CS Capstone Design Document

April 18, 2019

Slide Sentinel

Prepared for

Oregon State University

Dr Chet Udell ______________________ Signature ________ Date

Prepared by

Group 29
The Slide Sentinel

James Stallkamp
Lucas Campos-Davis
Kevin Koos

Abstract

This document outlines the design for the Slide Sentinel 2018-19 senior capstone project. This document includes both the design for the Slide Sentinel and LOOM integration this team will be working on. Outlined below is the design of the projects software for both hubs and the online client.

CONTENTS

Revisions: Revised April 2019

Section	Original	New
Slide Sentinel	N/A	Additional design elements describing the system hub
All	References to ONEhub	Integration of Freewave radio and Rockblock+ into LOOM
All	References to 4G	Client has opted to just use satellite communication
Slide Sentinel Visualization	N/A	Added section about an all in one web client integrating all work on networking into one
ONE Hub	Sections ONE hub software and interpreter	Moved into Slide Sentinel integration section
Slide Sentinel Enclosure	3D printed and made in house.	Combination of waterproof boxes and a mounting plate.
POSTproxy	N/A	Added section on design of the middleman site for Rock7 ->Google sheets
Integrated Website	N/A	Recently added. Describes some initial design choices made for the integrated website. Some features and design choices are still being determined.

1 Introduction

1.1 Purpose

The documents purpose is to provide an outline of the design going into the software components of the Slide Sentinel project.

1.2 Scope

Slide Sentinel project consists of a central hub, an array of sensors, and an online mapping visualization. Sensors will collect positional data whenever a change is detected, and sends that data to the central hub. Sensors will collect data on position, orientation, time, and status. The central hub is set to communicate with around 20 sensors in total. Data collected by the central hub will be stored locally for redundancy and sent to an online database using satellite communication where it can be analyzed and viewed in a map.

1.3 Intended Audience

The Slide Sentinel project intended for Weyerhauser and integrated components for LOOM is intended for the OPEnS lab headed by Chet Udell.

1.4 Definitions and Abbreviations

- IoT - Internet of Things
- LoRa - Long Range
- nRF - (Nordic) Radio Frequency
- GPS - Global Positioning System
- RTK - Real Time Kinematic (GPS)
- WAN - Wide Area Network
- LPWAN - Low Power WAN
- LTE - Long-Term Evolution
- NB-IoT - Narrow Band IoT

1.5 References

[1] G. L. Marissa Kwon, “Project description for capstone,” document hosted on OPEnSLab google drive.

[2] G. Lund, “Landslide monitor development targets,” document hosted on OPEnSLab google drive.

[3] M. Wright, “The open sound control 1.0 specification,” *OpenSoundControl*, 2002.

1.6 Document Overview

Each of the components listed below is focused on the software side of the component. Some components listed such as those referring to the hubs themselves have some substantial hardware design behind them outlined in other documents for the project. For Slide Sentinel, the components are the enclosure, online visualization, and the software running on the hub itself.

2 System Overview

The Slide Sentinel system will provide a way to monitor landslide-prone areas. In addition to passive monitoring uploaded on an interval, the Slide Sentinel system will detect sudden shifts in movement and make an immediate report back to the spread sheet. Besides the node and hub hardware component, the system will also provide a spreadsheet and map client online so users can view both the raw data and their relative positions on a map.

3 System Architecture

Slide Sentinel has a total of 4 components in total with 3 of them being focused on by the capstone team for design. There are the many sensors collecting positional data, the hub collecting all the data, the weather-resistant enclosure for the hub, and a Google maps visualization build from data stored in a Google sheets page. Below is a diagram of the components are their relationships to each other. The Freewave radio which deals with Hub to Node communications and the Rockblock+ for outgoing communication will be integrated into LOOM following their specific requirements.

