

Alternatives

None.

Diagram



Transfer (transaction) flux

Main flux

Pre-condition: the user must be capable of standard operations.

1. The user will press the "New transaction" button.
2. The system will request recipient details from the user.
3. The user should complete said details, most importantly the IBAN of the account.
4. The system will send a query to the database confirming all the details.
5. The database will send a confirmation of the details.
6. The system will prompt the user to enter any other transaction details, notably the amount of moeny transferred.
7. The user should complete said details.
8. The system will confirm the details with the database in order to validate the transaction.
9. The database will confirm the details.
10. The system will send the transaction in full for recording and processing, along with update requests.
11. The database will update itself and confirm the transaction taking place.
12. The system will relay the confirmation to the user and return to the main screen.

Post-condition: the transaction will have been done.

Alternatives

5.

1. If the database does not confirm the details of the account, return to step 3.
2. If the account found is in a different currency:
    1. The database will send the exchange rate of the two currencies along with the confirmation.
    2. The flux proceeds to step 6 as normal.
    3. The system will display the exchange in the second "details" page.
    4. The flux proceeds to step 10 as normal.
    5. The system will compute the update queries according to the exchange rate.
    6. The flux then proceeds as normal.

8.

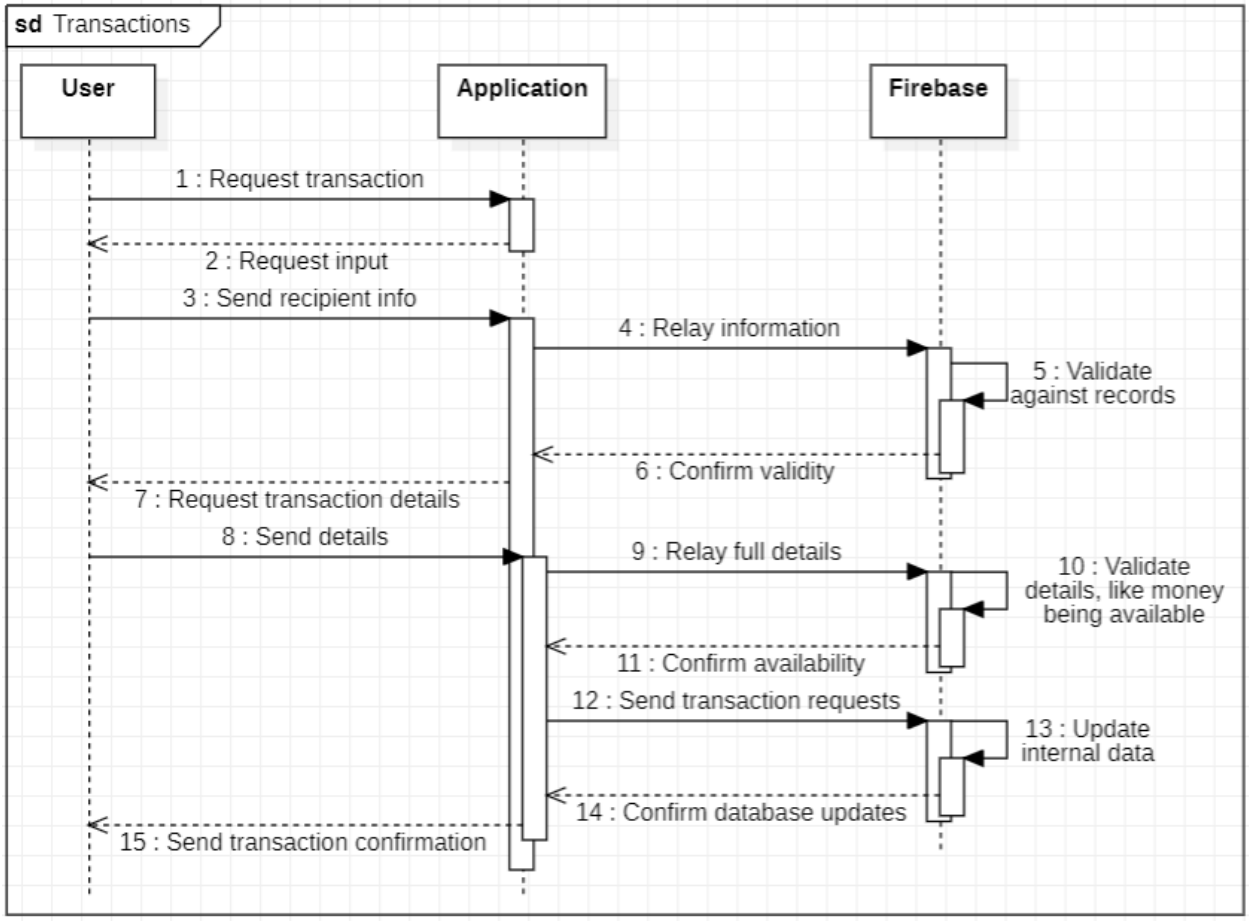
1. If the database does not confirm the details of the transaction (such as not enoug money being avaliable), return to step 6.

