Test Plan

Organizational Alert System

For

CS 895 MSE Project

Department of Computer Science

Kansas State University

Submitted to

Dr. Mitch Neilsen

Dr. Torben Amtoft

Dr. Scott Deloach

Submitted by

Angela Hall

3/11/2019

Introduction

This document provides the checklist to be used by peers for the formal technical inspection. The artifact inspected will be the Web API. The inspectors will verify that the Web API meets the critical requirements specified in the Vision Document and Test Plan.

Information for Inspectors

This project is designed to be a silent alert system for organizations. In the event of an active-shooter or other situation, any user can trigger a silent alarm and a pop-up message will flash on the screen of all computers in the organization. There are 4 customizable alarms and one quick message button. Client applications query the server every 8 seconds to retrieve the alarm state.

The full system consists of a web server, a web GUI, a Rest API, and a java-based client program. You will be evaluating the web GUI which interacts with the Rest API. Instructions for accessing the server will be provided in the email accompanying this document.

The web application implements the following critical requirements:

CR1. Each alarm button will trigger an alarm state
CR2. One button will stop all alarms
CR3. Users can customize alarm messages
CR4. Users can see a log of all alarm activity
CR5. Rest API provides most recent active alarm

Inspection Checklist

Inspection Item	Pass/Fail	Comments
CR1 Each alarm button triggers an alarm state Notes: 1. Click on “Send Alarm” on the top Navigation Bar 2. Press any Alarm Button 3. Verify that message appears under “current status of alarms” (may take up to 8 seconds). 4. Repeat for all alarm buttons.	Pass	It is disconcerting from a web design point to hover over a clickable area that does not indicate it is clickable. I would recommend using CSS to make the cursor style “pointer” or a different cursor from the text selector.
CR2 One button will stop all alarms Notes:	Pass	

Begin with one active alarm. On the “Send Alarm” page, the active alarm will be flashing in the status area. Click the “Stop Alarm” link Alarm should stop flashing		
CR3 Users can customize alarm messages Notes: 1. Click on the “Customize Application” link on the top navigation bar. 2. Change the messages/titles of the alarms. 3. Click on the “Save Alarm Messages” button 4. Return to the “Send Alarm” page 5. Verify that the alarms display the messages you created.	Pass	The CSS for this page is not user intuitive, It should be clear where the text boxes are at first glance. In addition, consider expanding divs for compartmentalizing different customizations.
CR4 Users can see a log of all alarm activity Notes: 1. Activate several alarms 2. Click on notifications 3. Verify that you can see a log of your activated alarm activity	Pass	Consider updating this table into a DataTable or Grid so that as more records come in the table can easily paginate.
CR5 Rest API provides most recent active alarm. Notes: The raw XML output of the Rest API can be seen in a browser by going to http://IPADDRESS:PORT/resources/alarms.php. Alternatively, the in-page notification on the Send Alarm page uses the API, so you can rely on that. Verify that the API output reflects the most recently triggered alarm.	Pass	You might consider changing or adding API functionality that to return a JSON object instead of XML.