User Manageable Digital Signage

You have all seen the digital signage on campus (UMB TV), it provides information about classes, events and other campus services. It also shows the time, the weather forecast and information pertaining to operating hours of food locations on campus.

Of course, the campus is not the only place using similar signage systems; train stations, concert venues and more all use some sort of solution to advertise or inform.

Not every use case would display the same information, but it is also unlikely that every venue would have software specific to them. While a concert venue may not need to show you the weather forecast, they may similarly to UMB advertising classes, advertise upcoming events.

The goal for this project is to provide a signage platform that the user can easily customize to display the information relevant for their use case with a layout of their choosing. In that sense this project is being left intentionally open-ended so that the group who works on it can focus on the features/use cases they find most interesting.

Given the timeline for the project, not every possible venue this could be deployed in can/should be considered at this time, but the expectation would be that at a later date further developments in those areas could expand the displayable information.

As you read what is below, think about what you may want to focus on. Schools, entertainments venues, or something else. Not everything makes sense for every possible deployment and so I am going to be a bit vague outside of a few must haves that crossover.

Below are some general bullet points about what might be done

- **Create modules** that can be displayed within a customizable layout
    - A user should be able to choose to display the weather (or whatever it may be) just as easily as they could choose to display rotating event advertisements.

- A few **required modules**:
    - Rotating "ad" content
        - Pictures, videos, animations etc
    - Time
    - Weather, upcoming calendar events or some other data that is obtained from a third-party API connection for proof of concept

- **Ideally the user could…**
    - Group devices to display specific modules/content
        - Suppose we are talking about UMB. UMB owns a license for this software but could enable TVs in Campus Center to show a different layout/content/module than University Hall is showing.

    - Easily adjust the layout to prioritize information
        - It shouldn't be that the clock can only ever take up a tiny spot in the corner while the rotating ads fill the majority of the screen.
        - This is flexible given the time constraints, but the more the user can choose when/where/how the content shows the better.
            - Ideally a user wouldn't choose to just use a rotating PowerPoint in lieu of this software

- The user **must have a friendly way** to add to/create the rotating content modules.
    - Choose Rotating Content Module -> Select which content -> reuse those selections in other places without completely recreating them each time

        - Do consider this scenario if possible/depending on your area of focus:
            - You are an entertainment company that owns multiple venues in the city. The upcoming events you advertise would be relevant to customers currently at any venue. The ads are fetched from somewhere in the cloud, maybe its not just a locally stored collection of PNGs/MP4s rotating from a specific folder
            - For this project, maybe that is a JSON file the user creates and then selects in a non-technical, friendly way (feel free to host the content locally on the server for now/assume the content is available for download from a URL)

    - Picture down the road the user selects the “March Events” as their ad content.
    - That JSON can be deployed across multiple venues and fetch the content based on it.

Things to consider but not required

- Eventually users would need accounts and things to manage this, but do not get bogged down by doing a ton of work on that end for the purposes of this project.
    - You may assume that this is being deployed to a single venue and hosted on site where access to the admin/layout management is controlled. The user does not need to have access to manage the system from anywhere in the world today…but someday something like that may be nice so try to leave the option open.

- What devices are showing this content?
    - We don’t need to decide this, but it could be a mini-Windows PC, RasperryPi, or similar.
    - Do assume that the device is relatively cheap/cost effective (We won’t have a discrete GPU and 128GB on RAM)

- How do those devices get registered to the system?
    - Keeping in mind that I would like to be able to identify devices for the purposes of grouping, you can assume that a little bit of setup is needed by the user to get started.
    - To simplify for your work, like many devices do, have the receiver devices (that would be plugged into TVs) automatically assume an IP/Port that the server is at.

- The server is individual to the venue/self-hosted
- Receiver will report its name/ID to the server so that a user can group it with the content and should receive/display
- The receivers can be automatically assumed to be legitimate/intended recipients of the content and do not need a way of registering to an organization.
- i.e. the server would not be host for both UMB and the TD Garden content so no need to do anything but let the user decide your content group
- It is fine to test on your laptops as the receivers so long as you stick to libraries that would work across Windows/Linux.

What language/library/frameworks?

- Leaving this totally up to the group.
  - Keeping in mind cross platform compatibility and some mechanism of being auto launched on the receivers end

For your reference

- UMB uses this currently if you would like to get some idea
  - https://www.visix.com/colleges-universities/