

Assignment offered by dr. João Moreira

After reading the description below, if you still need to know more about this assignment, please, contact Dr. João Moreira through the email below:

j.luizrebelomoreira@utwente.nl

IMPORTANT: Once you reach an agreement with your supervisor, contact the track chair [Leon de Vries](#) (or [Wallace Corbo Ugulino](#)) to register your option and block this assignment for you (and avoid other students wasting their time inquiring about the same assignment). We use a FIFO (first-in, first-out) approach: the first student that communicates their option to us will get the assignment. Thanks!

A few tips on how to approach your potential future supervisor (Dr. João L. R. Moreira):

- (1) Your future supervisor may be looking for different kinds of research related to this topic. One good question you can ask your potential future supervisor is what kind of research is she/he accepting. For instance, is this a Systematic Review of Literature (SRL)? Is this an empirical research (meaning that you do some sort of trial that is linked to some sort of observation, experimental analysis, or others involving practical observation/testing)? Does it require software development? Does it require the configuration and use of existing software?

ASSIGNMENT DESCRIPTION

DevOps - Testing the scalability of Remote Patient Monitoring systems

Remote Patient Monitoring (RPM) is an approach to monitor patients continuously, by capturing different physiological data and making this information available to the healthcare team. This information can be used to detect diseases early and even in real-time, avoiding patient health worsening and premature deaths. In the last few years, we have developed an RPM Platform, named HDash (formally HealthDash), which the main objective is to provide all the necessary infrastructure to collect, store and process Personal Health Records (PHR) captured in RPM scenarios. The platform is split into 3 modules: an SDK (HDash SDK) that provides libraries to integrate new devices; a mobile client app (HDash App) with a mobile interface used by patients to collect data; and a cloud application (HDash Cloud), which stores and processes the collected information. The infrastructure to deploy and run HDash Cloud is based on Rancher, a container orchestration software, and the current configuration of the cluster is depicted below (Figure 1). The cluster is composed of three virtual machines, one of them used by Rancher to orchestrate services, and the other two used to run HDash Platform.

Besides being used in a real-world scenario, we have a limited number of users using it (~35). Testing the elasticity and scalability of HDash can bring insights about the necessary infrastructure to run it at a large scale, and the problems we will have doing that. We also want to test different communication protocols beyond HTTP, such as MQTT, COAP, and WebSocket, comparing the benefits and drawbacks of each one. It is required to analyze which approach is better (vertical vs horizontal scaling or combining them) to each component in the architecture, as well as which are the main variables, e.g., response time, requests/second, network usage, memory usage, time taken to execute several tasks in HDash Cloud (deserialization, validation, persistence, and response serialization) and performance with several users (1k, 5k, 10k, 100k user simultaneously connected).

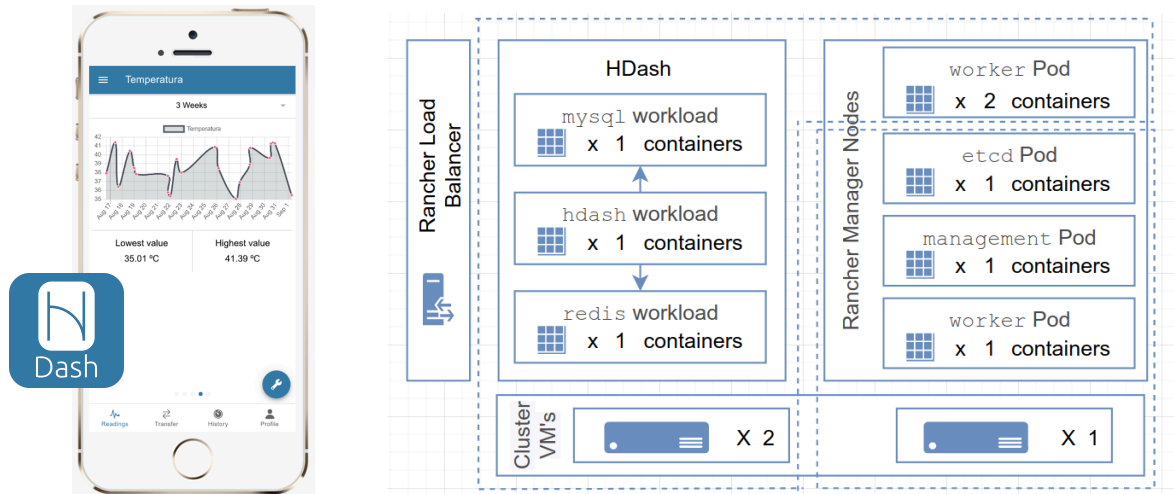


Figure 1. (a) HDash app, (b) current container cluster configuration running HDash Cloud.

This project consists of planning/executing/evaluating scalability tests to analyze the behavior of HDash Cloud, and also the effectiveness of the proposed infrastructure to run it. A literature review on scalability tests for this type of system should be conducted, as well as the definition of a scalability test plan for HDash that considers, for example, scaling up the components depicted in the picture and use tools to run load tests, e.g., JMeter, Locust and/or some other simulation tool. The tests should replicate user behavior while using HDash App, and cover records synchronization. The plan should be evaluated in a distributed infrastructure.