The Challenge

We'd like you to develop a "round-up" feature for Starling customers using our public developer API that is available to all customers and partners.

For a customer, take all the transactions in a given week and round them up to the nearest pound. For example with spending of £4.35, £5.20 and £0.87, the round-up would be £1.58. This amount should then be transferred into a savings goal, helping the customer save for future adventures.

Time:
Documentation or source code comments to help your reviewer orient themselves would also be appreciated! There is no hard deadline to submit our tech task, but most applicants get it back to us within 7 days.

Submitting: Share a private github repo with the user starlingtechtest - ideally we'd like to avoid lots of public solutions. Please include your name in the repository description, if it is not obvious from your github profile, so we know whose code we are looking at!

If you have questions, chat with us on slack.

API calls

To make this work, the key parts from our public API you will need are:

1. Accounts - To retrieve accounts for the customer
2. Transaction feed - To retrieve transactions for the customer
3. Savings Goals - Create a savings goals and transfer money to savings goals

We do expect to see your working here: please do not use any of the libraries out there which provide an sdk for interacting with our api.

What to build

Pick your favourite platform to develop for. If you have experience with multiple languages that's fantastic - you only need to submit for one and your choice won't impact potential roles with us - we have plenty of engineers who work across platforms / learn news ones here!

Here's what we imagine you will build for each platform, but you can diverge from this so long as you are prepared to explain what you've done and why:

- **Java:** Main method which runs the round up directly, or main method which starts an embedded web server with a REST resource we can invoke to trigger the round-up.

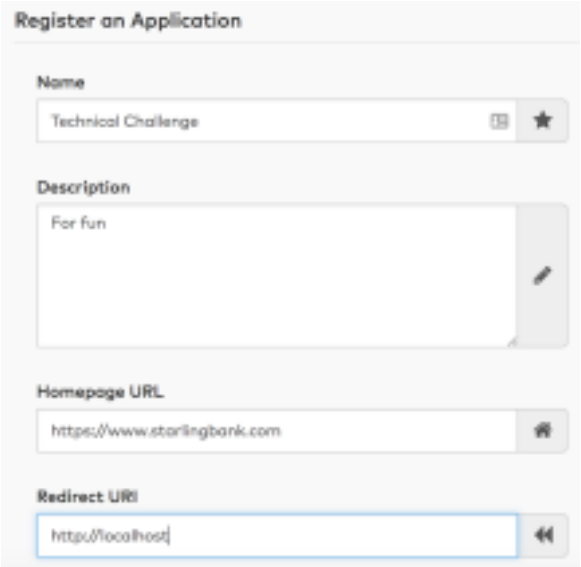
- **Android:** Simple application in Java or Kotlin which displays the amount to round up for a week with a button to perform the transfer.
- **iOS:** Simple iOS application written in Swift and using UIKit. It should display the amount to round up for a week of transactions with a way to perform the transfer into a saving goal.
- **JavaScript:** Simple web page which displays the amount to round up for a week with a button to perform the transfer. We use a lot of React & Redux but pick the libraries / frameworks you are comfortable with.

Getting started

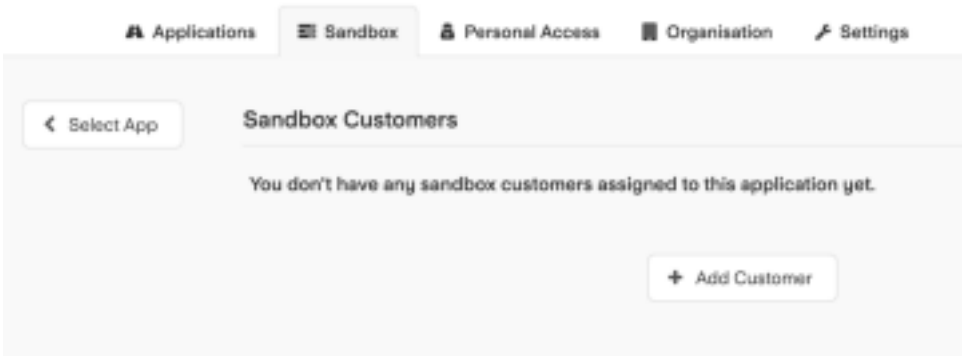
Don't worry about reading the top half of the API documentation - we aren't asking you to build out an OAuth implementation! To jump start to the point of being able to make API calls:

