A "Low Threshold Application" Streaming Video Server

Wesley Alan Wright, Center for Teaching and Learning

The University of Vermont

LTA: Low Threshold Application

"A Low Threshold Application (LTA) is a teaching/learning application of information technology that is reliable, accessible, easy to learn, non-intimidating and (incrementally) inexpensive."
Gilbert, Steven. "The Beauty of Low Threshold Applications – Campus Technology," February 12, 2002. https://campustechnology.com/Articles/2002/02/The-Beauty-of-Low-Threshold-Applications.aspx?Page=1.

We present a low cost, low threshold Streaming Video Server, built from a combination of open-source and commercial tools

Construction on campus

- Several IP network cameras installed to provide live monitoring of construction sites.

- *We needed a streaming server to rebroadcast multiple live rtsp:// streams.*

State of on-line video at UVM

- Primitive. We had a RealMedia Server and a Darwin Streaming Server both dating back to 1999. We were spending $3500/year support for RealServer. Both were obscure and difficult to use, and have been largely displaced by simple progressive download from campus web servers.
- *We wanted a YouTube-like streaming server to deliver on-demand content.*

Live Streaming: looked at 3 candidates

- Adobe Media Server, too hot!
- Red5, too cold!
- Wowza Streaming Engine, just right. Academic pricing for Wowza Streaming Engine perpetual license plus 3 years support < $1700.

CumulusClips

Wowza Streaming Engine streams both live and on-demand video. How could we leverage this by feeding it content? As if by magic, we stumbled across *CumulusClips.*

- http://cumulusclips.org/
- Open Source.
- YouTube clone where users can upload videos, rate videos, comment on videos.
- Uses php, MySQL.
- Reliable, accessible, easy to learn, non-intimidating.

Customized for UVM

- Implemented UVM SSO authentication (WebAuth).
- Wowza content folders shared via NFS with Cumulus server.
- Substituted JWplayer Premium as video player: $199 annual subscription; cloud hosted.
- Added UVM Watermark.
- Added MP3 audio and SRT captions support.


2015-11-17 10:45:00 AM STEMConstruction

Architecture
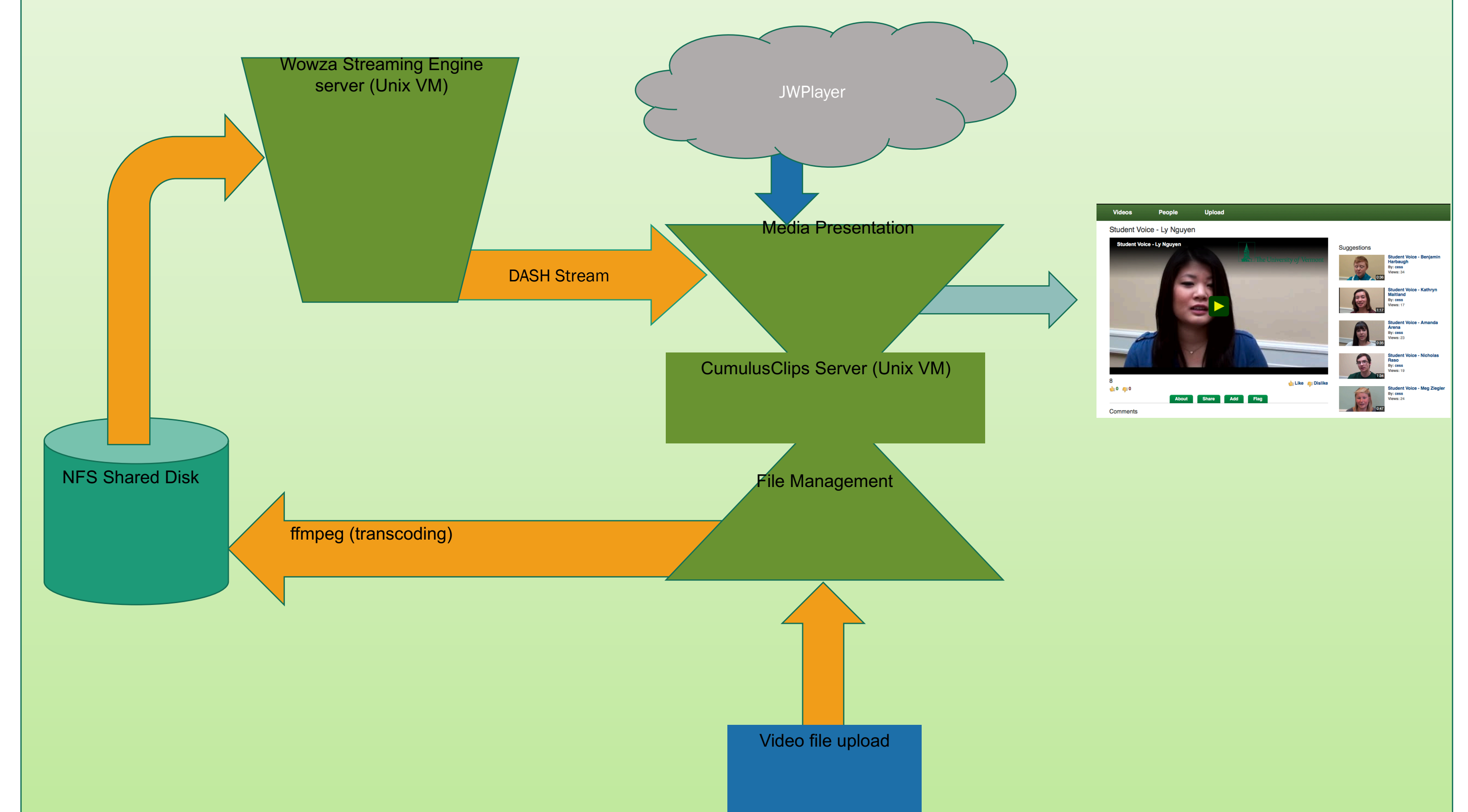
Features

- Accepts mp3, mp4, flv, wmv, rm (RealMedia), mov .
- Upload videos, tag and categorize.
- Uses ffmpeg to automatically transcode uploaded media to conformant MP4 files of multiple bitrates for consistent viewing experience.
- Optionally require UVM login to view.
- Add captions by uploading srt files.
- True dynamic adaptive streaming via Wowza Streaming Engine (http://www.wowza.com)
  - HTML5
  - Dynamic Adaptive Streaming over HTTP (DASH).
 - Apple HTTP Live Streaming (HLS)
- Upload or watch with mobile device.

