Team 17 – NATS Interactive Video Walls

Amanda Goodridge, Brandon Baugh, Will Schottler

Last updated: October 10, 2019

Software Architecture Specification

# Overview

The proposed system will be video walls to replace the old 70s looking publications displays in Life Science and Physical Science. The system will be a large touch screen or a combination of multiple screens that work together to allow students the ability to interact with the information that is displayed on the screens at the moment. For instance, there could be a rotating display of faculty and staff. If a student touches one of the faculty or staff members, their bio and a blurb regarding their recent research would appear on screen. Then the board could rotate to the disciplines and a student could touch one of those. Once a student touches a disciple, the screen(s) would show more information about that discipline, career opportunities, types of degrees offered, etc. If the system ends up being a video wall with multiple screens instead of just one huge screen, the project would be how to get these screens functioning properly together and how to display the information amongst multiple boards.

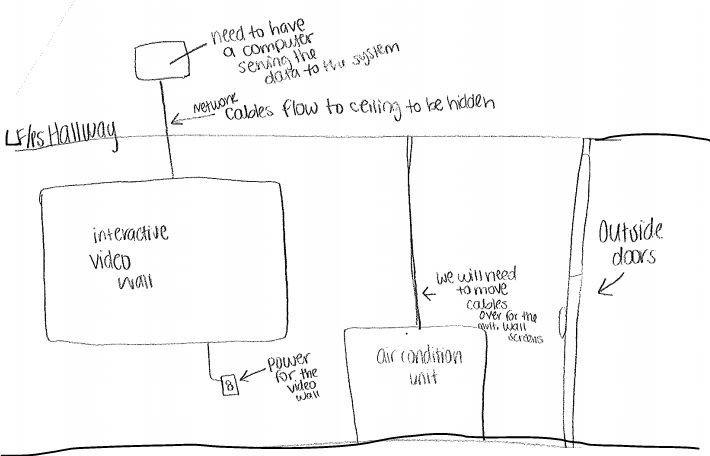
The ultimate goal of the proposed system is to modernize the NATS halls and bring them up to date with the 21st century. Our capstone team would need to creatively problem solve through cost estimating an entire package from hardware and possibly software. In addition, we would need to train the administration of this project and other faculty on how to use the system and update the software.

# Subsystem Decomposition

The interactive video walls will be made up of a tiled arrangement of monitors, panels, or projection screens. Tiling these multiple displays together is what will create our large, multi-HD display surface that students can interact with. When breaking this large system down, we have multiple systems that we will need to purchase. We will need to purchase the system software that provides an interface for controlling the displays, controller, and source content. We will need to purchase the hardware associated with the video walls such as the cables that will run Internet and power to the system. We will need a video wall controller that connects the content sources to the displays. This will let us control what is shown on the displays, when and where it appears, and how it looks. To conclude, we will need video wall display screens that provide a largescale, high-resolution “visual canvas” for our users.

# Hardware/Software Mapping

The hardware map illustrated below indicates the cables we will need to purchase and shows us that we will need to move the existing hardware (cables for the a/c). We will also need to purchase a server to host our data and software packages to create the interface of our system.



# Persistent Data Management

A visual application is an application that manipulates visual data as a part of its processing. Visual applications need to represent, manipulate, store, and retrieve both raw and processed visual data. Existing relational and object-oriented database systems fail to offer satisfactory visual data management support because they lack the kinds of representations, storage structures, indices, access methods, and query mechanisms need for visual data. With our system, we will have data management all done by one person so that it is visually symmetrical and does not confuse the users. This will ensure a better UI, which in return, will ensure a better UX for students, staff and faculty.

# Access Control and Security

Access control is a security technique that regulates who or what can view or use resources in a selected environment. In our case, any student, faculty, staff or visitor on the IU Southeast campus will be able to have access to our system. The physical access control limit is that it is located within the campus so unless you are on campus, you will not have access to the information displayed on the video board. This also relates to the logical access control limit of the computer networks, system files and data. The network will be secured and directed into the ceiling where the server will be located that holds our system flies. The system files will be located on a password protected server that will be stationed in a secure place so that unauthorized users cannot get ahold of it.

# Global Software Control

|  |
| --- |
| Global software control describes how the global software control is implemented. Each subsystem works together to accomplish the goal of getting the system to show the correct interface. This interface is controlled by the software and the software is designed from the database information that our project owner wants displayed. |
|  |

# Boundary Conditions

Boundary conditions describes the start-up, shutdown, and error behavior of the system. For our system, we hope to not have many errors, but we understand that errors are unpredictable. If errors do occur, we will have an error page display on the video board that states the system it is currently unavailable. For the start-up and shutdown of our system, we will have a power button that turns on and off the video wall and then another one that turns on and off the system. This will act as our hard reset button in case the system is glitching and needs a reboot.

# Key Personnel Information and contributions of each

Our team is composed of three members by the names of Amanda Goodridge, Will Schottler and Brandon Baugh.

Amanda Goodridge is the team lead. She provides guidance, instruction, direction, and leadership to the group for the sole purpose of achieving a complete project by the end of the academic year. She monitors the quantitative and qualitative achievements of the team and reports these results to the project managers (Dr. Finkbine and the Dean). She often works within the team, as a member, carrying out the same roles but with additional ‘leader’ responsibilities such as added documentation and reports that are to be completed weekly.

Will Schottler is the technical lead. He is responsible for leading the development of the project and the quality of the technical deliverables. His goal is to establish a technical vision and then turn that vision into a reality. He is also responsible for doing research and preparing prototypes of proof of concepts along with ensuring proper security for the technical aspects of our system.

Brandon Baugh is the system architect. His role is to analyze and recommend the right combination of IT components to achieve the project’s goal. He is also to help define and decide on the right IT strategy and approach that will best support long-term business plans and goals. He advises Will with the best tools, frameworks, hardware, software, and other IT elements to achieve the functional objectives.

# The Nine Basic Component Types Necessary:

* **Use Cases:** Are an ordered set of processes, initiated by a specific trigger (e.g., transaction, end of the day), which accomplish a meaningful unit of work from the perspective of the user.
* **Functions**: Are context independent processes that transform data and/or determine the state of entities.
* **Triggers:** Are the events that initiate Use Cases. There are three types of triggers: time triggers, state triggers and transaction triggers.
* **Data Stores and Data Flows:** Data stores are data at rest. Data flows are data in movement between two processes, a process and a data store, etc.
* **Data Elements:** Are the atomic units within data flows and data
* **Processors:** Are the components which execute the processes and events (i.e., computers and people)
* **Data Storage:** Is the repository in which the data stores reside (e.g., disks, tapes, filing cabinets)
* **Data Connections:** Are the pipelines through which the data flows flow (e.g., communications network, the mail)
* **Actors External Entities:** Are people or systems outside the scope of the system under investigation but with which it must interface