**Overview:**

This is the final task of our internship which tests your skills in planning, decomposing, learning and implementing new technologies under conditions of limited time. Communication, clarification of requirements and thinking about different nuances and details are also very important. The work will be assessed comprehensively, 100% completion of all tasks is not the only acceptable result.

**Task:**

**Deadline:** ~14 days

**Business Requirements, 3 main tasks:**

- When you open a certain product you should see how many users are viewing the same product in real time.
- When product quantity is changed (submitted order, canceled order, manual admin update) everyone viewing this product or having this product in a shopping cart (if you implement shopping cart in task 3) should have updated quantity in real time.
- Introduce a shopping cart so when the user puts the product in the cart, he automatically reserves it for a certain time (should be configurable). Other users in real time should see how many units of the product are reserved. If after the reservation there are no available products left, then it is impossible to add this product to the cart.

**Rules and definitions**:

- We introduce three main terms: in stock quantity (total amount of products), reserved quantity (what's already added to a shopping cart) and available quantity (what's left from in stock after all reservations), so Available = In Stock - Reserved
- Purchase process: add to cart -> submit an order

- You can’t submit an order with quantity more than in stock quantity
- You can’t add product to a shopping cart with quantity more than available quantity
- Technologies and libraries are up to you, but I expect websockets to be used to solve these tasks and the best way to demo your work is to have a simple UI that shows all described scenarios. You can extend your UI solution from 5.6 task

**Steps:**

- Decomposition and estimation of all tasks. This is where you start and the requirement is to send me a detailed report about how you decompose requirements into small steps and how much time each step will take. (I would expect this on the 2nd-3rd day)
- Think about the db structure, list of methods and the way in which you will implement the websocket mechanism in an existing web project. (Some schemas and diagrams should be prepared for the final assessment)
- Start implementing the features and think carefully about what you can do in a limited period of time. Successful demo is an important part of delivering the new features and it’s better to have 2 fully tested and ready for demo features than 3 features that are not working properly and not ready for the demo.
- The work ends at 12.00 the day before the assessment, all code and all documents and diagrams must be sent to the reviewers. We don’t push code after this time and we don’t update documents.