INFWEB01-D AND INFWEB21-D
Web Development – Exam 2 (second chance)

Year: 2020/2021

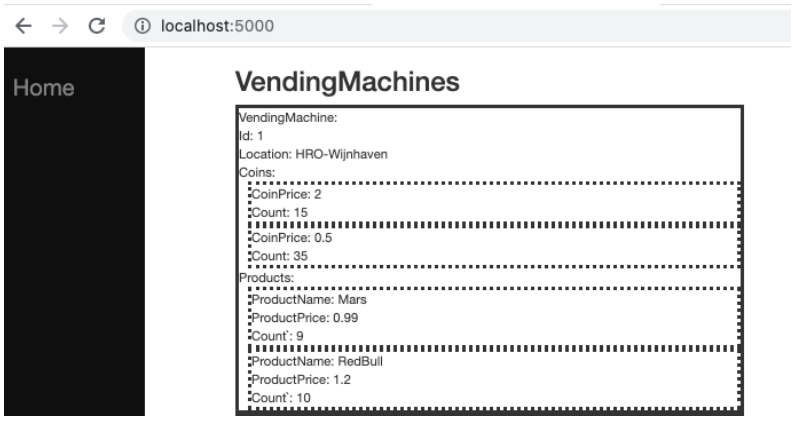
Student number: _____
Surname/Name: _____
Class: _____

Case description and the link:
Download the application from: https://github.com/hogeschool/development-web/tree/main/20-21/chance-2

The provided application for this exam is a simple vending machine preview application, where a user is able to visit the vending machines stored in the database (see below figure). In this application we see three entities: vending machine, product, and coin. A vending machine has a 1-to-Many relationship with coin and product, i.e., a vending machine contains many coins and stores many products.

In the figure below a vending machine located at the HRO-Wijnhaven contains:

- Coins: 15 coins each of value 2 euro, and 35 coins each of value 0.5 euro
- Products: 9 bars of Mars each of value 0.99 euro, and 10 cans of RedBull each of value 1.20 euro



Description of the view:
The layout of this application (see image above) contains a home-button. In home you can see the vending machines stored in the database. For each vending machine the lists of its stored products and coins are displayed underneath it. Note, to surround an element with a box (like in the figure above) you will need to add a styling to the container div. Example, *<div style={{border:"solid"}}>.. * will display a solid border around the elements contained in the div. You do not need to implement the exact above styling to score points. We will only score the functionalities.

We provide you an exam application, which is missing a series of code blocks. Each code block is mapped to specific points. To get the points for a specific block you need to fill it with valid/correct code. To pass the exam you need to acquire at least 5.5 points. The missing code blocks are marked in the exam application by a TODO comment[1]. In a TODO item you will also find the associated points. Note each TODO has a specific number necessary later for the submission (see submission rules in the following page). TODOs are distributed in the exam application as follows:

- Models
  - 4 TODOs 2 points
- Controllers:
  - 4 TODOs 3 points
- Views:
  - 7 TODO 5 points

[1] You can find all the TODOs by searching *TODO* in the search box of your IDE (in case of Visual Studio Code you can find the search box by clicking the magnifier icon on the left side of the IDE)

Submission form and rules
As you can see below (and in the application) each TODO is identified with a unique number. To accept your solution, you must write below the code you implemented in your exam application in the respective TODO slot. For example, if you wrote some code for TODO 5 in the exam application then you must copy it below in the TODO 5 slot. For this exam you do not need to upload the solution on N@tschool, but give these papers back to the attendant present in the room.
NB. Unreadable text in a TODO slot will give you 0 points for that specific TODO.

TODO 1 (0.5pt):

TODO 2 (0.5pt):

TODO 3 (0.5pt):

TODO 4 (0.5pt):

TODO 5 (0.5pt):

TODO 6 (0.5pt):

TODO 7 (1pt):

TODO 8 (1pt):

TODO 9 (1pt):

TODO 10 (1pt):

TODO 11 (1pt):

TODO 12 (0.5pt):

TODO 13 (0.5pt):

TODO 14 (1pt):

TODO 15 (1pt):