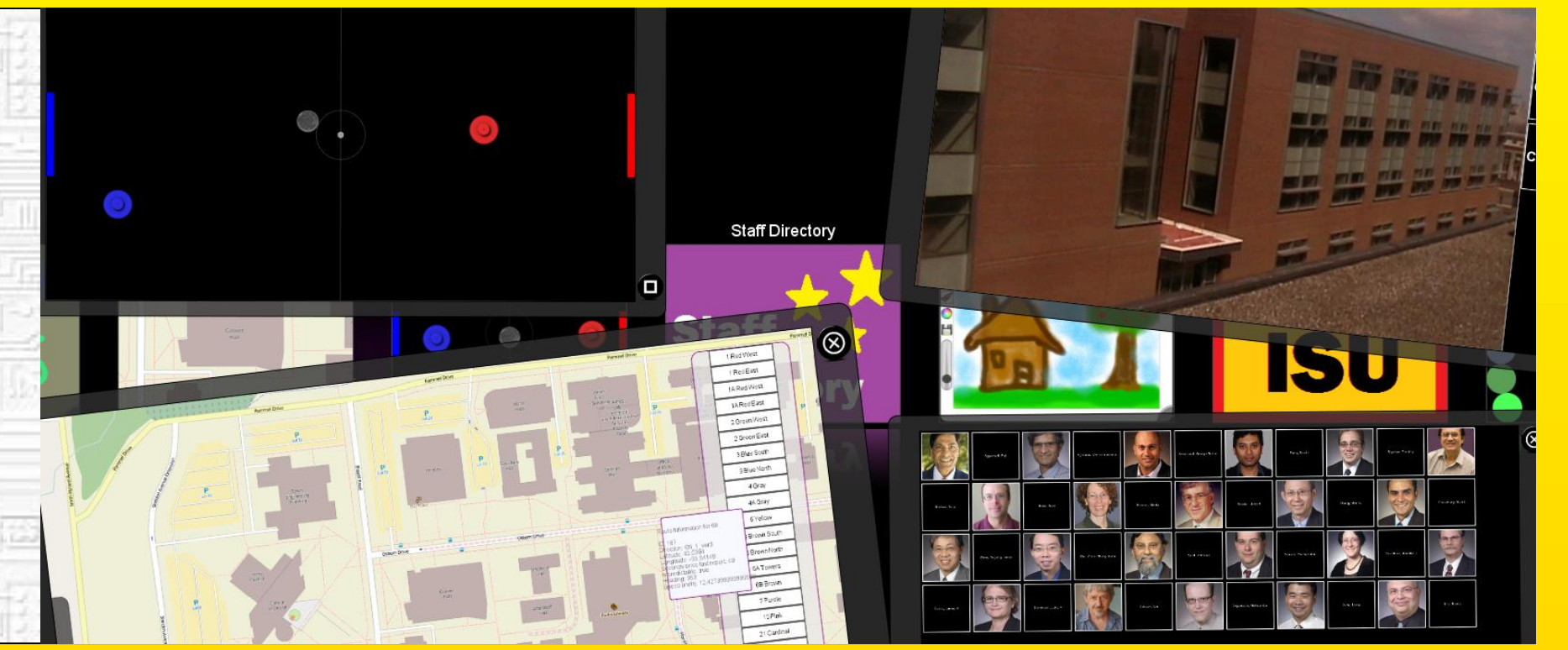


CyRIS May 14-04

Nathan Clague, Michael Krantz, Zachary Patzwald,
Maxwell Philips, Micah Stevenson, David Vriezen
Advisor: Dr. Manimaran Govindarasu Client: Brock Ascher



Introduction

The current content shown on the Union Pacific Multimedia Wall display does not demonstrate all the capabilities of the video wall. Our goal is to create multi-touch applications that entertain users and flaunt those capabilities. These applications will be integrated into the existing system. Additionally, we plan to allow future project owners to create their own content for CyRIS so that the system can be used as a learning platform.

To reach our goal of creating appealing content for the CyRIS system, we will:

- 1. Familiarize ourselves with the existing system**
 - Learn about the capabilities and limitations of the platform
 - Assess what applications would be feasible within the existing framework
- 2. Determine application requirements**
 - Survey students and staff regarding desired content
 - Evaluate the content present on other similar display installations across the country
- 3. Develop new content for the Union Pacific Multimedia Wall**
 - Implement an interface capable of handling multi-touch input
 - Create several applications to be launched via our interface
- 4. Test our content**
 - Test in-progress applications in a test environment
 - Test completed applications on the full system
- 5. Deploy content onto the Union Pacific Multimedia Wall system for daily use**

Design Requirements

Our project will operate in a Windows 7 environment with the 32-bit Java Virtual Machine.

Application	Functional Requirements	Non-Functional Requirements
Content Manager	<ol style="list-style-type: none"> 1. Able to generate multiple application windows and return control to the Intuiface presentation 2. Have multiple apps simultaneously accessing input and output 	<ol style="list-style-type: none"> 1. The content manager will match the look and feel of current Intuiface presentation
Campus / CyRide Map	<ol style="list-style-type: none"> 1. Map will be able to be panned, zoomed, and navigated 2. Map can be full-screen or scalable when needed 3. The map will show all currently-operating routes, buses, and bus stops 4. Data updates must remain within NextBus's data transfer limits 	<ol style="list-style-type: none"> 1. Map should be graphically pleasing and readable to the eye of the passers-by 2. The map should be resistant to service disruptions (network outage, NextBus API outage)
Feeds	<ol style="list-style-type: none"> 1. Feed will be populated by social updates approved by admin 2. Feed will automatically crawl while user using another app 3. Photos and links embedded in feed updates will be displayed 	<ol style="list-style-type: none"> 1. The automatic crawling feature will default to a useable speed 2. The app will be tested with at least ten different feed sources at one time to ensure reliability as well as client satisfaction
Staff Directory	<ol style="list-style-type: none"> 1. Directory will be populated and updated from the university website 2. Tapping an entry will display detailed information for that staff member 3. The staff directory will be searchable 	<ol style="list-style-type: none"> 1. Information shall include name, photo, courses, position, office location, corresponding major, phone number, and email
Webcam Application	<ol style="list-style-type: none"> 1. The application shall stream video from webcam feeds on campus 	<ol style="list-style-type: none"> 1. The application should be resistant to service disruptions

Design Implementation

Starting with a conceptual sketch nine months ago, our design has progressed over the course of two semesters to its current state today. See our block diagrams on the right for decomposition.

Acknowledgements

We would like to thank **Jason Boyd** for his help in deploying our project week after week. Our project is built with **MT4J – Multitouch 4 Java** (www.mt4j.org) and uses the **NextBus API** (www.nextbus.com).

Intended Use

- 1. The system will be used to display information relevant to the university**
- 2. The system will be used to display entertaining content, such as news channels, sports games, and interactive entertainment applications**

Intended Users

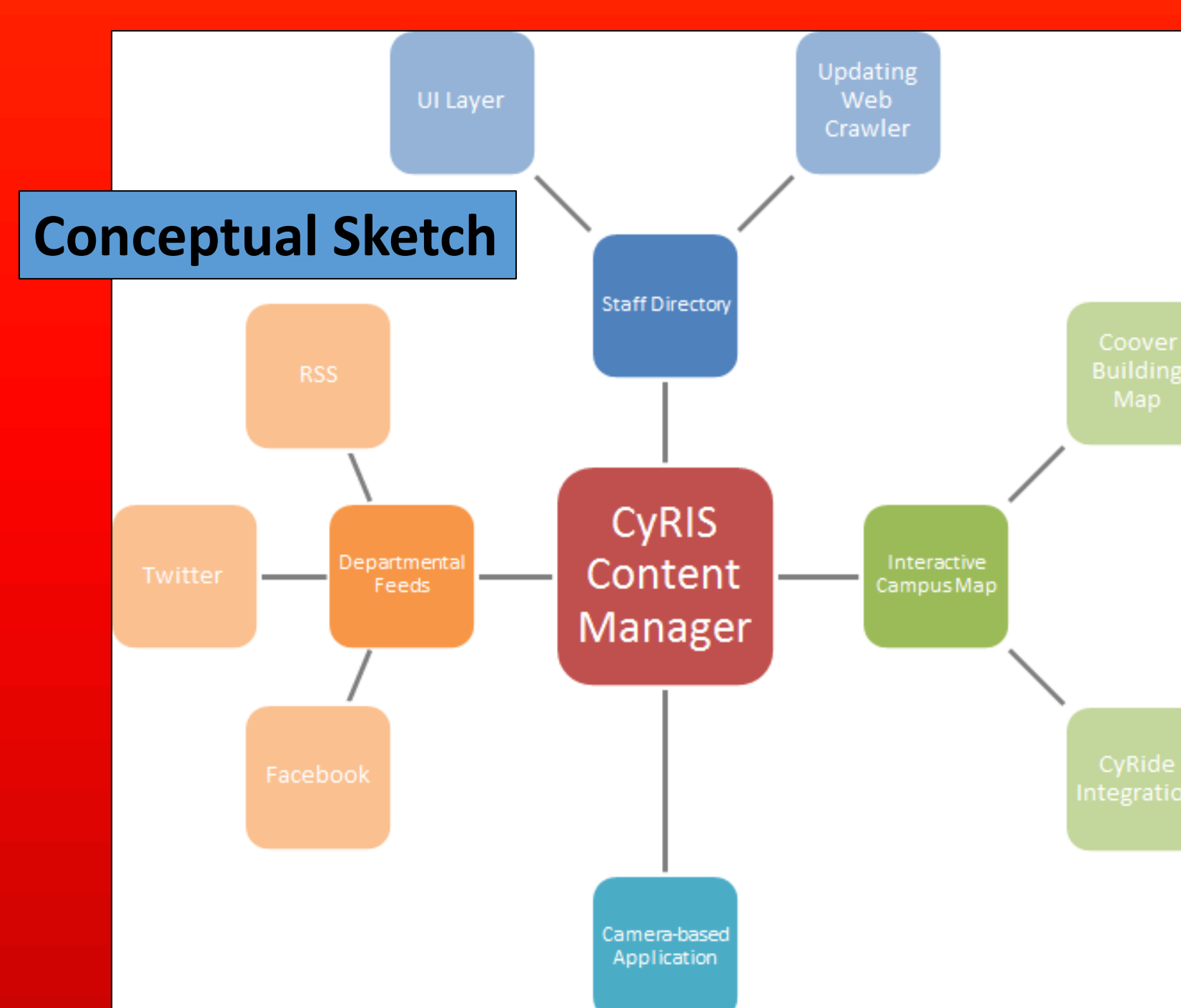
- 1. Coover Staff**
 - View brief, entertaining content going to/from classrooms and offices
 - Read news about campus events, weather, announcements, etc.
- 2. Prospective Students and Families**
 - View campus maps and information about the university
 - See the technological capabilities of the ECpE department
- 3. University Students**
 - Read news about campus events, weather, announcements, etc.
 - View brief, entertaining content going to/from classrooms
 - View campus maps and bus routes

Testing

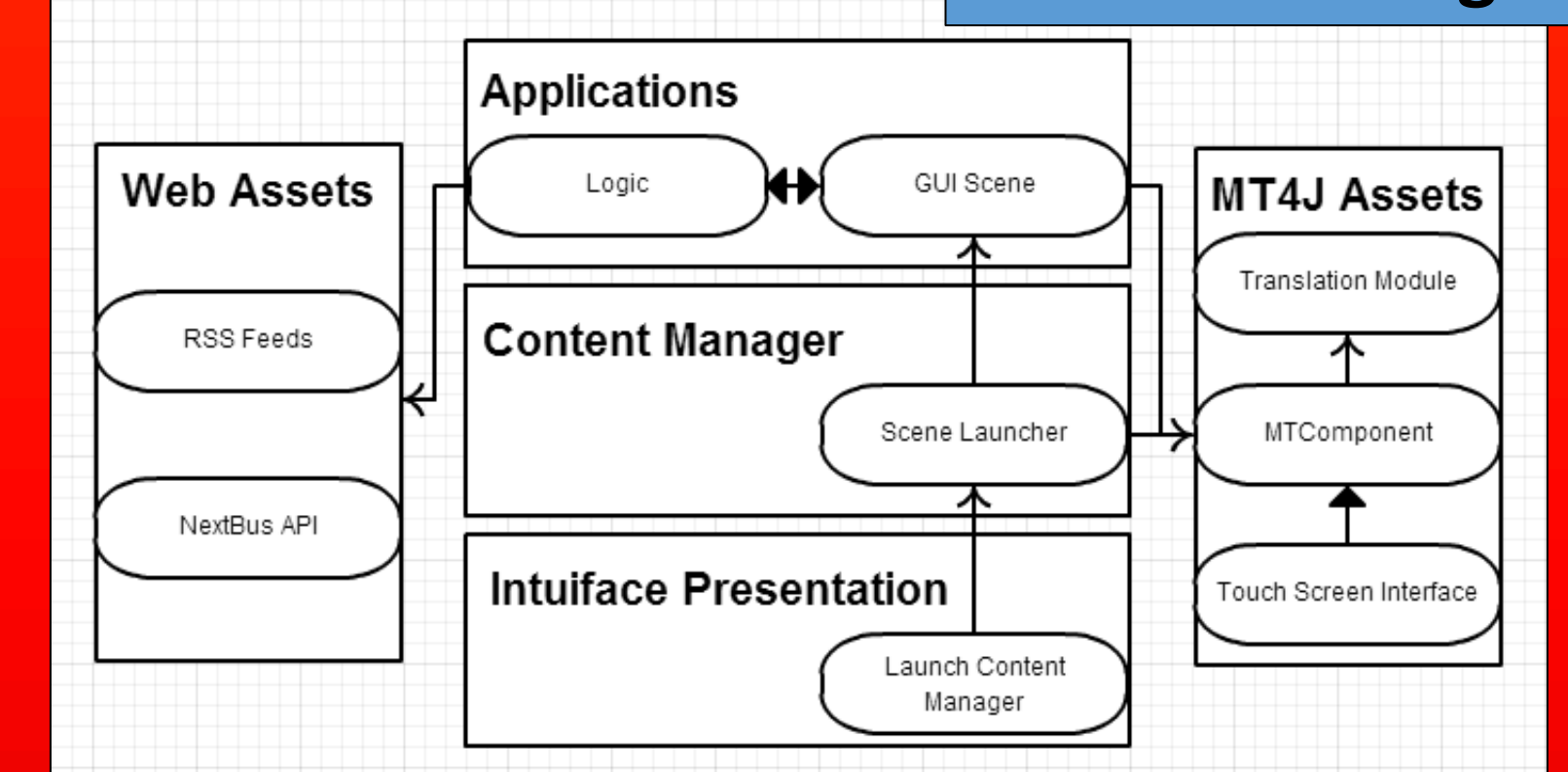
We have used a Windows 7 touch screen monitor provided by the ECpE department as a development and testing environment throughout our senior design project.

Specific testing strategies we have employed include:

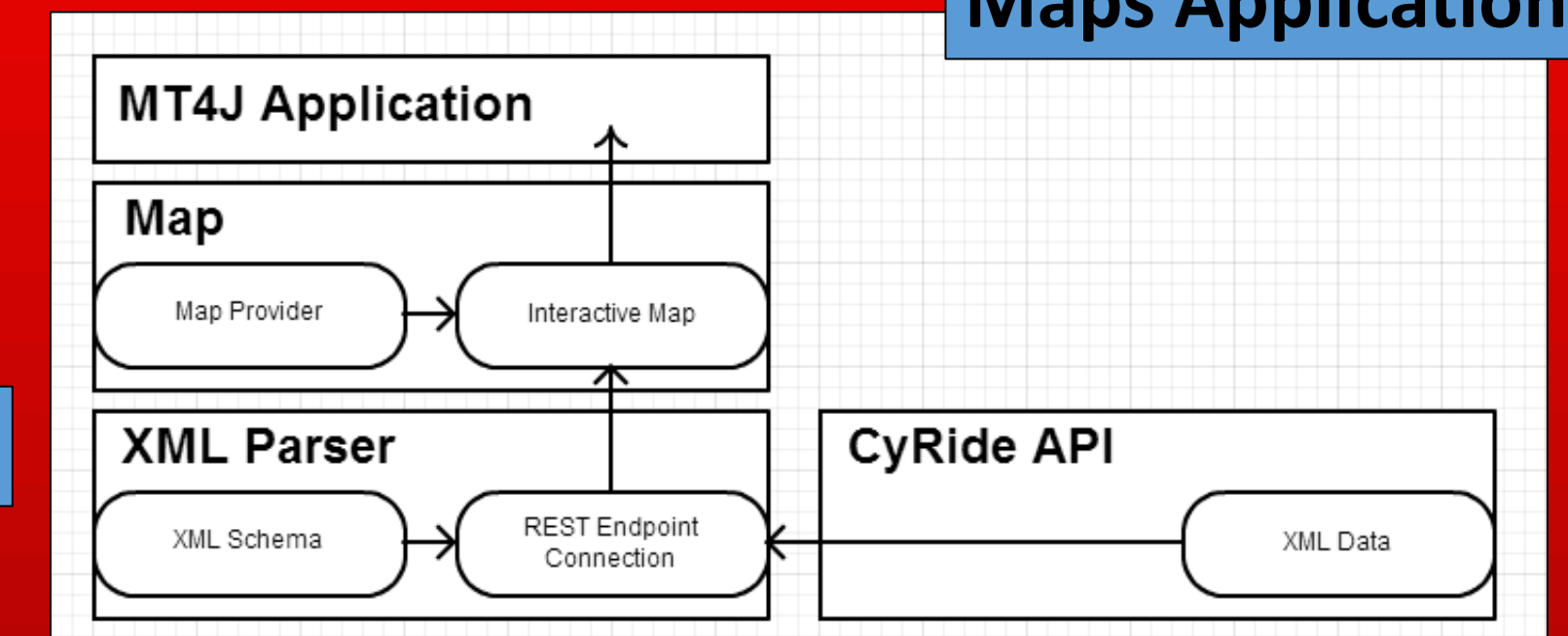
- 1. Functional Testing**
 - Demonstrating current progress to our advisor and client in our test environment and in the production environment
 - Adjusting development goals according to feedback gathered
- 2. Integration Testing**
 - Performed after completing development on new functionality
 - Testing the behavior of the system as a whole to ensure new development is compatible with existing features
- 3. Load Testing**
 - Used for applications that require an internet connection
 - Tests are performed to ensure correct behavior even in periods of extensive use
- 4. UI Testing**
 - Input sequences are delivered to our applications via third-party software
 - Afterwards, the actual state of the system is compared to an expected state



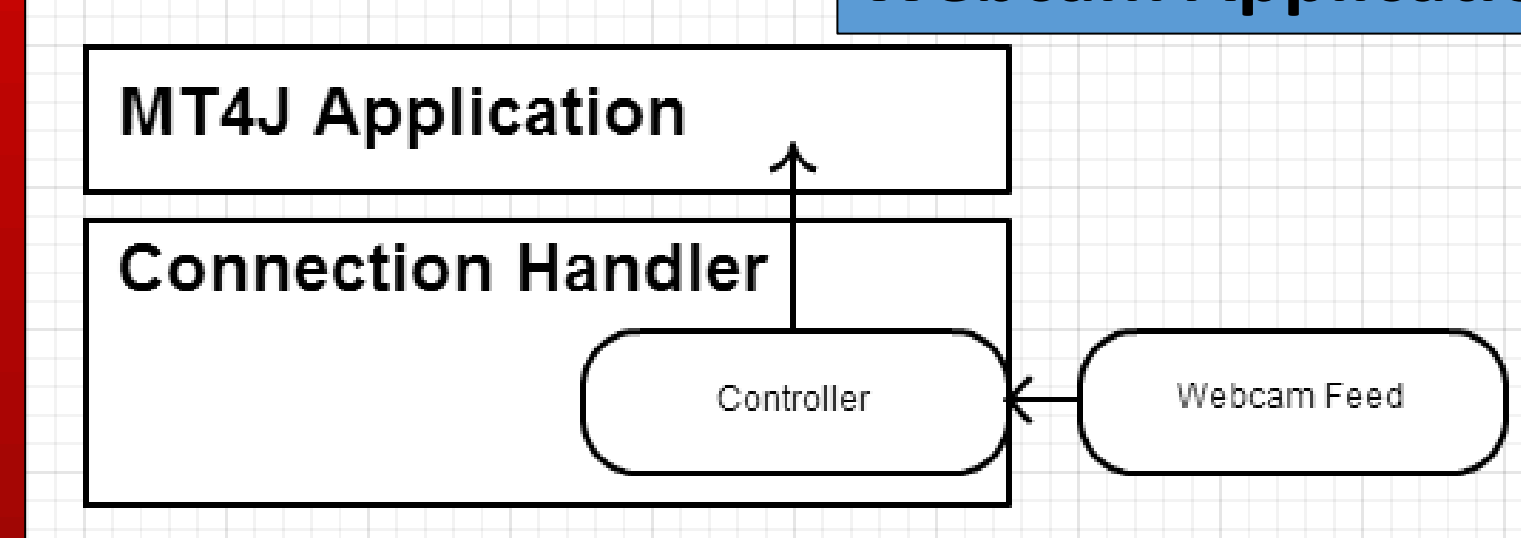
Content Manager



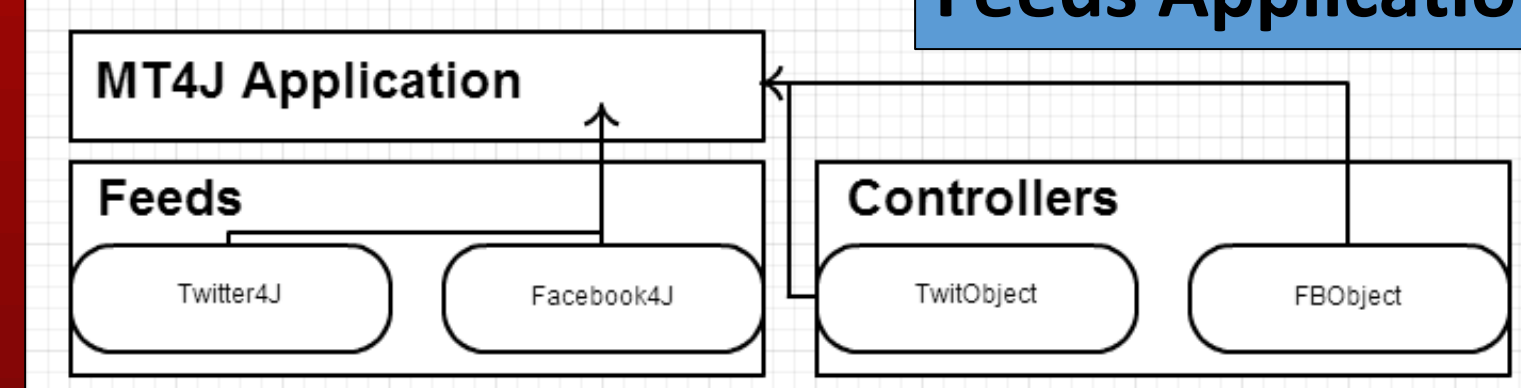
Maps Application



Webcam Application



Feeds Application



Staff Directory Application