Growth suggests user acceptance
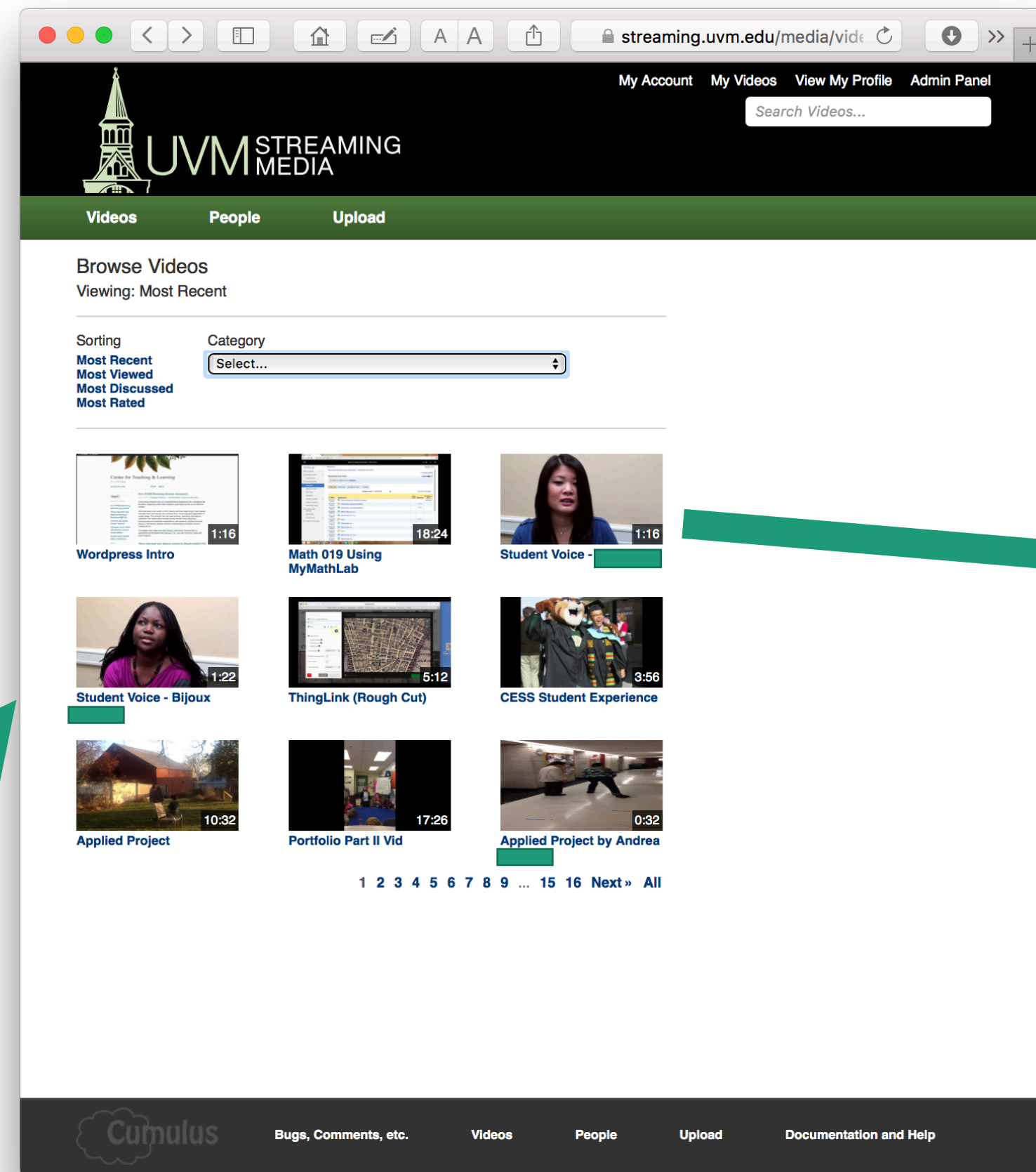

Media Files

Contributors


