IoT Smart Calendar

Advisor: Dr. Malinowski

Jason Morris, Cole Lindeman

Presentation Contents

- Introduction
- Prior Work
- Functional Requirements
  - System Level Diagram
  - Subsystem Diagrams
- Efforts Completed
- Parts List
- Schedule for Completion
- Future Discussion
- References

Introduction

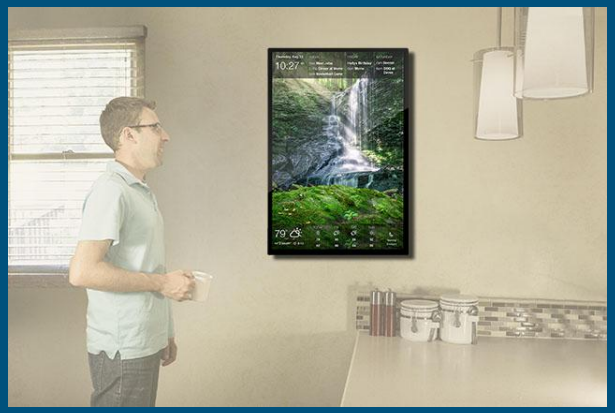
An Internet of Things Smart Calendar

- Wall mounted Smart Calendar placed outside of the Professor’s office
  - Displays Calendar for people who stop by
  - Displays advertisements for people walking by
  - Allow users to leave messages
  - Users can also access professor’s announcements
- Interface with sensors
  - Track and record motion
  - Sense if people are stopped at calendar or walking by
- Communicate with the Internet
  - Retrieve urgent announcements
  - Send messages, alerts and tracking data
  - Automatic updates

Prior Work

Raspberry Pi Framed Informational Display

- Powered by Raspberry Pi 2
- Displays Google Calendar and local weather
- Turns off display at night
- Buttons on side toggle Google Calendar between monthly, weekly, and daily views



DAKboard

- Customizable wall display
- Can showcase photos, calendar events, and weather
- Allows user to customize information to be displayed
- Everything is done through web interface

Functional Requirements

Modes of Operation:

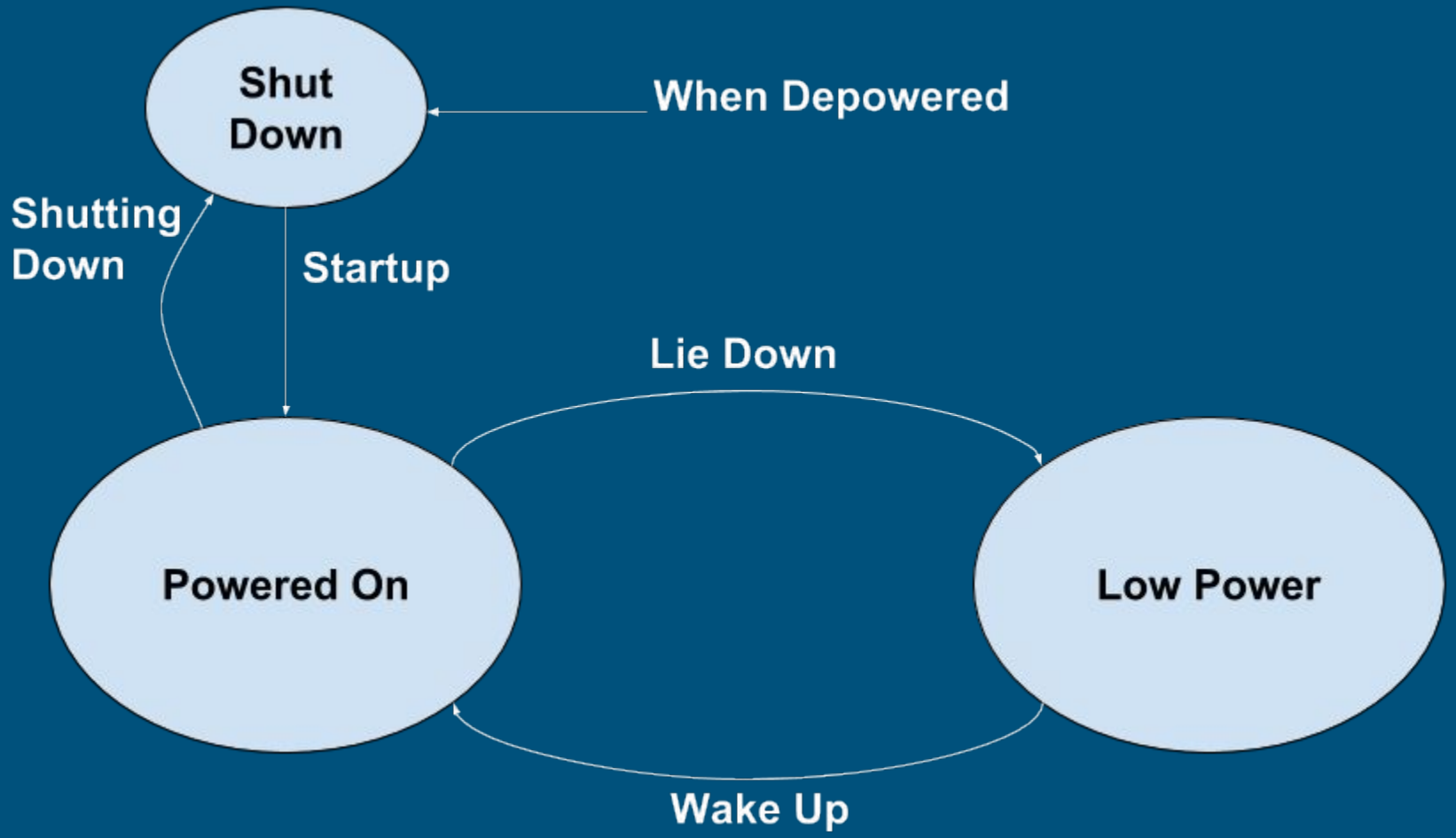
- Shut Down
  - Everything is completely powered off
- Startup (transition)
  - GUI is loaded in kiosk mode, display is briefly off and sensors are turned on
- Powered On
  - Display is on and capable of showing showing calendar, advertisements, messenger service, or announcements
  - Users can manually switch what the display shows
  - Switches to advertisements automatically when idle
  - When idle, switches to calendar if user stops at calendar
  - Perform periodic updates of the calendar, advertisements and announcements

Functional Requirements

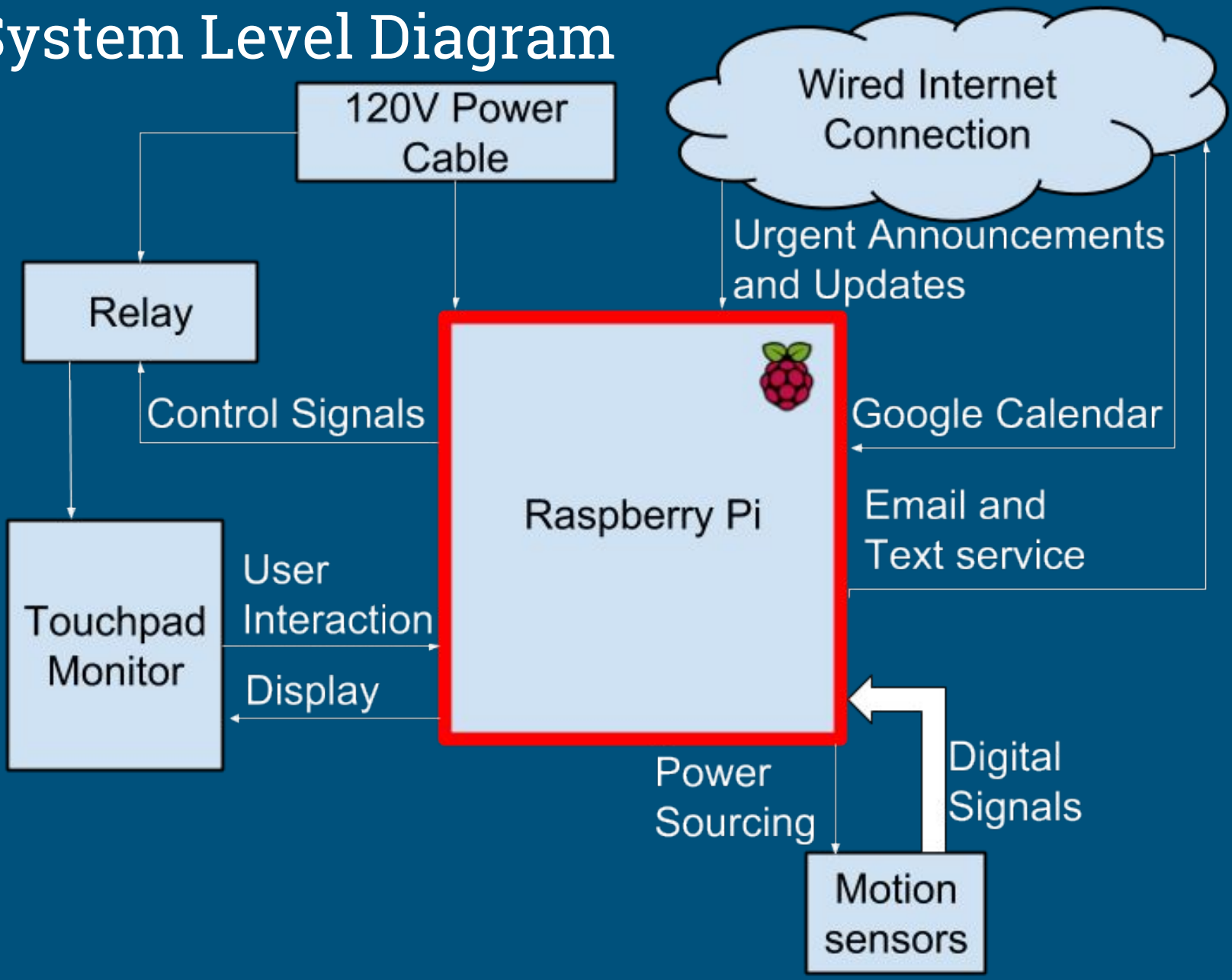
Modes of Operation Continued:

- Lie Down (transition)
  - Transition Smart Calendar to Low power mode to shut off display
- Low Power
  - Smart Calendar is saving power, display is off, sensors stay on
-
- Wakeup (transition)
  - Transition calendar from Low Power mode to Powered On mode to turn on the display
- Shutting Down (transition)
  - Send tracking data to internet
  - Shut down all inputs and outputs
  - This is for safely restarting the system

