Known problems

Due to Expo, I was unable to use Google/Facebook/Apple logins. This is because firebase requires specific code per platform, so I would need to detach expo and change things for IOS and Android separately. The biggest problem that comes with this is that I do not have a Macbook or Android phone, meaning I wouldn't be able to test with any real device.

To combat this for the time being, I created a user selection page where there a different examples of different types of users.

Some functionalities have not been fully implemented or are hardcoded for the time being, For example, the posts further down the page are hardcoded for the time being but can easily come from a simple call to the service layer which can pull posts from mongodb
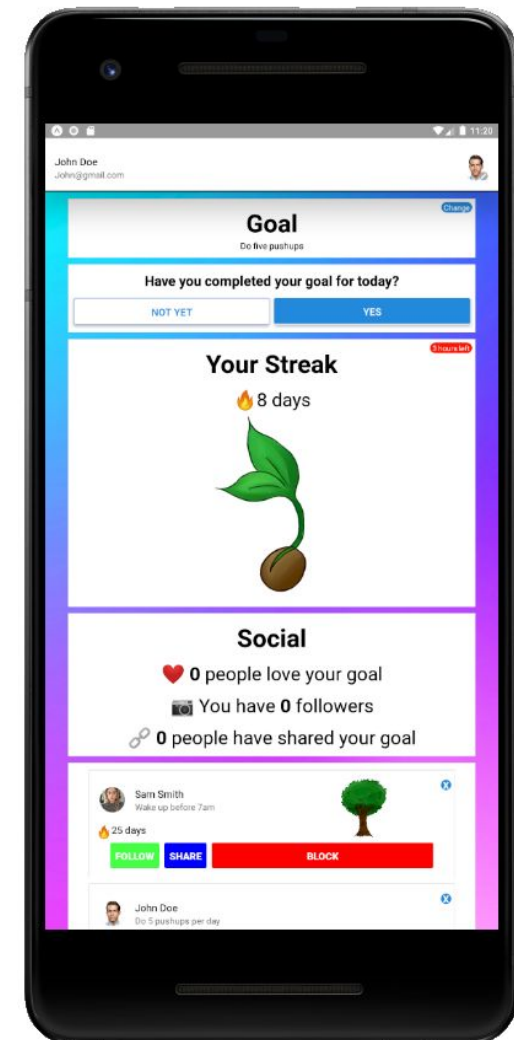
Note:

IOS testing was done on a real IPhone XR running IOS 12.4

Android testing was done on a Nexus S API 29

Moleculer running locally, Mongodb running on external server

Highlighted cells represent failed tests

Client	AIT	Date	01/02/2020
Program Name	Sapling	Supervisor	Aeron Chamara

Test case #	Description	Test Script	Results	
			Expected	Actual
1. Moleculer				

1.1	Correct Authentication with tokens	- Open service-layer
		- Run "npm run dev"
		- Open Postman
		- Make call to http://localhost:3000/api/userValidation/validate
		- Parameters sent will be
		- {userID: "0c814ba9-6642-4695-8439-4b27a02bf0cf", loginToken: "ae89f698-d5d0-4302-aeb8-69a71077e0eb"}
		
		-
			An object of the users data will be returned	{
				    "code": 200,
				    "valid": true,
				    "newUser": false,
				    "user": {
				        "_id": "5e343af4e00dc13874c265a8",
				        "userID": "0c814ba9-6642-4695-8439-4b27a02bf0cf",
				        "loginToken": "ae89f698-d5d0-4302-aeb8-69a71077e0eb",
				        "fname": "John",

				``
				        "lname": "Doe",
				        "goal": "Do five
				pushups",
				        "streak": 8,
				        "goalCompleted":
				false,
				        "goalCompletedTime":
				"2020-01-31T16:13:22.646Z",
				        "following": [],
				        "followers": [],
				        "loving": [],
				        "lovers": [],
				        "sharedby": []
				    }
				}
				``

1.2	Authentication with incorrect token	- Open service-layer
		- Run "npm run dev"
		- Open Postman
		- Make call to http://localhost:3000/api/userValidation/validate
			Response will have a valid: false	Response was {
				``
				    "code": 200,
				    "valid": false,
				    "newUser": false,
				    "user": {
				``

Client	AIT	Date	01/02/2020
Program Name	Sapling	Supervisor	Aeron Chamara

		- Parameters sent will be - {userID: "0c814ba9-6642-4695-8439-4b27a02bf0cf", loginToken: "someincorrecttoken"} -		"_id": "5e343af4e00dc13874c265a8", "userID": "0c814ba9-6642-4695-8439-4b27a02bf0cf", "loginToken": "ae89f698-d5d0-4302-aeb8-69a71077e0eb", "fname": "John", "lname": "Doe", "goal": "Do five pushups", "streak": 8, "goalCompleted": false, "goalCompletedTime": "2020-01-31T16:13:22.646Z", "following": [], "followers": [], "loving": [],

| | | | ```
        "lovers": [],
        "sharedby": []

    }

}
```<br><br>This is a huge problem because it said valid:false but it still returned all the user details.<br>Fix: Move where mongodb is called |
| 1.3 | Authentication with incorrect Token **AFTER FIX** | - Open service-layer<br>- Run "npm run dev"<br>- Open Postman<br>- Make call to http://localhost:3000/api/userValidation/validate<br>- Parameters sent will be<br>- {userID: "0c814ba9-6642-4695-8439-4b27a02bf0cf", loginToken: "someincorrecttoken"}<br>- | Response will have a valid: false | ```
{
    "code": 200,
    "valid": false,
    "newUser": false
}
``` |

Client	AIT	Date	01/02/2020
Program Name	Sapling	Supervisor	Aeron Chamara

1.4	Authentication with a userLogin that doesn't exist	- Open service-layer
		- Run "npm run dev"
		- Open Postman
		- Make call to http://localhost:3000/api/userValidation/validate
		- Parameters sent will be
		- {userID: "someNewUserID", loginToken: "sometoken"}
		-
		-
			Response.valid will be false but response.newUser will be true	`{`
				`    "code": 200,`
				`    "valid": false,`
				`    "newUser": true`
				`}`

1.5	Adding in a new user	- Open service-layer
		- Run "npm run dev"
		- Open Postman
		- Make call to http://localhost:3000/api/userValidation/firstInsert
		- Parameters {userID: "SomeID",loginToken:"SomeToken",fname: "Foo",lname: "Bar",goal: "Do a test", }
		-
			Should return a new user object with extra information. This information will also be saved to mongodb	`{`
				`    "code": 200,`
				`    "user": {`
				`        "userID": "someID",`
				`        "loginToken": "someToken",`
				`        "fname": "Foo",`
				`        "lname": "Bar",`
				`        "goal": "Do a test",`
				`        "goalCompleted": false,`
				`        "goalCompletedTime":`
				`"2020-02-01T12:57:24.264Z",`
				`        "streak": 0,`
				`        "following": [],`
				`        "followers": [],`

Client	AIT	Date	01/02/2020
Program Name	Sapling	Supervisor	Aeron Chamara

<table>
<tr><td></td><td></td><td></td><td>

```
        "loving": [],
        "lovers": [],
        "sharedby": []
    }
}
```

</td></tr>
<tr><td colspan="4" align="center">**2. Application**</td></tr>
<tr>
<td>2.1</td>
<td>Screen Navigation on **IOS IPhone XR** real device</td>
<td>
- Open Application project
- Run "npm run start"
- In the metro bundler browser window, scan the QR code and expo will download the app bundle
- Click the Continue with Google button
- Observe results
</td>
<td>The screen should navigate to the user picker screen</td>
<td>The screen navigated to the user picker screen</td>
</tr>
<tr>
<td>2.2</td>
<td>Screen Navigation on **Android Nexus S** emulator</td>
<td>
- Use gradle to run an android emulator
- Open Application project
- Run "npm run start"
- In the metro bundler browser window, click the run on android
</td>
<td>The screen should navigate to the user picker screen</td>
<td>The screen navigated to the user picker screen</td>
</tr>
</table>

Client	AIT	Date	01/02/2020
Program Name	Sapling	Supervisor	Aeron Chamara

		- device/emulator button - Click the Continue with Google button - Observe results			
2.3	"Signing in" as an existing user on **IOS IPhone XR** real device	- Open Application project - Run "npm run start" - In the metro bundler browser window, scan the QR code and expo will download the app bundle - Click the Continue with Google button - Select John Doe - Test to see if goal is the same as in mongodb	The goal should be "do five pushups"	When clicking on John Doe, the screen seemed to freeze before an alert saying "Something went wrong, try again later" Problem: The service-layer was running, however since the phone is a different device, I couldn't use localhost and needed to use the IPv4 address of my Wireless LAN adapter Wi-Fi	
2.4	"Signing in" as an existing user on **IOS IPhone XR** real device **AFTER FIX**	- Open Application project - Run "npm run start" - In the metro bundler browser window, scan the QR code and expo will download the app bundle - Click the Continue with Google button - Select John Doe	The goal should be "do five pushups"	The goal said Do five pushups	

Client	AIT	Date	01/02/2020
Program Name	Sapling	Supervisor	Aeron Chamara

		- Test to see if goal is the same as in mongodb		
2.5	"Signing in" as an existing user on **Android Nexus S** emulator	- Use gradle to run an android emulator - Open Application project - Run "npm run start" - In the metro bundler browser window, click the run on android device/emulator button - Click the Continue with Google button - Select John Doe - Test to see if the goal is the same as in mongodb	The goal should be "do five pushups"	The goal said Do five pushups Note: Due to the emulator running locally, I did not have the same problem that I did with IOS
2.6	Deleting one of the social posts on **IOS IPhone XR** real device	- Open Application project - Run "npm run start" - In the metro bundler browser window, scan the QR code and expo will download the app bundle - Click the Continue with Google button - Select John Doe - Scroll to social posts	The page should update and the post should be gone	The post was gone. However due to the posts being hardcoded, it would just come back if the user were to close and reopen the application

		- Click the x button and observe results		
2.7	Deleting one of the social posts on **Android Nexus S** emulator	- Use gradle to run an android emulator - Open Application project - Run "npm run start" - In the metro bundler browser window, click the run on android device/emulator button - Click the Continue with Google button - Select John Doe - Scroll to social posts - Click the x button and observe results	The page should update and the post should be gone	The post was gone.

Prepared by: Nick Hulley   01/02/2020