11.

1. If the database does not confirm the update process:
    1. Display an error message and a "continue" button.
    2. The user should press the button.
    3. The system will exit the flux and return to the main screen.

Note: the postcondition will not be fulfilled.

Diagram



Account extract flux

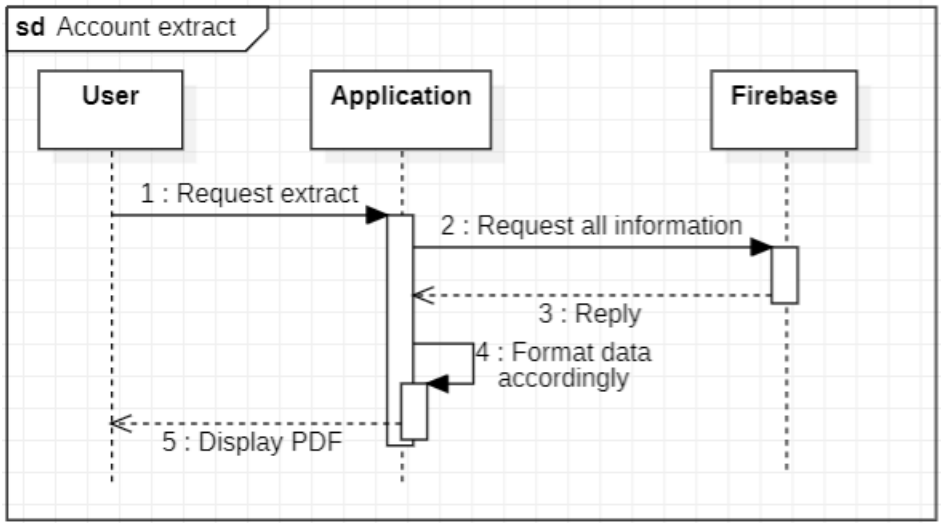
Main flux

Pre-condition: the user must be capable of standard operations.

1. The user will press the "Account extract" button.
2. The system will request recent transaction data from the database.
3. The database should relay the data to the system.
4. The system will process the data into the standard viewing format, in a widely recognized electronic format.
5. The system will save the data to the filesystem and display it to the user.
6. The system will return to the main screen.

Post-condition: the user will have a copy of their recent transactions for viewing and printing.

Diagram



Account deactivation flux

Pre-condition: the user must be capable of standard operations, have the account be empty and have at least two accounts.

1. The user will request deactivation of an account.
2. The system will check whether the user still has currency in their account with the database.
3. The database will reply with a confirmation.
4. The system will relay the appropriate updates to the database.
5. The database will update itself and send confirmation.
6. The system will display success, and the user will enter their account should they only have one, or be able to select an account to view otherwise.

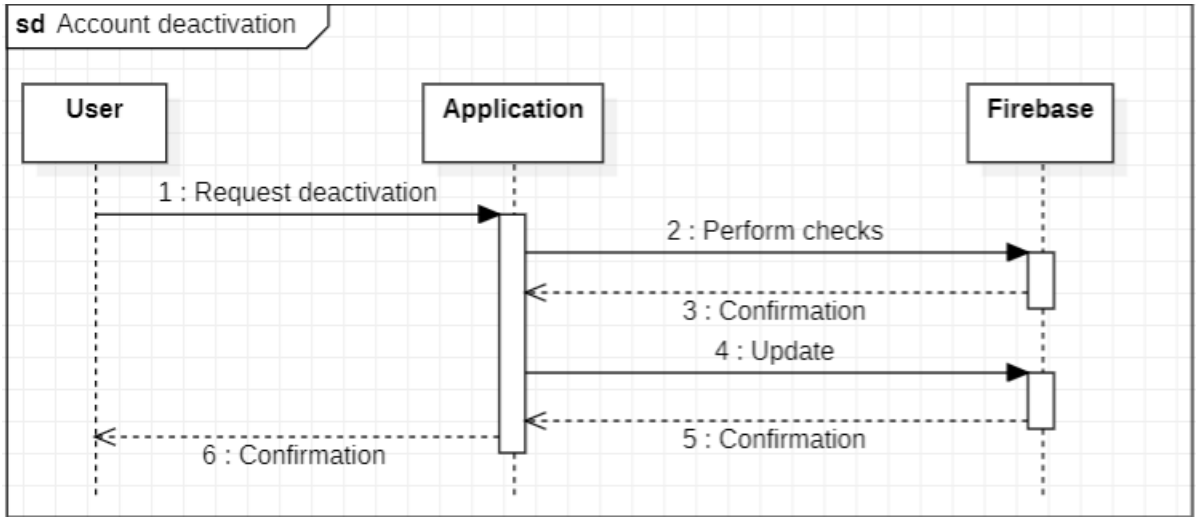
Post-condition: the account will be deactivated, remaining in the database only for tracing purposes.

Alternatives

3.

    1. If the system does not receive the confirmation:
        1. The system will display an error message.
        2. The system will exit the flux, returning to the main screen.

Diagram



User deactivation flux

Pre-condition: the user must be capable of standard operation.

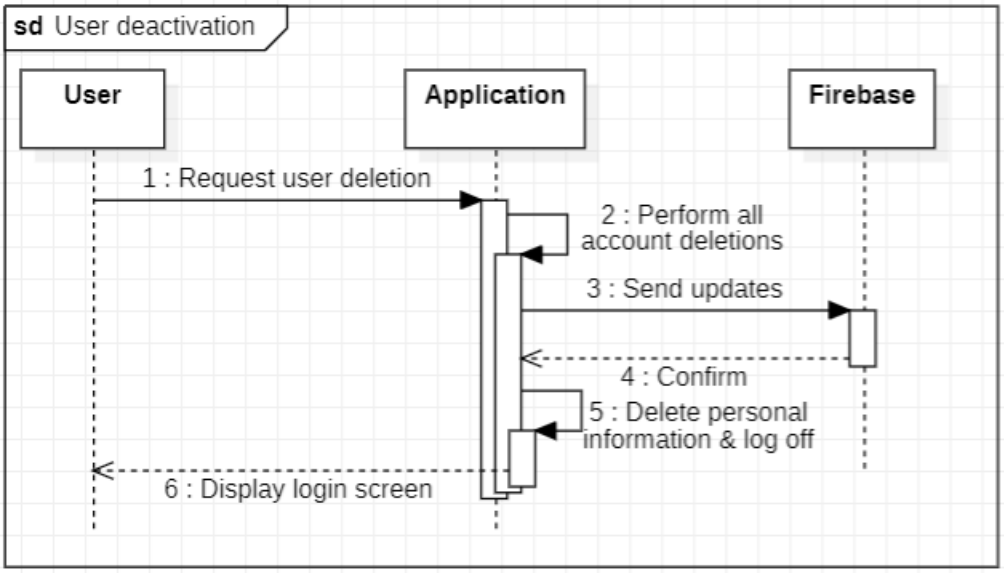
1. The user will request their deactivation.
2. For each account the user has active, steps 2 - 3 of the "Account deactivation" flux will be taken, disregarding the third precondition.
3. For each account the user has active, steps 2 - 5 of the "Account deactivation" flux will be taken, disregarding the third precondition.
3. The system will relay updates to the database requesting deactivation of the user.
4. The database will update itself and relay confirmation.
5. The system will display success and then the login screen.

Post-condition: the user will only be registered in the system for tracing purposes and will no longer be able to login.

Alternatives

2.
    1. If the system does not receive the confirmation, follow branch 3.1 of the "Account deactivation" flux.
3.
    1. If the system does not receive the confirmation, follow branch 3.1 of the "Account deactivation" flux.

Diagram



Deposit creation flux

Main flux

Pre-condition: the user must be capable of standard operation.

1. The user will request the creation of a deposit.
2. Proceed with steps 1 - 5 of the "Account creation" flux.
3. The system will retain the IBAN of the new account.
4. Proceed with steps 6 - 11 of the "Transfer" flux.
5. The system will display a confirmation and return to the main menu.

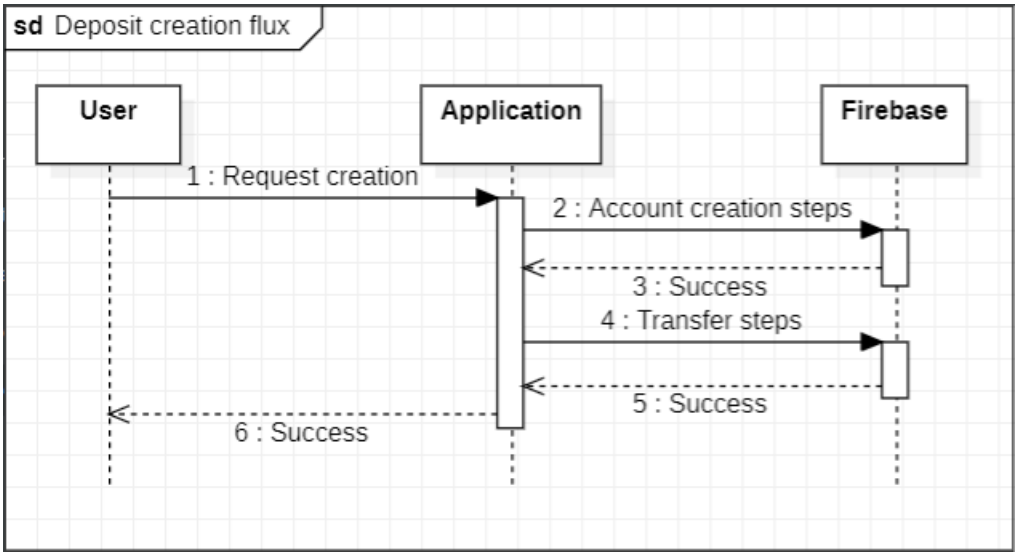
Post-condition: the user will have created a new account dedicated to deposits.

Alternatives

2.
    1. If any exception happens, follow the corresponding steps in the "Account creation" flux.

4.
    1. If any exception happens, follow the corresponding steps in the "Transfer" flux.

Diagram


sd Deposit creation flux
User
Application
Firebase
1 : Request creation
2 : Account creation steps
3 : Success
4 : Transfer steps
5 : Success
6 : Success