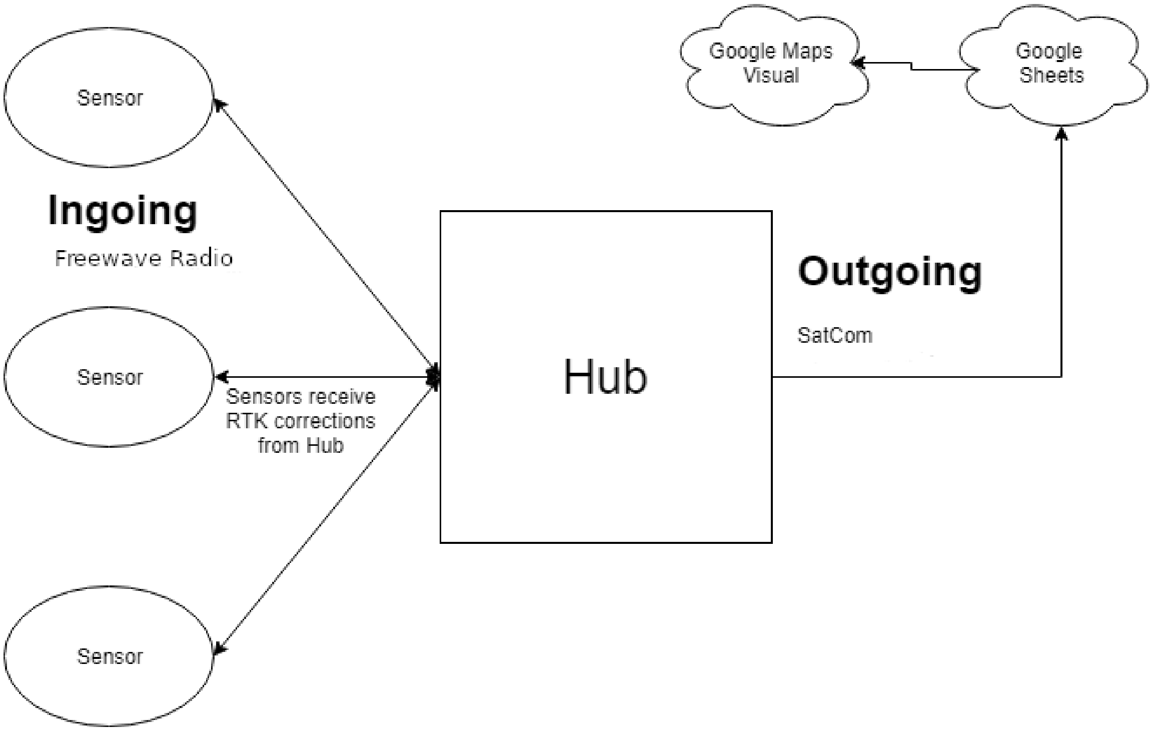

Fig. 1. Relational Diagram of the Slide Sentinel project

4 Major Design Components

The following is a component overview of the components in the Slide Sentinel.

5 Slide Sentinel

5.1 Slide Sentinel Hub

The Slide Sentinel Hub will collect data from an array of nodes that report positional data whenever a positional change is detected. The Slide Sentinel Hub will receive data with the nodes over LoRa and communicate RTK correctional data for 1cm precise positioning. The data will then be stored locally on an SD card and also sent to the online visualizer via satellite communication. Technicians will be able to access the data stored on the SD card if the wireless communications fails. The Slide Sentinel Hub will be based on an Adafruit Feather M0 with a number of additional modules outlined in the bill of materials which include devices such as real time clock, GPS evaluation board, SD card reader, Rockblock+ and a custom board for interfacing with it.

Based off a set interval

5.2 Slide Sentinel Data Client

The Slide Sentinel Data Client is a google spreadsheet with a script running in it as a web app which will receive data from the hub and properly parse the data. Data received is based off the STI 030 NMEA string specification and is parsed to readable forms of the data. Data will be ordered in a logical way and added to the spreadsheet in descending order by time. Since the strings received do not have either an absolute time stamp or a date, the data for each message received is figured out from the recorded UTC time and the local received time from the application. An additional sheet will be present which will filter out the most precise positional fixes based off the type of positional fix it was and the RTK ratio.

5.3 Slide Sentinel Map Client

The Slide Sentinel Visualization will be the only user interface designed by the capstone team. It will be a web page that will be portable to most web browsers. It will display the data that is reported by the Hub on a projection of the area that the sensor nodes are located using the Google maps API. The visualizer will be able to show how the sensor nodes have shifted over time at certain time stamps. In addition to displaying the data visually, the web client will make the raw data available for inspection through the Google sheets API.

5.4 Slide Sentinel Enclosure

The enclosure will not be made in house using 3D printing but will consist of a water proof project box outfitted with cable coverings for the antenna and wires. The Rockblock+ is a waterproof version of the Rockblock and will stay outside the enclosure for better visibility of the sky. A mounting plate for the inside of the project box will be design be a member of OPEnS which will give a secure foundation in the project box to mount all the hardware too.

5.5 LOOM Integration

The code designed for use in the Slide Sentinel system for the Freewave radio and the Rockblock+ satellite communication module will be integrated into the LOOM library once the code for the hub communication has been finalized. Since LOOM 2 is set to be released soon we will be integrating it into this version of the library. This version is very different on a structural level but is assumed to have the same specifications as the previous LOOM version with OSC bundles for communication and specific interfaces to be filled out. Work will be coordinated closely with Kamron when the code is ready to be integrated.

5.6 POSTproxy

The small node app was an unexpected addition to the project which came about as a solution to an incredibly small but frustrating problem. The problem was that Rock7 will expect only a 200 http status response back from the URL given to the site. On the other hand, the URL provided by google for pushing data to a spreadsheet with the script will always redirect the request to a one time use link for security purposes. Unfortunately, this redirection results in a 302 response back to Rock7 which it interprets as not having sent the message so it'll start retrying on an interval. Additionally the message will actually get through to the sheet and be properly displayed but will continuously get duplicate data. We originally we're using Pushingbox to this end but it is being phased out due to security concerns and limitations. Our solution was a site to proxy the post requests by capturing it, responding with an OK, and sending the message to a specific link. This app was built on NodeJs and basic packages. Due to the need for these system to be used by more than just Weyerhause, a UI was added to add new URLs to send data to using a unique key. Additionally the site allows for changing the URL a specific key points to.

5.7 Integrated Website

As a continuation of the cloud infrastructure build out for the data and map client, both will be integrated together into a single site. Work has only begun as of mid April. The site is to be build on a MERN stack. (MongoDB, Express, React, Nodejs) This set of frameworks was chosen to be used due to the well defined modality of its components which can be tooled for reuse. A future goal of this site is to fork off a version for use with LOOM users particularly for use with the upcoming version 2 of it. Additionally this site is set to be continued later by other developers in OPEnS past the expiration of the capstone team so using a well defined framework is advised.

Since the project has gained interest of other parties than just Weyerhauser, this site will have to manage a number of Slide Sentinel users using a number of whole systems each. Documentation will be included on how add additional features to the project. The site will be integrated with a cloud service to allow for easy deployment of the site.

Integration of code from the geo-visualization application used in the OPEnS evaporametor will be used in front end for this site. The POSTproxy project will lay the groundwork for some of the API needed by the sites back end. The map client will be integrated into the Slide Sentinel front end for mapping nodes on the map. Some of the spreadsheet code can be used for the Slide Sentinel front end in converting values and date calculations.