Week 6 Status Report

SE 491: Team sddec19-24
DevOps Framework for IOT
Client / Faculty Advisor: Lotfi Ben Othmane

Team Members:

Yussef Saleh	*Team Leader*
Matt Bechtel	*Chief Designer / Engineer*
Brett Wilhelm	*Assistant Programmer / Engineer*
Chakib Ahlouche	*Assistant Programmer / Engineer*

Git repositories for extensions mentioned below:
https://git.ece.iastate.edu/sddec19-24/iot-deployment-manager/
https://git.ece.iastate.edu/sddec19-24/iot-deployment-service/

- **Weekly Summary**

This week we decided on our issue documentation structure, meaning how we will document milestones, tasks, and issues for the project. We decided on using gitlab's built in issue control, so we can document our project plan, step by step, within our version control service. This will provide convenient access to specific documentation about each piece of functionality that is built out.

Beyond our git setup work this week, we also began development on the actual extension for our project, the IoT Deployment Service and the IoT Deployment Manager. To kick off the development, we created both pieces as base express servers (express is a very simple framework for web servers written in NodeJS). Once that was done, we created a very basic file transfer protocol for the two services, it goes as follows: A get request is made to the IoT Deployment Service containing the filename to transfer from the IoT Deployment Service to the IoT Deployment Manager as a query parameter. The IoT Deployment Service (service) then reads the file from disc, and sends a post request to the IoT Deployment Manager (manager) with the file in the payload. The manager then receives the request, extracts the file from the payload, and saves it to the specified file store directory. This little example, the basic file transfer, is the precursor to a piece of functionality for our extension, the credential file and start script upload. The credential and start script upload, will, when complete, allow a user to upload their git credentials and a start script for the project housed in the given git repository (the one the credentials were given for) to the manager, via the service, which will allow the manager to pull down new code and spin it up when there are updates.

With the beginning of our extension coming together we also were faced with some new design decisions that we must think about for next week, such as ‘How will we verify that the service is what is making requests to the manager?’, ‘How will we securely communicate from the service to the manager, even after the services identity is verified?’. This will be something to think about next week.

- **Past Week Accomplishments**
  - Git setup
  - Successfully created the basic file transfer over http between the service and the manager
  - Created project structure for issues and milestones in GitLab

- **Pending Issues**
  - Yussef, Brett, and Chakib need to set up their local development environments and begin dev work

o Individual Contributions

Name	**Individual Contributions**	**Weekly Hours**	**Total Hours**
Yussef	Finished researching about Docker and the required software for this project and got all the previous code to make sense and configured it on the local machine. I also researched about a way to generate express server for future deployment.	6	21
Matt	Created repositories for both the service and the manager. Generated base express servers for both the service and the manager. Designed and created the file transfer protocol mentioned above in the weekly summary. I was able to successfully complete the task, and now, in our repository, both the service and the manager have routes to handle their piece of the file transfer (detailed above in the summary). When the file transfer portion was complete I decided to do some work to containerize our manager so that it may be easily tested as if it running on an IoT device. I was able to get the image in a runnable state, meaning that you can run the container, allow it to serve on a port, and then make a request to the service, and the service will transfer the file to the manager running on the docker container.	8	33
Brett	Looked at repository. Furthered local development setup. Got a better understanding of how the previous code worked.	5	19
Chakib	Checked out the deployment service and manager, installed node and played around with the service and manager for a bit, simply transferring files to test it	2	15

- **Plans for Next Week**
    - Now that the file transfer is working, we plan to add onto it so that a git credential and start script file may be transferred from the service to the manager, and in turn the manager will use the git credentials to pull down the given repository, and spin it up. For next week, a git file should be able to be uploaded, the file read, and the credentials pulled from it, and then the manager should also be able to pull down the code. Were unsure as to the design of this yet but it is likely we will just use an npm package for the git integration.
- **Summary of Advisory Meeting**
    - Discussed our future plans and how we will follow agile development. We demonstrated our basic file transfer from the service to the manager, and got feedback on it. We also discussed our plans for next week which we mentioned above.