System Design Document

V1.1

*Overview*

*The EduCase is a fully functional classroom inside a portable case. It contains multiple computers (Raspberry Pi & Android) loaded with content, tutorials, tools and information which educators can use in any area of the world.*

*The project is fully self-contained and self-powered and also includes provisions for communication with the outside world using either cellular data or a dedicated portable satellite dish and receiver linked to the "Outernet" satellite data provider. It also houses it's own LCD display and utilizes an LED projector enabling the entire classroom to see the content.*

*The EduCase can be deployed to refugee camps, rural areas, developing nations or anywhere else around the world and used to provide content, communications, courses and material we take for granted in the western world.*

*The EduCase can also be deployed into disaster/relief zones and used for sharing news, communication or things as simple as an impromptu movie theater.*

**

# INTRODUCTION

## Purpose and Scope

**This document is a work in progress and will be updated OFTEN. Thanks for your understanding!**

The EduCase will be able to deploy technical training, medical knowledge, communication, news, and content from all fields into any area of the world regardless of infrastructure.

## Project Executive Summary

The project is open to all user modification and improvements. At its core the Raspberry Pi computer(s) provide the content and computer power needed to share information via the on board screen and projector.

This can be extremely useful for aid workers, teachers, refugee camp employees and many more.

### System Overview

The Raspberry Pi computers are used to store the information, lessons and content. This can interface with the optional "Outernet" satellite downlink system or via cellualr data by tethering to the Android phone.

On board batteries supply power which can be replenished via mains power, solar panel, automotive power, generator or any other suitable source.

On board LCD display can be used for single users or deploy the included LED projector for displaying on any flat surface in order to share with a group.

Parts Video:

<https://youtu.be/J0L335KiNEU>

isual and Audiong Display either the Lighthouse wifi hotspot or to the cellular network via the Android phoneng the volume ava

### Design Constraints

Space in the case is limited so care had to be taken to include as many peripherals as possible without exceeding the volume available.

On board battery supply is limited to conserve weight but can easily be expanded if the situation requires. Lead acid batteries can also be substituted for LiPo units to reduce weight by capacity ratio.

**Rasbian Image License**: [Debian Free Software Guidelines](https://en.wikipedia.org/wiki/Debian_Free_Software_Guidelines) (DFSG)

### Future Contingencies

Expansion to control systems may require further modification.

## Document Organization

Basic explanations with expansion into subsystems as necessary.

## Points of Contact

Support provided via the forum at <http://mkme.org> as well as the GIT repository.

https://github.com/MKme/EduCase-Portable-Classroom

Support for the libraries used to support the code is provided by the original authors only. Restrictions for use can be found on their respective support pages (linked within the Arduino Source Code and Raspberry Pi Files)

Forum Thread:

<http://mkme.org/forum/viewtopic.php?f=32&t=702>

## Project References

Initial design completed spring 2016. Testing and updates ongoing via Hackaday website:

<https://hackaday.io/project/11010-educase-portable-classroom>

## Glossary

**Arduino**- Arduino is an open-source computer hardware and software company, project and user community that designs and manufactures microcontroller-based kits for building digital devices and interactive objects that can sense and control the physical world.

The project is based on a family of microcontroller board designs manufactured primarily by SmartProjects in Italy, and also by several other vendors, using various 8-bit Atmel AVR microcontrollers or 32-bit Atmel ARM processors. These systems provide sets of digital and analog I/O pins that can be interfaced to various expansion boards ("shields") and other circuits. The boards feature serial communications interfaces, including USB on some models, for loading programs from personal computers. For programming the microcontrollers, the Arduino platform provides an integrated development environment (IDE) based on the Processing project, which includes support for C,C++ and Java programming languages.

# SYSTEM ARCHITECTURE

Raspberry Pi Rasbian image provides basic operating system

## System Hardware Architecture

## Control System Architecture (Aurora Firmware)

For control and monitoring of the battery system there is an optional microcontroller control and supervision module. This consists of the Arduino Nano, temperature sensor, Real Time Clock and voltage monitoring.