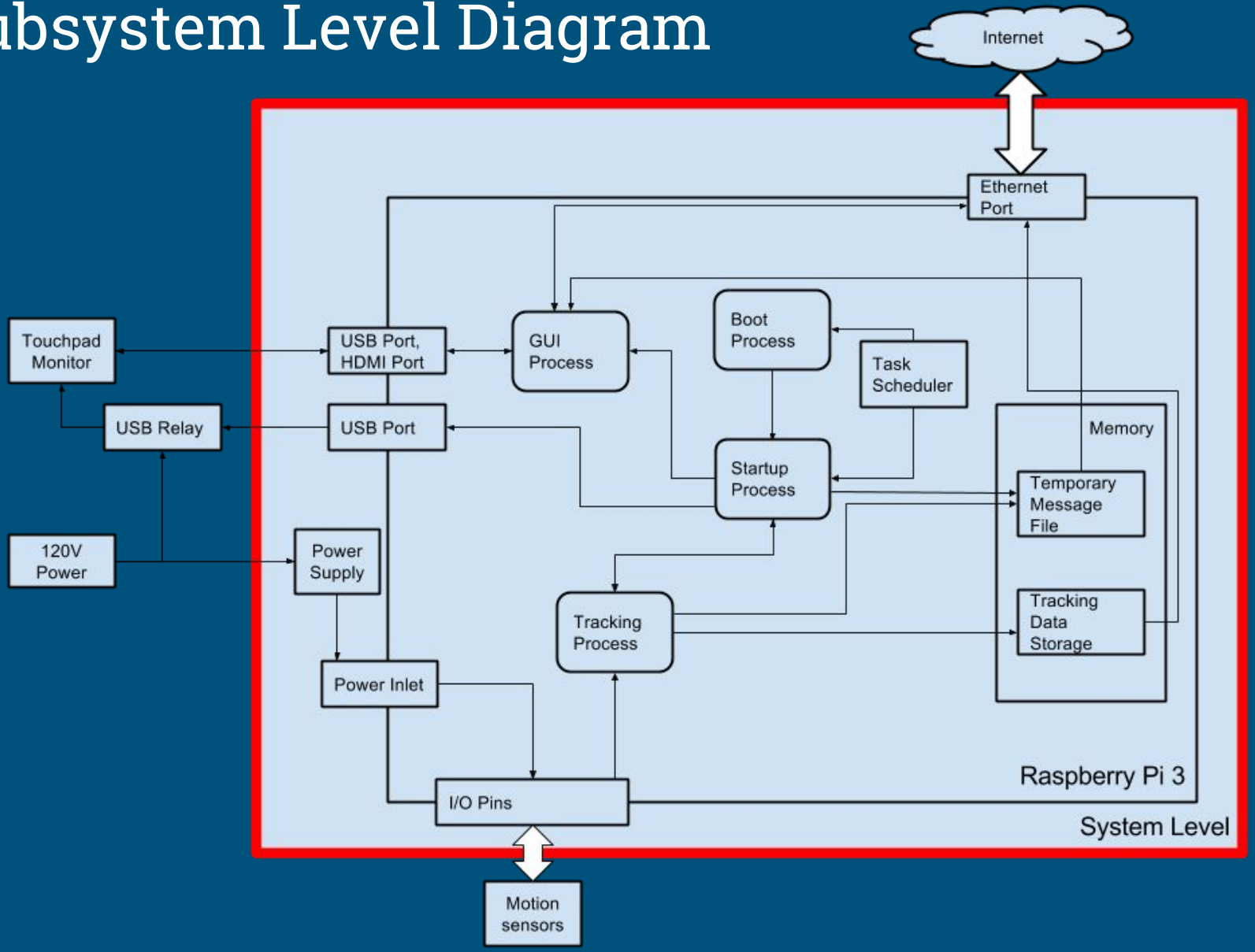
Functional Requirements



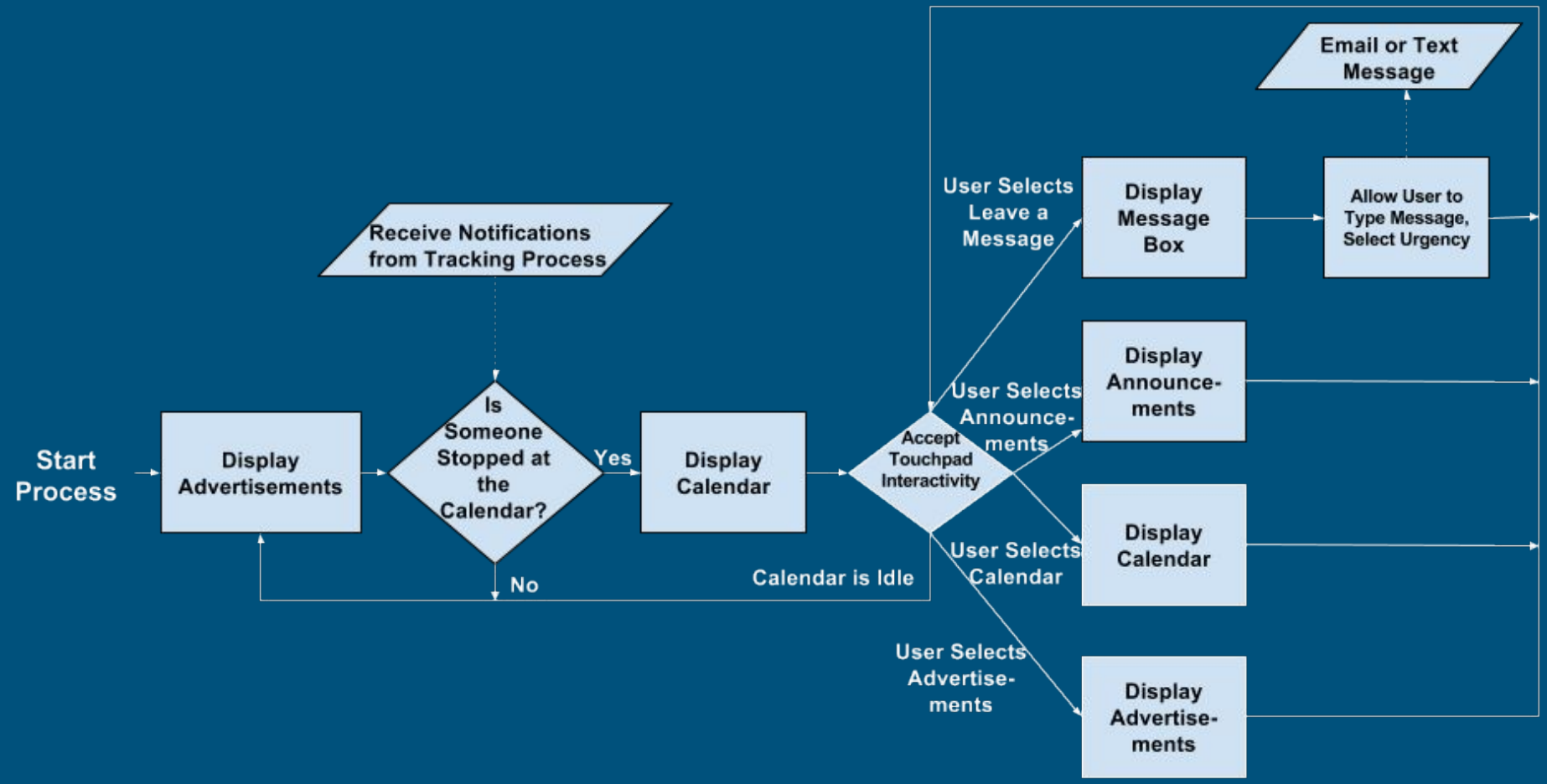
System Level Diagram



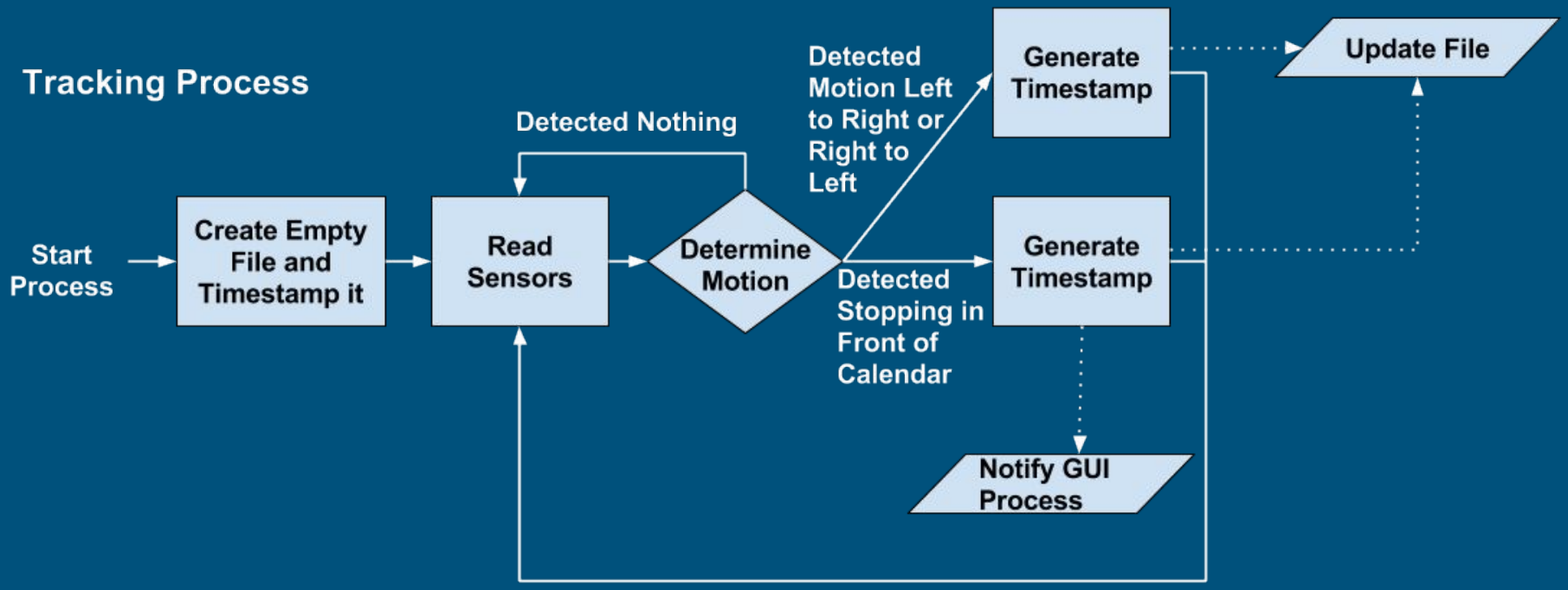
Subsystem Level Diagram



GUI Process Subsystem



Tracking Process Subsystem



Efforts Completed

GUI Subsystem

- Made with HTML, CSS, Javascript, and PHP
- Contains tabs for displaying the professor's Calendar, advertisements, announcements, and messaging screen
- Displayed in Mozilla Firefox on bootup
  - Firefox extension R-kiosk is used to load Firefox in kiosk mode
