EECS6432: Containerized Orchestration of IoT Lighting Systems

Daniel Parreira
York University
Toronto ON, Canada

Pedro Casas
York University
Toronto ON, Canada

Shane Segal
York University
Toronto ON, Canada

Abstract

We have implemented a dockerized, automatically load-balancing IoT system and associated web-infrastructure for a networked smart-lighting system. Our system can be applied to public spaces with shared resources, potentially containing individual actors with competing goals - all while working to balance a global setpoint that best satisfies the goals of all independent actors in the system.

Created with horizontal scalability in mind, our software architecture is well-suited towards rapid expansion of both users and resources. Our embedded IoT devices are connected to our GraphQL API through a Kafka pub-sub system, ensuring easy scalability. Our various interaction methods (voice, web app, EEG Reader) connect through our Pub-Sub architecture where appropriate, which connects them to our orchestrated containerized server architecture. Our PID based load-balancer ensures that there are enough server resources available and that the requests coming from each available endpoint are satisfied as much as possible in the face of possibly competing goals.

1. Introduction

Public spaces with adaptive smart technologies present interesting problems regarding the satisfiability of concurrent requests for shared resources. We explore this problem in the context of a smart lighting system in a space such as a university library or public co-working space. Prior to the introduction of smart technologies, users might be limited to working in a globally lit environment, or a locally lit environment that did not effectively consider the preferences of those around them, or of the energy usage policy of the building or organization.

Our system enables authenticated and authorized users to make luminance-level requests about the local lighting level. Our system balances this request with the requests of spatially co-located individuals who have made prior requests. These requests can either be trivially compatible with each other (e.g co-located users making similar requests) or be incompatible (brightness preferences outside of 10-15%).

As IoT technology has only recently emerged as a powerful force in the technology space, there are many situations that are in a suboptimal state. We attempt to address one such situation through the application architecture presented here. Our solution works to combine the global needs of a building such as the overally energy usage or brightness level, as well as local needs of a user such as the local brightness level being optimal for the situation at hand. Local optimality for the system at hand can include such things as brightness for the time of day, color of the light produced. or appropriate brightness for the task. In addition, we also focus on accessible and innovative models of user interaction with the goal of reducing the barrier for use of our system by people of differing physical abilities.

The research challenges we faced include several disparate factors. The load balancing of our servers through adaptive software is an open problem in software engineering and one that we apply very modern techniques in software engineering in order to deal with it. We then have the challenge of balancing user-local preferences with potentially contradictory preferences of other co-located users as well as with global preferences that can be set on a location-wide basis. And finally, we have combined new and experimental human computer interaction techniques into an integrated technological solution.

2. Models, Architecture, and Algorithms

2.1. Models

When considering the models in our software system, it is useful to seperate them into the server software and the IoT software. Our server architecture is composed of a