IAS Project

Team Requirement Report

Application Manager And Scheduler

Group 6 Team 3

Submitted by -
Abhijeet Srivastava(2020202011)
Shubham Singh(2020202002)

Table of Contents

Introduction

Application Manager – This module will provide application development and deployment, user authentication and authorization services. It also provides the user interface to interact with the platform.

Scheduler – This module will be responsible for scheduling and sending the command to run the algorithms as per the schedules defined by the user.

Intended Use –

- Provide a UI to users for interacting with the platform like development and deployment of applications and .
- Check for validity of application zip, application configuration file and sensor configuration file for sensor registration.
- Scheduling the algorithms of applications.
- Providing an interface to register sensors on the platform.

Assumptions and Dependencies –

- Only the platform administrator can register the sensors on the platform.
- The configuration files need to be in pre-defined format.
- The application to be uploaded on the platform needs to be in a zipped file which also contains the application configuration file.

Features and Requirements –

Functional Requirements –

1. **Authentication and Authorization –** Application developers and end users will be able to register on the platform and access applications based on their access rights.
2. **Validation –** The configurations files for applications and sensor registration will be validated by the platform before adding them to the application repository and sensor registry.
3. **User Interface –** The user interface will provide a way for the user to interact with the platform. The user interface will be in the form of a Web GUI.
4. **Scheduling –** The algorithms of applications deployed on the platform will be scheduled according to the schedule provided by the user.

Non-Functional Requirement –

1. **Robustness –** The platform can handle the configuration files uploaded and handle the wrong formats.
2. **Security –** Authorization will prevent the users who do not have access rights to see or modify the application.
3. **Ease of Accessibility –** The web GUI will provide access to the platform through the internet. The Rich Interactive Web GUI will allow users to access the platform more effectively.

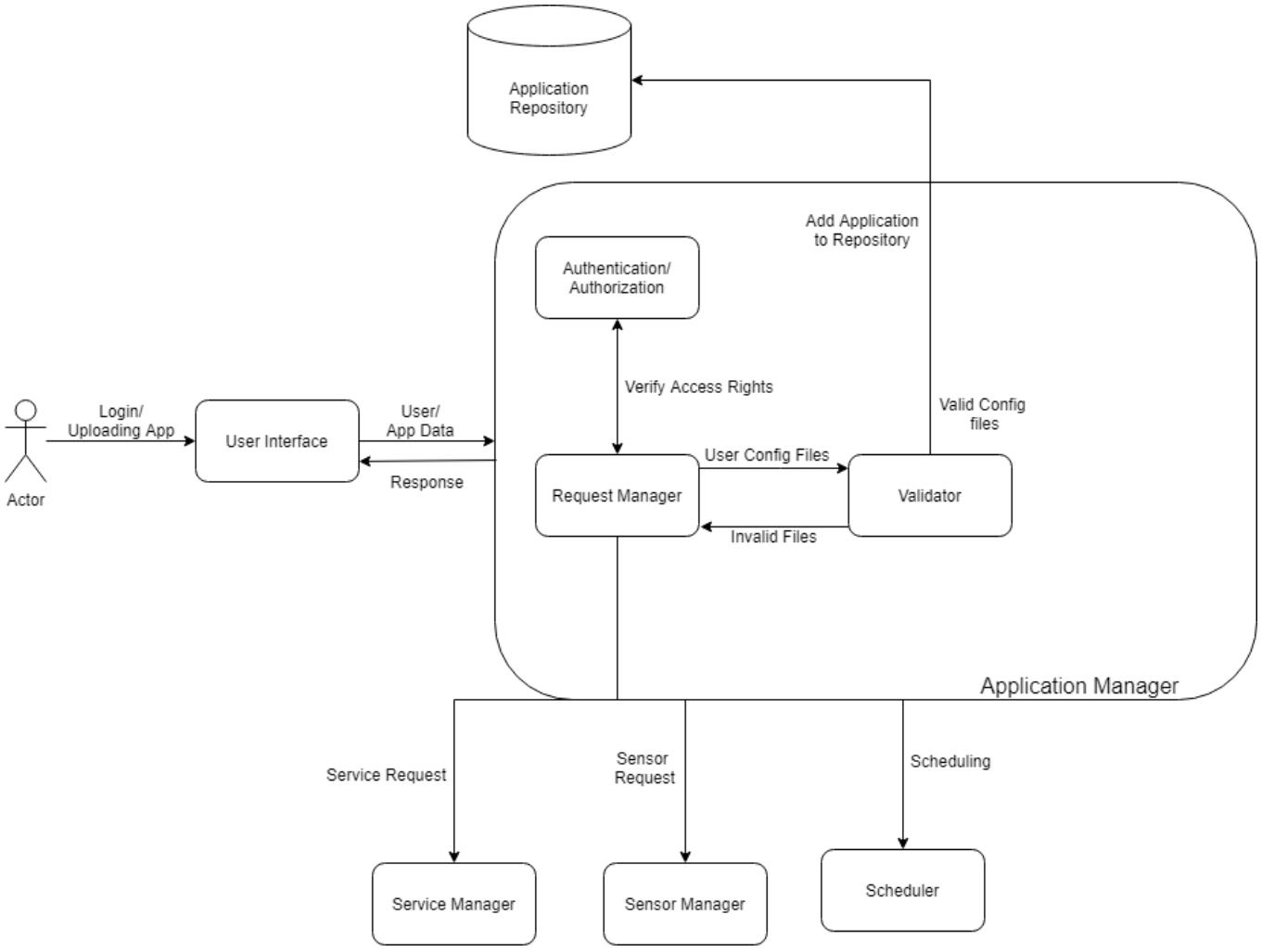
Systems -

1. Application Manager

1.1 Functional Overview –

This part of the project deals with handling applications uploaded by the user along with authentication and authorization of the user. The interface will be in the form of Graphical User Interface. The developer can sign-up/sign-in from the interface and then upload the code/algorithms of the application. This module will also provide an interface to manage those applications and view the algorithms running on the platform by application developers and end users respectively.

