Software Design Specification

*Group 26 - Finbourne Technology*

Daniel Whelan

Adam Mulvihill

Xiaolei Zhang

Haojun Xing

Tamunotonye Harold Karibye

Daniel Madaghjian

**Table of Contents**

[**1. Introduction**](#_5i9pe9vcrm7o) **2**

[1.1 Overview - Purpose of System](#_h7e1bmpdm8t) 2

[1.2 Scope](#_2brub5aoq2ie) 2

[1.3 Definitions/Abbreviations](#_o3ua29tr0r27) 2

[1.4 References](#_329otok5h194) 2

[**2. System Design**](#_ff2ijca4fx4r) **3**

[2.1 Design Overview](#_3od9ftssb5ou) 3

[2.1.1 High Level Design Overview](#_pmtgy7kg142e) 3

[2.2 System Design Models](#_bq72f4k4hbps) 3

[2.2.1 System Context](#_b2sc23zaq4j3) 3

[2.2.2 Use Cases](#_pqkzntptpz1p) 4

[2.2.3 System Architecture](#_z6gosfc9v87j) 4

[2.2.4 Class Diagram](#_pc8hw2n1321k) 5

[2.2.5 Sequence Diagram](#_lrh4b1xjyqx7) 6

[2.2.6 State Diagrams](#_2qc4dskv1nc7) 7

# 

# 1. Introduction

## 1.1 Overview - Purpose of System

The purpose of the system is to allow users to gain access to statistics related to the database such as queries, transactions, and index usage, as well as to retrieve information from certain configured system tables, without gaining access to client data present in the database. The purpose of this is to increase security, as the current procedure to access these statistics and queries is to request elevated access permissions, which gives full access to the database including client data.

Team Objectives

* Develop a service that will run inside a database cluster in parallel with each instance of a database to analyse them.
* Develop an authenticated REST API that will allow access to the service.
* Use PostgreSQL and C# for development.

## 1.2 Scope

We have been tasked with creating a service to analyse a database for table statistics and index usage. The service also should have the ability to keep track of the number of running queries and open transactions, and the ability to run a given query through an explain analyse clause. This service will run inside the cluster that the database is in. The service will poll the Postgres database inside the Kubernetes cluster and in turn will store the information in a file system that must be accessible through an authenticated REST API.

## 1.3 Definitions/Abbreviations

API - Application Programming Interface - An interface or communication protocol between different parts of a computer program

REST - REpresentational State Transfer - An architectural style for providing standards between computer systems on the web, making it easier for systems to communicate with each other

## 1.4 References

Finbourne Technology (2022), “*Project 8\_Finbourne”*, Retrieved via CSU33013 Blackboard page.

# 2. System Design

## 2.1 Design Overview

### 2.1.1 High Level Design Overview

A C# monitoring system that is spun up when a Postgres database is created inside a Kubernetes cluster. This will poll the database and collect statistical information relevant to the database and store said information in a file system. This will then be accessed by a C# REST API that will be run from a Finbourne Technology team member that requires access to the data.

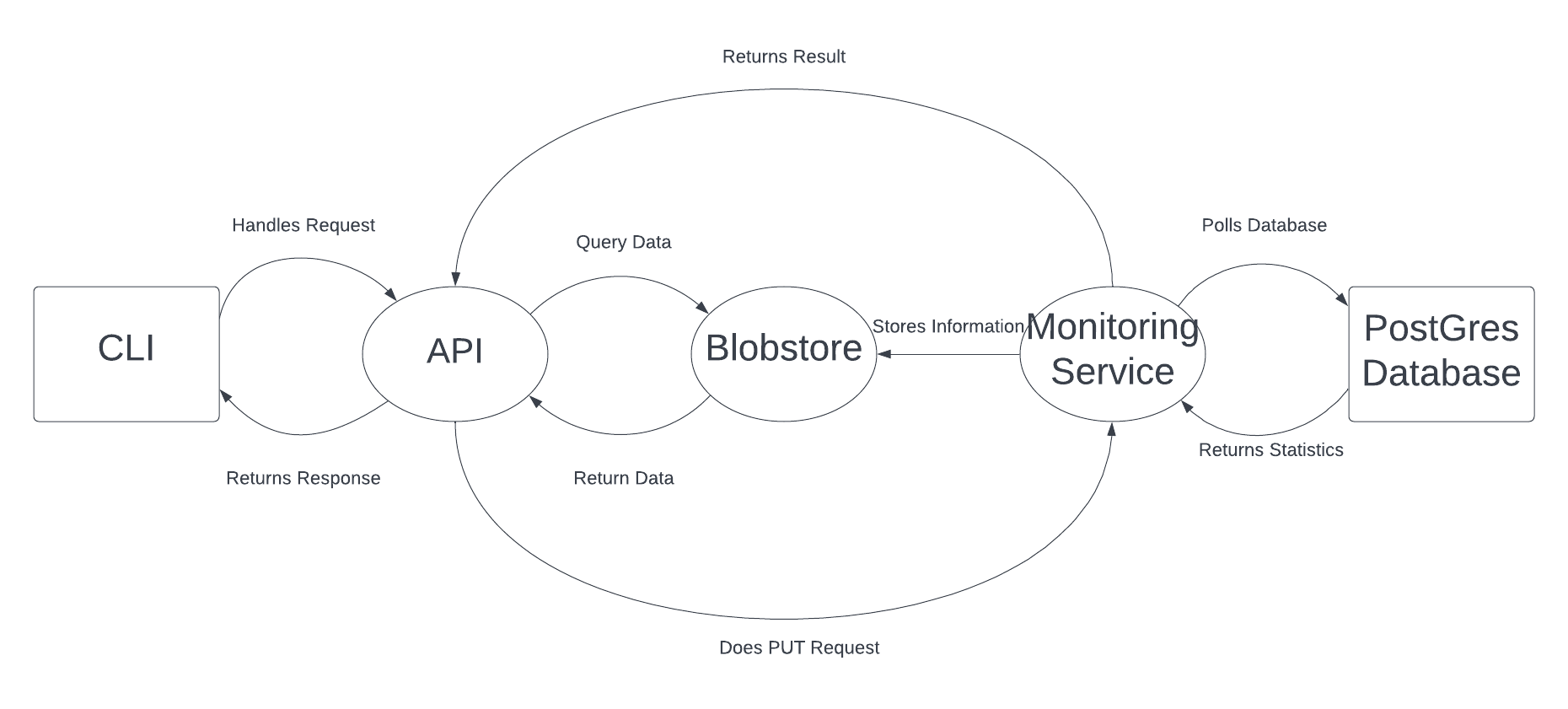
Diagram

Description automatically generated

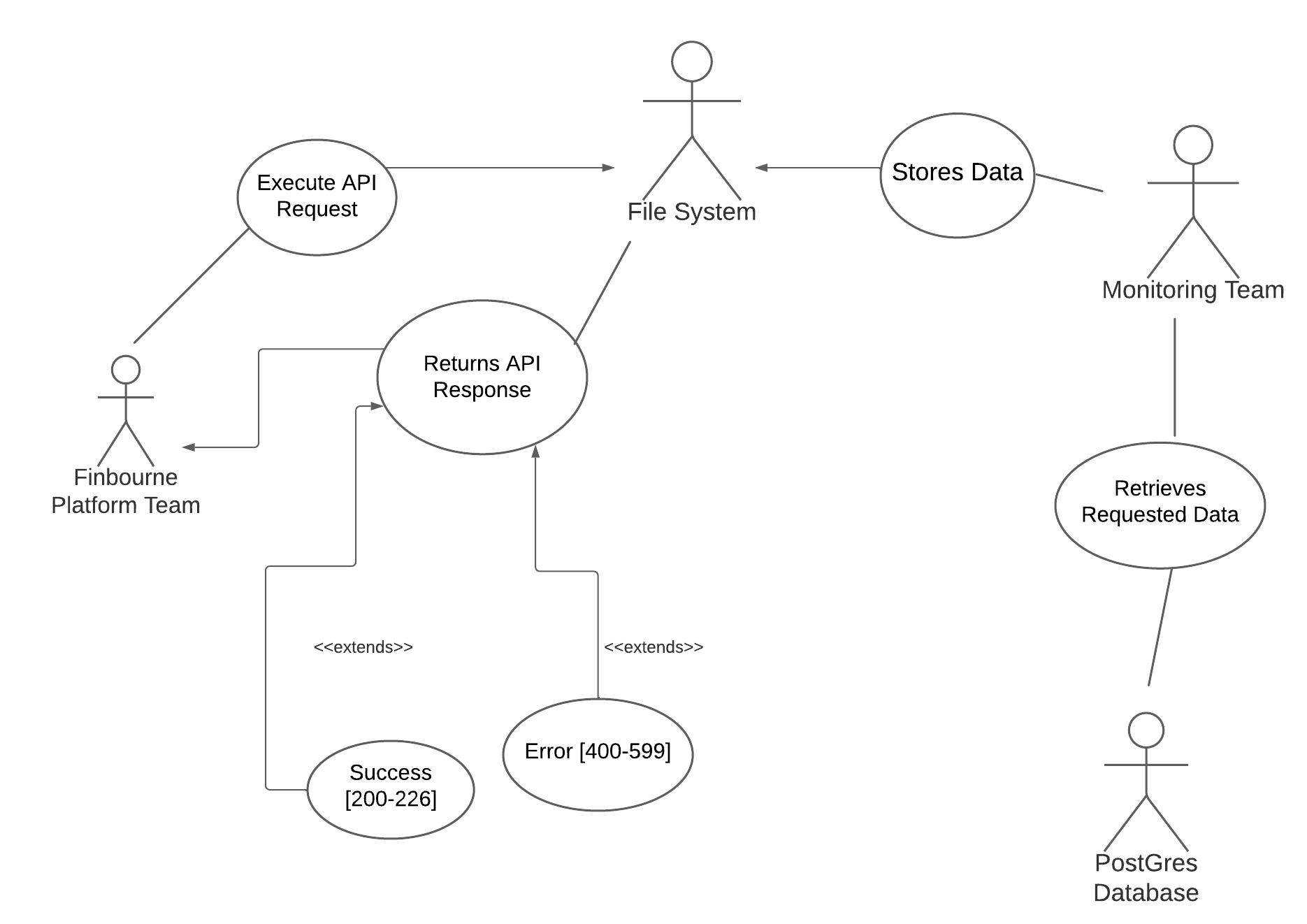
Diagram from Finbourne highlighting how they would like the proposed system to operate.

