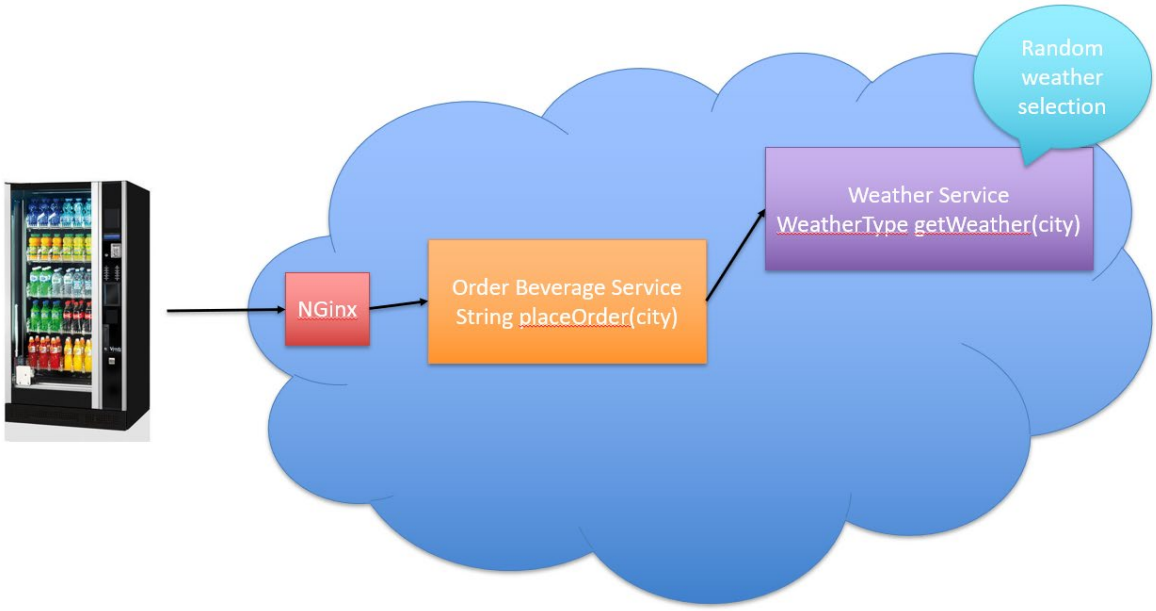
Lab assingment: Microservices

Download the vending machine application from github, build and run. You can run the application on AWS EC2 virtual machine. Following figure is showing the high-level application structure of current vending machine implementation.



In this assignment you will modify the application to add few additional features.

1. **Modify** the **weather-service** microservice:

   Replace the random weather selection mechanism with odd-even mechanism. With the new implementation, the weather type will be decided based on city id. If the city id is an odd integer, return WARM as the type of the weather. If the city id is even, return COLD weather type.

2. **Create** a new service "**beverage-preference-service**" and a function called **getBeverage(BeverageType btype)**:

   The function takes an input of BeverageType. BeverageType is an Enum declared in the vending-machine.thrift file. If the input parameter is BeverageType:Hot then randomly select a beverage between {"cappuccino", "latte", "espresso"} and return the name. Otherwise randomly select a beverage between {"lemonade", "ice tea", "soda"} and return the name.

3. **Update placeOrder(city)** function in order-beverage-service:

The Order-beverage-service's **placeOrder**() will call the weather-service's getWeather function as usual. However, after receiving a feedback from getWeather() RPC, it will call the **getBeverage(BeverageType btype)** to request the name of the beverage and send it back to the Nginx reverse proxy.

4. **Create** a docker-hub account and upload the image.
5. **Upload** the working code on github.
6. **Run** the script script/generate_request.sh from the client and store the result in the output.txt file.
7. Take a **screenshot** showing all the four services are running (docker ps -a) including the new "**beverage-preference-service**".
8. Write a brief **report** (one page) to describe the major steps you did and a brief reflection on what you learned.
9. **Submit** the output.txt, screenshot, docker-hub link of the image, Github link of the source code and the report.

Following figure is showing the high-level architecture of the updated application: