

Video Content Scheduling Application Design Document

Siqiao Chen

Regis University

Reversion Table		
Date	Author Name	Summary
10/28/2012	Siqiao Chen	Initial documentation, created TOC, project summary, use cases and reference page.

Table of Content

1. Summary	-----	4
2. Use Case	-----	5
a. Use Case Diagram	-----	5
b. Use Case Actors	-----	6
c. Use Case List	-----	7
3. Reference	-----	11

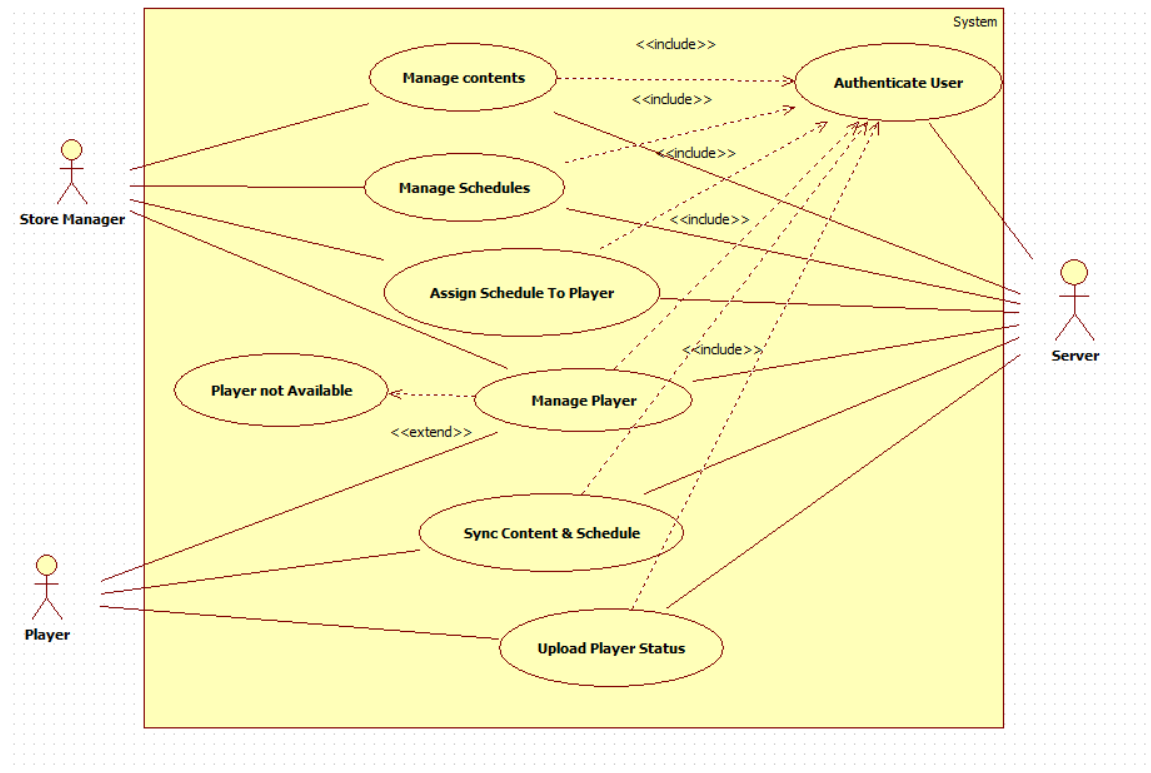
Project Summary

My project will be a web video content scheduling application for broadcasting/advertising service. This service should allow end user (probably store owner or restaurant manager) to upload video/animation/image contents, schedule them to multiple displays, and push the schedule video contents to the client controllers which actually broadcast the contents. Also, the user should be able to control some of the functions of display controller remotely, like turn on/off the display, add real time message, etc.

The player software on display controller will try to sync with the server all the times with a defined interval. It downloads new schedule, necessary contents and commands from server side. It plays content according to schedule and executes the commands from manager.

Use Case:

a) Use Case Diagram:



b) Use Case Actors

Store Manager:

The manager should upload and manage the contents for the local display. The manager should add and manage schedule using the uploaded contents. The manager should also be able to manage player at some degree. For example, the manager should be able to view current connection status, turn on/off the display and monitor current content.

Server:

The Server stores the user account for both store managers and in-store players. It also stores schedule and contents. Both manager and player need to pass authorization on server before doing anything.

In-Store Player:

In-Store player will try to sync schedule, contents, and diagnostic status with the server within defined interval.

c) Use Case List

Use Case: Manage Player

Primary Actor: Store Manager, Server, Player

Stakeholders and interests:

- Manager: wants to be able to monitor/manage in-store player.
- Server: wants to provide secure, reliable connections.
- Player: wants to update its status to server and execute commands from store Manager without delay.

Preconditions: The networking is ok between manager's computer and server, and the server is currently connected with player.

Success Guaranteed: The Manager successfully controlled player.

Main Success Scenario:

1. Manager opened Browser and get to server's Welcome page.
2. The manager enters this username and password into corresponding text boxes.
3. The server page verifies user and redirects manager to user menu.
4. Manager navigates to Manage Player page.
5. Player page displays player's latest status and options.
6. Manager assigns command (Update/Retrieve current play file/Shut-down Screen) to server
7. Server issues the command for player

Extension

- 3a. Invalid user name or password

1. Server redirect user to a page which notifies the customer of invalid user name or password

2. Server redirect user back to welcome page

5a. Player connection is lost

1. Server displays the “Connection is lost” message on player section.

Special Requirements:

- Player must always try to connect with server.

Technology and Data Variation list:

- How does the player stay connected with the server?

Frequency of Occurrence: approximately 1/day.

Open issues: N/A

Use Case: Sync Contents and Schedule

Main Success Scenario

- The player log into server. The player checks for new schedule. Server provides the most current schedule. The player downloads and parses the schedule and find out the needed contents. The player query missing contents from server. Server provides content for download. The player downloads contents. The player uploads “Sync Finished” status.

Alternate Scenarios

- The player cannot log into server
Player log info into its local log. Wait for 1 minute, and then try to login again.

- No new Schedule

The player uploads “Sync Finished” status. Wait for 3 minute, and then try to login again.

Use Case: Manage Contents

Browser displays welcome page. Manager enters username and password. Server verifies password and redirect manager to user menu. Manager selects manage contents page. Server redirects manager to content management page. Manager manages (create, read, update and delete) contents. The server responses result page. Manager logs off the website. Browser displays welcome page.

Use Case: Manage Schedule

Browser displays welcome page. Manager enters username and password. Server verifies password and redirect manager to user menu. Manager selects manage schedule page. Server redirects manager to schedule management page. Manager manages (create, read, update and delete) schedule. The server responses result page. Manager logs off the website. Browser displays welcome page.

Use Case: Assign Schedule to Player

Browser displays welcome page. Manager enters username and password. Server verifies password and redirect manager to user menu. Manager selects Assign Schedule to Player page. Server redirects manager to player scheduling page.

Manager assigns/removes created schedules to player. The server responses result page. Manager logs off the website. Browser displays welcome page.

Use Case: Update Player Status

The player logs into server. The player generates diagnostic information. The player uploads the diagnostic information to server. Server responses information received. The player uploads “Status Update Finished” status.

Use Case: Authenticate User

The player/manager input username and password to server. Server verifies password and username with the local database. Server gains player/manager to access restricted contents.

Reference