

# Monitoring Report

# Table of Contents

1 Introduction.....	3
2 Rationale.....	4
2.1 As-Is Concept.....	4
2.2 As-Is Realization.....	4
2.3 Actors.....	4
2.3.1 Goals.....	7
2.3.2 Needs.....	7
2.3.3 Pain Points.....	7
3 Prototype.....	9
3.1 Requirements.....	10
3.2 Measures of Effectiveness.....	10
3.3 Use Cases.....	10
3.3.1 Scenarios.....	10
3.4 To-Be Concept.....	15
3.4.1 Storyboards.....	16
3.5 To-Be Realization.....	21
3.5.1 List of Potential Tools.....	22
3.5.1.1 Competitive Analysis.....	23
3.5.2 Paper Prototype.....	25
4 Test Plan.....	32
4.1 Test Cases.....	32
4.2 Approach.....	32
4.2.1 Paper Prototype Testing.....	32
4.2.2 High-fidelity Prototype Testing.....	39

# 1 Introduction

The CAE SE Technical Working Group identifies performance as a primary concern for the OpenCAE environment. To address this concern, we attempt to characterize performance across CAE subsystems for the purpose of identifying problem areas in performance in real-time. This topic is what we refer to as monitoring. The CAE SE Team will conduct research with the CAE DevOps, AppOps, and Services Teams to identify needs and pain points in the context of monitoring. Research with these teams will drive work in integrating existing monitoring solutions and prototyping new technologies if necessary. We will manage a series of collection and contact efforts to provide the rationale for our system specification. We will then prototype concepts via activities and storyboards for the purpose of needs validation with users. Collect List of as-is strategies for diagnosing system performance problems List of monitoring/logging tools currently used by team List of open-source/commercially available tools that support scenarios similar to our own Contact Brad Clement Cin-Young Lee Sophie Wong Patrick Leung Jason Han Doris Lam Methods Interview Needs Validation



Name: Susan

Position/Employer: Europa Systems Engineer/JPL

Susan is a systems engineer with the Europa team at JPL. She is a member of the CAE user community and uses many CAE applications and services for her work. She relies on the performance of these services to ensure productivity of her job. When she encounters issues, she submits a ticket for resolution. Once the issue is resolved she resumes her normal work.

- DEVOPS ENGINEER -

- WHAT IS THE PROBLEM?
- SITUATIONAL AWARENESS
- MACHINE PROBLEMS
- UPTIME/DOWNTIME
- DEVELOPS <sup>AND TESTS</sup> CODE FOR INTEGRATION
- ~~INTEGRATES + DEPLOYS~~ SOFTWARE

- DEVELOPER -

- HOW DO I FIX THE PROBLEM?
- INVESTIGATION
- MACHINE + PROGRAM PROBLEMS
- RESOURCE UTILIZATION
- DEVELOPS <sup>AND TESTS</sup> CODE FOR APPLICATIONS / SERVICES
- ~~BUILDS~~ SOFTWARE

## 2.3.1 Goals

DevOps goals:

- Ensure that infrastructure for all devices and servers are performing properly
- Be able to know and quickly resolve issues relating to network issues
- Be aware at all times of the state of the infrastructure

Software goals:

- Be aware of application downtime and immediately respond
- Be aware of machine and network status
- Ensure that application metrics are running properly

Operations goals:

- Be aware of network status
- Be able to identify whether problems are machine specific or pertain to a larger network
- Submit tickets for issues that arise

User goals:

- Be aware of any upcoming issues/resets
- Remain updated on system status
- Have issue tickets resolved quickly
- Remain updated on issue ticket status

## 2.3.2 Needs

DevOp needs;

- Need to receive warnings of any issues that arise before or as they happen
- Need to diagnose and identify the problem quickly for resolution
- Need to be able to view and interpret dynamic data graphs

Software needs:

- Need to receive warnings of any issues that arise before or as they happen
- Need to diagnose and identify the problem quickly for resolution
- Need to be able to respond quickly to user issues

Operations needs:

- Need to know system and network status
- Need to receive warnings of application issues
- Need to view dynamic graphical data
- Need to monitor proactively

User needs:

- Need issues to be resolved quickly
- Need awareness of system to avoid issues
- Need to be aware of issue status

## 2.3.3 Pain Points

### Visibility

Monitoring dashboards are the landing page for engineers to observe the status of the system. The complexity of entire networks often make these dashboards highly cluttered and clunky. In order to support greater visibility, dashboards need to be easily comprehensible and easily navigated so engineers understand the state of the network at all times.

### Alerting

One problem with alerting in performance monitoring is that alerts are often numerous and non-descriptive. Alerts need to be informative, contextual, and salient. Unimportant alerts can unnecessarily capture attention if not of major severity. Engineers need a way to differentiate non-urgent from urgent and critical alerts. Additionally, alerts are often too uninformative, with no context as to what data the alerts refer.

### **Data Displays**

Because engineers rely so heavily on graphical data, the display of this data is important in how engineers find and interpret trends. Monitoring systems and dashboards often contain confusing or unadjustable graphs. Engineers need dynamic graphs to fully understand the time range and trend of the data.





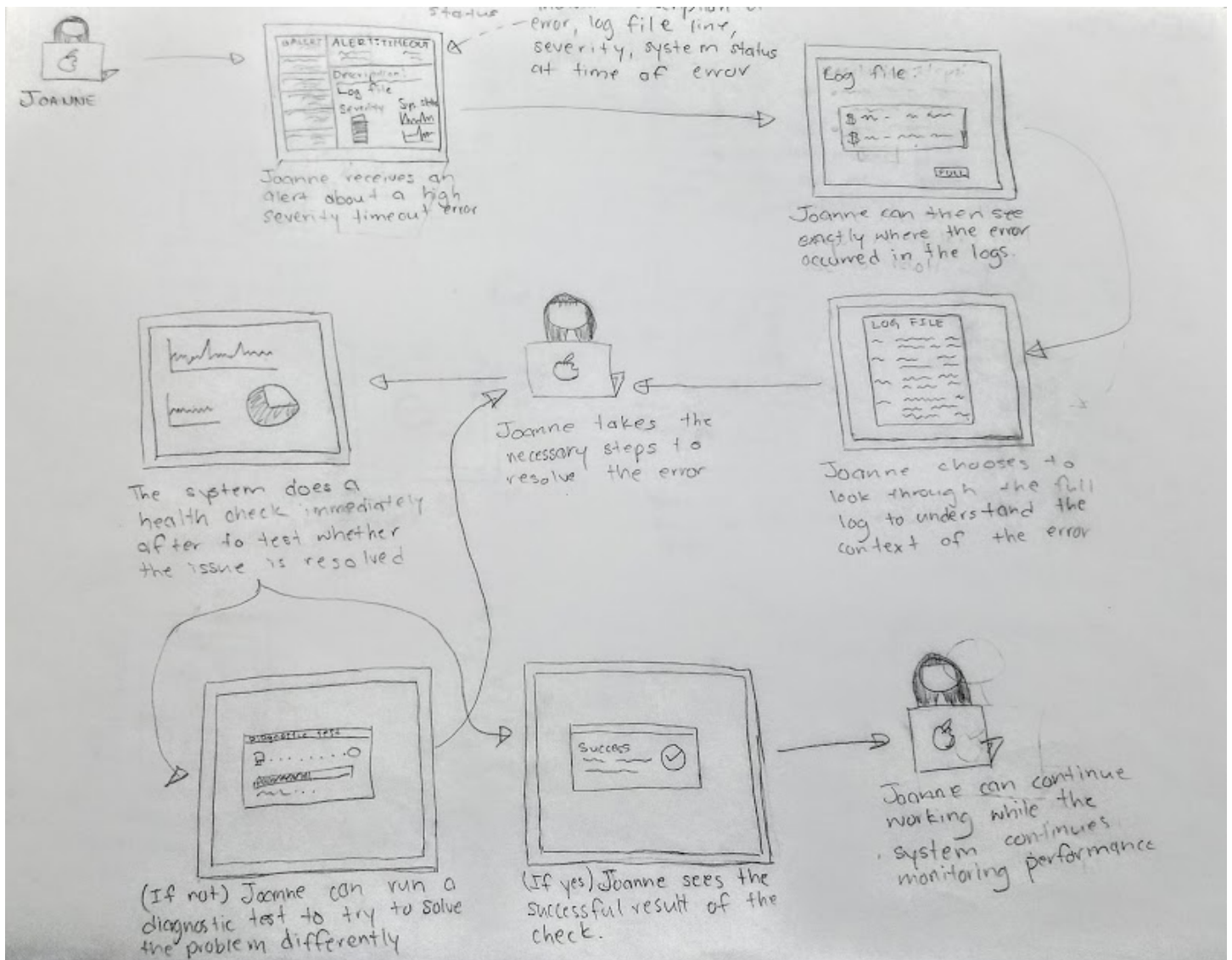
### 3.1 Requirements

### 3.2 Measures of Effectiveness

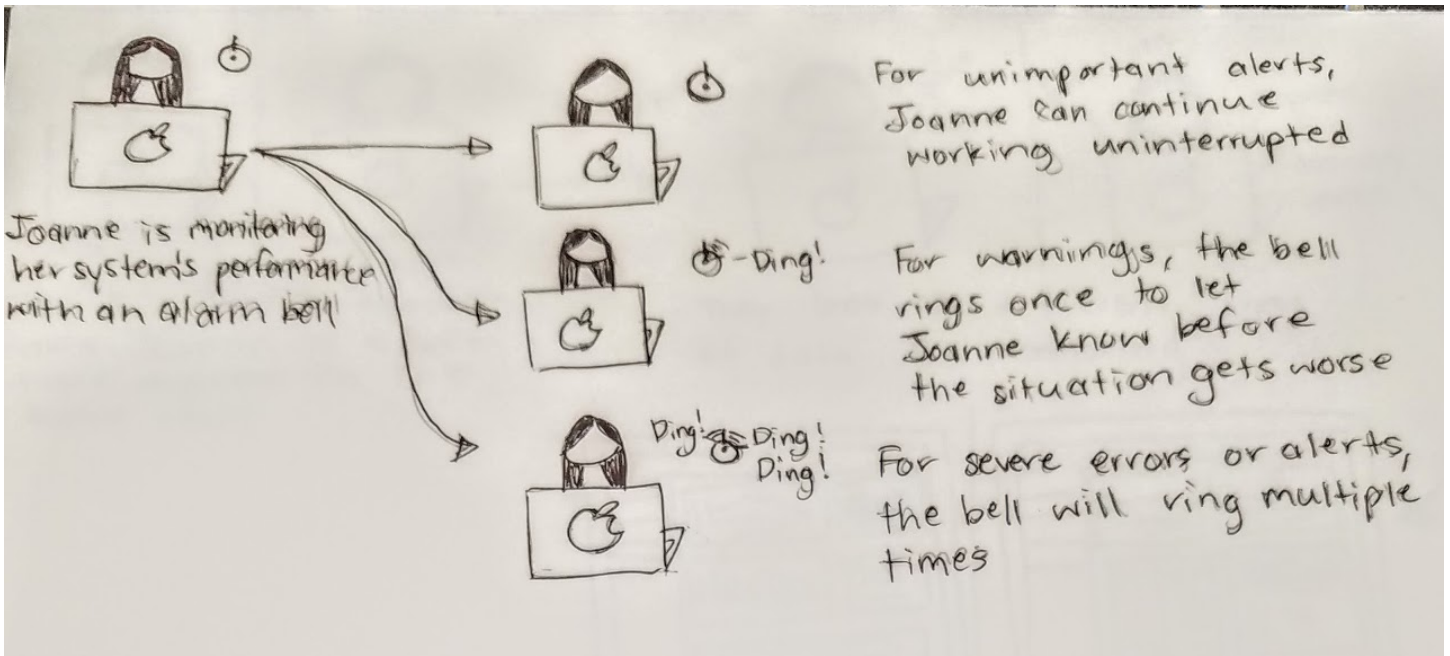
### 3.3 Use Cases

#### 3.3.1 Scenarios

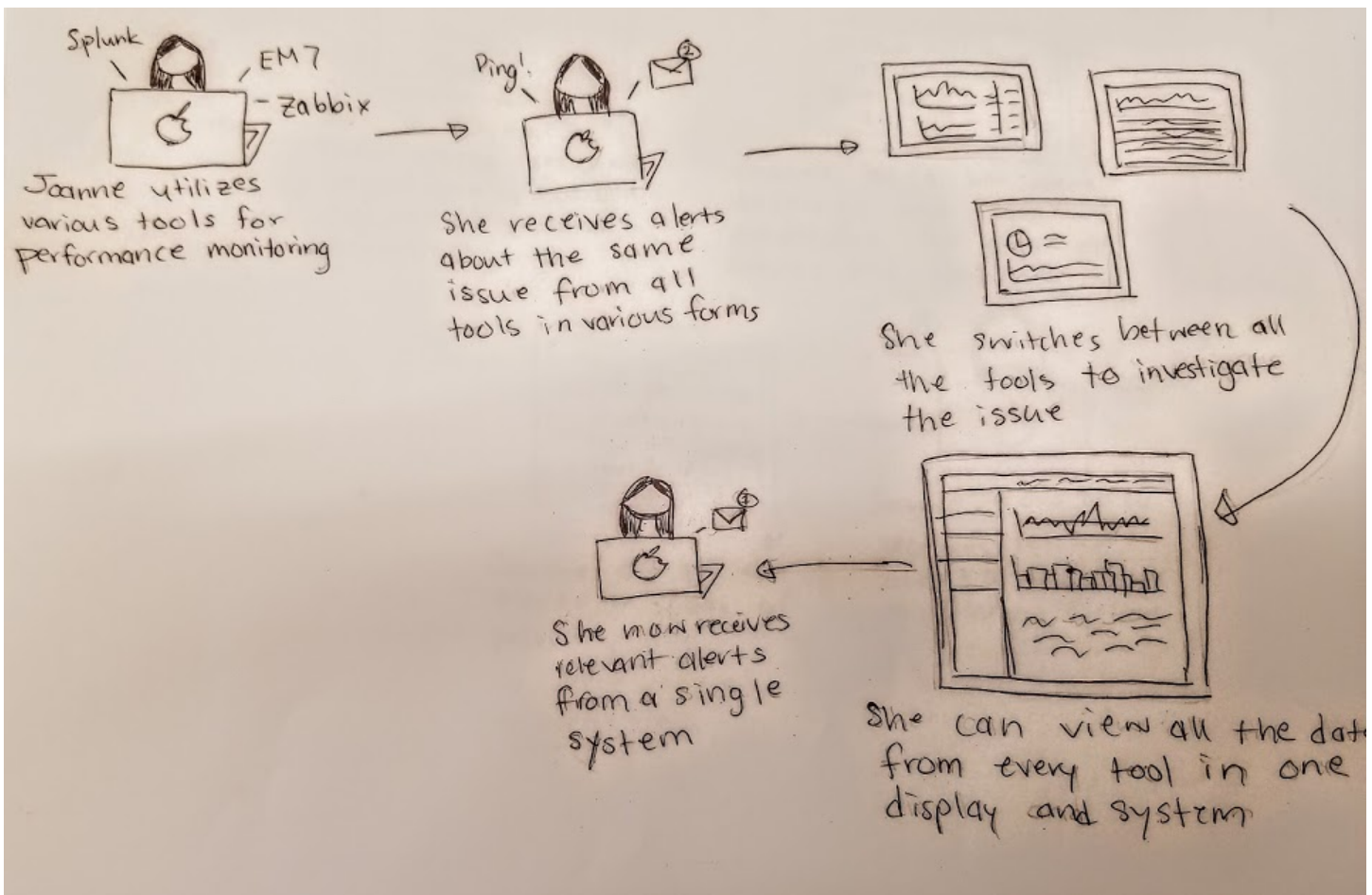
Scenarios are elaborations of use cases. Storyboards are depictions of those scenarios that can be tested with users. These storyboards depict situations related to monitoring that users realistically deal with and present a potential conceptual solution.



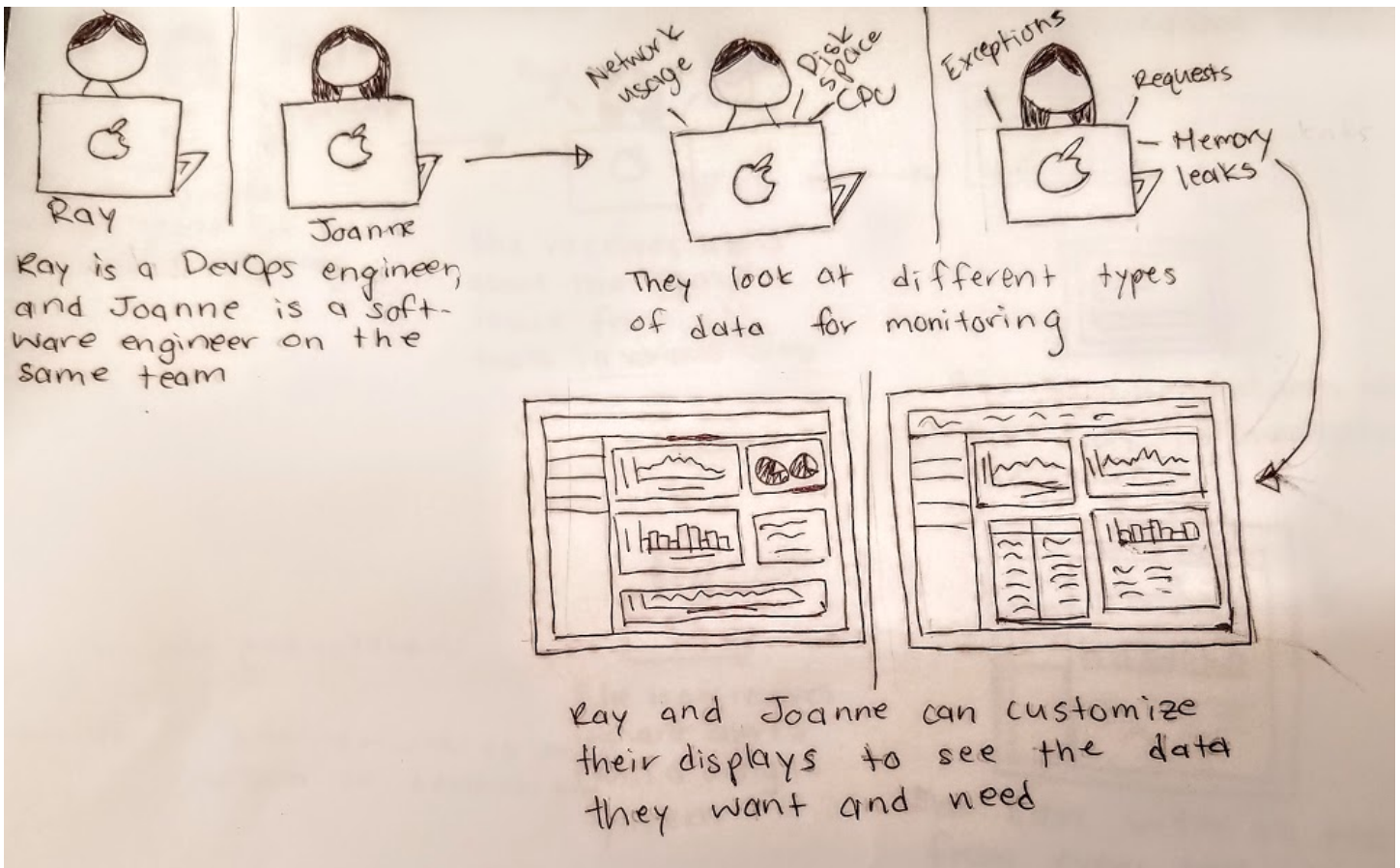
Log handling and analysis: in this scenario, the user analyses system logs to to resolve a monitoring issue.



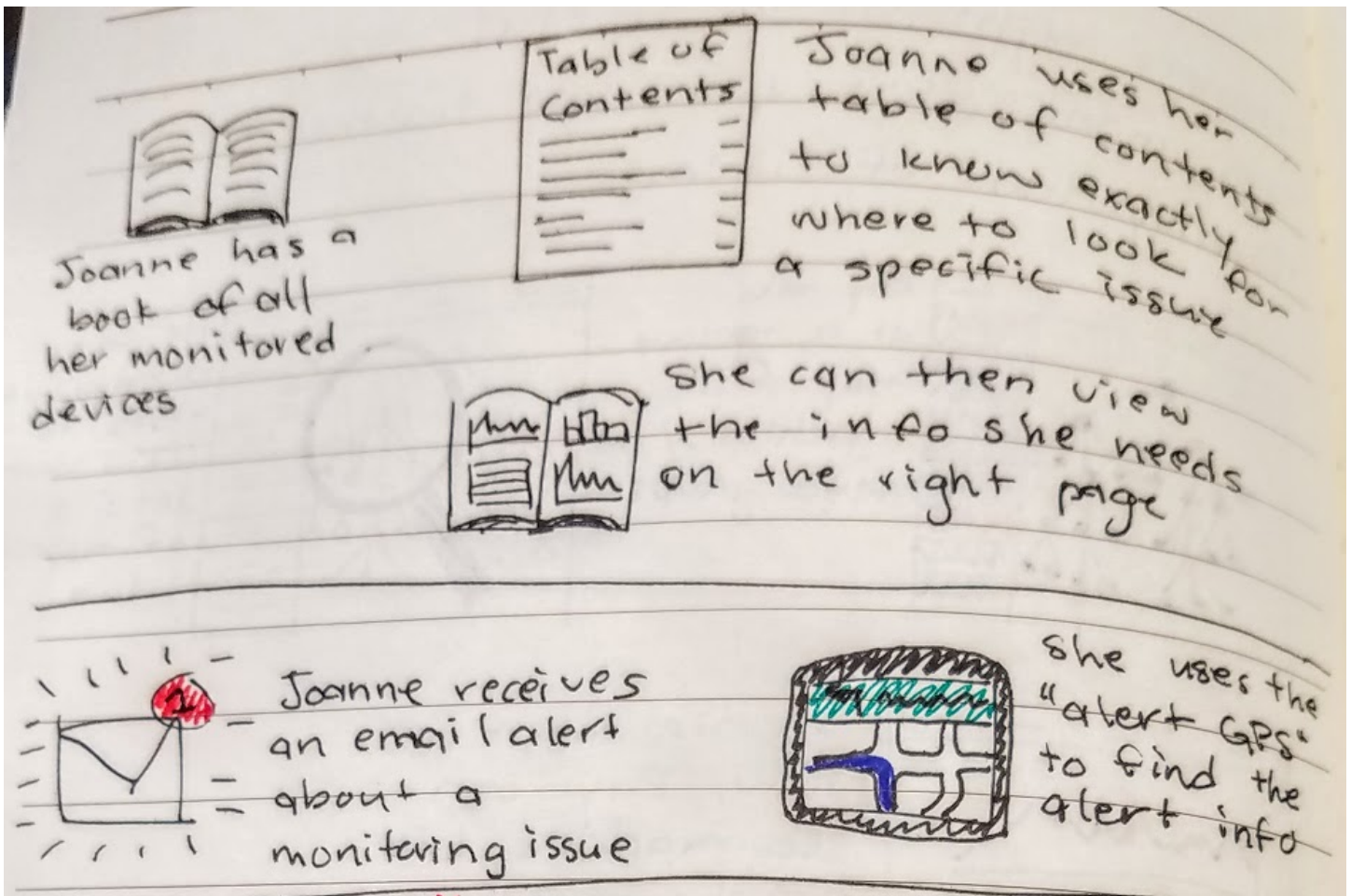
Severity handling: Depending on the severity of the issue, different alarm actions should occur so the user knows which issues require immediate attention versus those that don't.



Integration of tools: monitoring of complex networks requires multiple tools, all of which have their own alarm settings. The user in this instance can view all alerts and information from a central hub.

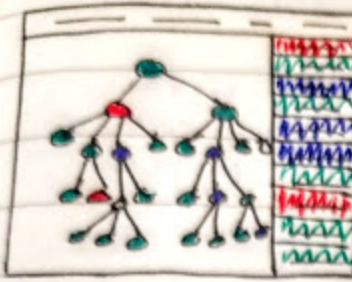


Role-based dashboard displays: Because different users have different needs in monitoring, a single dashboard display may not be adequate for meeting everyone's needs. Custom, role-based dashboard displays allow users to see the information they need, specific to them.



a. Navigation of Data: The user can find the exact location and context of data needed to address a specific issue.

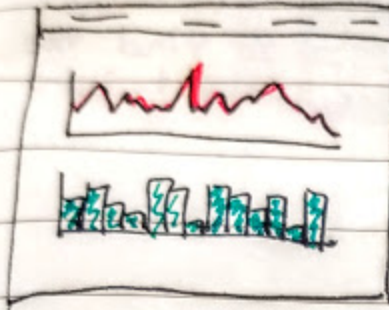
b. Contextual alerts: Alerts contain a navigating link or source to take users exactly to the data relevant to the alert.



Ray looks at a topology map of all machines / device groups



He uses a magnifying glass to view more details of a specific issue



In his magnified view, Ray can see machine data graphs + other specific info



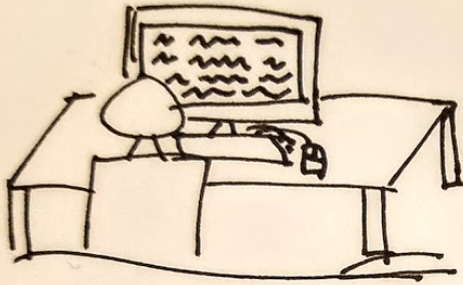
Ray has a map of his entire system of devices + networks



He uses the map to see where he needs to drill down further

Topological Zoom: The user can drill down on a specific issue from a holistic view to a more granular view.

**License Monitoring**



Kevin is a license manager and has to view the statuses of all these licenses



He uses a funnel to find specific license info he needs

With numerous licenses and features, users have difficulty filtering through all licenses to determine problems or usage patterns. A funnel that filters through this data can help the user find the specific information he needs.



Kevin has a forest of licenses. When a user hits a denial, Kevin doesn't know where to find the right license to help.



Kevin finds a pair of glasses that color codes the license trees



w/ the color coded license trees, Kevin finds the license + user he needs to address

When a customer hits a license denial, he reports it to the user, who must go through a lengthy process to try to figure out what license needs investigation and why. Quickly finding the exact license and information he needs will more efficiently aid the productivity of the customer.

### 3.4 To-Be Concept

Clickable Monitoring Prototype: [https://cae-ems.jpl.nasa.gov/share/page/site/site\\_\\_18\\_0\\_5\\_8630260\\_1465427006119\\_760709\\_47553/document-details?nodeRef=workspace://SpacesStore/ff385f58-413a-46a1-b910-fe90bf63eb70](https://cae-ems.jpl.nasa.gov/share/page/site/site__18_0_5_8630260_1465427006119_760709_47553/document-details?nodeRef=workspace://SpacesStore/ff385f58-413a-46a1-b910-fe90bf63eb70)

Monitoring Prototype: [https://cae-ems.jpl.nasa.gov/share/page/site/site\\_\\_18\\_0\\_5\\_8630260\\_1465427006119\\_760709\\_47553/document-details?nodeRef=workspace://SpacesStore/91e2dde6-69c6-446c-b100-93957d9e2c5d](https://cae-ems.jpl.nasa.gov/share/page/site/site__18_0_5_8630260_1465427006119_760709_47553/document-details?nodeRef=workspace://SpacesStore/91e2dde6-69c6-446c-b100-93957d9e2c5d)

#### System Monitor

- Availability Monitor
  - SNMP
- Hardware Monitor
  - CPU
- Disk Monitor
  - Disk space
  - I/O performance
- Interface Monitor
  - Packet loss
  - Errors

#### Application Monitor

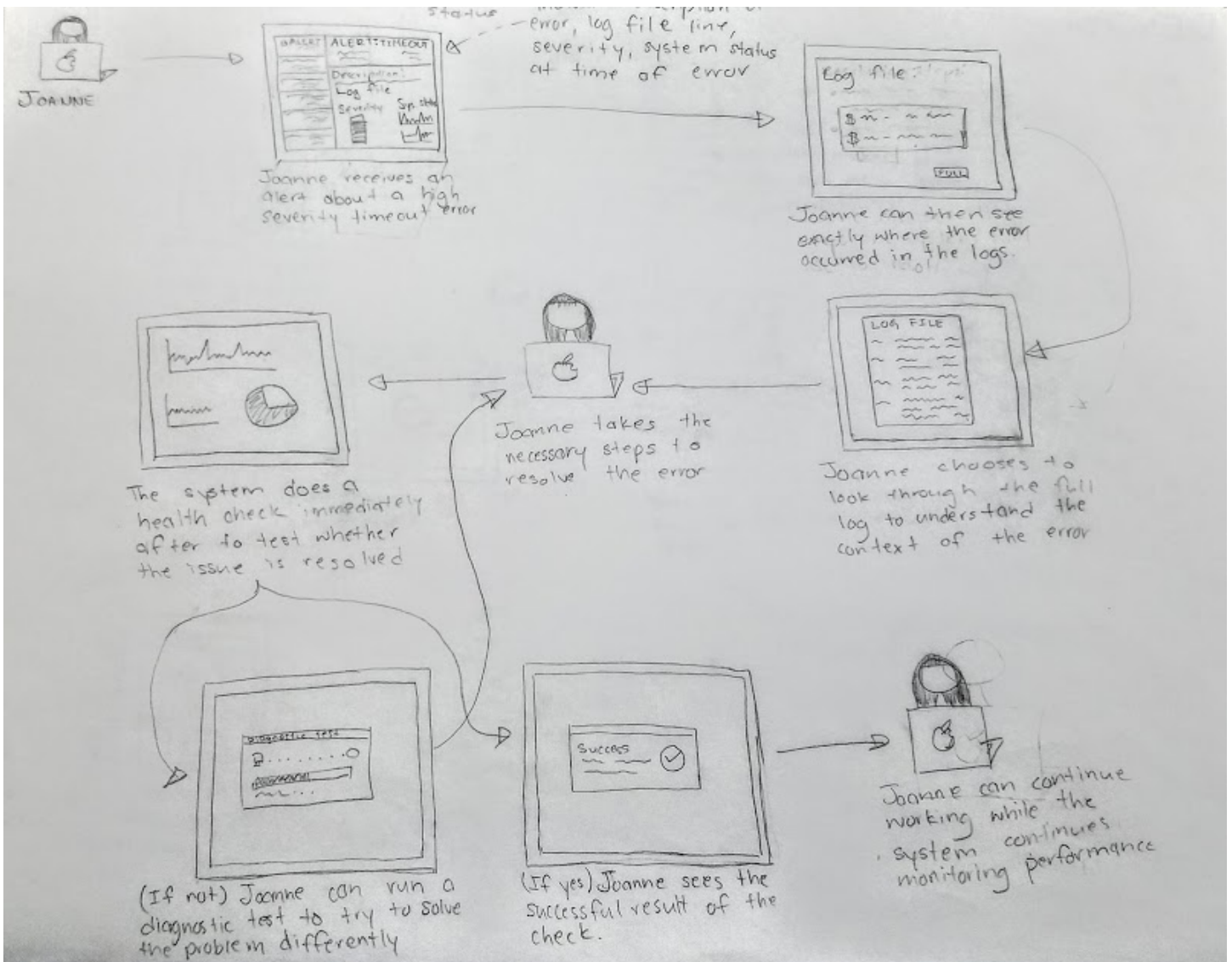
- Component Monitor
- End-User Monitor
  - Web Analytics

#### Log Event Manager

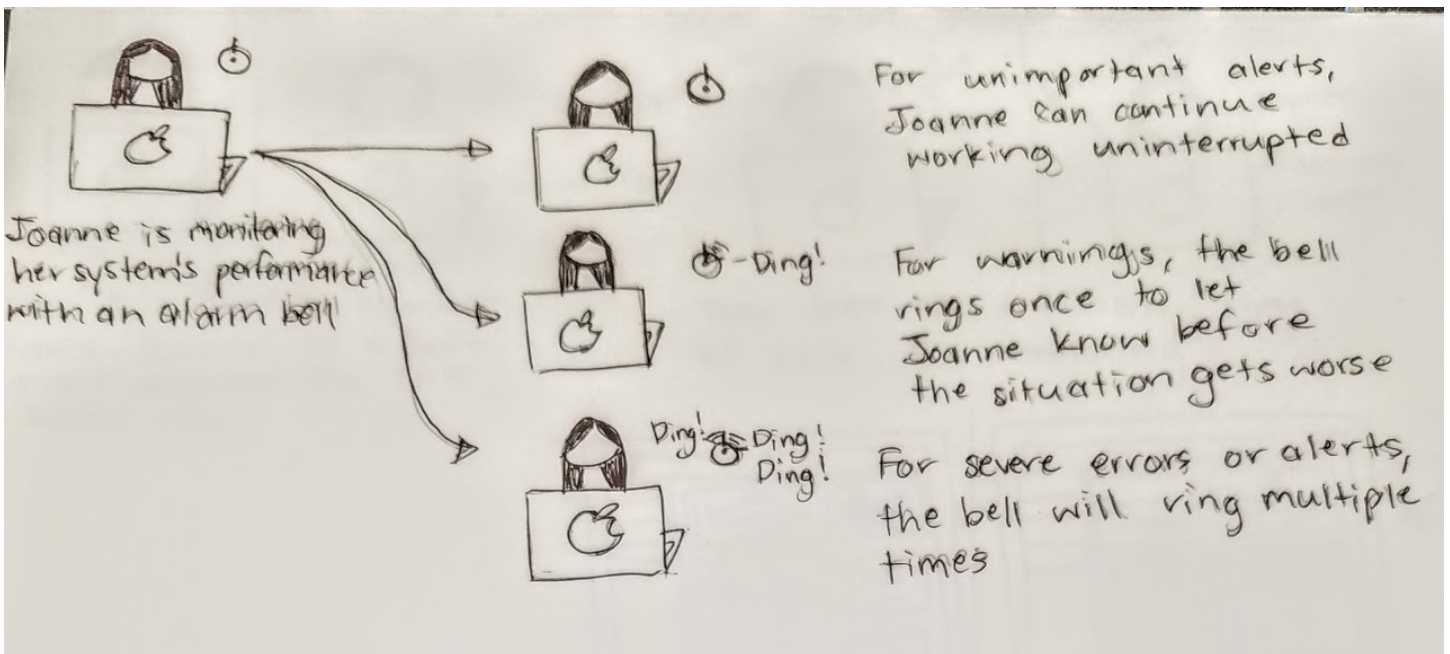
- Log event visualizer
- Query tools

### 3.4.1 Storyboards

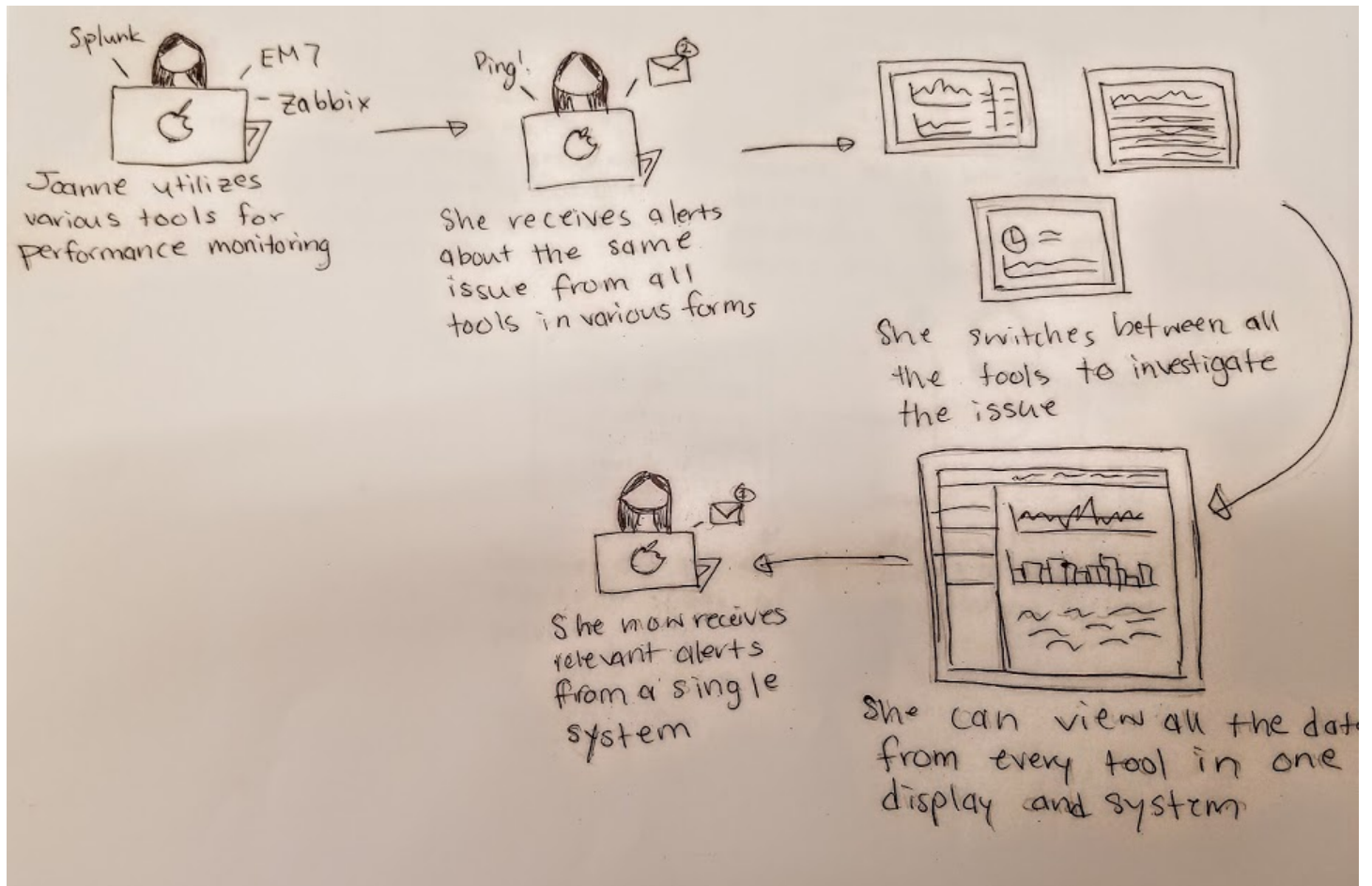




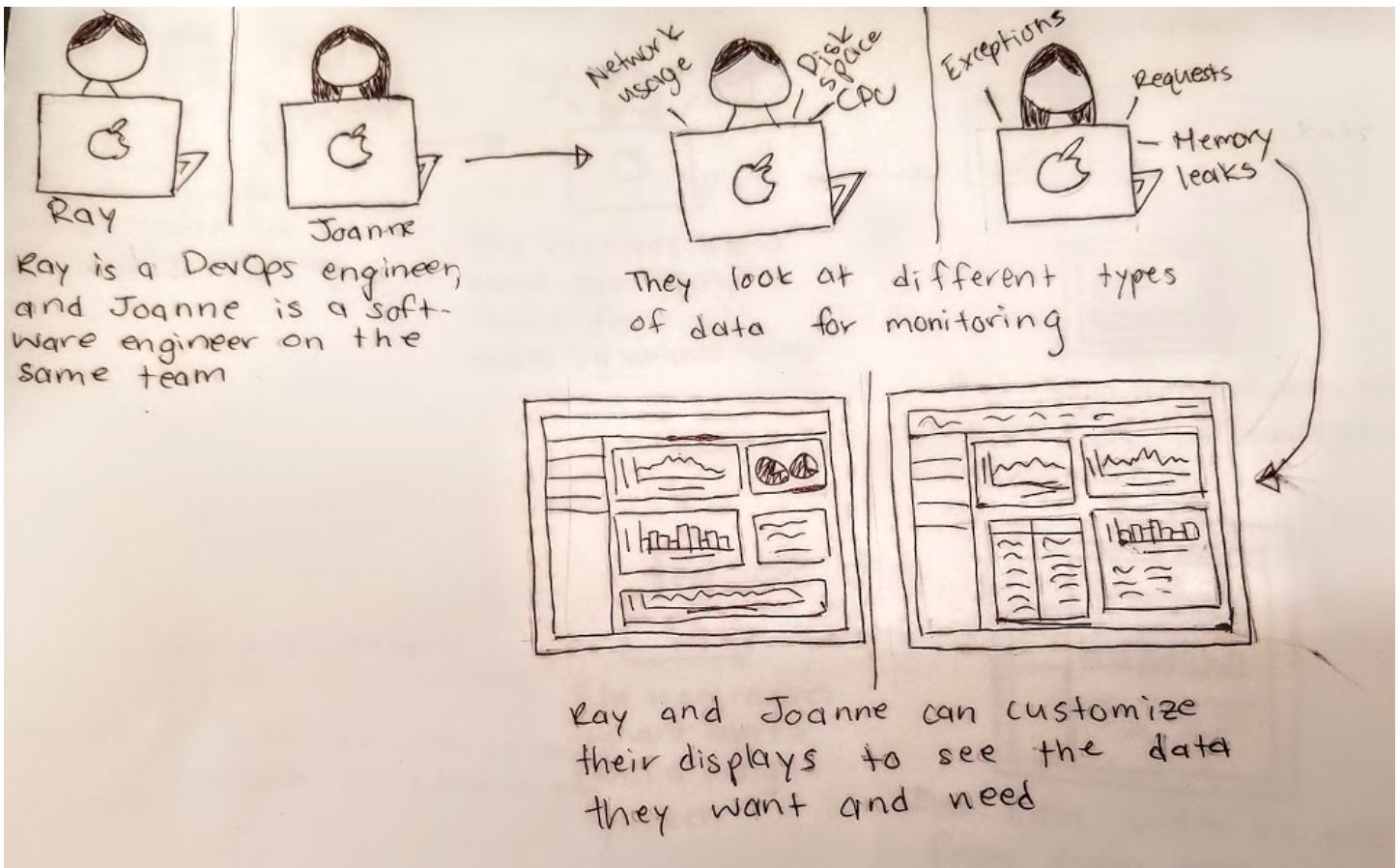
Log handling and analysis: in this scenario, the user analyses system logs to resolve a monitoring issue.



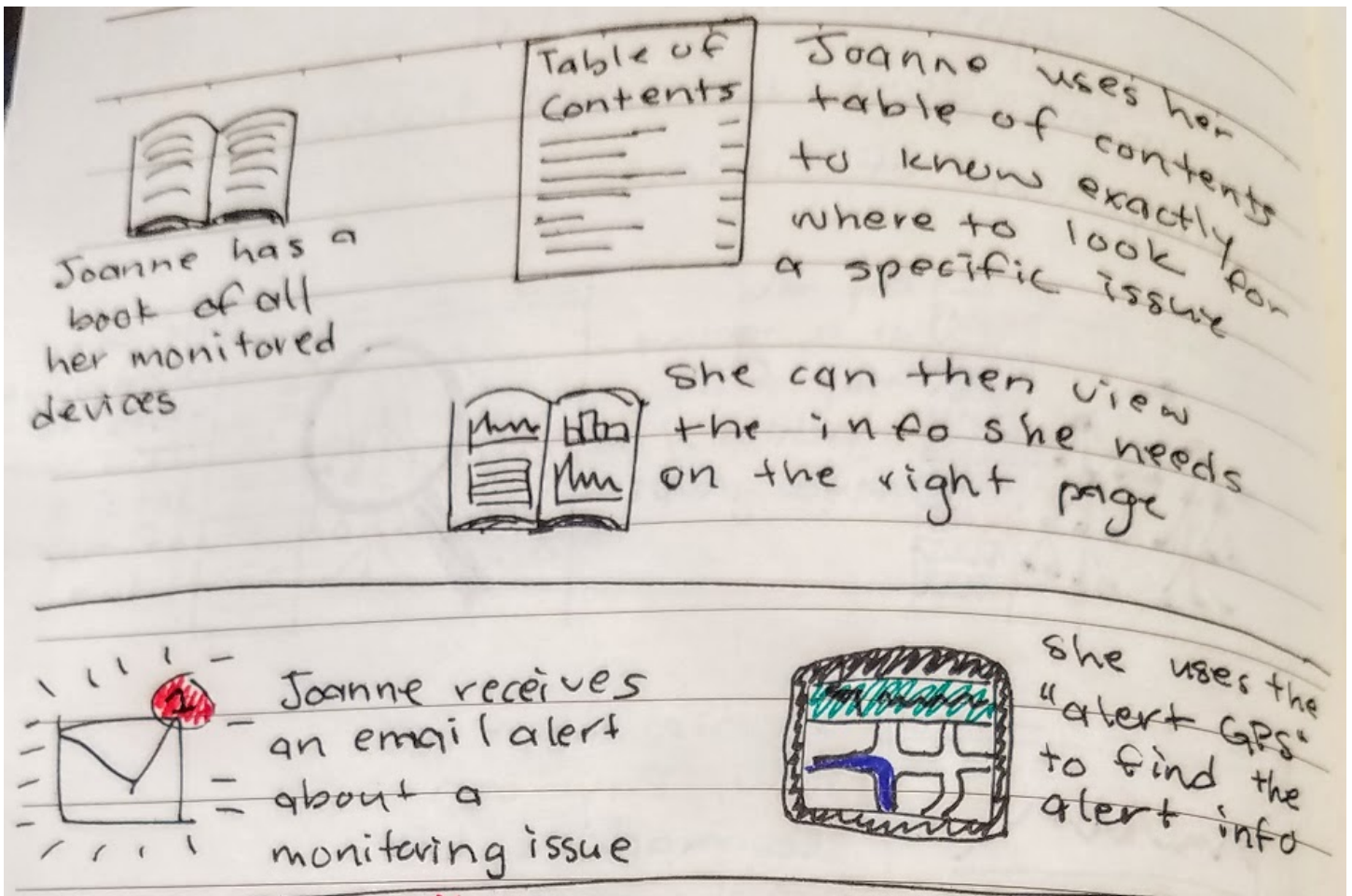
Severity handling: Depending on the severity of the issue, different alarm actions should occur so the user knows which issues require immediate attention versus those that don't.



