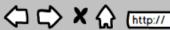




## Welcome to the test assignment

- Create a fake payment page following the wireframes on the next 2 pages.
- The Frontend must be done using Angular2 or ReactJS and the backend using a Python service.
- Please create good structures in the code, that will be part of the review.
- All actions on the page should be checked for correctness using the API.
- Ask questions if you have any or give ideas how to do things better.
- Write some documentation how to test the code on Mac or Linux PC.

Good luck!



Product scanner and Account

separate tools for which should be

connector are

billed.

Create an

appropriate

structure on the

backend side.





# Billing

### **Product scanner**

Number of products

Total number of products to scan.

#### Crawls per month

How often should the product be scanned per month.



Off (\$0)

 $$150 \times 2 = $300 / month$ 

100 (\$50)

2 (2x)

#### Account connector

Connected accounts

Off (\$0)

1 (\$10) 2 (\$17.50)

200 (\$75)

4 (3x)

5 (\$30)

500 (\$150)

8 (5x)

10 (\$50)

1000 (\$250)

30 (15x)

Total number of connected accounts



\$17.50 / month

Grand total: \$317.50 / month (3)



Cancel

Update plan

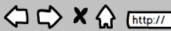
(a) When a slider is updated, the total amounts are uptated too in (1), (2) and (3).

(b) Every update is sent to the API to be validated.

When Update plan is clicked, the data should be sent to the backend API, wait for 2 seconds and return OK.

During the wait the buttons should be disabled to no allow double clicks.

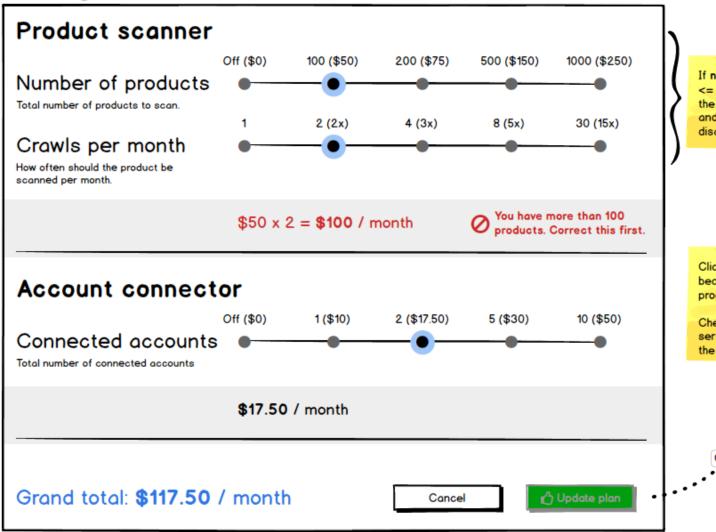








## Billing



If number of products <= 200 is selected, then the API returns an error and the Update plan is disabled.

Click here is not allowed because of the 100product limit.

Check must also be done server side if somehow the click happend.

