Sri Lanka Institute of Information Technology

PROJECT REGISTRATION FORM

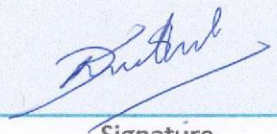
The purpose of this form is to allow final year students of the B.Sc. (Hon) degree program to enlist in the final year project group. Enlisting in a project entails specifying the project title and the details of four members in the group, the internal supervisor (compulsory), external supervisor (may be from the industry) and indicating a brief description of the project. The description of the project entered on this form will not be considered as the formal project proposal. It should however indicate the scope of the project and provide the main potential outcome.

PROJECT TITLE (As per the accepted topic assessment form)	A Network Science Based Approach for Optimal Microservice Governance
RESEARCH GROUP **(as per the Topic assessment Form)**	Distributed and Parallel Computing
PROJECT NUMBER	2020-021

PROJECT GROUP MEMBER DETAILS:

	STUDENT NAME	STUDENT NO.	CONTACT NO.	EMAIL ADDRESS
1	Saranga S.A.G (GROUP LEADER)	IT17016230	071 491 4133	it17016230@my.sliit.lk
2	De Silva N.	IT17006880	078 919 5560	nishithadesilva123@gmail.com
3	L.S. Jayasinghe	IT17012966	076 939 8256	sanjayajayasinghe54@gmail.com
4	M.V. Lakshitha	IT17410250	077 836 5271	lmvithanage@gmail.com

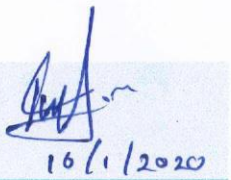
SUPERVISOR Details

Name	Signature	Date
Dr. Dharshana Kasthurirathna		08. 01. 2020

CO-SUPERVISOR Details

Name	Signature	Date

EXTERNAL SUPERVISOR Details

Name	Affiliation	Contact Address	Contact Numbers	Signature/Date
Lahiru Maramba Widanage	Pearson Lanka (Pvt) Ltd	Pearson Lanka (Pvt)Ltd Orion City, Rigel Building No. 752, Dr. Danister De Silva Mawatha Colombo 09	0712180051	 10/1/2020

ACCEPTANCE BY CDAP MEMBER

Name	Signature	Date

PROJECT DETAILS

Brief Description of your Research Problem:

Research problem - In deploying Microservices through Kubernetes, there is no efficient and effective way for developers to evaluate and monitor the effectiveness and viability of a microservice deployment and identify possible performance bottlenecks. Furthermore, developers are not able to optimize their deployment such that they can make the optimal use of their deployed microservices in the cluster.

Throughout the years, with the introduction of many cloud-based platforms such as Azure, AWS and Google which primarily incur cost based on resource utilization, it has become an increasing need for developers and system administrators to monitor, evaluate and optimize their Kubernetes deployments such that they could be able to maximize the usage of their available resources without sacrificing performance. However, maintaining this balance between performance and resource utilization is quite challenging for developers and system administrators since this process requires constant monitoring of metrics and finetuning multiple variables and paraments in their deployment configurations such that the optimal performance criteria are met.

Although current Application Performance Management (APM) tools and platforms in Kubernetes allow gathering and observation of these metrics there is no effective solution for developers and system administrators to evaluate the gathered metrics and get a clear idea on the effectiveness and viability of microservice deployments and identify the possible performance bottlenecks.

Key statements from which the above-mentioned research problem was identified, and ultimately led to the identification of the research objectives, can be found in the paper “Performance engineering for microservices: research challenges and directions”, published by Robert Heinrich and his team. The extensively detailed research paper, clearly highlights the importance of performance evaluation with regard to microservices and the importance of monitoring and monitoring solutions through the use of APM tools, especially in those that are deployed in containers. This paper further goes on to state the need for APM tools deployed in containers to include additional measures to monitor microservices such that they could be used as input for resilience mechanisms and creation of auto-scaling policies, and therefore, has since been one of the key inspirations for the initiation of this research project.

Description of the Solution:

- First, we will select a Kubernetes cluster that is currently deployed and identify the dependencies among microservices using tools such as Istio and Prometheus.
- Then, based on data gathered using those tools we hope to derive metrics and obtain quantitative measures to determine the dependency strength among the interconnected microservices.
- Based on the findings obtained from the previous step we hope to generate a real-time dependency chart to represent the strength among the interconnected microservices
- We also hope to use tools such as Chaos Monkey to evaluate the resilience of the microservice cluster and using the dependency chart developed previously, we will evaluate the resiliency of a particular microservice in the cluster.
- Meanwhile, we will also perform thorough analysis on the microservice cluster to evaluate the auto-scaling and load balancing policies used, and based on the dependency chart generated previously, we hope to develop an algorithm using AI and Machine Learning to optimize the current load balancing an auto-scaling policy such that the developed algorithm could be used to predict future loads and automatically scale the cluster in time.
- Furthermore, an optimization algorithm will also be developed using AI and Machine Learning, which takes into account all the data gathered from all other components to come up with an optimal deployment strategy.
- Lastly, all the data will be fed to a dashboard so that the user will be able to view all the findings.