The firmware (Arduino code) running on the controller is branded Aurora and found on GitHub

*Aurora Logo:*



## Internal Communications Architecture

Raspberry Pi wifi can be tethered to either the Lighthouse wifi hotspot or to the cellular network via the Android phone

# FILE AND DATABASE DESIGN

All files and updates via GitHub:

<https://github.com/MKme/EduCase-Portable-Classroom>

## System Files

* Raspberry Pi running the Raspbian operating system. Android phone and MK809 running Android OS with automatic updates.

## Non- System Files

* Remote updates and file downloads from the Outernet interface as well as standard data delivery over cellular data.

# HUMAN-MACHINE INTERFACE

1. LCD Monitor
2. LED Pico Projector
3. Android Phone Display
4. Battery Monitoring Display (Nokia 5110 LCD)

## Inputs

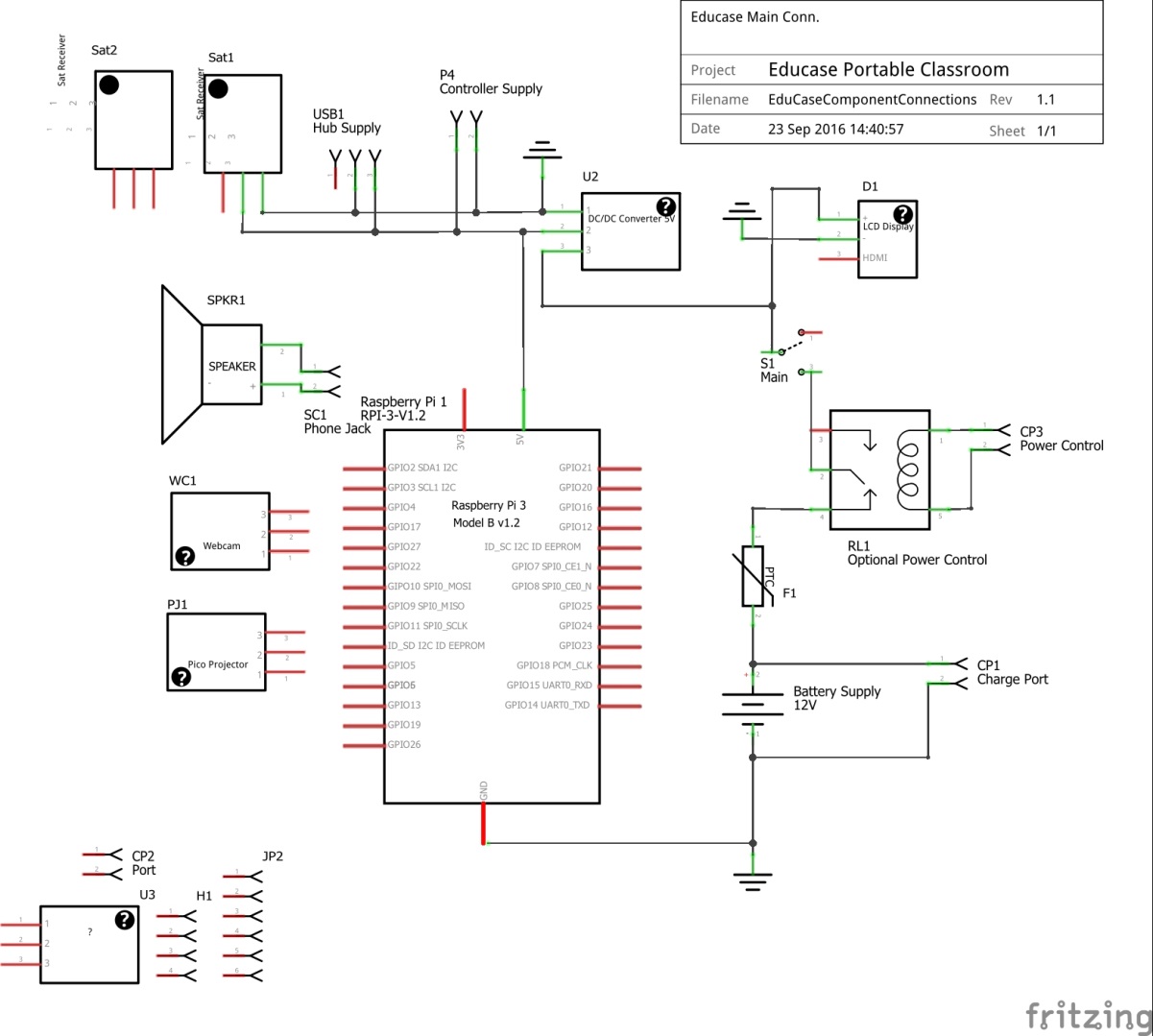
* Wireless Keyboard
* Bluethooth Keyboard
* Wireless Mouse
* Touch pads on Tablet PC & Cellular Phone

