Summary of the work done from week 17 (14.10.2019) to week 20 (10.11.2019)

During the seventeenth week of the internship I continued the previous week work of changing the HickariCP database connection pool size of Ballerina and ran few tests to check the performance of Ballerina Database Read with different connection pool sizes.
After that I ran the performance test by directly hitting the database with JMeter to measure the Database side latency. From that we can easily find the Ballerina side latency values. Then I started to work on the Bayesian implementation to tune MySQL database. I searched for a dynamic MySQL system variable that can be tuned online without restarting the server and found its bounds, then I implemented those bounds in the code. After that I researched all the different Dynamic system variables and other system variables that requires MySQL service restart. Then I worked on changing those variables using bash and finally created a bash script that can get the values of the variables and also set those variables. Finally I ran tests for tuning several parameters individually.

In the next week I continued the previous week tests for tuning some parameters. Then I analyzed the Open Banking Flight Recording files to find the performance bottlenecks in their code. After discussion with the mentor about my analysis, we had a meeting with the Open Banking team on which areas in the code they have to improve. After that I got the code for one of the research papers I read called OtterTune. I analyzed the code for Gaussian Process implementation. The code was in Django so I learned Django and installed and configured PostgreSQL which is used in their code base. Then I searched about the different system variables that can be tuned for PostgreSQL.

In the nineteenth week, I have started studying Python Fabric and Django Celery which are used in their code implementation. They have used standard database benchmarks like TPCC, OLTP to collect server performance scores. Therefore, I learned how to set them up and run the benchmarks in a database. There are around 63 variables exposed by PostgreSQL to indicate the status/performance of the database, they are using that as well in order to measure the performance, so I was finding more information about them also. Debugging Django is entirely different from debugging a normal Python program, so I searched for a way to debug it in PyCharm and debugged the code to see the execution flow of the program. They have implemented a database client that collects the server performance values like throughput and also apply the recommended configurations to the database. I was going through their code in order to find a way to run their client. They are using RabbitMQ as the message broker in maintaining the asynchronous tasks given to celery. It was not running first due to some misconfigurations, I fixed the configurations issues and successfully ran the whole Ottertune program that can check the performance of the database and recommend configurations that can improve the overall performance by almost 50%.

During the twentieth week, I was working on the previous work of Ottertune by making it to work with MySQL, earlier they have support for PostgreSQL. The program won’t predict the correct values if it doesn’t have a dataset already available. Therefore, I went through their code to find how to make dataset and how to upload it. They are using Random Sampling to randomly get some configurations and also Latin Hypercube Sampling which covers most of the points. And, finally I was able to upload the dataset to Ottertune. Then, I started working on WSO2 Enterprise Integrator threadpool size tuning project. And, I was going through their code structure and the dependent modules.

……………………………….
Signature