Dataviz project: Progress notebook

Notes for the first idea: real-time weather monitoring system for Morocco :

- Initial idea we had was about a weather monitoring solution for morocco.
- We started looking for APIs that could get us real-time data providing the location.
- Looked into multiple solutions like scrapping windguru, windy api (paid)…
- Found this API that returns important data such as temperature, pressure, humidity, wind speed & degree, cloud, sunset, sunrise etc..
- Started building a front-end using react and d3js of the map of morocco with most cities clickable and the ability to zoom on a selected province
- Research ways to display the country of morocco using d3js, implemented various solutions, but the most success we had was with using a geojson of morocco that is divided by provinces
- Looked multiple sources for data about Moroccan cities, with little to no luck. All the datasets we found didn't match the map (some city pointes were in ocean or in spain and algeria)
- A friend provided us with a json list of pretty accurate lats & longs of most moroccan cities.
- Started integrating this dataset with the map and it works
- Pushed the code to github and used github pages to host the result
- Github project
- Published result
- Started researching how to fetch data from rapid apis, how to set headers, what type the result is…
- Stopped with this idea and went to the other idea of monitoring smart cities

Notes for the second idea: real-time silo monitoring system using IoT devices :

- https://wokwi.com/projects/351600989253403224
- https://github.com/artiumrexbellator/smart_farm
- We decided to select mqtt,spark,influxDB architecture,due to the handling of data while streaming,since spark can separate streaming jobs among several workers.
- Some modifications are made on a spark project that was already working on docker (1 master and 3 workers).
- Editing docker project to adapt new libraries for jupiterLab (jupiter is used for execution).
- Fixing mismatched versions of spark libraries.
- Choosing emqx as a serving platform of mqtt broker through ssl.
- Searching for the appropriate mqtt jars for our spark version.
- Engineering data communication structure between Wokwi simulator and spark.
- The creation of an influxDb bucket on a cloud called farm to save measurements coming from spark https://europe-west1-1.gcp.cloud2.influxdata.com/.
- The retrieval of data from influxData cloud to grafana using flux language.
- The build of the dashboard on grafana.