Jared Miller

CSC 483

201820

Project Proposal - Digital Signage

**The Situation**

Currently at UM-Flint there exists signage at various locations providing event information, advertisement for student groups, pathfinding and weather info. The current solution requires a full Windows computer to run each sign, bringing with it all the requirements of maintaining a Windows system, and also leaves the signs open to human interference in the form of [unauthorized persons] attaching a keyboard/mouse to the computer and performing other actions. The computers also require network connectivity, necessitating access to an ethernet interface (expensive to run new ones) or wireless access (not native to most desktop hardware).

To provide an alternative solution, we propose the creation of a system fulfilling a similar role, but able to run on cheaper hardware with software created solely for the purpose of signage. The main candidate platform is the Model B Raspberry Pi 3, a low-cost SoC computer featuring a native 802.11 wireless interface, HDMI output (usable on most display platforms), and a sane programming environment allowing easy duplication simply by cloning an SD card.

**Requirements/Features**

* Low cost hardware
* Ability to function wirelessly
  + Low/nonexistent idle bandwidth requirements
* Ability to display images, video and dashboard style data (weather, announcements)
* Can display multiple items at once, with user defined layouts that scale to fit the display medium
* Ability to schedule content in advance
* A central command server/application
  + Ability to set up content layouts for various display devices
  + Push layouts, schedules and content to players
  + Granular authentication (X can alter these signs, Y can modify layouts, etc)
  + Not required to by "always-on"
* Easy to use for users with little training

**Customers**

* U of M Flint campus
* Any university, business, or other organization that might utilize signage on premises

**Benefits of this solution**

* Lower installation costs (cheaper hardware, no need to run cables)
* Lower energy requirements ([4 Watts](https://www.pidramble.com/wiki/benchmarks/power-consumption) beats anything a Windows computer can do)
* It's solid state! (No fans, hard drives, or PSUs to die from age/dust)
* Easier to physically secure/lock it inside a box away from tampering

**Development requirements**

* 1-2 boards for prototyping, display equipment can be found anywhere
* Creation of command system
  + Web interface?
  + Authentication system
  + Upload of content
  + Content storage/management
  + Scheduler
  + Layout manager
* Creation of player system
  + Select candidate base OS
    - Stripped down OS, to minimize overhead
    - Add only necessary packages to fulfill goals
  + Development of display system
    - Need to display images, text and videos at arbitrary screen locations and scales
  + Find a reliable source of external info for weather
  + Access to third party data feeds?
    - RSS/XML/JSON stream for announcements
    - ICS to display calendar/events
  + Interactivity if we're feeling good about things
    - Touch screen interfaces (e.g. Pavilion, UCEN)
    - Path-finding, details for specific items
  + TV power management: turn it off at night

**Notes**

[Cheapskate way of doing things](https://www.danpurdy.co.uk/web-development/raspberry-pi-kiosk-screen-tutorial/): Set up our files, output some HTML to the FS to tie it together, let chromium at it. Would have to do some kruft to serve off the local filesystem? Or just set up a static http server, dump in files, bind to ::1 and point chromium at it. Refresh every while in a once. How would videos perform? Some randos say it is alright with local content. Layout would be dirt easy, at least.

Or we could figure out how X works...