# GrabOff – an offline peer to peer (p2p) mobile money transfer

## Problem

The current payments applications rely heavily on the stable internet connection which makes it difficult to tap onto markets where internet is not stable.

## Solution

The solution is to provide an offline version of wallet in place of a single centralized wallet, which can take the responsibility of money transfers without the internet during transactions.

## Detailed Architecture

The architecture contains the following concepts:

Online wallet

Offline wallet

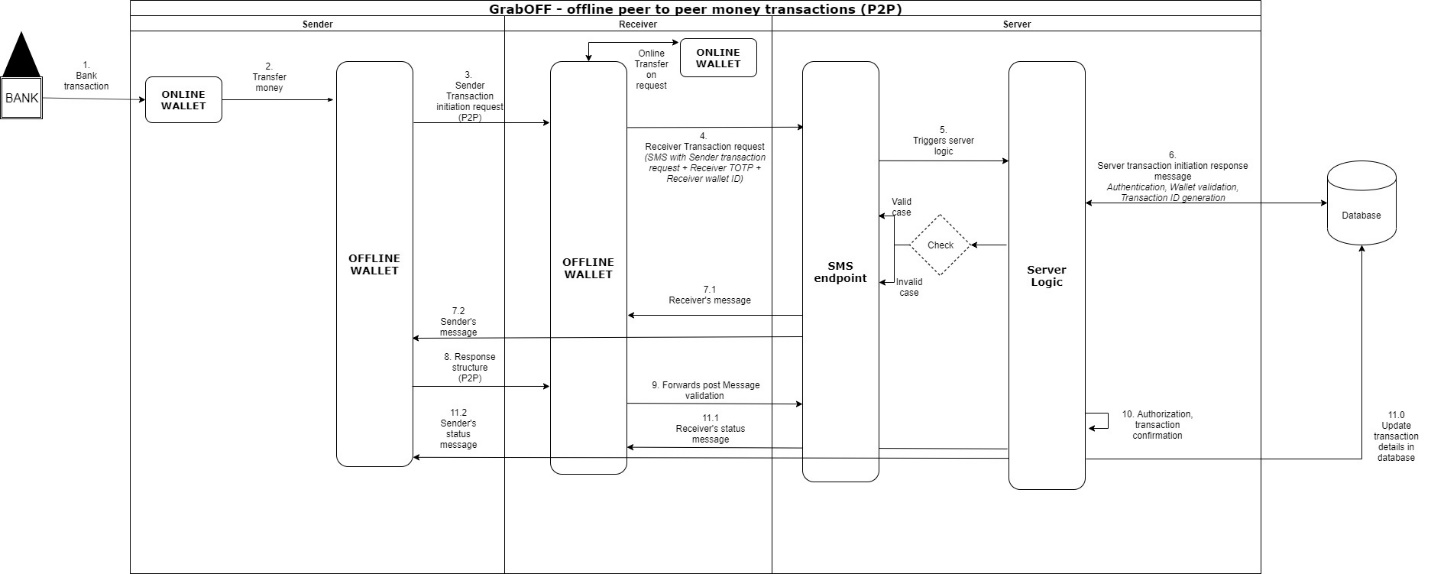
Client based authenticator (Time based OTP)

Peer to Peer WIFIDIRECT connection

Service: (SMS Client + Server logic)

Database

## Sequence diagram:



## Request Response Structures:



## Workflow

### Money addition into wallet

The application has two wallets:

Online wallet: Sender adds the money into the wallet via Bank

Offline wallet: Money can be loaded from the online wallet for offline transactions

### Offline P2P transaction initiation

Here the sender initiates the transaction by creating a WIFI direct connection

Example:

3\_sender\_transaction\_initiation\_request:

{

sender\_offline\_wallet\_id: 'xaxasdcadscsdsf1',

sender\_transaction\_id: 'qweq1131sqda21', <-- auto generated at sender

sender\_totp: 789456,

requested\_amount: 500,

}

### Receiver requests server

Once, the receiver receives the request from sender via P2P, server sends the following request to the server via SMS.

Example:

4\_receiver\_transaction\_request:

{

sender\_transaction\_initiation\_request: <3\_sender\_transaction\_initiation\_request>,

receiver\_offline\_wallet\_id: '456445sdadscsdsf1',

receiver\_totp: 964741

}

### Server logic trigger

Once, the server receives the SMS from client, the server logic gets triggered and the following events happen.

Authentication: Both sender and receiver are validation using the wallet ids.

Request validation: The request is validated by the checking the feasibility

Transaction ID generation: A unique transaction ID is generated at the server

Status Generation: A request status is generated

### Response to Sender and Receiver

Both sender and receiver would receive the following responses.

Receiver:

6\_server\_transaction\_initiation\_reponse:

{

server\_transaction\_id: 'as456a484sa5d',

status: {

authentication\_status: SUCCESS | FAILURE

}

}

Sender:

{

OTP: <OTP>

}

### Sender communicates to Receiver with payload containing OTP + PIN

8\_response\_structure:

{

server\_transaction\_id: 'as456a484sa5d',

sender\_OTP: 4548941,

sender\_wallet\_pin: 5848

}

### Transaction Authorization + Confirmation

TEAM NAME: CRONS

TEAM MEMBERS:

Sai Manohar Boidapu

Manikant Kella

Viswanath Reddy Sanepalle