# CHAPTER 3
# RESEARCH METHODOLOGY

This chapter depicts all of the steps that were used in this project in detail. Starting from gathering and studying the related journals to the final stage of the research. The steps of the project as follows :

## 3.1. Literature Study

The first steps of this project is to gather and study journals, papers and research related to microservices architecture and monolith architecture. The journals is used to provide knowledge about the characteristics of both architecture, what challenges that may face when moving from 1 architecture to another architecture and used as references for this project.

## 3.2. Implementation of  Microservice Architecture

The backend of the application that used in this project was first created using monolith architecture, the backend of the application runs as HTTP Server with many services that use a single database using MySQL and use Go as the programming language. To perform the implementation, there are 2 stages to be carried out, such as: Decomposition, Rebuild API Function.

### *3.2.1.* Decomposition

In a monolithic architecture, all services are encapsulated into a single web service, so the modules cannot be executed independently. Therefore to implement the microservice architecture, the services that are in the monolith architecture need to be decomposed into separated components, the components consist of user service and list service. The main goal of the decomposition process is to create the same services with the same core code but with different architecture and some adjustment, so that the comparison of both architectures can be done fairly.

### *3.2.2.* Rebuild The Service

After decomposing the service into small components, the services need to be adjusted, from previously all of the services are in 1 base code to being a separated small component that runs independently. Each service that previously only used Go Language will be rebuild using Go Kit framework.

## 3.3. Testing and Analyzing

The monolith architecture and the microservice architecture will undergo several load tests and will be compared. The comparison will be done to measure the average response time of both architectures.