- Calendar page
  - Contains professor's Google Calendar embedded into the web page
  - External links removed using AdBlock Plus Firefox Extension
- Advertisements page
  - Alternates between displaying different advertisements every five seconds
  - Picture changing functionality done using JavaScript
  - Python script made that grabs any of the advertisement picture names in the images folder and throws it into the advertisement HTML page while replacing the old images

Efforts Completed

GUI Subsystem Continued

- Mali Messenger page
  - Allows user to type in and send message to either professor Malinowski's phone or email
  - Firefox extension FxKeyboard installed so user can type message on touch screen
  - Message is sent to a PHP script using HTTP post
  - PHP script then sends this message to professor's email or phone number

Updates

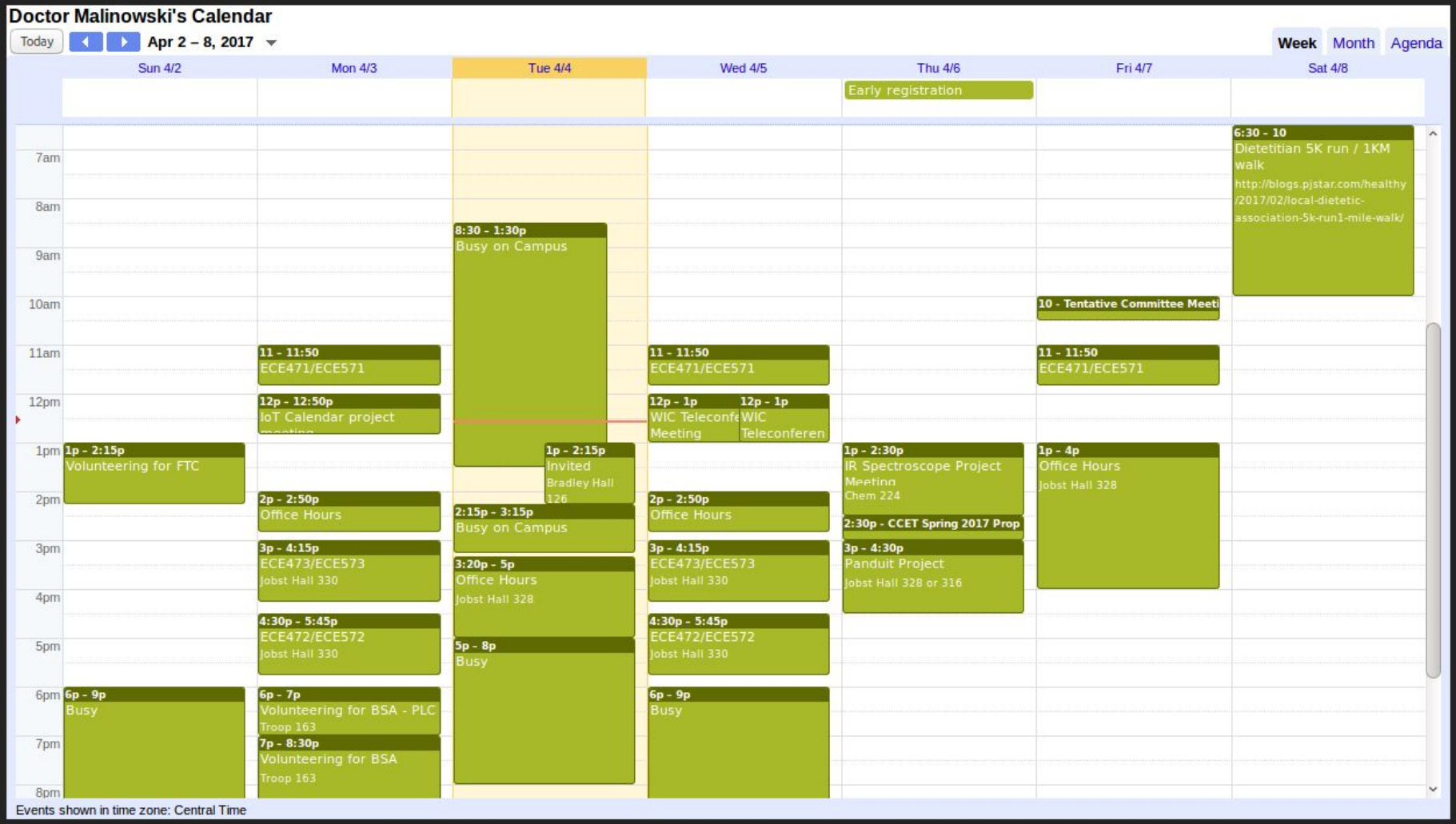
- Git is being used for updates
  - Command added to crontab on the Raspberry Pi that does an automatic git pull from the Calendar's master branch once every ten minutes

