

Software developing



**ASSIGN
BUSTER**

The software was developed iteratively by submitting module by module. The requirements were changing time-to-time and the software had to go many changes throughout the development. Small chunks were developing from time-to-time which required some changes to be incorporated in the system.

In the meanwhile the developed modules were tested and the feedback was collected continuously to incorporate in our system.

The initial version of the software was released with some simple functionalities and the changes and feedback and requirements were updated which added some advancement to the software we developed.

2. 2 Architectural Strategies

2. 2. 1 Programming Language

Python- As python is one of the world's powerful programming languages it gives some of the built-in modules for development which makes system faster and easy for development. The classes and methods are developed using python. The prediction model uses some of the libraries in python.

PromQL- The query for extraction and generation of the graphs has been written in PromQL query language

2. 2. 2 Future Plans

As it comes to the future it will be taking the corrective actions automatically which means using AI agents for handling all the aspects offailureand

recovery of the system. The enhancement includes chatbot implementation for limited set of queries about the usage stats and analysis of the data.

2. 2. 3 User Interface Paradigm

The user will be provided with the dashboard for the results and reports generated. The dashboard provides various features like querying on the data and stats about the usage of resources and various functionalities. The predictive analysis will be shown in a console of the IDE PyCharm. The user will be given set of values through which the user get an idea about the usage.

2. 2. 4 Error Detection and Recovery

ErrorDetection is carried out by user testing and slack bot has been setup to report the bug in the system. The different datasets are used for testing the ARIMA model has been carried out to test the efficiency of the system. Recovery has been done by alerting the user about the crash in the system using slack automated system and the systems stable state (previous state) will be restored.

2. 2. 5 Data Storage Management

The data are extracted from the exporters and stored in a csv file. The extraction happens between an interval of 5 sec. As the data will be not accessed frequently and modified the data is stored on the stable storage within the machine running the programs.

2. 2. 6CommunicationMechanism

Prometheus used http protocol to communicate with its client system and members. Message passing mechanism will be used to communicate with the exporters for the extraction of the raw data about usage of the resources. Grafana uses http protocol for extraction of the data from prometheus. The data will be passed by prometheus to grafana using the endpoint '/metrics'.

2. 2. 7 Graph Generation Mechanism

The prometheus tool uses a query language called PromQL used for aggregating the extracted data and based on those factors the graphs will be generated. 2. 3 System Architecture As it comes to system architecture typical style has been used which is separate modules and microservices has been used to build the system. Figure

2. 2 System Architecture

2. 4 Data Flow Diagrams2.

4. 1 Data Flow Diagram- Level 0 Figure

2. 3 Data Flow Diagram- Level 0 Initial step is to collect the data from the system (AWS) and the data are stored in CSV file for further analysis.

Prometheus is used for real time monitoring of the AWS instances and generation of usage graphs.

2. 4. 2 Data Flow Diagram- Level 1Figure

2. 4 Data Flow Diagram- Level 1 Exporters are installed for extracting the metrics from the AWS instances , which is then used by Prometheus

<https://assignbuster.com/software-developing/>

monitoring tool for the usage graph generation and the extracted data will be stored in the CSV for further analysis

2. 4. 3 Data Flow Diagram- Level 2Figure

2. 5 Data Flow Diagram- Level 2

Different exporters are installed to get the metrics from different instances, where each exporter will be used by Prometheus to get the data for graph and usage stats generation. Predictive analysis will be done on the stored data using the ARIMA model.