Main expected outcomes of the project:

The main expected outcomes of this research are as follows:

- To create a weight-based dependency network using metric analysis which highlights dependencies between the interconnected microservices.
- To create an improved auto-scaling policy integrating load-prediction analysis.
- To improve the performance of microservice deployments through resiliency analysis.
- To increase the performance in microservice deployments through the combination of the above-mentioned approaches.

WORKLOAD ALLOCATION

MEMBER 1	Saranga S.A.G – IT17016230

Creation of the dependency map as well as the metrics to evaluate the dependency strength of a microservice cluster and working on configuring auto-scaling of the cluster based on the load-prediction algorithm.

Key tasks in the development of the above process include:

- Setting up and configuring the initial Kubernetes microservice cluster for metric analysis.
- Setting up and configuring the relevant tools such as Prometheus and Istio etc. in order to retrieve the required metrics.
- Identify and retrieve the required metrics from the various metrics analyzing tools.
- Analyze the retrieve metrics such that it could be used to determine the interdependencies between the deployed microservices.
- Development of custom metrics using the retrieved metrics which can be used to represent the interdependency between microservices.
- Development of a solution in order to display the interdependencies gathered using metrics in order to display as a dependency map.
- Performing configuration of the auto-scaling of the cluster based on the load-prediction algorithm.

MEMBER 2	De Silva N. – IT17006880

Developing the algorithm that will be used to predict future loads for automatic-scaling of the cluster.

Key tasks in the development of the above process include:

- Retrieval of the necessary metric data from pods in the cluster in order to perform the load. prediction based on the developed dependency map.
- Development of solution in order to store the gathered metric data for time series analysis.
- Preparing and manipulating the extracted data in order to establish a time series. (data cleaning)
- Development of a time series data set based on the historic metric data retrieved, such that future loads could be predicted.
- Selection of the most appropriate time series forecasting model for load prediction.
- Development of the algorithm with the help of Machine Learning, for the time series using the selected model in order to predict future loads.
- Finetune the developed model in order to get the most accurate prediction.

MEMBER 3 L.S. Jayasinghe – IT17012966

Development of the optimization algorithm which will be used to come up with the optimal deployment strategy.

Key tasks in the development of the above process include:

- Retrieval of the analyzed outputs from the load prediction, dependency analysis map as well as resiliency evaluation process.
- Formatting and data manipulation of the data received from the above-mentioned output processes, in order to be fed as inputs to the optimization algorithm.
- Development of final optimization algorithm making use of Machine Learning, which takes in inputs from the load prediction, dependency analysis map and resiliency evaluation process, and proposes the suggested optimal deployment strategy.
- Integrate with the UI dashboard in order to display the optimal deployment strategy to the user.
- Evaluate the effect of the developed optimization algorithm with respect to the performance of the microservice cluster.

MEMBER 4 M.V Lakshitha – IT17410250

Evaluating the resilience of the microservice cluster through the use of a "Chaos Engineering" approach with the help of tools such as Chaos Monkey by performing targeted attacks on the identified microservices with the help of the developed dependency map and, also the task of the development of the business intelligence dashboard which displays this findings of the optimization process.

Key tasks in the development of the above process include:

- Using the dependency map, perform the development of a system which analyses the resiliency of the cluster using "Chaos Engineering" tools such as "Chaos Monkey".
- Identification of the microservices which show low resilience from the above-mentioned process.
- Evaluating the deployed microservice cluster and coming up with proposed suitable measures in order to improve the resiliency of the deployed microservice cluster.
- Evaluating the effect of the proposed resiliency measures with respect to the performance of the microservice cluster.
- Development of the UI dashboard which displays the final findings of the optimization process to the user and the displaying of the optimal deployment strategy to the user.

DECLARATION

"We declare that the project would involve material prepared by the Group members and that it would not fully or partially incorporate any material prepared by other persons for a fee or free of charge or that it would include material previously submitted by a candidate for a Degree or Diploma in any other University or Institute of Higher Learning and that, to the best of our knowledge and belief, it would not incorporate any material previously published or written by another person in relation to another project except with prior written approval from the supervisor and/or the coordinator of such project and that such unauthorized reproductions will construe offences punishable under the SLIIT Regulations.

We are aware, that if we are found guilty for the above-mentioned offences or any project related plagiarism, the SLIIT has right to suspend the project at any time and or to suspend us from the examination and or from the Institution for minimum period of one year".

	STUDENT NAME	STUDENT NO.	SIGNATURE
1	Saranga S.A. G (GROUP LEADER)	IT17016230	
2	De Silva N.	IT17006880	
3	L.S. Jayasinghe	IT17012966	
4	M.V. Lakshitha	IT17410250