Integration of tools: monitoring of complex networks requires multiple tools, all of which have their own alarm settings. The user in this instance can view all alerts and information from a central hub.

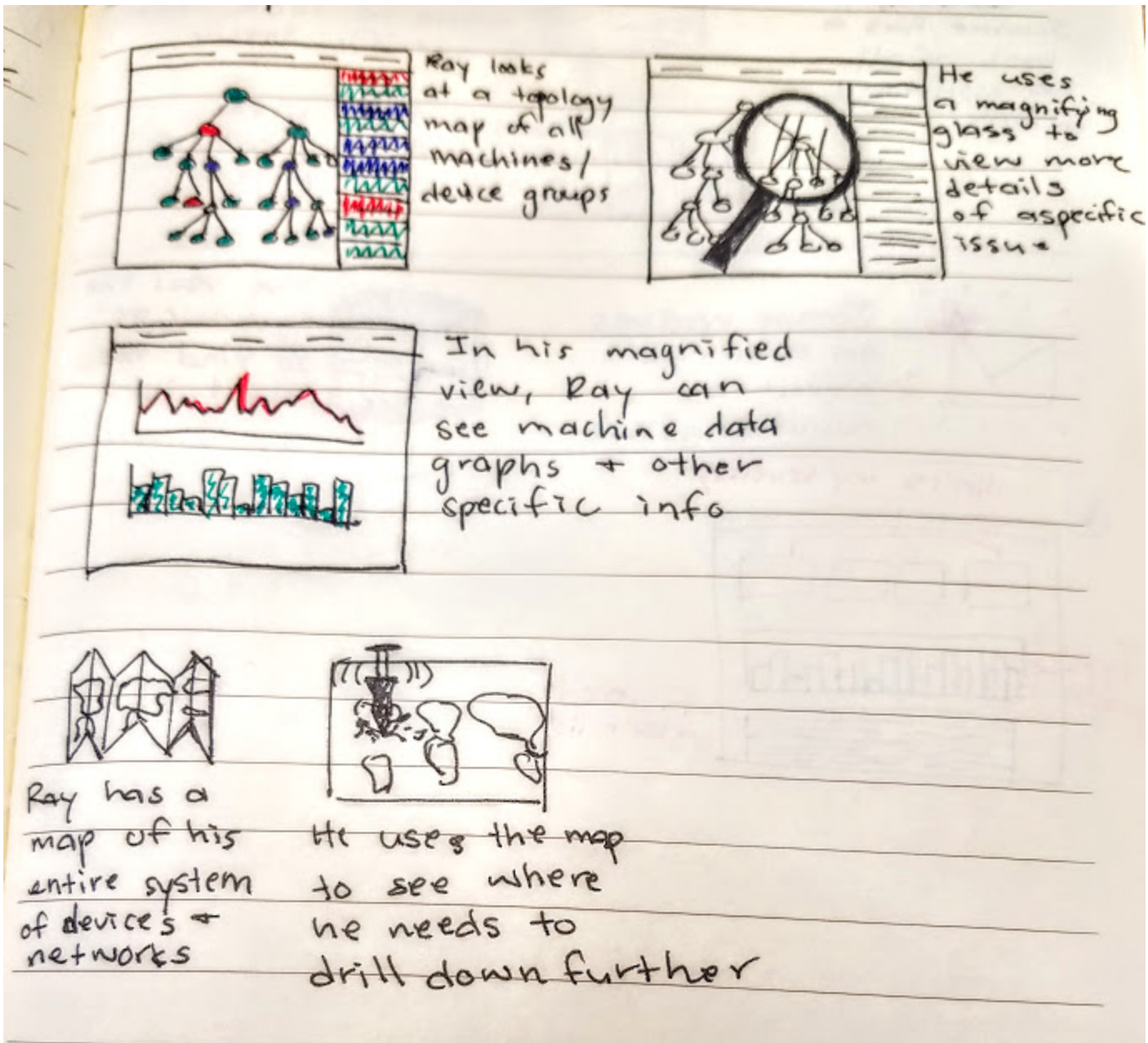


Role-based dashboard displays: Because different users have different needs in monitoring, a single dashboard display may not be adequate for meeting everyone's needs. Custom, role-based dashboard displays allow users to see the information they need, specific to them.



a. Navigation of Data: The user can find the exact location and context of data needed to address a specific issue.

b. Contextual alerts: Alerts contain a navigating link or source to take users exactly to the data relevant to the alert.



Topological Zoom: The user can drill down on a specific issue from a holistic view to a more granular view.

## 3.5 To-Be Realization

### Semantic Zoom Topology

Tom Sawyer provides an interface for viewing network topologies. By integrating this with our own network and monitoring needs, we can view the entire network of components and quickly view their statuses, quickly navigating to as-needed information.

### Dashboards

[Grafana](#) provides Graphite-based metrics for custom dashboard creation and time-series visualizations. This could be useful in monitoring, as it would give dynamic visualizations of data trends. These dashboards would also be useful for end-users who are not necessarily monitoring or resolving issues but need to be aware of system status.

### Time-Series Visualizations

Time-series visualizations are important for monitoring as they allow for highly granular and updated views of data input/output.

### Annotations

Related to logging of incident resolutions and visibility, Grafana allows for annotations, which are time-linked notes that can be added to specific graphs. These annotations can link to external sources, allowing for contextualized incident resolution.

### *Analytics*

Given Grafana's capabilities, it is able to measure web analytics in addition to collection system metrics. It integrates with a variety of data sources and applications, which allows it to provide web analytics data.

### *Monitoring*

Grafana's dynamic time-series visualizations are helpful for monitoring as engineers can view data trends as they happen and can revisit time points to investigate issues. These graphs need to be comprehensive and filterable.

### **Incident Management**

Various tools provide incident management services to integrate alerts from monitoring tools into a single location. These services would aid in reducing false alerts, and would aid in contextualizing alerts to specific data. Some incident management tools: VictorOps, PagerDuty, ServiceNow (ServiceNow integrates with EM7)

### **Automatic Discovery**

A system that automatically discovers servers, devices, applications, etc. and automatically connects and configures them together to give a holistic view of the network.

## **3.5.1 List of Potential Tools**

### [Piwik](#)

- Identify spikes
- Analyze traffic
- Features specified by piwik.com
  - Top keywords and searches
  - Top page URLs
  - Page titles
  - Operating system
  - Browser marketshare
  - Screen resolution
  - Desktop vs. mobile
  - Engagement (time on site, pages per visit, repeated visits)
  - Custom variables
  - Top entry/exit pages
  - Downloaded files

### [Apache Benchmark \(ab\)](#)

- Scaling/load balancing
- Benchmarking for Apache HTTP servers

### [Blitz.io](#)

- Load testing i.e. test with up to X number of virtual users

### [JVisualVM](#)

- JVM profiling
- GUI run from the command-line
- Comes with Java
- Currently used with NoMagic ticketing
  - .nps (CPU sampling file)

### [Splunk](#)

- Machine data digestion

- e.g. logs, configurations, data from APIs, message queues, change events, the output of diagnostic commands, call detail records
- Features specified by splunk.com
  - Collects and indexes log and machine data from any source
  - Powerful search, analysis and visualization capabilities empower users of all types
  - Apps provide solutions for security, IT ops, business analysis and more
  - Enables visibility across on premise, cloud and hybrid environments
  - Delivers the scale, security and availability to suit any organization
  - Available as a software or SaaS solution

### [Nagios](#)

- Searches log data efficiently
- Analyzes network traffic
- Alerts to issues and allows queries of logs
- Adaptable and scaleable

### [Consul](#)

- Provides service discovery
- Health checks for servers

### [Sensu](#)

- Compatibility with existing monitoring plugins
- Monitors servers, services, and application health, sends notifications

### [Logstash](#)

- Centralizes data processing from logs
- Standardize log formats

### [Kibana](#)

- Visualization of data
- Data analytics
- Integrates w/ Logstash
- Summary of streaming data

### [Zabbix](#)

- All-in-one open-source system
- Minimize downtime w/ proactive monitoring
- Monitor usage trends to manage capacity
- Notifications of errors
- Visualizations of system status
- Customization of monitoring

### [AppDynamics](#)

- Server and application monitoring
- False alert prevention
- Integrated dashboard
- Database visibility
- Server visibility
- Analytics

### Stackify

- Monitors apps & servers in one platform
- Developer-centric
- log viewer
- App performance and usage metrics
- Code analysis

## 3.5.1.1 Competitive Analysis

Software	Faster Config	Good Interface	Better Graphics	Cost Effective	Free	Automated Correction
Nagios		X		X	X	
Zabbix	X	X	X	X	X	
Solarwinds		X	X	X		
EM7	X	X	X	X		
Open NMS	X	X	X	X	X	X

## NAGIOS

Features:

- Web interface allows users to check network health from anywhere
- Reports on trends, availability, alerts, notifications,
- Monitors network redundancies and failure rates

Pros:

- Extensive set of plug-ins

Cons:

- GUI isn't very good
- Steep configuration learning curve

## ZABBIX

Features:

- Combines both monitoring and trending functionality
- Web monitoring function allows monitoring of performance over time

Pros:

- Open-source
- Well-designed GUI and overall concept
- Good alerts

Cons:

- Not suitable for 1000+ nodes because of PHP performance and GUI limitations
- Lack of real-time tests
- Complicated templates and alert rules

## SOLARWINDS

Features:

- Graphical visibility into user's network
- Monitors wireless devices for security and other issues
- Reduces difficulty in managing items

Pros:

- Excellent UI
- Customizable network mapping
- Native VMware support



Cons:

- Unable to configure alerts from web console
- Clumsy "Group Dependency" config
- Reporting doesn't have good ad-hoc reports
- Features SNMP only

## EM7

Features:

- Rapid deployment and optimized operations
- Lower "total cost of ownership"
- Support for entire system by single vendor
- Future enhancements added to one coherent system
- No modules; all functionality in base offering
- No costly integration
- Built-in dynamic firewall
- Automated back-up

Pros:

- Cost-effective compared to other large-scale solutions
- Faster installation
- Robust GUI and simple navigation

Cons:

- Doesn't collect network-flow info
- Doesn't provide topology map nor can it correlate network and system outages

## OPENNMS

Features:

- Easy installation
- Event and notification management
- Features thresholding
- Alarms and automation
- Sends notifications regarding noteworthy events

Pros:

- Free licensing
- Full featured and highly flexible
- "Path outages" and "minimize excessive alerting"
- Reasonable support costs

Cons:

- Steep learning curve
- Interface not very intuitive
- Requires learning and modifying various config files for customization

## 3.5.2 Paper Prototype

### Alert Customization

Customization 1

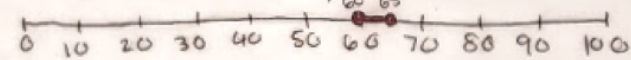
## Main Servers Applications Databases

Alerts
Alert List Alert Settings Archive
Logs
Settings

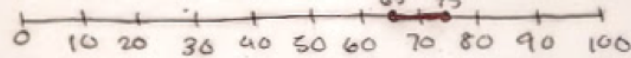
① Select metric:  or Add Custom

② Set alert conditions

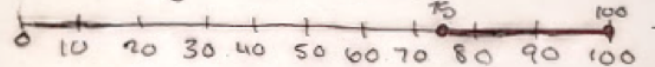
% for low severity



% for moderate severity



% for high severity



③ Set time conditions

Alert if metric exceeds threshold for more than  minutes.