Authenticated Users
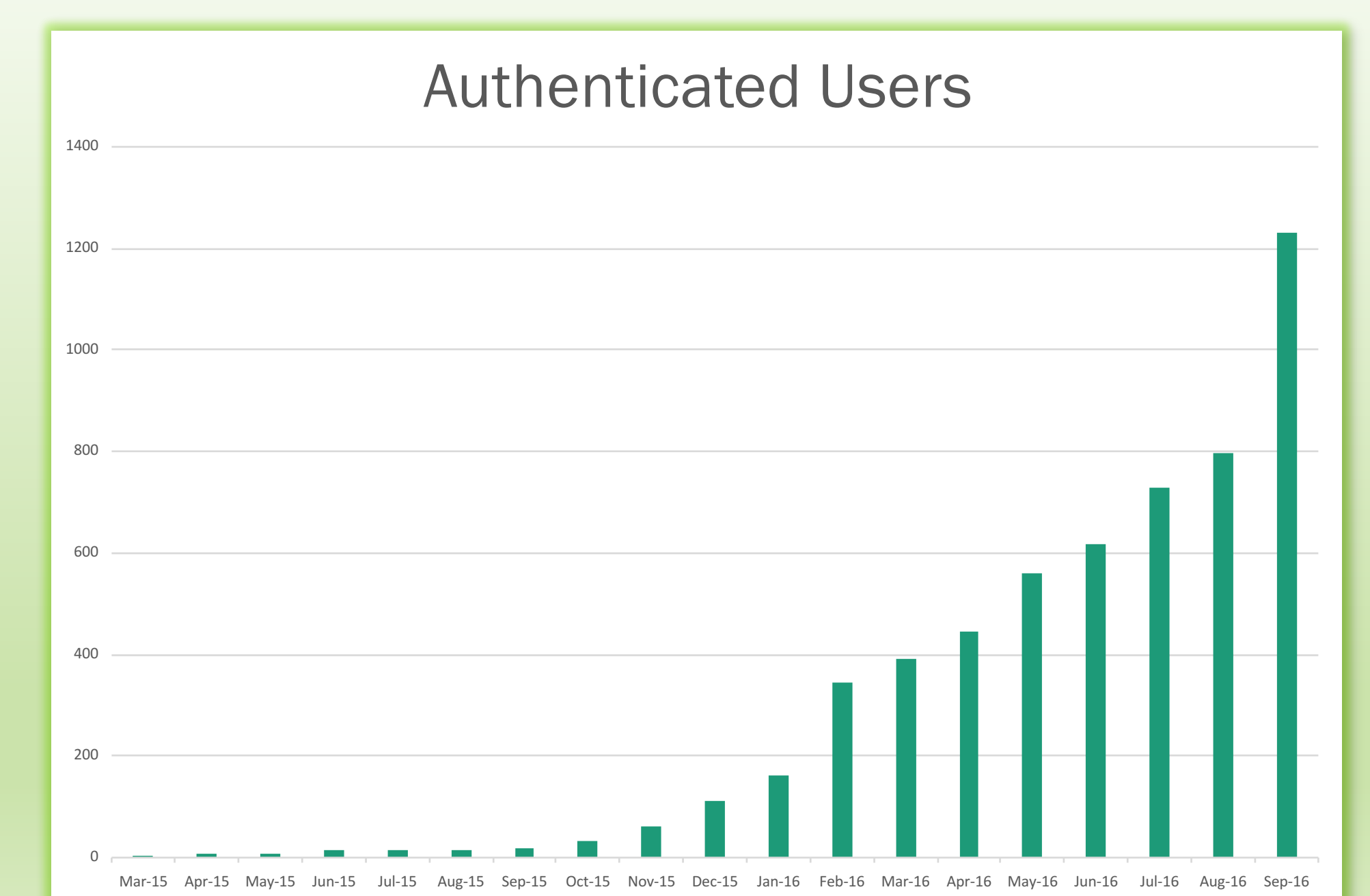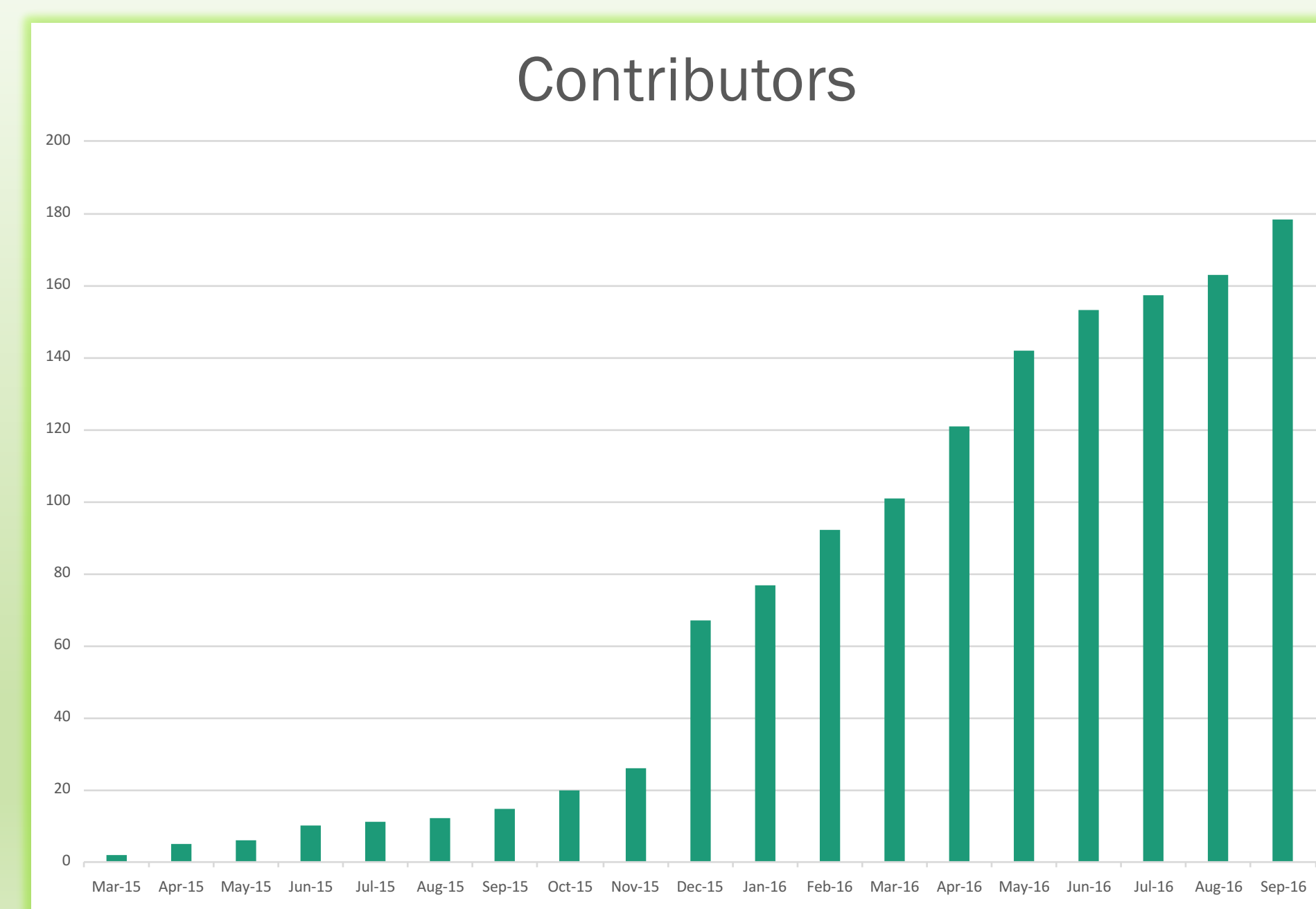