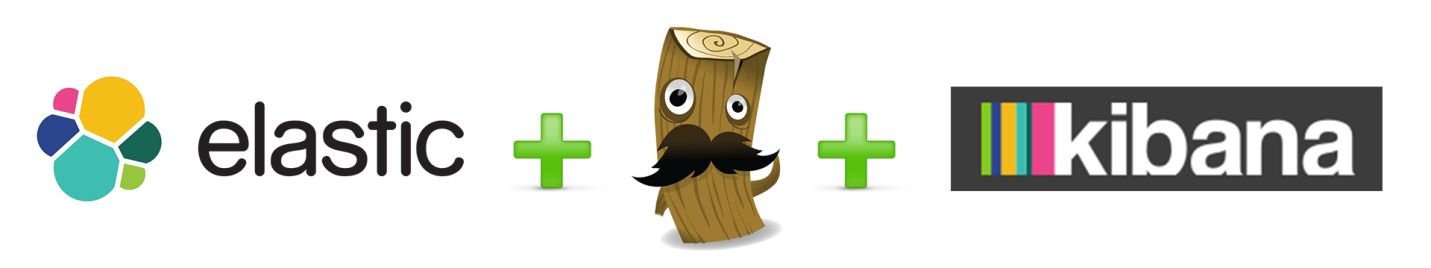
##### ELK Stack



Software Requirements Specification

MedVoice

July 8, 2106

**Table of Contents**

1. Introduction.

**1.1. Purpose.**

**1.2. Scope.**

* 1. **Definitions, acronyms, and abbreviations**
  2. **References.**
  3. **Overview.**

1. **Overall description**
   1. **Product Perspective.**
      1. **System Interfaces.**
      2. **User Interfaces.**
      3. **Hardware interfaces.**
      4. **Software interfaces.**
      5. **Communication interfaces.**
      6. **Memory constraints.**
      7. **Operations.**
   2. **Product functions.**
   3. **User characteristics.**
   4. **Constraints.**
   5. **Assumptions and dependencies**
   6. **Apportioning of requirements**
2. **Specific requirements.**
   1. **External interfaces.**
   2. **Functions.**
   3. **Performance requirements.**
   4. **Logical database requirements.**
   5. **Design compliance.**
   6. **Software system attributes.** 
      1. **Reliability.**
      2. **Availability.**
      3. **Security.**
      4. **Maintainability.**
      5. **Portability.**
   7. **Organizing the specific requirements.**
      1. **System mode.**
      2. **User Class.**
      3. **Objects.**
      4. **Feature.**
      5. **Stimulus.**
      6. **Response.**
      7. **Functional hierarchy.**
   8. **Additional comments.**
3. **Supporting information:**
   1. **Table of contents and index.**

**1 Introduction**

* 1. **Purpose**

The purpose of this document is to explain in detail an ELK stack. It shall explain the uses of each of the individual parts ElasticSearch, Logstash, and Kibana and how they work together to make an ELK stack.

* 1. **Scope**

To easily display log data, which can be difficult to decipher due to a lack of standardization, and make it easy to understand for business/analytic purposes. The ELK will take log data, from multiple source/formats, generalization the format into a json file, store the data in ElasticSearch and display this information in Kibana.

**1.3 Definitions, acronyms, and abbreviations**

|  |  |
| --- | --- |
| SRS | Software Requirements Specification |
| ELK | ElasticSearch Logstash Kibana |

**1.4 References**

[1] IEEE Recommended Practice for Software Requirements Specifications Software Engineering Standards Committee of the IEEE Computer Society. 1998.

[2] http://blog.arungupta.me/getting-started-elk-stack-wildfly/

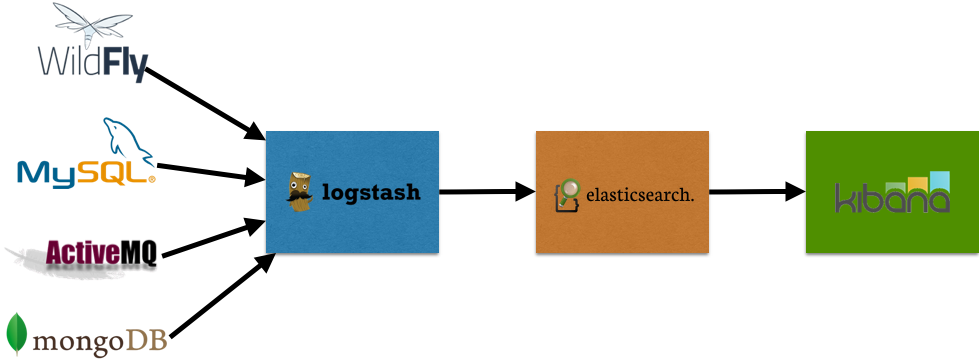
**1.5 Overview**

1. **Overall Description**

This section will give an overview of the whole system.

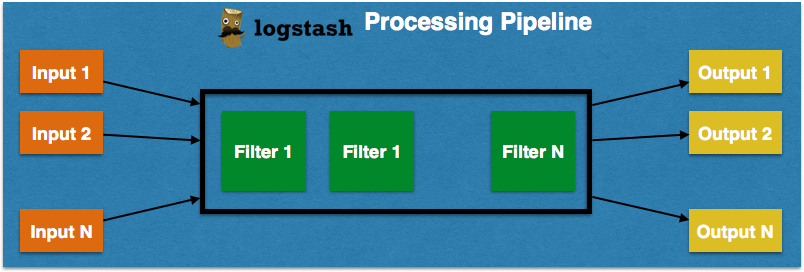
**2.1 Product Perspective**

ELK stack consists of 3 major parts ElasticSearch, Logstash, and Kibana.



ElasticSearch stores the data and make it searchable in an almost real-time.

Logstash collects logs/events data and ships them to ElasticSearch.



Kibana frontend for the whole system, that visualizes the data for analytics purpose.

**2.1.1 System Interfaces**

The interfaces with be several connected servers

**2.1.2 User Interface**

Kibana dashboard through a Web browser.

**2.1.3 Hardware Interfaces**

Mouse and Keyboard

**2.1.4 Software Interfaces**

Viewed in a web browser

**2.15 Communication Interfaces**

**2.1.6 Memory Constraints**

**2.1.7 Operations**

Continuous, can be accessed at anytime

**2.2 Product Functions**

Users log data will be taken in by Logstash, stored in ElasticSearch, then become visualized by Kibana for analytics purpose.

**2.3 User Characteristics**

Data analysists, upper management

**2.4 Constraints**

Multiple users could be logged in at the same time. Internet is a constraint since Kibana is hosted on a Webpage. A work around would be to download the data to the mobile app, however internet connection would still be required to update.

**2.5 Assumptions and Dependencies**

MongoDB, can be accessed through a web browser

* 1. **Apportioning of Requirements**

**3 Specific Requirements**

**3.1 External Interfaces**

Web page that has graphs, charts, data which can be resized and scrolled through.

3.2 Functions

3.3 Performances Requirements

3.4 Logical Database Requirements

3.5 Design Compliances

3.6 Software System Attributes

3.6.1 Reliability

3.6.2 Availability

3.6.3Security

3.6.4 Maintainability

3.6.5 Portability

3.7 Organizing the Specific Requirements

3.7.1 Systems Mode

3.7.2 System Mode

3.7.3Objects

3.7.4 Features

3.7.5 Stimulus

3.7.6 Response

3.7.7 Functional Hierarchy

3.8 Additional Comments