Efforts Completed

Background Services

- Scripts made with Python
  - Also uses Bash, Cron, and other Linux utilities
- Boot Process
  - Controls the Pi on boot
- Startup Process
  - Contains multiple instruction sets
    - Starts/ends other services
    - Controls the monitor
- Tracking Process
  - Reads sensors over GPIO
  - Communicates with other processes
  - Tracks and saves activity

Calendar | Advertisements | Mali Messenger | Announcements




DR. A. MALINOWSKI
ELECTRICAL & COMPUTER ENGINEERING
328

Parts

- Waveshare 10.1 inch 1280x800 IPS LCD Capacitive Touchscreen with case
  - $118.99
- Raspberry Pi 3 with power supply, case and heatsinks
  - $51.94
- Sandisk 32GB microSDHC card with normal SD card adapter
  - $10.59
- Emy passive infrared motion sensor detector modules
  - $5.49
- Ethernet, HDMI, USB and digital I/O cables*
  - $14.89
- SMAKN LCUS-1 type USB relay module*
  - $0.00

Total:$201.90

*Provided in house and is not included in price

Schedule

		Planned Schedule		Actual Schedule	
Week		**Jason's work**	**Cole's work**	**Jason's work**	**Cole's work**
1/15/17	1/21/17	**Spring Semester begins** Write Python code to host HTTP web server for AJAX to communicate with		Setup Raspberry Pi Setup local Apache2 web server	Researched temporary file storage systems and AJAX
1/22/17	1/28/17	Write XML code using AJAX to direct browser	Continue writing Python code to communicate with AJAX	Continued setup of local web server Troubleshooted message sending with PHP Script	Implemented temporary file storage system Implemented AJAX and Python web server
1/29/17	2/4/17	Write HTML code to direct browser back to ads when idle for long enough	Setup Raspberry Pi Setup monitor for Pi	Replaced Chrome browser with Firefox	
				Continued fixing email message sending	Researched commands for monitor control