## 2.2 System Design Models

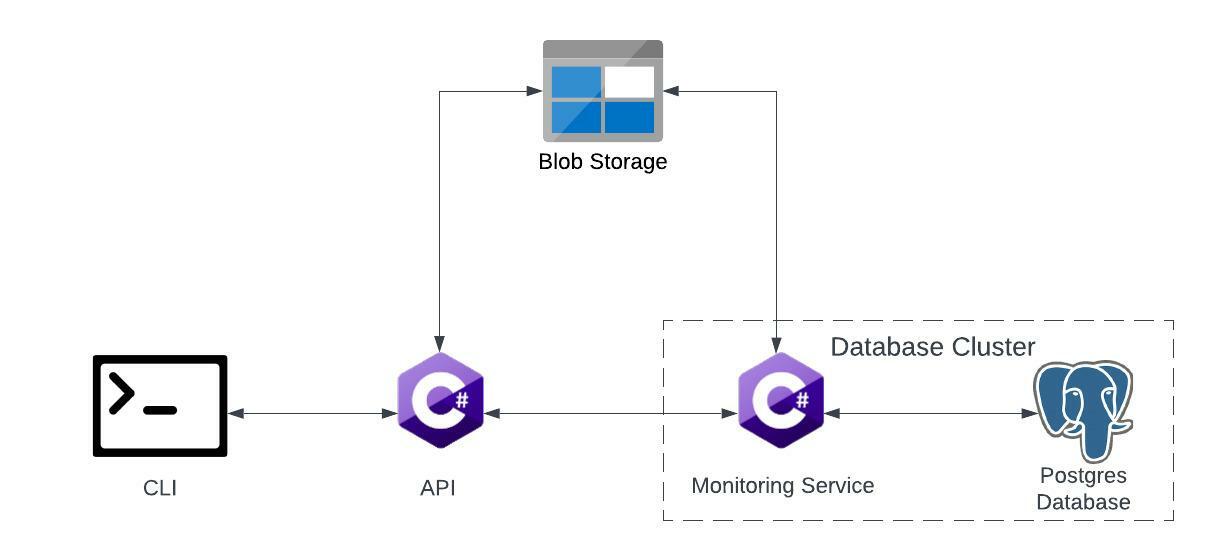
### 2.2.1 System Context



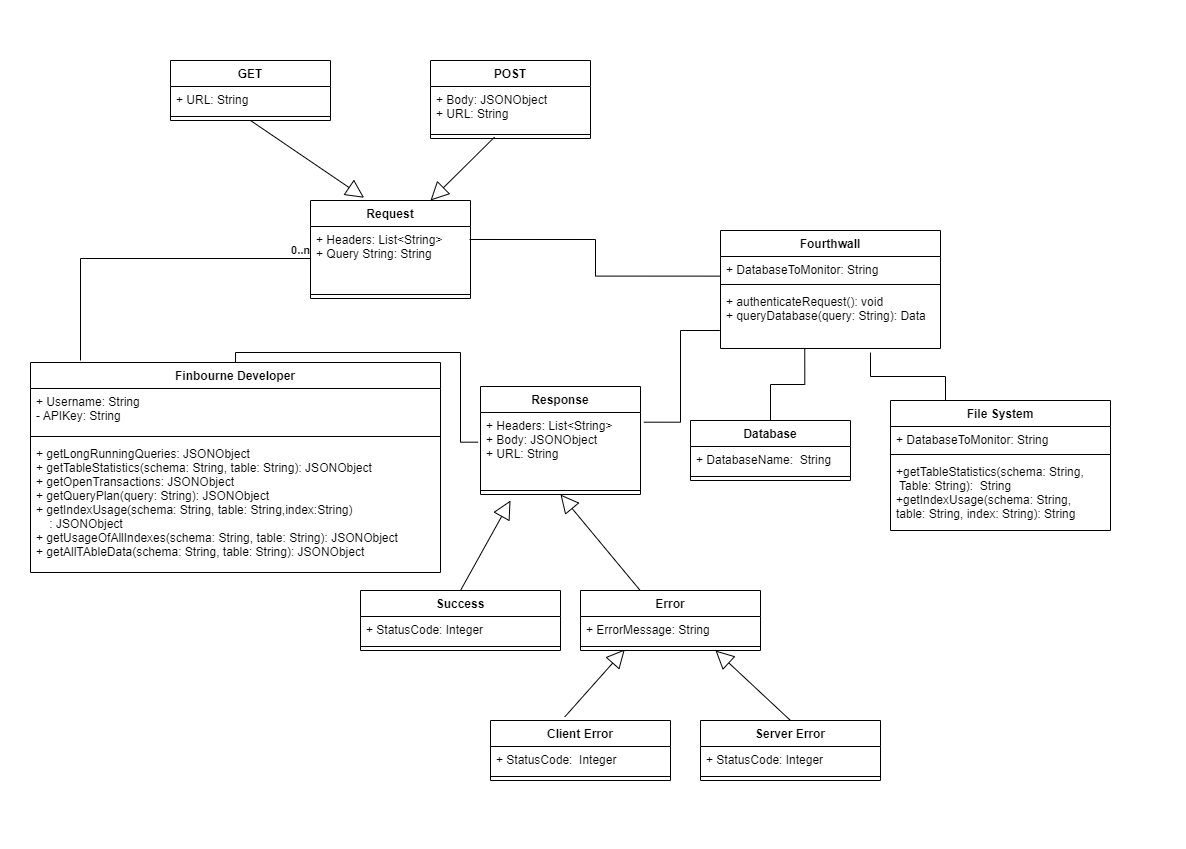
### 2.2.2 Use Cases



### 2.2.3 System Architecture



### 2.2.4 Class Diagram



### 2.2.5 Sequence Diagram

### 

### 

### 

### 2.2.6 State Diagrams