1.2 Block Diagram –



1.3 Sub-Systems –

1.3.1 **Authentication and authorization module** – This module handles the authentication of user and permissions of the user on the platform. There are 3 types of permission levels platform administrator, application developer and end user which have different access permissions.

1.3.2 **Request Manager** – This module handles the request from the end user for execution and scheduling algorithms.

1.3.3 **Validator –** This sub-system will validate the requests made by the users.

1.3.4 **Algorithm Manager** – This module manages the algorithms uploaded by developers. It includes saving the algorithms in the application repository and fetching them for execution.

1.3.5 **User Interface** – This module provides the user interface to the platform which includes signing in and a dashboard for maintaining and monitoring the applications.

1.4 Services –

1.4.1 **Access Management –** This part of the platform provides access to services based on one rights of the user.

1.4.2 **Dashboard –** This part of the platform provides a dashboard to users for monitoring the tasks of an application.

1.4.3 **Application Repository –** It provides a storage service for applications uploaded by the user.

1.5 Interaction with other parts –

The module will interact with monitoring service to get status of the tasks of an application and display the results.

The module will interact with the scheduler to send algorithms' information which is to be scheduled.

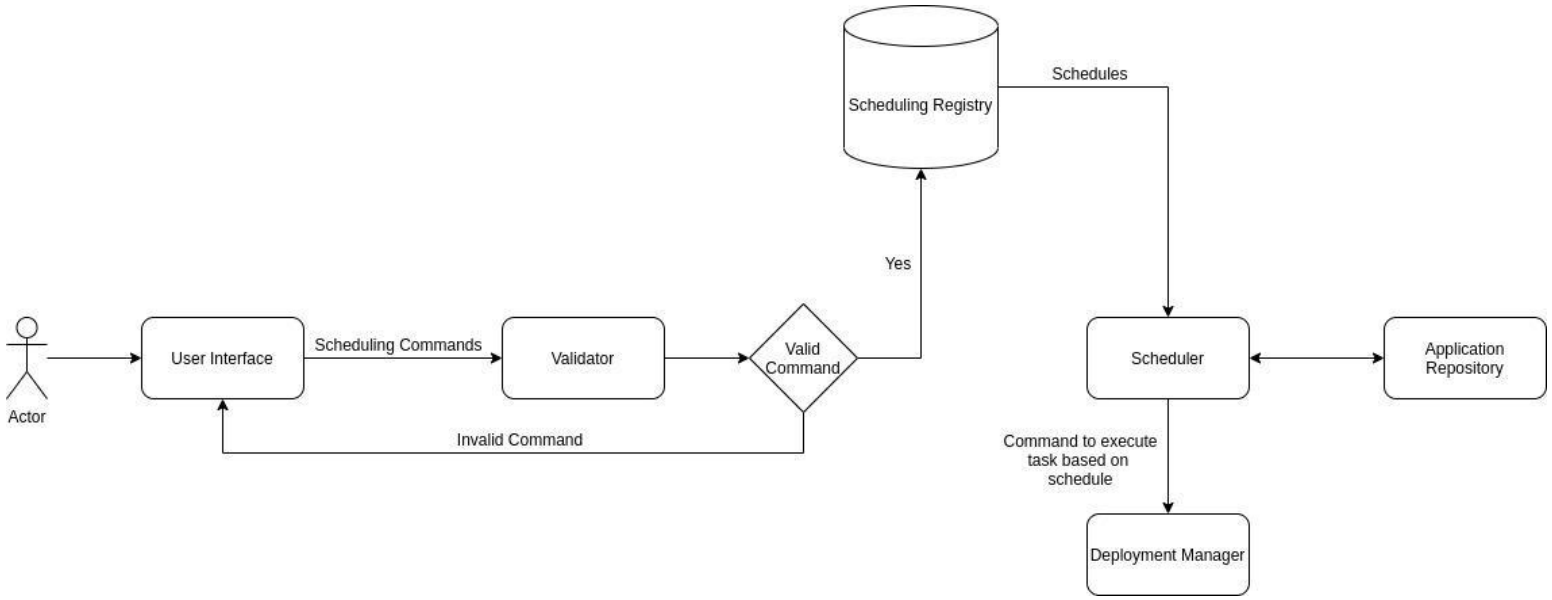
Request Manager will interact with sensor manager and service manager by creating requests for data.

2. Scheduler

2.1 Functional Overview –

This component of the platform helps in scheduling algorithms of applications based on the need of the user. The scheduler can schedule the tasks for one-time execution or scheduled executions based on given time. The executions can be a recurring.

2.2 Block Diagram –



2.3 Services –

2.3.1 Scheduling an algorithm for execution at a particular time.

2.3.2 Providing a scheduling registry to store the schedules of all algorithms in an application.

2.4 Interaction with other parts –

Scheduler receives a request for scheduling from the Application Manager and generates an execution command for the deployment manager at the given schedule.