# ITWS-4500 Web Science Systems Development Project Proposal

## Group 2 Members

Andrew Aquino, Dan Bruce, Jason Kim, Blake Lingenau, Jeremy Simon

## Problem

Paper menus at restaurants are not easy to change. An unreasonable monetary and labor cost is incurred when formatting, and producing new menus following even small changes, such as adding or removing a single item. Menus also cannot be changed remotely.

Existing commercial digital menu systems are expensive, up to and exceeding $1,000 for systems powering two displays, not including the price of the monitors themselves. On the more cost-effective side, the likes of PowerPoint do not look good, and do not represent menu items in easily mutable data structures. Current solutions don't allow for integration with websites to have consistent menus with those in-house.

## Solution

For our semester project, we propose an easy-to-access and -update, web-based open-source software platform for allowing restaurant managers to display their products on the screen(s) of their choosing. The product also allows for updating menus from a mobile device as well as customers viewing the menu from their own mobile device.

The physical menus will be customer-supplied television screens each run with a dedicated Raspberry Pi driver. The Pis will be configured to automatically open a browser and navigate to the menu display on boot. Additionally, WebSockets may be used to force a menu update, rather than forcing the restaurant owner to shut down and restart. Furthermore, other features may be accessible on the Pis from the menu dashboard, including the ability to blank the screen between certain hours (useful for OLED displays), as well as shut down the televisions using RS-485, IR blaster, or USB control during off-hours.

Menus can be edited with dishes separated into sections, with comprehensive information, called "attributes" about each item including price, allergens, and gluten-free or low-calorie status, among others, which restaurant managers can choose whether or not to display. This menu item data can be edited at any time, each time regenerating the menu visible to the public. In addition, the layout of the menu can be arranged intuitively by dragging and dropping dishes and sections. We will provide an interface for managers to quickly and easily make changes to menu styles such as fonts and colors by defining variables within a CSS preprocessor.

## Technologies

HTML5, SCSS
WebSockets
JavaScript/JQuery

Node.js
Raspberry Pi
Raspbian
Television control (stretch goal)

## Task Assignments

Each group member will be familiar with and help with building the application-level logic. Specific assignments are as follows:

Backend - Blake, Jeremy
Frontend - Andrew, Jason
Hardware - Dan

## Timeline

| Date | Milestone |
| --- | --- |
| 2/6 | Project name chosen, program architecture established |
| 2/20 | Front-end prototyping complete, basic functionalities are implemented |
| 2/27 | Back-end prototyping complete to handle those basic functionalities |
| 2/27-3/6 | Ensure front-end and back-end integrate in a presentable format, practice for presentation |
| 3/6 | Midterm presentation: Conception, plan and prototype complete |
| 3/20 | Hardware implementation, will be able to display on a TV from a Raspberry Pi |
| 4/7 | Mobile-responsiveness for menu complete |
| 4/17 | All functionalities implemented by this point, begin working on presentation and final edits |
| 4/24 | Final presentation: Solution complete and working |

We will be meeting weekly between our deadlines to ensure that we are all staying on pace.