1. Sign-up for a Starling developer account and verify / secure your account.

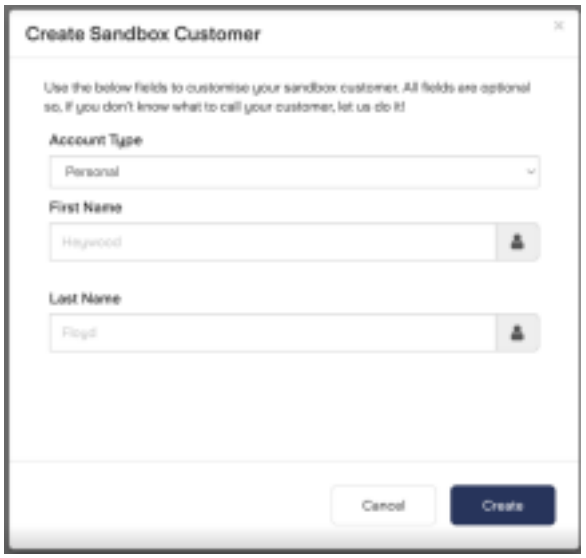
2. Create an API application (you can use any old URLs as they won't be used for local development). Applications are how we track API keys and usage.



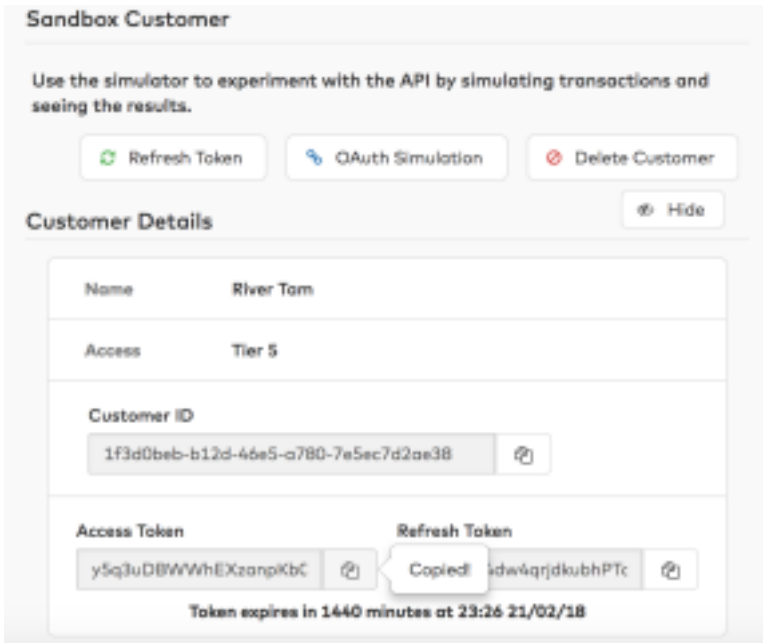
3. Create a sandbox customer for your application. This creates a new active customer account in our test bank, and simulates them granting API access to your



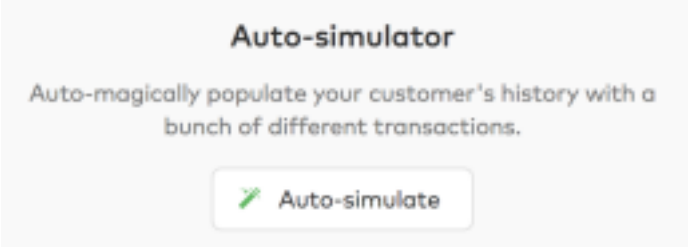
application.



4. Copy the **access token** for the customer. This is effectively a username / password for the customer's account, specific to your application and the level of access they have chosen to give you.



5. Click the auto-simulate button for the customer. This makes about 30 transactions on the customer account to give a reasonable history to play with.



That's it! You can now make API calls for the customer with your favourite HTTP client.

Base URL for resources: https://api-sandbox.starlingbank.com
Required headers: Accept: application/json
Authorization: Bearer {yourAccessTokenFromAbove}
User-Agent: Your Name

Example with curl:
curl --get https://api-sandbox.starlingbank.com/api/v2/accounts -H "Accept: application/json" -H "Authorization: Bearer {yourAccessToken}"

Example with Postman: