Project Plan
AppDynamics Platform Configuration Tool

The Capstone Experience
Team Evolutio
Kp Inuaeyen
Jon Dressel
Ian Guswiler
Cameron Rasico
Ben Haase

Department of Computer Science and Engineering
Michigan State University
Spring 2019
Functional Specifications

• Empower Evolutio’s Consultants and Clients to Utilize Cisco AppDynamics
• Easily Configure, Deploy and Manage Applications
• Retrieve and Report Basic Application Metrics Via Amazon’s Alexa
Design Specifications

• Web Application
  ▪ Allow Easy Transfer of Dashboards and Health Rules between Applications

• Notification System
  ▪ Notifies user when there are changes to source environment with sub environments grouped to it

• Amazon Alexa
  ▪ Let Evolutio Clients easily access analytics from AppDynamics
Screen Mockup: Dashboard/Health Rules

Controller: http://35.92.134.8090/controller/
Application: 1 (Default to Current)
Tier: 2 (Default to Current)
Node: 5 (Default to Current)

Deployment Mappings

Import  Export  Deploy

Drag File for Import
Screen Mockup: Deployment
Screen Mockup: Grouping
Screen Mockup: Settings

```
Settings

Controllers

Add

http://35.9.22.134:8090/controller/  Edit  Delete

Set Controller:
Set User:
Set Account:
Set AppID:
Set Password:

Save Settings
```
Technical Specifications

• Web App UI
  ▪ Built on React Framework and Bootstrap CSS

• RESTful Web Service
  ▪ Built using Spring Boot Java
  ▪ Uses Java HTTP Clients to communicate with AppDynamics API

• AppDynamics Controller
  ▪ Access point for AppDynamics API
  ▪ Receive HTTP Request from RESTful web service
  ▪ Send HTTP response containing JSON data

• Amazon Alexa
  ▪ Custom Skill to Communicate with AppDynamics API via HTTP requests
System Architecture
System Components

- **Hardware Platforms**
  - Linux VM for AppDynamics
  - Amazon Echo Dot Device

- **Software Platforms / Technologies**
  - Node.js
  - React
  - Bootstrap CSS
  - Java
  - Spring Boot
  - RESTful API
  - AppDynamics Controller API
Risks

• Java Application into REST Service
  ▪ **Description:** Unsure of feasibility of exposing Java app to REST service
  ▪ **Mitigation:** Researching and testing with Java RESTful frameworks

• Alerting the Front-End when a Controller Connection Fails
  ▪ **Description:** Loss of connection results in losing rule configuration work without the front-end knowing
  ▪ **Mitigation:** Research and design an HTTP response JSON structure that includes errors

• Developing Portable, Locally-Executable Web App
  ▪ **Description:** Unsure as to how we can develop a web app that can be run locally
  ▪ **Mitigation:** Investigating frameworks that allow for locally executable web apps

• Detecting Changes to an Application Environment
  ▪ **Description:** Unsure how to detect changes to an AppDynamics application environment
  ▪ **Mitigation:** Research AppDynamics' Alert and Respond function

• Configuring Alexa with API Request
  ▪ **Description:** Our Alexa Skill will need to be able to initialize HTTP request from AppDynamics Metric API in order to receive metrics while passing sensitive credentials.
  ▪ **Mitigation:** Investigate how credentials are stored within Alexa Skills. Research how to initiate and retrieve data through http request within an Alexa Skill
Questions?