④ Set alert type

Low severity:  Add another

Moderate severity:  Add another

High severity:  Add another

⑤ Set priority

The user adds CPU Utilization as a metric and sets alert conditions using a slider for low, moderate, and high severity alerts. The user can then set what type of alerts to receive given the severity of the alert.

Customization 2

## Main Servers Applications Databases

Alerts

Alerts List  
Alert Settings  
Archive

Logs

Settings

Set metric:  or Add custom

Set conditions

- Critical: Alert if metric   % for more than  minutes
- Moderate: Alert if metric   and  % for more than  minutes
- Low: Alert if metric   and  % for more than  minutes

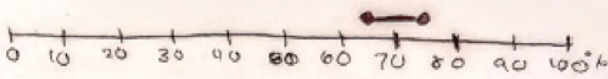
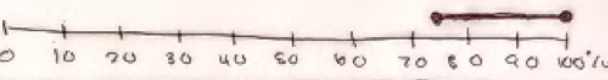
Set alerts

- if metric is Critical
- if metric is Critical
- if metric is Moderate
- if metric is Low
- + Add alert

The user adds CPU Utilization as a metric and can set what is considered critical, moderate, or low. The user can then add what type of alert to receive given the level of severity.

Customization 3

## Main Servers Applications Databases

Alerts	Add metric: <input type="text" value="CPU Utilization"/> or <u>Add Custom</u>
Alerts List Alert Settings Archive	Set alert levels
Logs	Minor level: 
Settings	Major level: 
	Set alert conditions
	<input type="checkbox"/> Alert is cleared <input type="checkbox"/> Alert has existed for more than... <input checked="" type="checkbox"/> The alert severity is... <input checked="" type="checkbox"/> The metric exceeds threshold for more than... <input checked="" type="checkbox"/> The metric is below threshold for more than... <input type="checkbox"/> The metric occurs... <input type="checkbox"/> The time of day is between... <input type="checkbox"/> The day of the week is... <input type="checkbox"/> The date is... <input type="checkbox"/> Other...
	Select action
	<input checked="" type="checkbox"/> Send email to... <input checked="" type="checkbox"/> Send text message to... <input type="checkbox"/> Write alert to log <input type="checkbox"/> Suppress alert <input type="checkbox"/> Execute script <input type="checkbox"/> Other...
	Rule description
	<div style="border: 1px solid black; padding: 5px;"> <p>For <u>minor level</u> alerts...  <u>Send email to expert@jpl.nasa.gov</u>  <u>and send text message to (555)600-2000</u>            if metric exceeds threshold for <u>more than 30 minutes</u>            or if metric is below threshold for <u>more than 30 minutes</u></p> </div>

The user specifies adding a CPU Utilization metric. The user can set thresholds for what is considered minor or major severity. The user can then select from a check list various alert conditions and the action that should occur with each condition. The rules for each alert condition are then displayed.

## Dashboard

# Main Servers Applications Databases

Alerts

Logs

Settings

## OVERVIEW

### System Summary

<b>72%</b> CPU Utilization	<b>*86%</b> Memory Usage	<b>64%</b> Disk I/O
-------------------------------	-----------------------------	------------------------

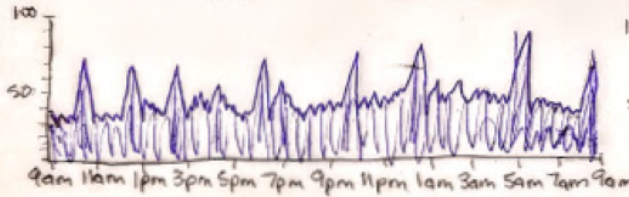
### Server Health

<b>10</b> Healthy	<b>0</b> Warning	<b>1</b> Severe
----------------------	---------------------	--------------------

## METRIC GRAPHS

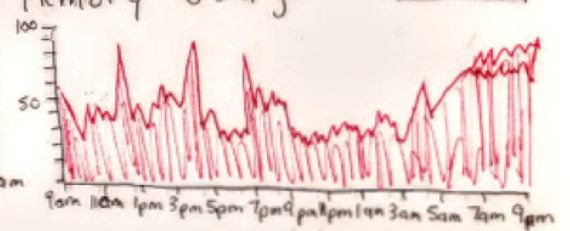
### CPU Utilization

Last 24 hrs



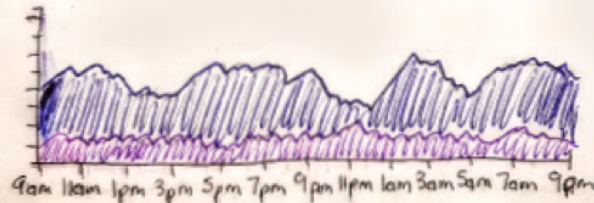
### Memory Usage

Last 24 hrs



### Disk I/O

Last 24 hrs



### Recent Alerts

~~~~~	~~~~~	~~~~~	~~~~~	~~~~~	~~~~~
~~~~~	~~~~~	~~~~~	~~~~~	~~~~~	~~~~~

### Top Users

~~~~~	~~~~~	~~~~~	~~~~~
~~~~~	~~~~~	~~~~~	~~~~~
~~~~~	~~~~~	~~~~~	~~~~~
~~~~~	~~~~~	~~~~~	~~~~~

The user can view the main dashboard for a quick text and visual summary of system status so the user can quickly glance at overall system health.

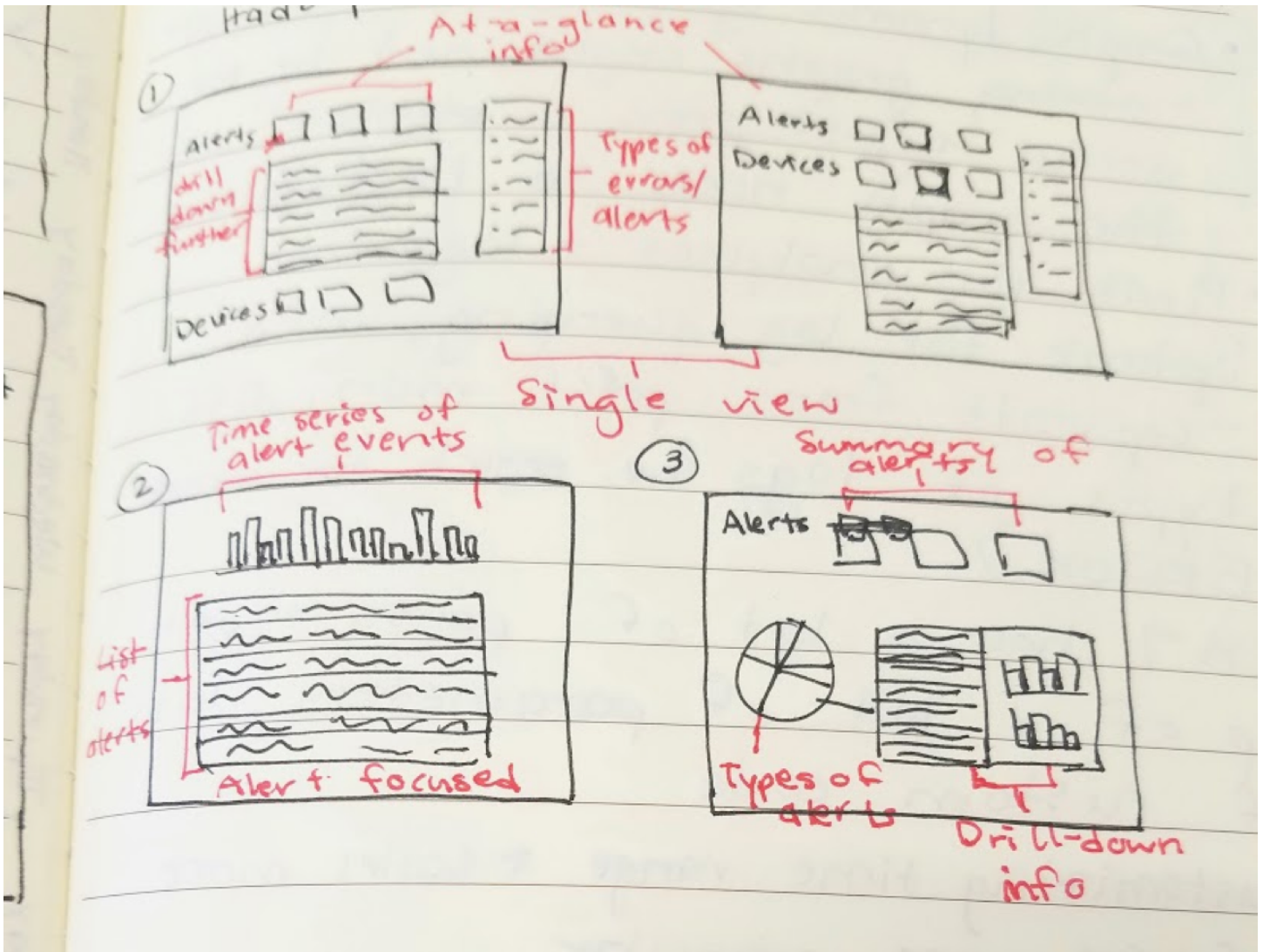
Servers

## Main Servers Applications Databases

Alerts	SERVERS			
Logs	Summary			
Settings	<div style="display: flex; justify-content: space-around; align-items: center;"> <span>● Healthy: 10</span> <span>⚠ Warning: 0</span> <span>✖ Severe: 1</span> </div>			
	Server List:			
	Server Name	Health	Latest Activity	Actions
	cae-ems-bamboo	✖ Severe	7/20/2016 3:02PM	<a href="#">VIEW</a>
	cae-ems-maple	● Healthy	7/20/2016 5:52PM	<a href="#">VIEW</a>
	cae-ems-apricot	● Healthy	7/18/2016 10:17AM	<a href="#">VIEW</a>
	cae-ems-pine	● Healthy	7/18/2016 6:01PM	<a href="#">VIEW</a>
	cae-ems-evergreen	● Healthy	7/20/2016 12:22PM	<a href="#">VIEW</a>
	cae-ems-redwood	● Healthy	7/16/2016 4:45PM	<a href="#">VIEW</a>
	cae-ems-fig	● Healthy	7/19/2016 11:05AM	<a href="#">VIEW</a>
	cae-ems-sequoia	● Healthy	7/18/2016 7:13PM	<a href="#">VIEW</a>
	cae-ems-palm	● Healthy	7/15/2016 7:20AM	<a href="#">VIEW</a>
	cae-ems-oak	● Healthy	7/19/2016 3:36PM	<a href="#">VIEW</a>
	cae-ems-apple	● Healthy	7/17/2016 8:09AM	<a href="#">VIEW</a>

To dig deeper into the source of issues, the user can view a server list and see a quick summary of server health.