## Outputs

* HDMI video
* Stereo Audio
* Wireless LAN
* Bluetooth Data

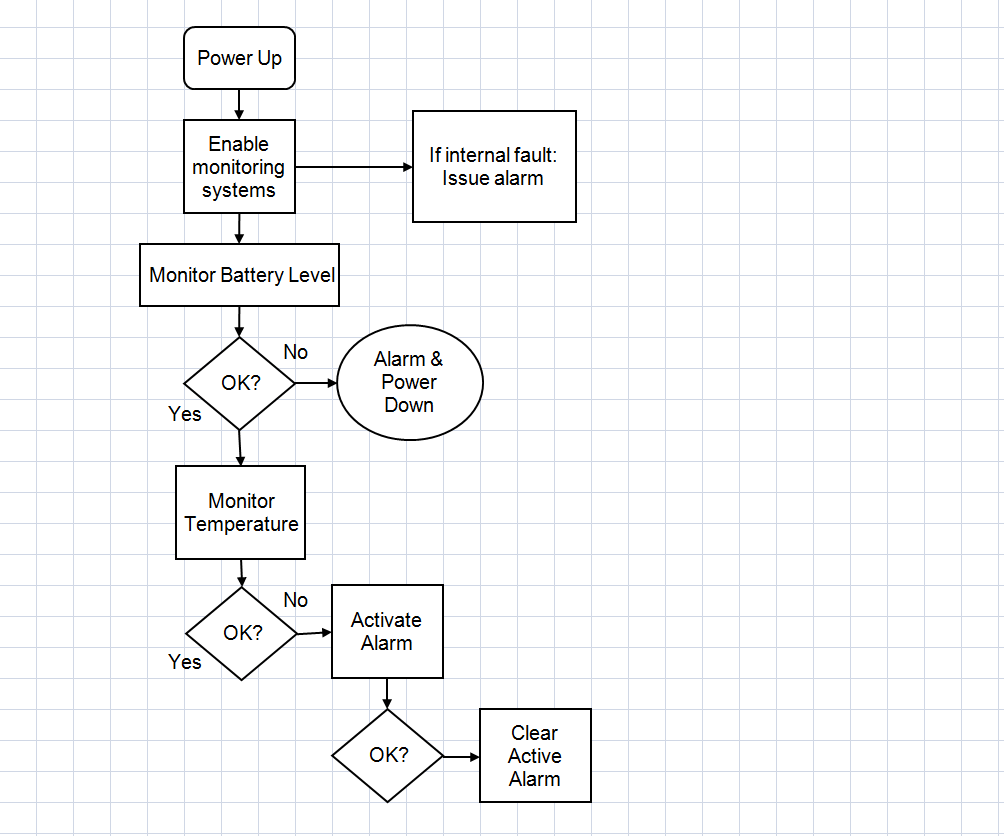
# DETAILED DESIGN

## High Level Schematic Diagram



## Software Detailed Design

Software design for all programs used is beyond the scope of this document. Software logic for the monitoring and control systems can be summarized in the blow functional flow.

*Functional description of the control logic:*

## Internal Communications Detailed Design

* Wireless networking via 2.4 ghz utilized for file transfer between systems
* I2C communication utilized for Real Time Clock
* SPI utilized for remaining components

# EXTERNAL INTERFACES

* Cellular Mobile data
* Outernet satellite receiver system

## Interface Architecture

NA

## Interface Detailed Design

NA

# SYSTEM INTEGRITY CONTROLS

NA