Schedule

		Planned Schedule		Actual Schedule	
Week		**Jason's work**	**Cole's work**	**Jason's work**	**Cole's work**
2/5/17	2/11/17	Write JavaScript for ads that "follow" passersby	Figure out reading, writing, and permissions for I/O pins Connect sensors to Pi	Cleaned up messaging Got Firefox running in Kiosk mode on bootup Wrote script to grab ad files and put them into ads page	Added and configured Firefox extensions
2/12/17	2/18/17		Write Python script to poll I/O pins Write Python script to enable and disable I/O pins	Wrote JavaScript to redirect page to calendar if idle Added command that does frequent Git updates Rewrote image grabbing script Added announcements page	Wrote JavaScript to whitelist allowed pages

Schedule

<table>
<tr><td colspan="2"></td><td colspan="2">Planned Schedule</td><td colspan="2">Actual Schedule</td></tr>
<tr><td colspan="2">Week</td><td>Jason’s work</td><td>Cole’s work</td><td>Jason’s work</td><td>Cole’s work</td></tr>
<tr><td>2/19/17</td><td>2/25/17</td><td rowspan="2">Find method to upload text files
Write script to use method to upload tracking text file</td><td>Write Python script to track movement with IR sensors</td><td>Blocked external links on google calendar</td><td>Tested motion sensors and GPIO pins</td></tr>
<tr><td>2/26/17</td><td>3/4/17</td><td>Write Python script to compile movement information into a text file</td><td colspan="2">Talked to lab directors about mounting Smart Calendar</td></tr>
<tr><td rowspan="2">3/5/17</td><td rowspan="2">3/11/17</td><td rowspan="2">Write Python script to send commands to Ajax using movement information</td><td rowspan="2">Write Python script to communicate with door lock</td><td>Split and stripped power cable to measure devices</td><td>Started work on background script drafts</td></tr>
<tr><td colspan="2">Measured all power ratings and gave them to lab directors</td></tr>
</table>

Schedule

		Planned Schedule		Actual Schedule	
Week		**Jason's work**	**Cole's work**	**Jason's work**	**Cole's work**
3/12/17	3/18/17	**Spring Break**			
3/19/17	3/25/17	Test Internet communication	Write script for sleep/wakeup process	Started work on website	Refined background scripts
3/26/17	4/1/17	Test mount setup for project		Worked on web site and poster.	Further refined background scripts
4/2/17	4/8/17	Mount project		Finished project poster	
				Continued work on website	Added USB relay control

Schedule

		Planned Schedule		Actual Schedule	
Week		**Jason's work**	**Cole's work**	**Jason's work**	**Cole's work**
4/9/17	4/15/17	Spare time in case of changes		Finished project poster and went to Student Expo	
4/16/17	4/22/17	Spare time in case of changes		Worked on final report and presentation	

Future Directions

- Professor GPS tracking
  - Messages professor if student is outside door
- Replace Git updates for ads and announcements
  - Custom application with GUI
  - Google Drive
- Additional interactivity
  - Weather information
  - Weather alerts
  - Game
- Companion phone app
  - Contains main Smart Calendar features

References

Archambault, Michael. "DAKboard Is a Customizable Wall Display for Photos, Calendar Events, and Weather." PetaPixel. N.p., 19 Aug. 2015. Web. <http://petapixel.com/2015/08/19/dakboard-is-a-Customizable-wall-display-for-photos-calendar-events-and-weather/>.

Barrett, Daniel J. *Linux Pocket Guide*. Sebastopol, CA: O'Reilly, 2004. Print.

"jQuery.ajax()." *Ajax jQuery API Documentation*. JQuery Foundation, n.d. Web. <http://api.jquery.com/jquery.ajax/>.

Kmccb. "Raspberry Pi Framed Informational Display - Google Calendar, Weather, and More.." Imgur. N.p., 07 Apr. 2016. Web. <http://imgur.com/gallery/z94Vr>.

"Linux Documentation." *Linux Documentation*. N.p., n.d. Web. <https://linux.die.net/>.

“PHP 5 Tutorial” PHP 5 Tutorial. W3 Schools, n.d. Web. 2016. <http://www.w3schools.com/php/default.asp>

"Python 2.7.12 Documentation." *Python 2.7.12 Documentation*. Python Software Foundation, n.d. Web. 2016. <https://docs.python.org/2.7/>.

Questions?


Calendar
Sun Mon Tue Wed Thu Fri Sat
1 2 3 4
5 6 7 8 9 10 11
12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 29 30 31