## Summary Landing Pages



Variations of summary landing pages. Each reduces a front page to simply a list of alerts or health status. Some allow for single pane drill down info or broad info for further investigation.





## Main Servers Applications Databases

Alerts

Alerts List  
Alert Settings  
Archive

Logs

Settings

Set metric:  or Add custom

Set conditions

- Critical: Alert if metric   % for more than  minutes
- Moderate: Alert if metric   and  % for more than  minutes
- Low: Alert if metric   and  % for more than  minutes

Set alerts

- if metric is Critical
- if metric is Critical
- if metric is Moderate
- if metric is Low
- + Add alert

In this version, the user can similarly set thresholds for different severity levels, but instead uses drop down menus and text boxes to specify the thresholds. The user can set the type of alert to receive for each given severity level.

Version 2:

Main Servers Applications Databases

Alerts

Alerts List  
Alert Settings  
Archive

Logs

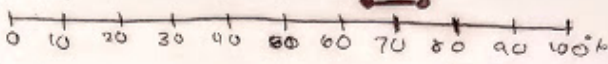
Settings

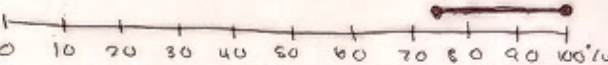
---

Add metric: CPU Utilization or Add Custom

---

Set alert levels

Minor level: 

Major level: 

---

Set alert conditions

<input type="checkbox"/> Alert is cleared	<input type="checkbox"/> The metric occurs...
<input type="checkbox"/> Alert has existed for more than...	<input type="checkbox"/> The time of day is between...
<input checked="" type="checkbox"/> The alert severity is...	<input type="checkbox"/> The day of the week is...
<input checked="" type="checkbox"/> The metric exceeds threshold for more than...	<input type="checkbox"/> The date is...
<input checked="" type="checkbox"/> The metric is below threshold for more than...	<input type="checkbox"/> Other...

---

Select action

<input checked="" type="checkbox"/> Send email to...	<input type="checkbox"/> Suppress alert
<input checked="" type="checkbox"/> Send text message to...	<input type="checkbox"/> Execute script
<input type="checkbox"/> Write alert to log	<input type="checkbox"/> Other...

---

Rule description

For minor level alerts...  
Send email to expert@jpl.nasa.gov  
and send text message to (555)600-2000  
if metric exceeds threshold for more than 30 minutes  
or if metric is below threshold for more than 30 minutes

The user can set minor and major severity level thresholds. This version gives the greatest amount of customization with a checklist as well as the greatest amount of customization of alert actions.

### Testing

To test which version of alert customization best suits our team, we plan to show each version to users. Metrics to examine:

- What level of customization is needed?
- Which customization method is the most usable?
- How should severity levels be handled?

In comparing results, we will examine what types of alert customization best fits our needs whether that be a single version or a combination of all versions presented.

### Findings

From user testing with the various alert customization, we found that the current system used for monitoring has been sufficient in customizing alerts. The focus will instead shift to the information contained within alerts and how that information is conveyed. We will also focus on dashboard displays of information and how these displays affect awareness of monitoring situations.

### Dashboard Displays

# Main Servers Applications Databases

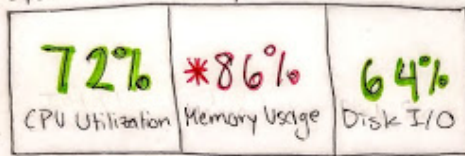
Alerts

Logs

Settings

## OVERVIEW

### System Summary



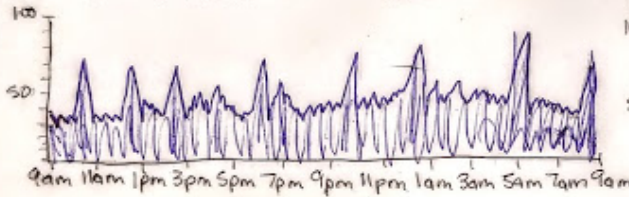
### Server Health



## METRIC GRAPHS

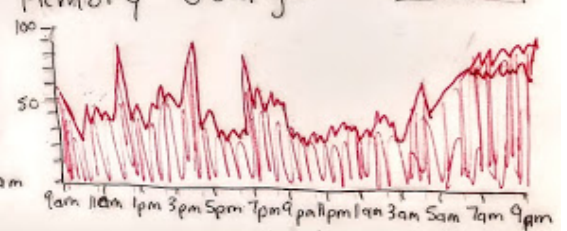
### CPU Utilization

Last 24 hrs



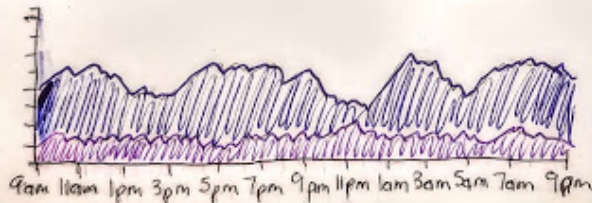
### Memory Usage

Last 24 hrs



### Disk I/O

Last 24 hrs



### Recent Alerts

~~~~~	~~~~~	~~~~~	~~~~~	~~~~~	~~~~~
~~~~~	~~~~~	~~~~~	~~~~~	~~~~~	~~~~~

### Top Users

~~~~~	~~~~~	~~~~~	~~~~~
~~~~~	~~~~~	~~~~~	~~~~~
~~~~~	~~~~~	~~~~~	~~~~~
~~~~~	~~~~~	~~~~~	~~~~~

The dashboard displays the current system status and vital stats as well as some basic machine data graphs.

## Main Servers Applications Databases

Alerts

## SERVERS

Logs

## Summary

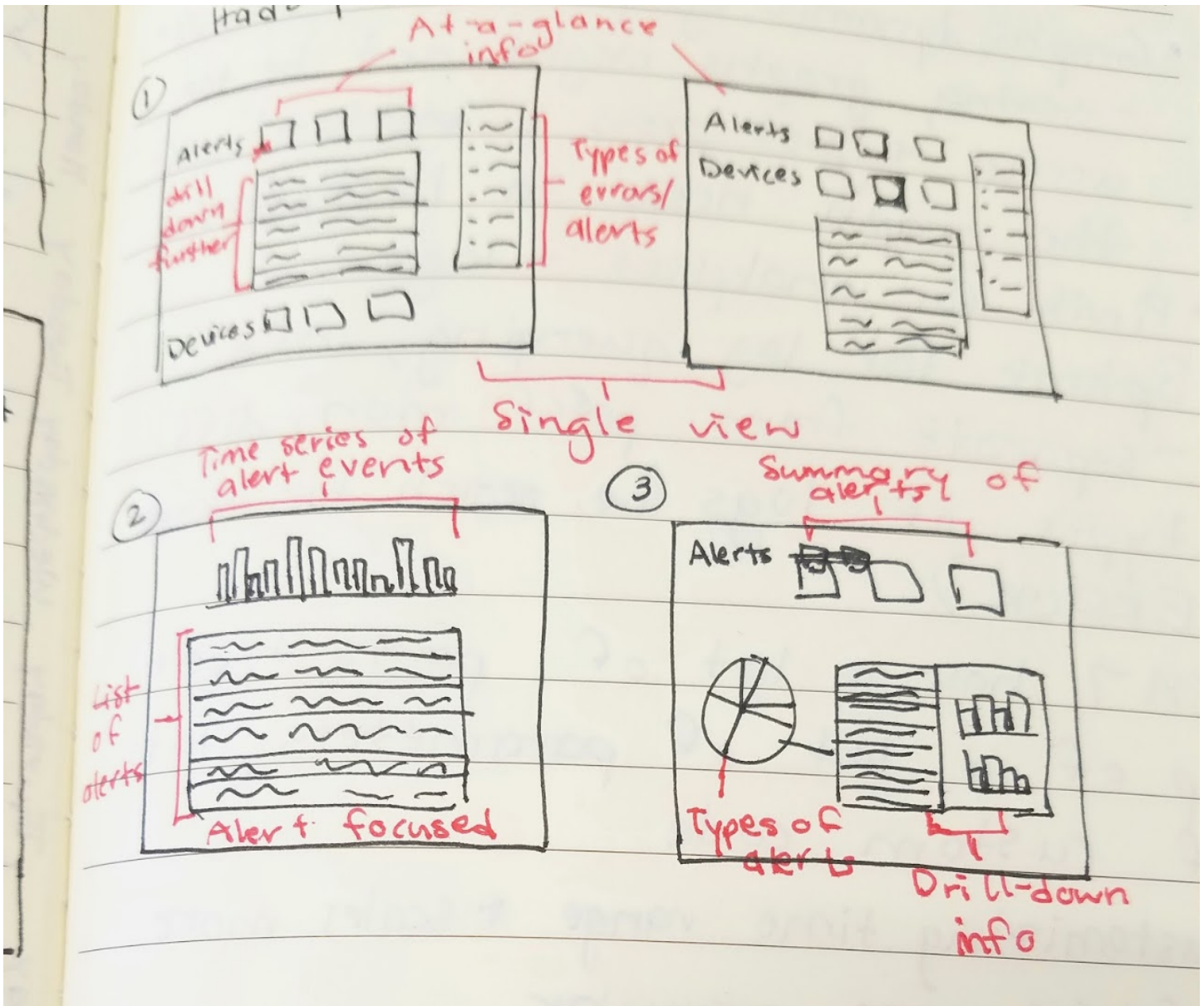
Settings

● Healthy: 10    ⚠ Warning: 0    \* Severe: 1

## Server List:

Server Name	Health	Latest Activity	Actions
cae-ems-bamboo	* Severe	7/25/2016 3:02PM	<a href="#">VIEW</a>
cae-ems-maple	● Healthy	7/20/2016 5:52PM	<a href="#">VIEW</a>
cae-ems-apricot	● Healthy	7/18/2016 10:17AM	<a href="#">VIEW</a>
cae-ems-pine	● Healthy	7/18/2016 6:01PM	<a href="#">VIEW</a>
cae-ems-evergreen	● Healthy	7/20/2016 12:22PM	<a href="#">VIEW</a>
cae-ems-redwood	● Healthy	7/16/2016 4:45PM	<a href="#">VIEW</a>
cae-ems-fig	● Healthy	7/19/2016 11:05AM	<a href="#">VIEW</a>
cae-ems-sequoia	● Healthy	7/18/2016 7:13PM	<a href="#">VIEW</a>
cae-ems-palm	● Healthy	7/15/2016 7:20AM	<a href="#">VIEW</a>
cae-ems-oak	● Healthy	7/19/2016 3:36PM	<a href="#">VIEW</a>
cae-ems-apple	● Healthy	7/17/2016 8:09AM	<a href="#">VIEW</a>

A list of servers shows the status of each server/device and an immediate view of the status of all servers/devices at once to give an at-a-glance view of the overall health of the servers/devices.



