Heimdall

• • •

An open-source server monitoring application

"HAME-doll"

● ● ●

(Noun) (Norse Mythology) Ever-vigilant guardian of the god's stronghold Asgard, known for the ability to see/hear everything

Project Purposes

- Provide free, open-source means of monitoring remote servers (or any machine connected to the web)
- Explore possibilities of application monitoring, such as bytecode instrumentation

Goal Statements

- Design multi-platform monitoring agent
- Create an online platform, accessible by web and mobile
- Offer end-users the ability to setup monitoring agents on the desirable servers
- Collect data and aggregate it to provide meaningful analytics

Team Members

- Kyle Cullion
  - culliokw@mail.uc.edu
- Michael Keenan
  - keenanmj@mail.uc.edu
- Vivek Kunapareddy
  - kunapavk@mail.uc.edu
- Zack Steck
  - steckzt@mail.uc.edu
- Dr. Kenneth Berman
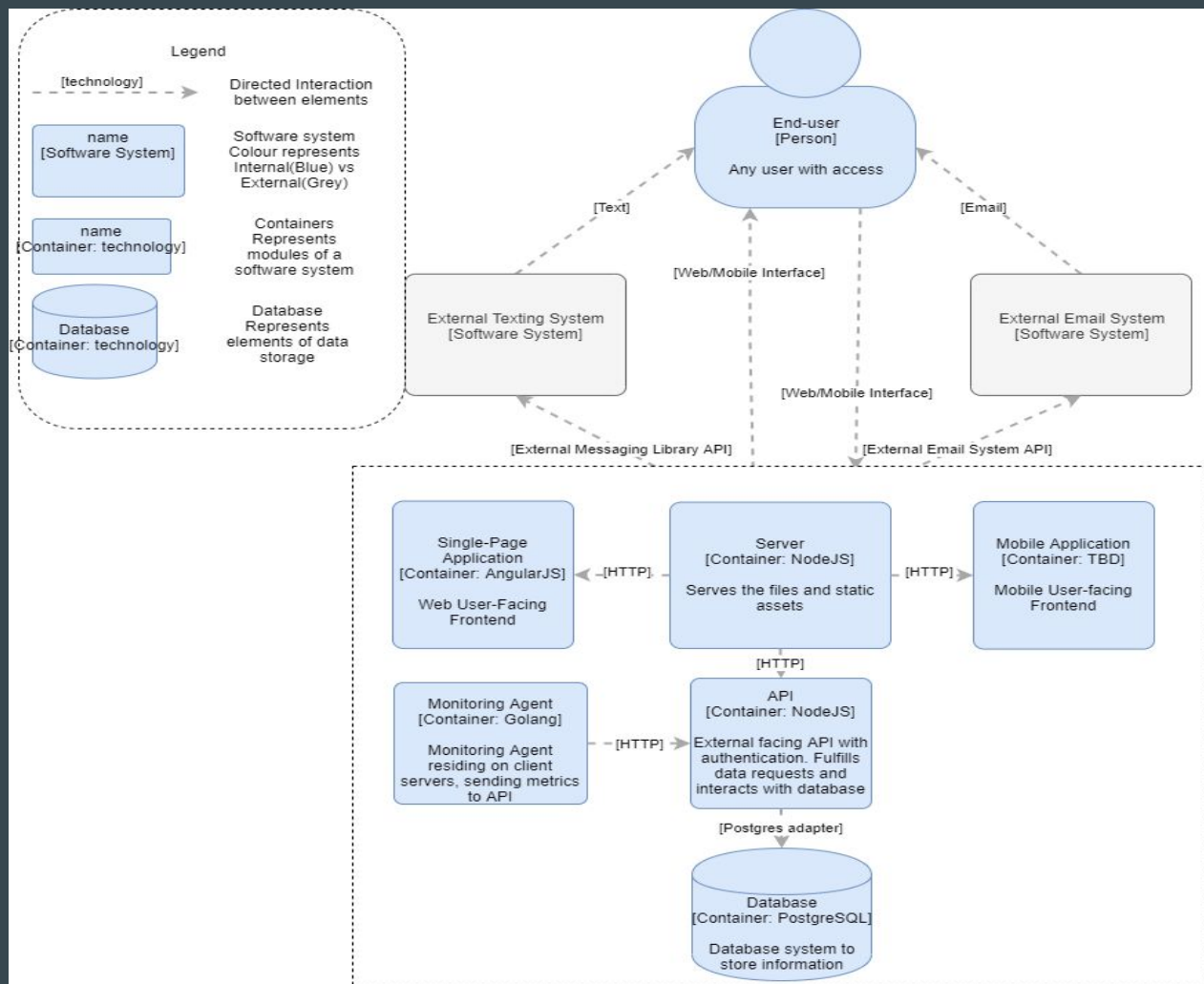 - Project Advisor

Project Description

Project Abstract

This project attempts to bridge the gap between small, specific open-source server-monitoring applications and expensive proprietary applications. We will provide the users with a mobile application, web application, and server monitoring agent for analyzing server uptime/performance. The monitoring agent will rest on target servers and communicate data back to the web server for aggregation and visualization to the user. Users will be able to customize the alert criteria, defining new rules and criteria on which alerts will be communicated to them.

User Stories

- As a user in Heimdall, I want to login to the website, so that I can view my agent's metrics
- As a user in Heimdall, I want to see a forgot password link, so that I can reset my password in the event I lose the current password
- As a user in Heimdall on the Dashboard page, I want to see a list of my registered agents, so that I can choose which agent to monitor
- As a user in Heimdall on the Dashboard page, I want to be able to click an agent on the list of registered agents, so that I can be shown the metrics for that agent
- As a user in Heimdall on the Dashboard page, I want to see a graph of the CPU usage of the selected agent, so that I can watch for spikes in percent usage

Design Diagrams

Legend

[technology] ----→ Directed Interaction between elements

name [Software System] — Software system Colour represents Internal(Blue) vs External(Grey)

name [Container: technology] — Containers Represents modules of a software system

Database [Container: technology] — Database Represents elements of data storage

End-user [Person] — Any user with access

External Texting System [Software System]

External Email System [Software System]

[Text]

[Email]

[Web/Mobile Interface]

[Web/Mobile Interface]

[External Messaging Library API]

[External Email System API]

Single-Page Application [Container: AngularJS] — Web User-Facing Frontend

Server [Container: NodeJS] — Serves the files and static assets

Mobile Application [Container: TBD] — Mobile User-facing Frontend

[HTTP]

[HTTP]

[HTTP]

Monitoring Agent [Container: Golang] — Monitoring Agent residing on client servers, sending metrics to API

API [Container: NodeJS] — External facing API with authentication. Fulfills data requests and interacts with database

[HTTP]

[Postgres adapter]

Database [Container: PostgreSQL] — Database system to store information

Project Constraints

- Application level monitoring
- OS level separation of agents
- OS security
- Application configuration

Current State of Project

- Preliminary design complete
- Choice of technologies complete
- Proof of concept for agent complete
- Proof of concept for web server complete

Expected End of Term Progress

- Functioning prototype
- Confer with advisor for refinement
- Research into application monitoring
- Research into aggregation

Project Timeline

Task ID	Task	Owner	Start Date	End Date	Milestone
0	Identify performance metrics provided by linux kernel to be parsed by monitoring agent.	Zack Steck	12/14/2019	12/21/2019	
1	Develop initial database schema	Kyle Cullion	12/14/2019	1/14/2019	
2	Create Register, Login, and Dashboard pages and deploy for other team members to connect to.	Michael Keenan	12/14/2019	1/14/2019	
3	Design initial master server in Nodejs	Vivek Kunapareddy	12/14/2019	1/14/2019	Web Server Complete
4	Research methods of obtaining similar metric from Windows machines.	Zack Steck	12/21/2019	12/31/2019	
5	Develop fault-tolerant method of obtaining metrics from kernel files for transmission to master server.	Zack Steck	01/01/2019	1/14/2019	Server Monitoring Agent Complete
6	Develop aggregation on the backend based on incoming metric	Kyle Cullion	1/14/2019	1/21/2019	Database Complete
7	Deploy test environment to preview UX design	Michael Keenan	1/14/2019	1/21/2019	
8	Expand server setup to accept agent requests	Vivek Kunapareddy	1/14/2019	1/21/2019	
9	Create fault-tolerant channel between monitoring agent and master server.	Zack Steck	1/14/2019	1/21/2019	
10	Develop baseline and std-dev values on backend for incoming metrics	Kyle Cullion	1/21/2019	1/28/2019	
11	Identify all metrics that need to be displayed on the Dashboard page of the web app.	Michael Keenan	1/21/2019	1/28/2019	
12	Refine initial server setup to allow file serving	Vivek Kunapareddy	1/21/2019	1/28/2019	
13	Investigate data using EDA on recovered metrics.	Zack Steck	1/21/2019	2/28/2019	Data Analysis Complete
14	Design simple and intuitive way for users to switch between registered agents on the web app.	Michael Keenan	1/28/2019	02/04/2019	
15	Investigate platform-specific instrumentation for application monitoring	Vivek Kunapareddy	1/28/2019	2/11/2019	
16	Refine server to include ORM consistent with schema	Kyle Cullion	1/28/2019	2/11/2019	
17	Design registration and onboarding flow for new customers on the web app.	Michael Keenan	02/04/2019	2/21/2019	Web Application Complete
18	Refine server setup to server agent files, either with dynamic or static configuration	Vivek Kunapareddy	2/11/2019	2/21/2019	
19	Expand backend framework to serve mobile and web-app	Kyle Cullion	2/11/2019	2/21/2019	

Member Roles and Responsibilities

- Kyle Cullion
  - Backend developer
  - Data Analysis and Aggregation
- Michael Keenan
  - Frontend Developer
  - Energy drink supplier
- Vivek Kunapareddy
  - Backend Developer
  - Data aggregation, Frontend display
- Zack Steck
  - Backend Developer
  - Data Analyst

Expected Demo at Expo

- Agents installed on multiple machines (remote and local)
- Show frontend displays for metrics
- Demo of agent installation
- Display alerting
- Present data aggregation