

Live Video Streaming

Written By: **Cole Herzog, David Dec, Richard Bird**

Last Updated: **3-Oct-2014**

Version: **1.0**

Introduction

This document will introduce the concepts behind a live video streaming website and server. This website will give video streamers (**content providers**) a chance to broadcast their daily activities. Content providers must be authenticated prior to streaming content. Watchers (**consumers**) will also be able to connect and view the live streams of the content providers. Consumers can watch while authenticated or anonymously. A chat will be included (one chat per stream) and can only be accessed by authenticated consumers and content providers.

Server Capabilities

The Live Video Streaming servers will allow for:

- Creating an account
- Authenticating previously created accounts
 - requires a unique username
 - requires a unique password
 - 6-18 alphanumeric characters
- Connecting to streams of content providers
 - Multiple streams may be watched simultaneously
 - Anonymous consumers may still watch
 - Chat will be available to authenticated consumers and content providers
 - Volume control and video quality options will be included
- Content Providers may choose to record and save streams

Front-End Capabilities

The Live Video Streaming website will have:

- Main Page with overview of popular content providers
 - Will only display content providers currently streaming
 - If no providers are streaming it will display popular recorded videos
- Individual pages for each stream (**channels**)
 - consumer version
 - A live and stream specific chat room

- Chat room customization is planned
 - A Recently Recorded Videos section
 - A video box containing the content being streamed
- content provider version
 - A record option
 - A Stream Delay option
 - A preview of the content currently being streamed on their channel
- Account information page for authenticated users

Languages and tools

The currently planned languages and tools are:

- Front-End
 - Dart
 - HTML5
 - CSS3
- Back End
 - NodeJS
 - C