Various designs for an alert dashboard to show quick info about the severity and types of alerts.

### Testing

We want to test how users comprehend data on the dashboards and what aspects they find useful for their needs. We will be testing the following aspects:

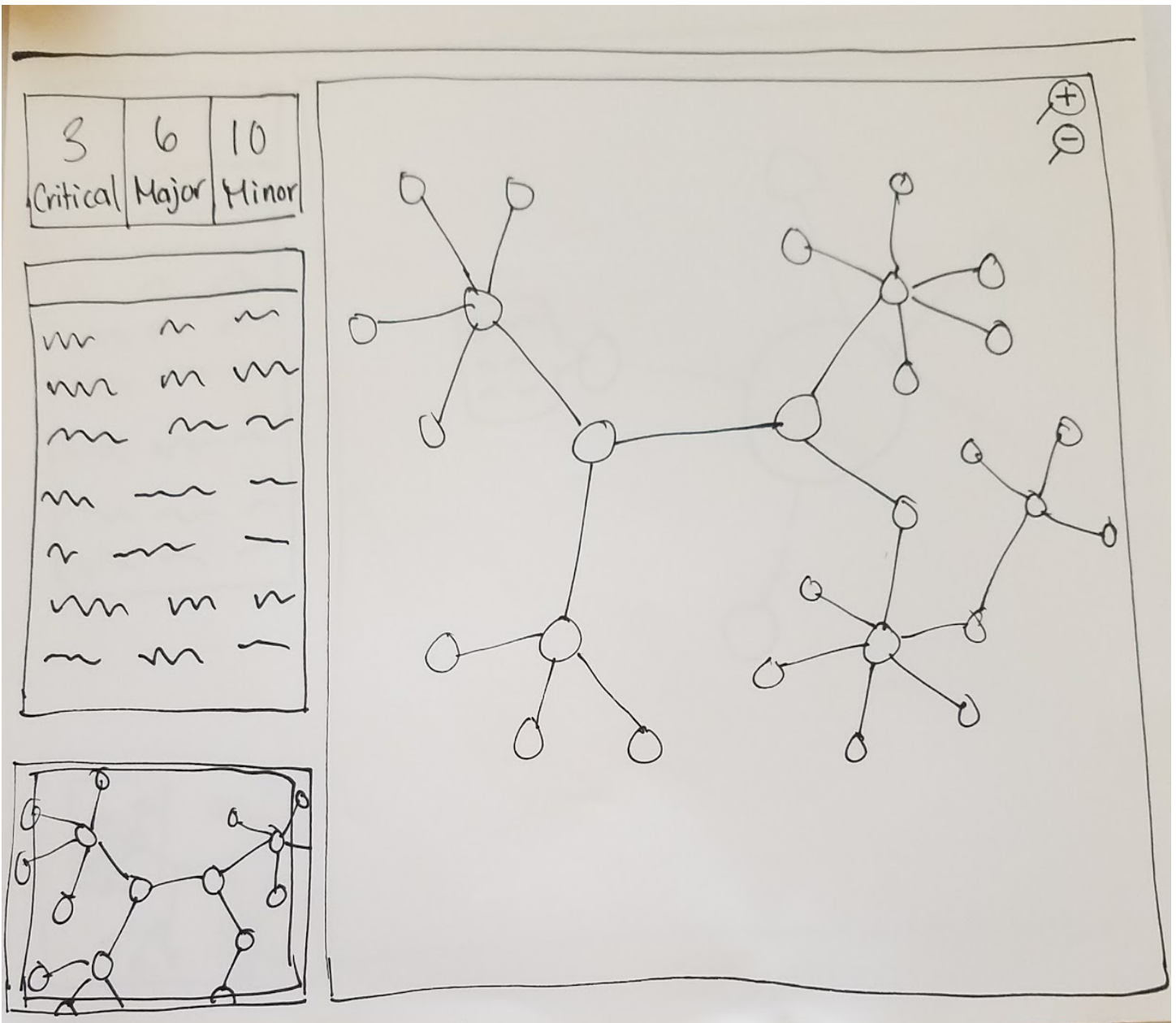
- What features are needed in a dashboard display?
- What is the best way to present data in a dashboard?
- What level of visibility do you want to see on the landing page dashboard?

### Findings

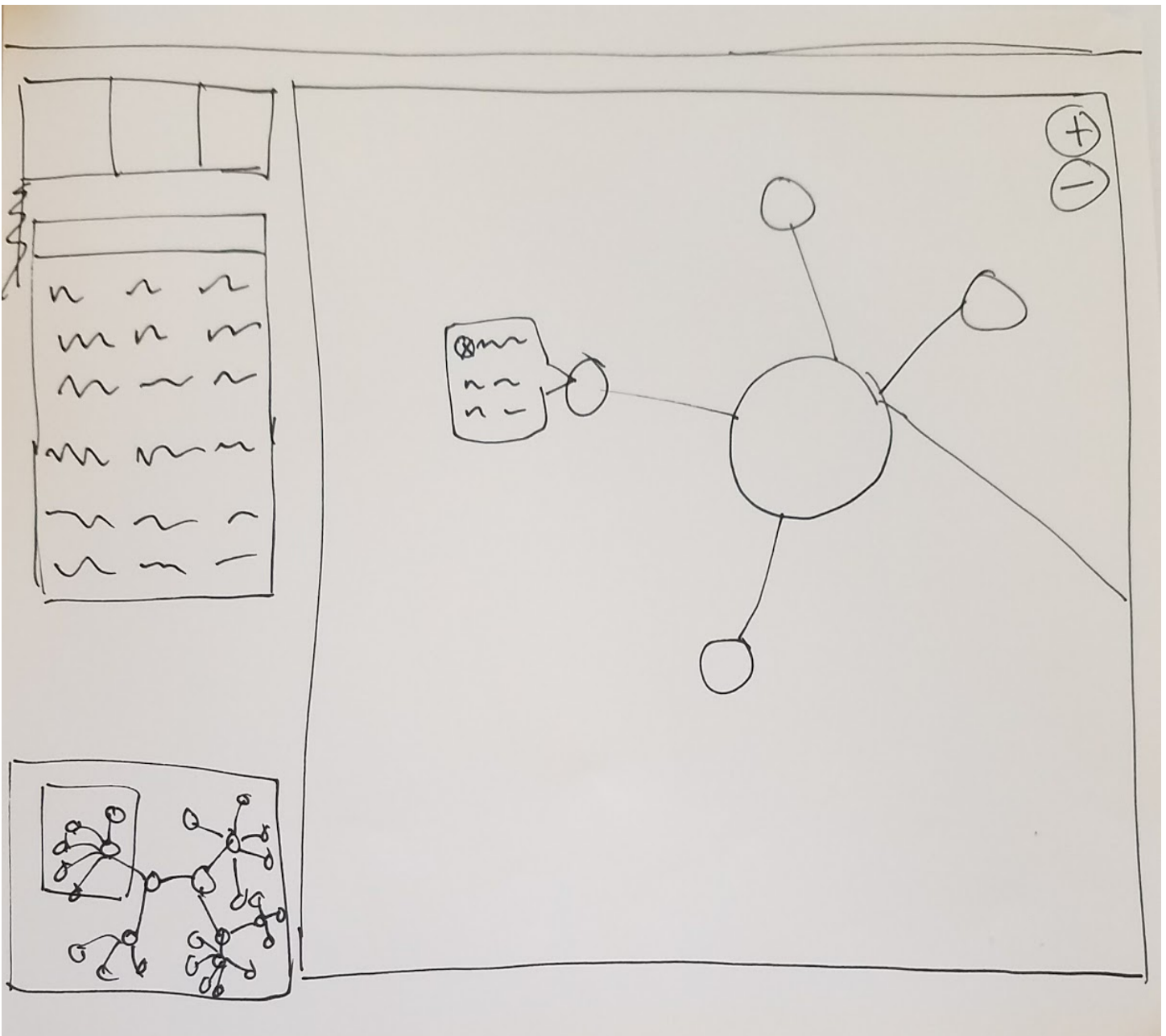
In presenting the dashboard displays to users, we found graphs are the most important part of a dashboard as graphs are what reveal important trends in data. All users felt that the dynamic display of graphs was the most important part of visual dashboards.

DevOps users liked the summary bar but preferred the summary to contain alerts. They stressed the importance of dynamic graphs. Operations users wanted to see all important metrics on the dashboard screen so they can see whether their current processes are overloading the system or whether there are any metrics they need to be wary of. They also wanted an operational log of some sort to show recent events and installations.

### Semantic Zoom/Topological Map



The dashboard displays a network topology with a contextual mini-map and a details list. Meant to capture important information on a single dashboard view



Clickable zoom to look at specific elements of the network and information about that element.

*Testing*

## 4.2.2 High-fidelity Prototype Testing

We developed high-fidelity prototypes in Sketch that mimic real-world monitoring and tested this prototype with users to understand what needs it addresses and what pain points remain. The main areas we tested with the high-fidelity prototypes are:

- Semantic zoom dashboards
- Dynamic graphing
- Contextual alerts

### Semantic Zoom Dashboards

We want to test whether a semantic zoom feature would be useful for the team, examining how it affects situational awareness and interactions between the user and the system. We will test two options of dashboards: a topological zooming dashboard and a more traditional list view dashboard, looking to see which is easier to navigate and which provides the most awareness of the overall system.

### Graphical Dashboards

We also want to test the design of graphical displays of data on a single dashboard. The graphs need to be easily comprehensible and granular enough to display trends. We will utilize Grafana to display sample monitoring graphs and test user reactions to this graphing style.

### **Contextual alerts**

A common problem noted by our engineers is that alerts do not contain enough information or context to aid them in investigating issues. We will test how users react to linking more context to alerts, which should improve investigation and resolution of issues.

### **Findings**

- The semantic zoom feature is neat, but for DevOps users, they don't to view all information at a topological level, and thus, it might be unfeasible or undesirable to place a topological map or list view into a single view
- Other users liked the topological zooming as this would give them an overall picture of the ecosystem and greater context as to the root of issues. They want to view how different services and servers are linked together so they can see how different components are affected.
- Users instead wanted a concise at-a-glance summary of open alerts or problems that are categorized by type and severity.
- Users agreed that contextualized alerts would be very useful. Linking specific areas of machine data to specific alerts would allow users to quickly identify where the problem is and what information is relevant to it.
- Graphs not only need to easily show trends but need to be dynamic, meaning that users can adjust the time range and scale to view both holistic and granular graph data.
- Graphing styles and colors don't matter as much as the adjustability of the graph
- Users liked the flexibility and handling of the Grafana graph displays, which will be useful for different teams given the customizability and open-source nature of the site.
- Users want to be able to calculate metrics on-the-go.

### **Priorities**

1. Navigatable topological map: displays overview of how components are connected together and they live in the ecosystem
2. Summary landing page: design a separate landing page that concisely displays information at-a-glance and is alert-focused
3. Log handling: provide a design of an interface to handle analytical logs and monitoring logs and visualizations of log trends, and provide a recommendation of a log handling tool
4. Integrating everything together: provide a holistic specification of integrating the various tools in our team's monitoring workflow into a single system