

CIB Digital Tech – Front End Assessment

TIME FOR THIS ASSESSMENT: 4 HOURS

You are writing a small part of a banking system for Acme Bank:

- Acme bank runs only 2 types of accounts, a **Savings** and **Current** account.
- Most of the behaviour in these 2 types of accounts is very similar
- For the purpose of this exercise, we will not look at all the functionality, we will only implement the "withdraw" and "balance".

Savings Account:

A savings account must have a minimum balance of greater than 0 to perform a **withdraw Current Account**:

A **current** account can have an **overdraft** limit (the maximum overdraft limit allowed on a **current** account by Acme bank is **R500.00**

This means that a current account can have both positive and negative for withdraw.

Implementation guidelines:

Your job is to write a basic front-end implementation of **withdraw** and **balance**. The code needs to work correct at a basic level. It does not have to be 100% perfect in terms of catering for live environment.

Please feel free to comment areas that you may implement in more detail in a live environment.

Languages:

- Javascript
- HTML5
- Typescript (optional)
- ESNext (optional)
- CSS / SASS (optional)

Javascript Frameworks (choose at least one)

- Angular 4+
- React
- Vue

It is completely acceptable to work in any other framework you feel comfortable with.

Best of luck!

CSS frameworks (optional)

- Bootstrap
- Foundation
- Material

Business Rules

- One cannot withdraw more than the balance on savings accounts
- The maximum **overdraft** limit allowed on **current** account is R500
- Display alert('Success') on withdraw button click
- Display inactive withdraw button e.g. saving account equals -R20.00
- Calculate the **balance** for all accounts

General

- Are all the requirements set above met?
- Can the project run e.g. **npm run**?
- Is the page working without any JS errors?

Data source:

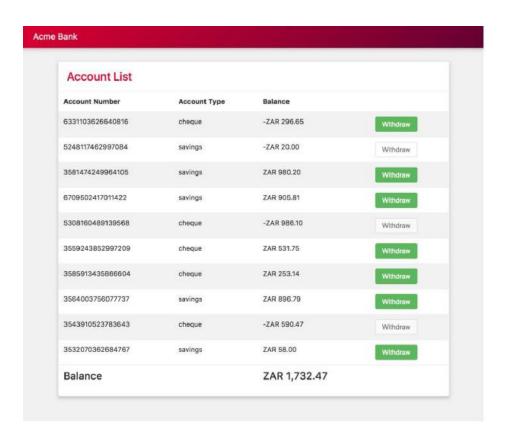
Account List (GET) http://localhost:8080/api/accounts

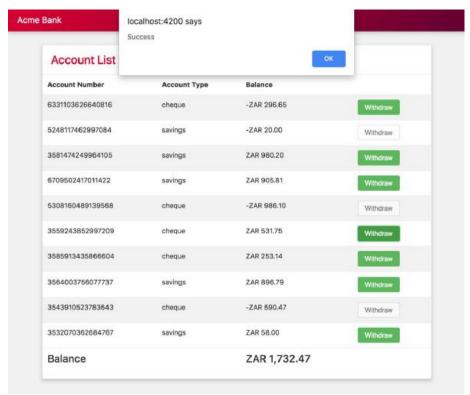
You will be handed a node server with the assessment which you will need to start the API server. Download API server: https://github.com/cibfrontend/mock-api-server

```
Example of JSON
[
{
    "account_number": "6331103626640816",
    "account_type": "cheque",
    "balance": "-296.65"
},
{
    "account_number": "5248117462997084",
    "account_type": "savings",
    "balance": "-20.00"
}
```

Use Case:

The wireframes below act as a guideline, it is entirely up to you to interpret this spec in any way you see fit





Kindly share the link with your completed assessment with us within 3 working days upon receiving the assessment.