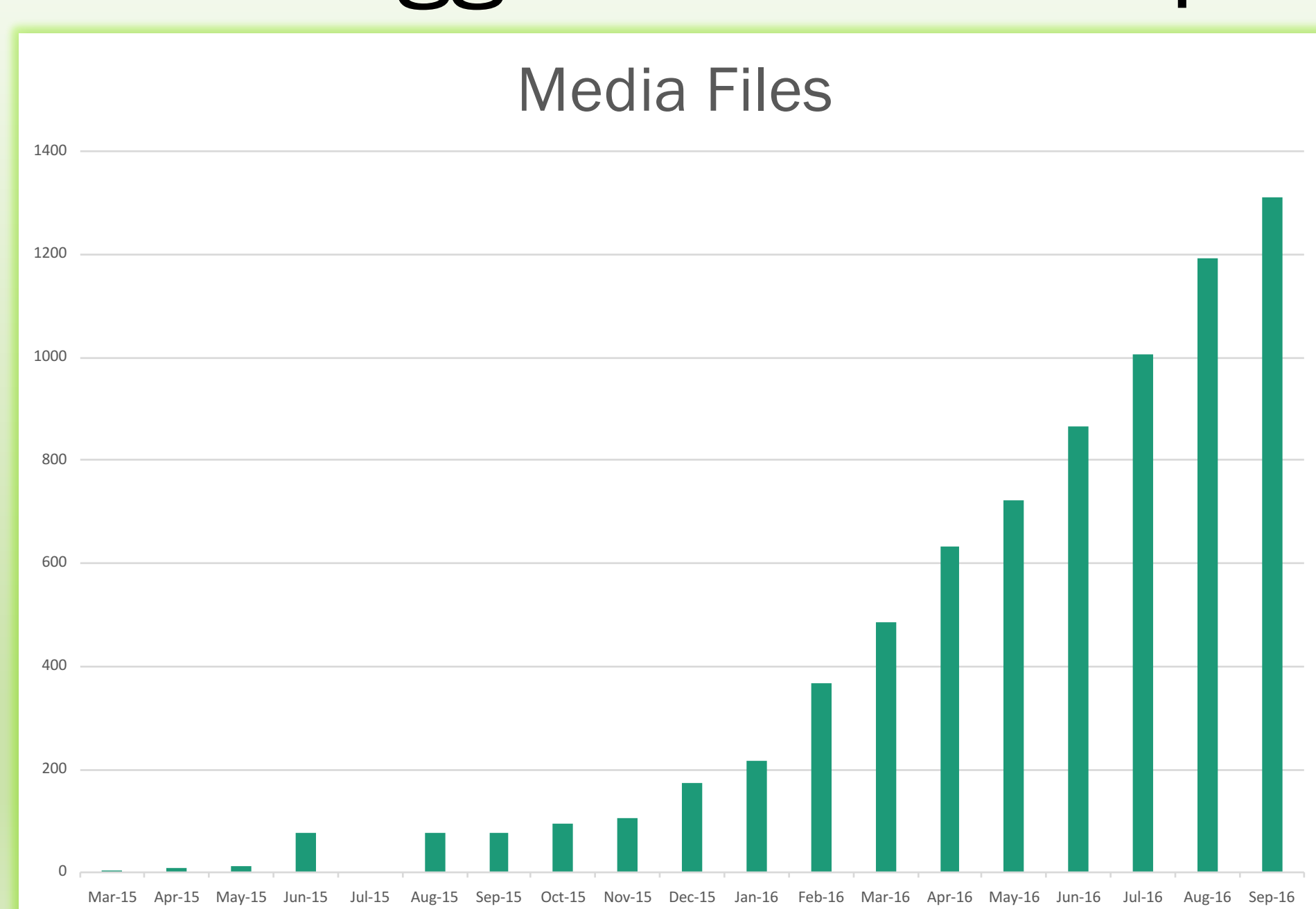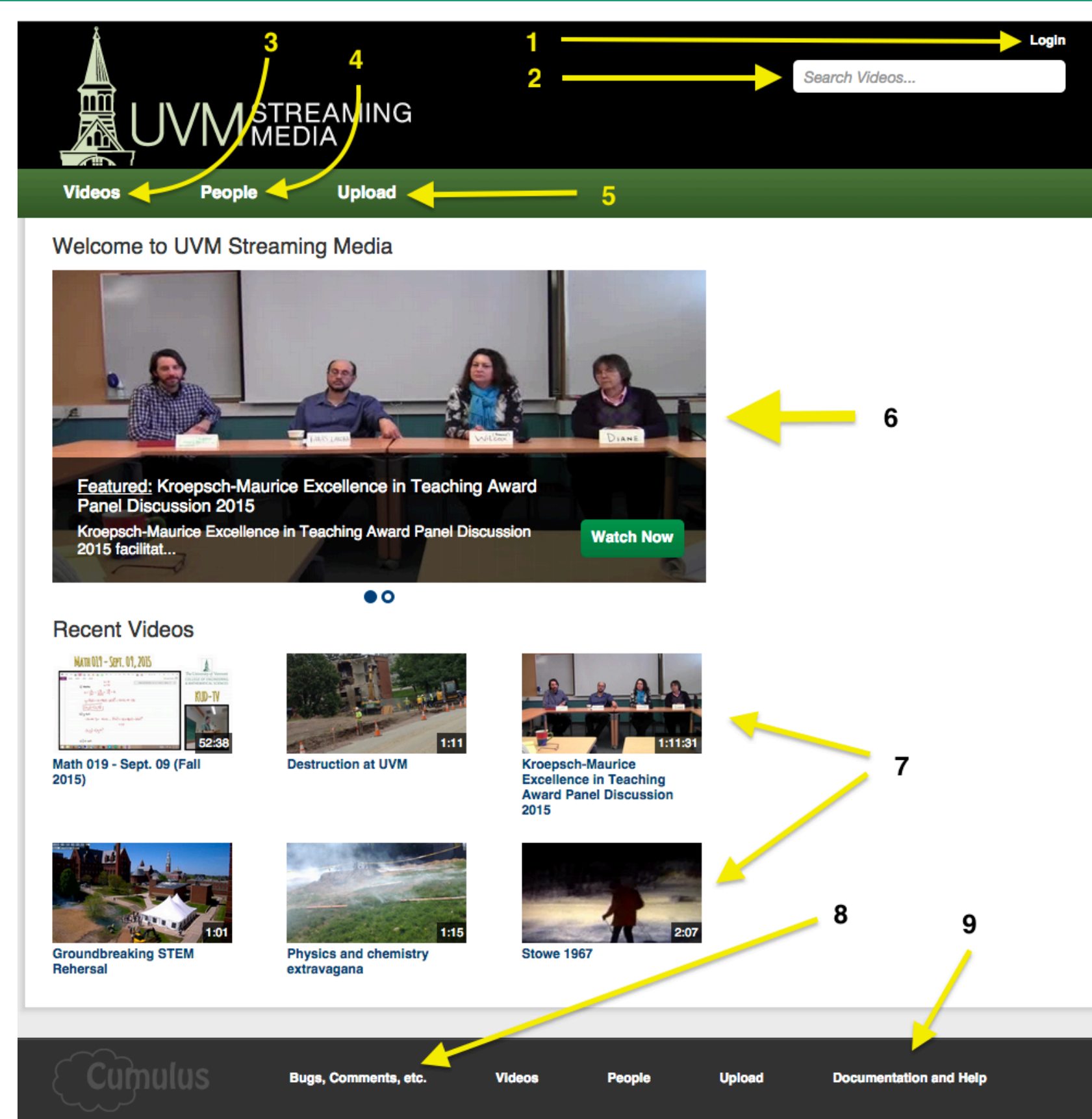
Tour

1. Log on with your UVM NetID and password (same as for Blackboard, Webmail, MyUVM, etc.) Login is necessary to view UVM-restricted videos or to upload a video.
2. Search for videos by owner's name, NetID, video titles, tags, and descriptions. Search will not find videos flagged as UVM restricted if user is not logged in. Search will not find videos flagged as private.
3. Browse videos: if not logged in, only videos not flagged as UVM restricted or private will be visible. If logged in, all videos not flagged as private will be visible.
4. Browse contributors: if user is not logged in, this feature displays a list of users who have uploaded at least one video not flagged as UVM-restricted nor private. If user logged in, it displays a list of users who have uploaded at least one video not flagged as private.
5. Upload a video.
6. Slideshow cycles through public videos flagged by system administers as "featured." Suggestions welcome!
7. Shows the 6 most recent public videos.
8. Submit bug reports, comments, and suggestions.
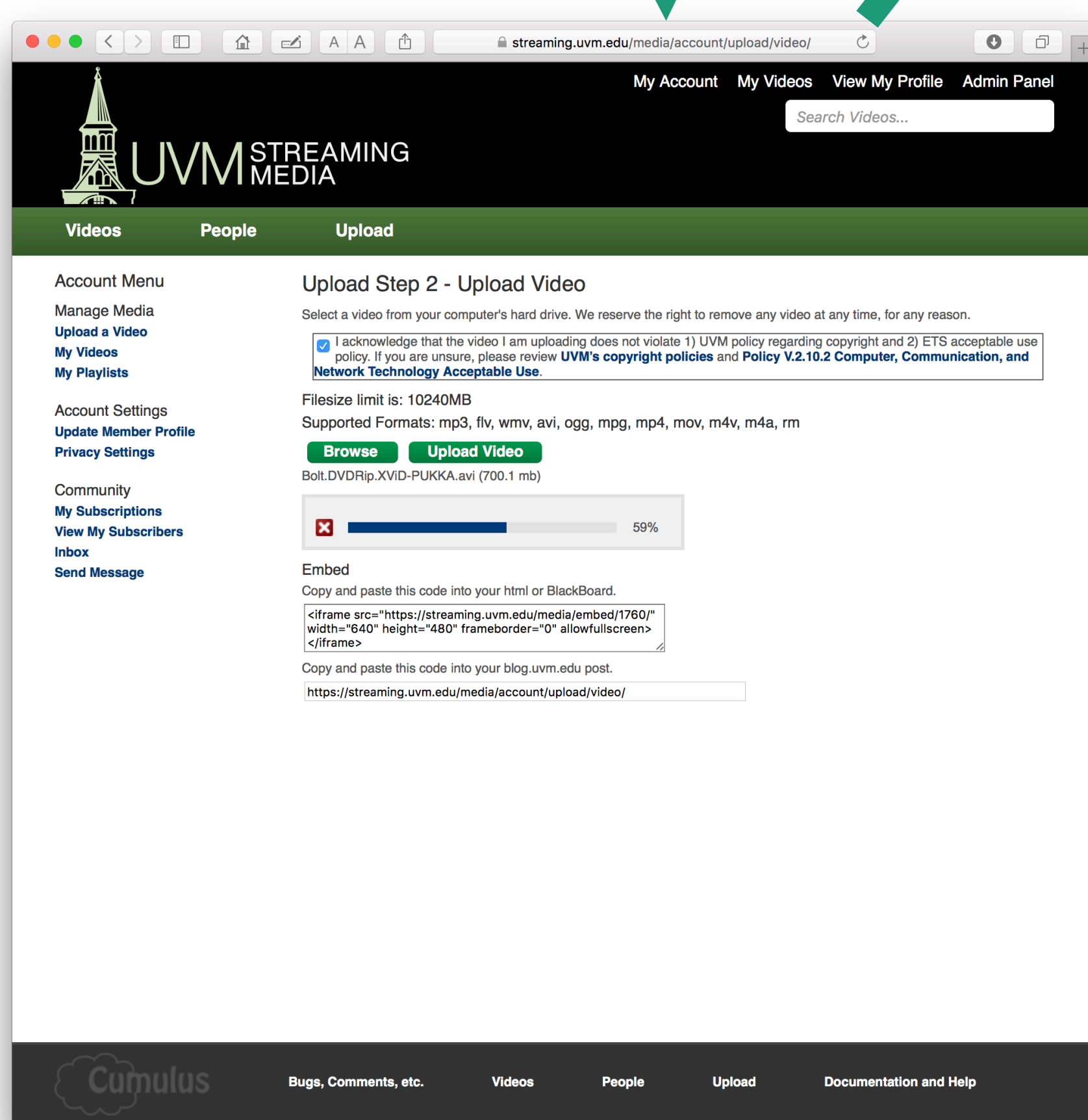9. "Documentation and help" brings you there.
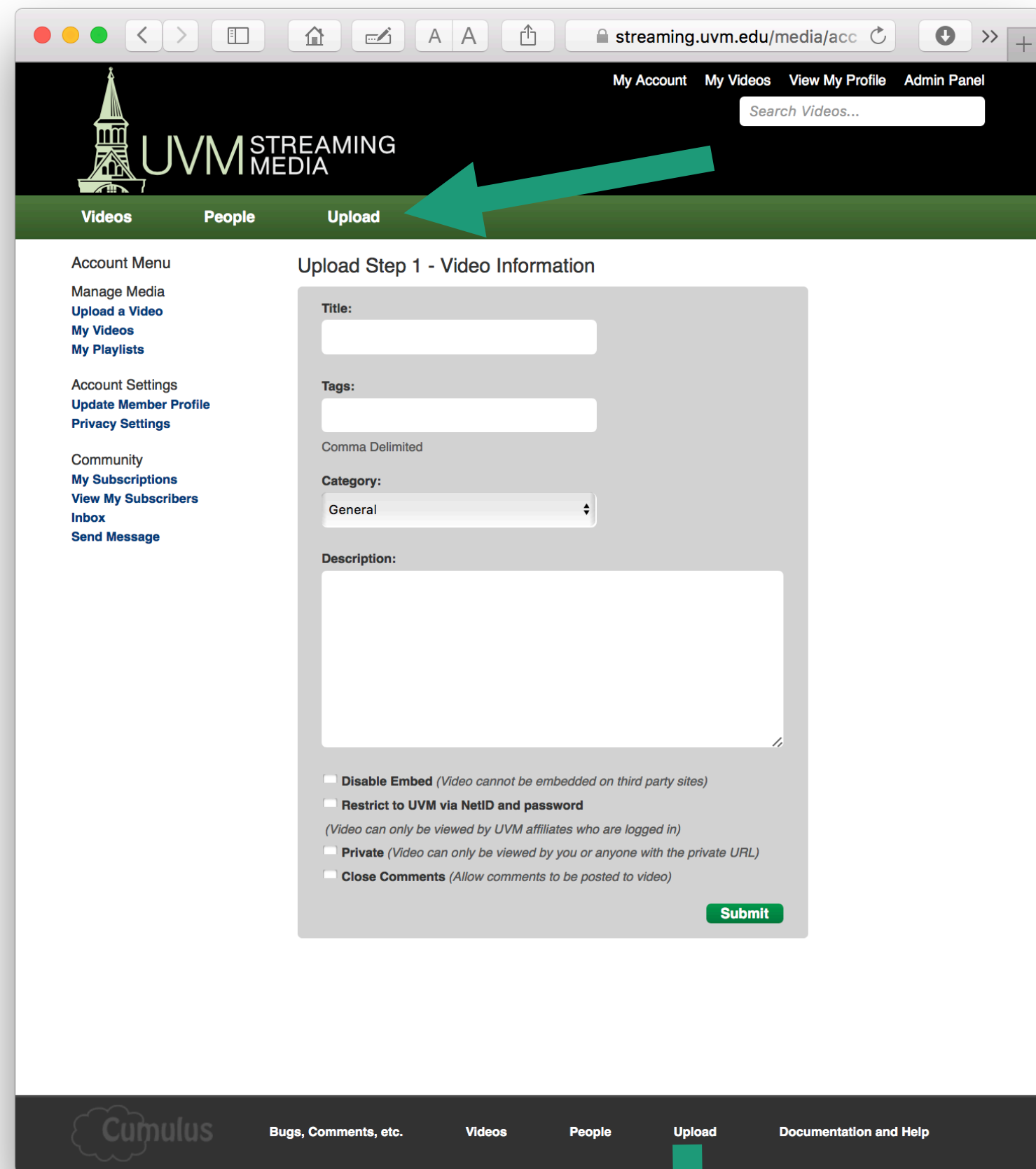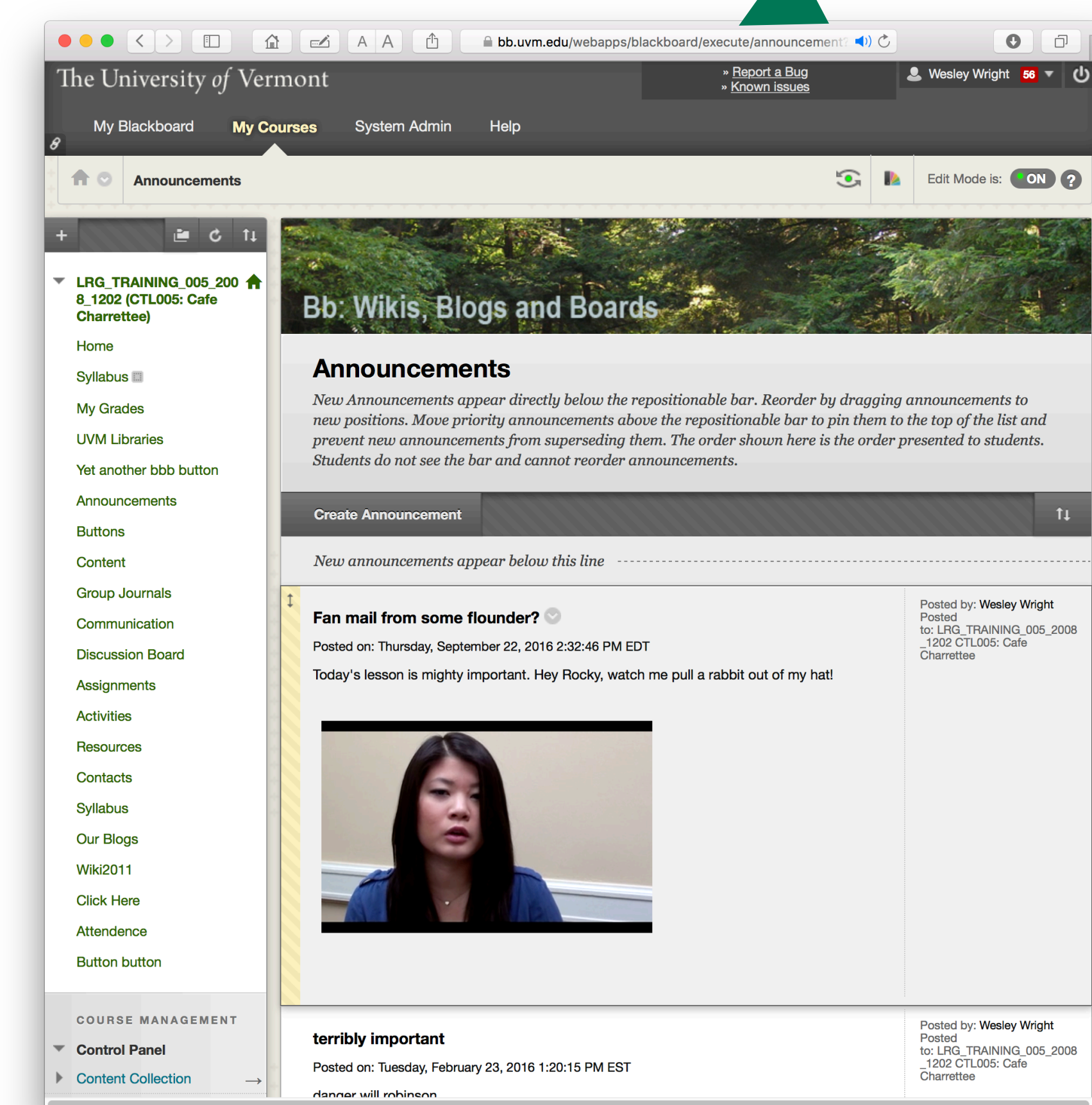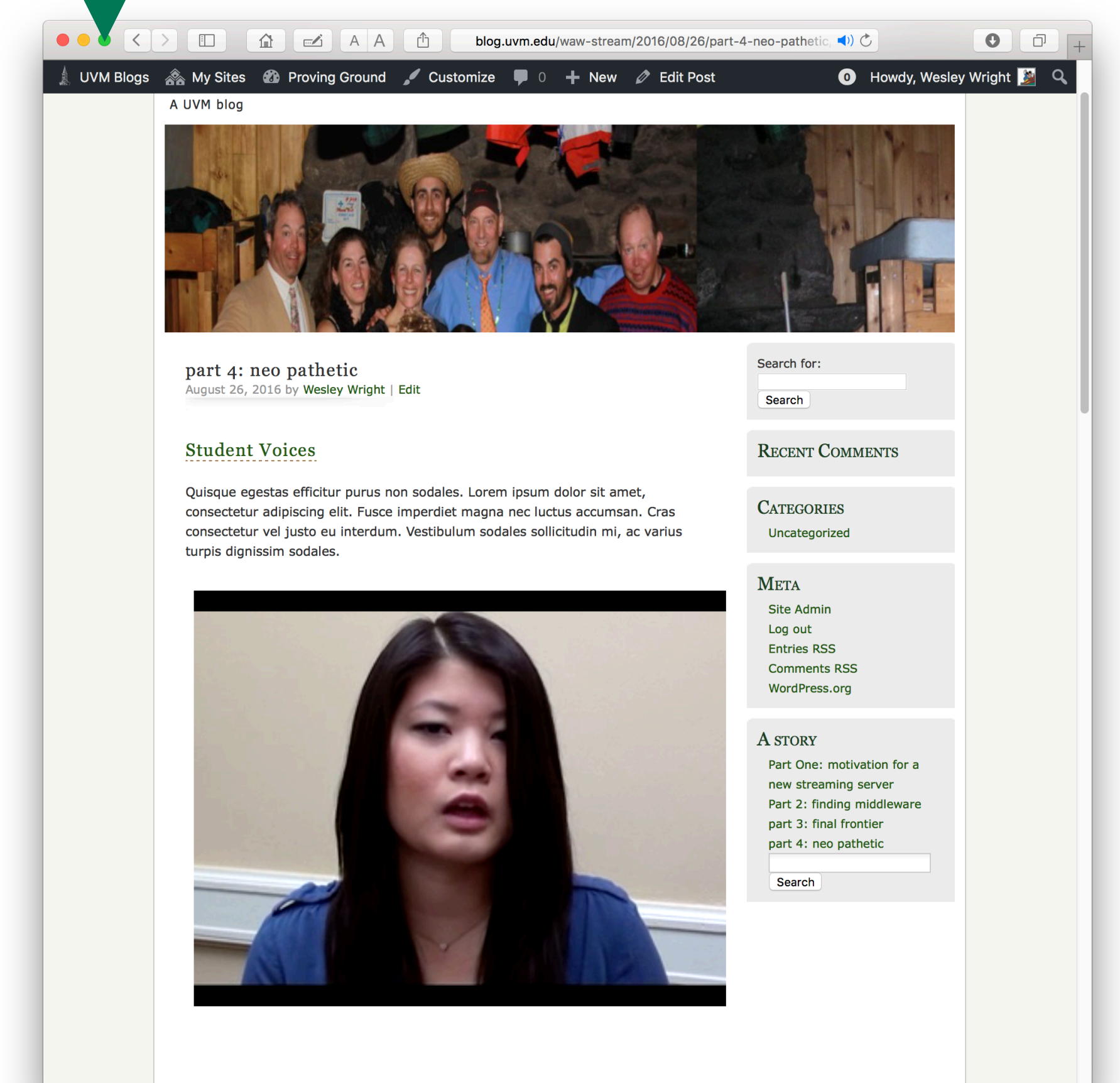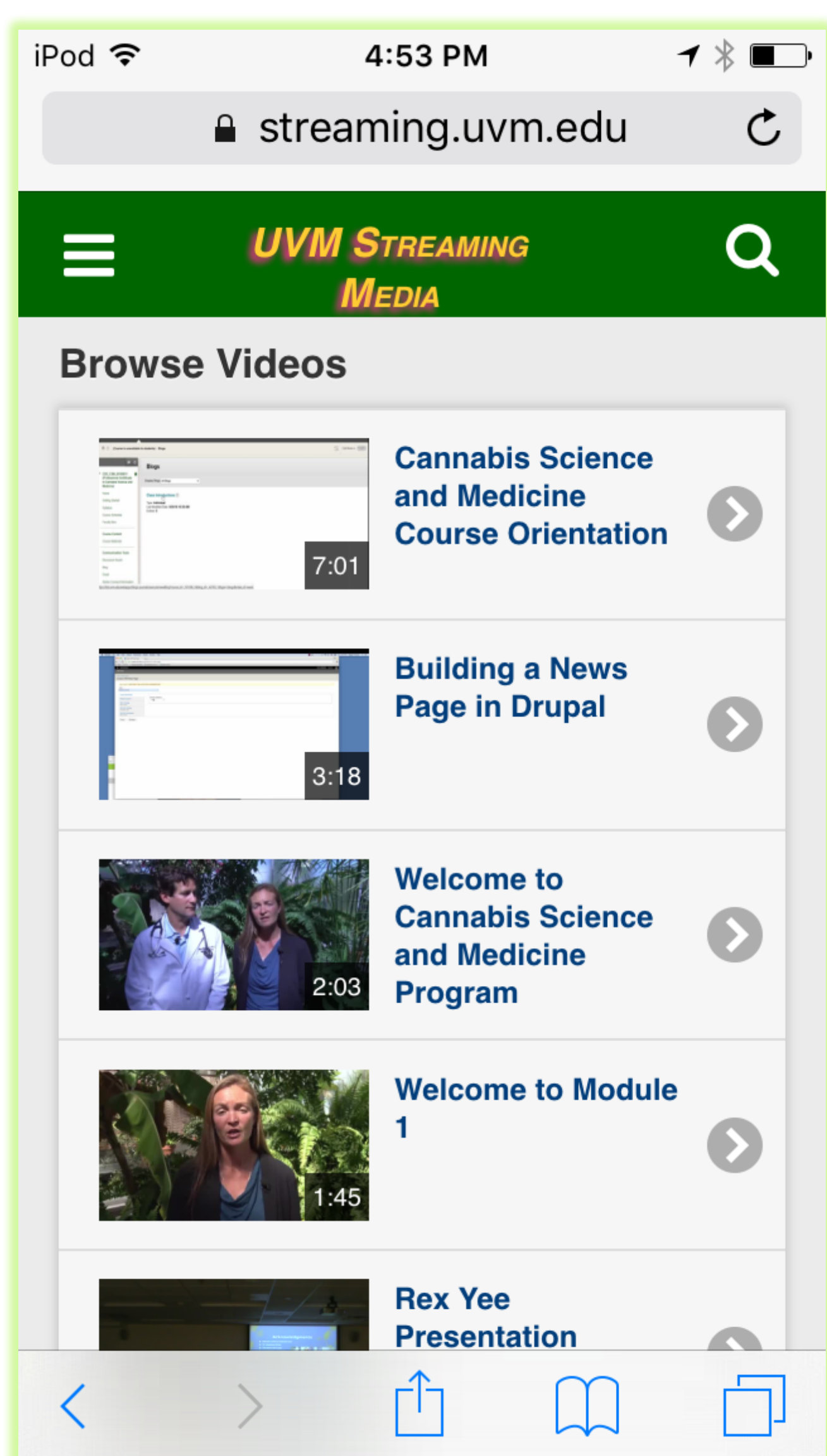
Upload Media Metadata

Browse Videos
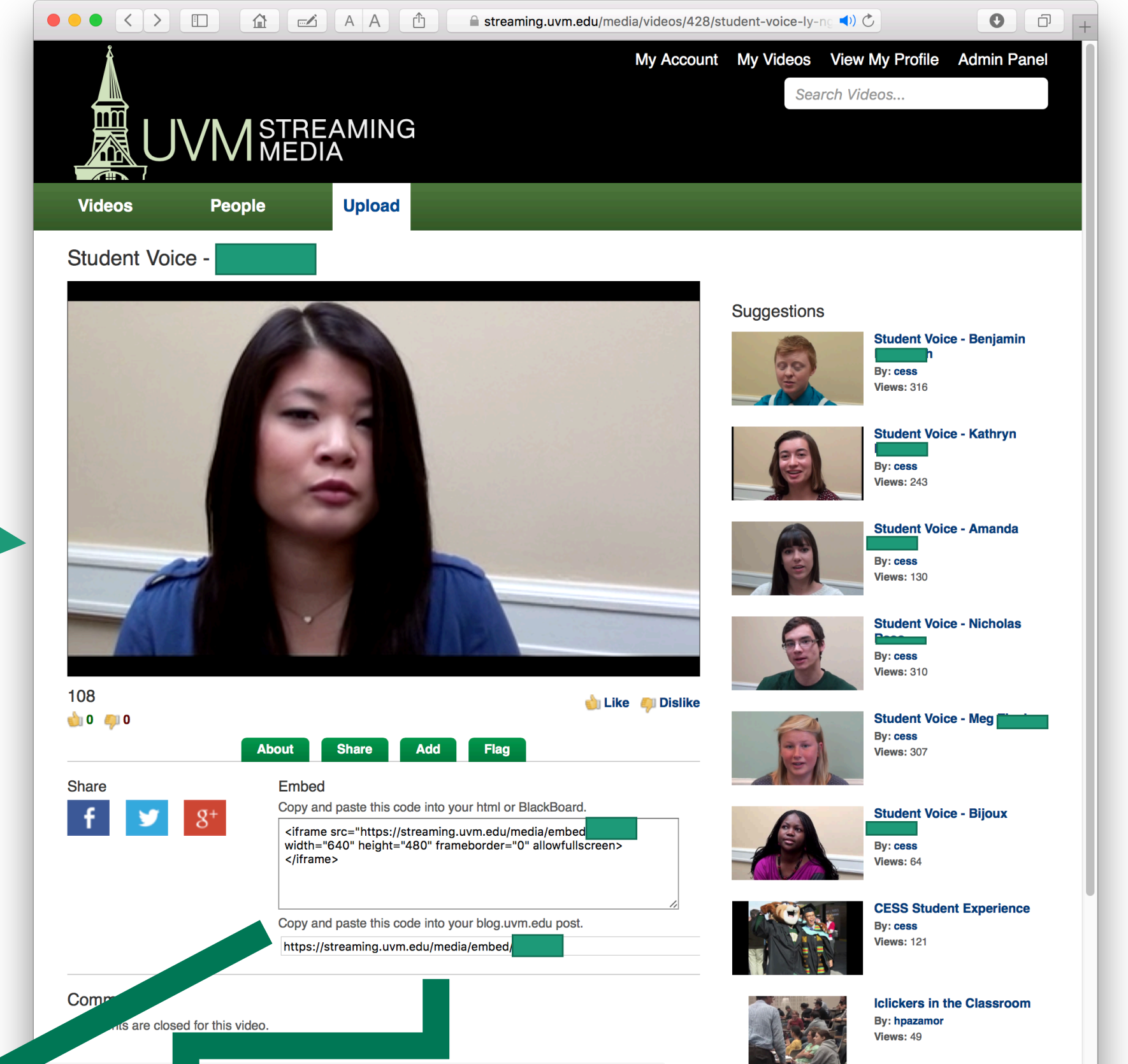
Watch Videos

Upload Media File

Embed in Blackboard (iframe)
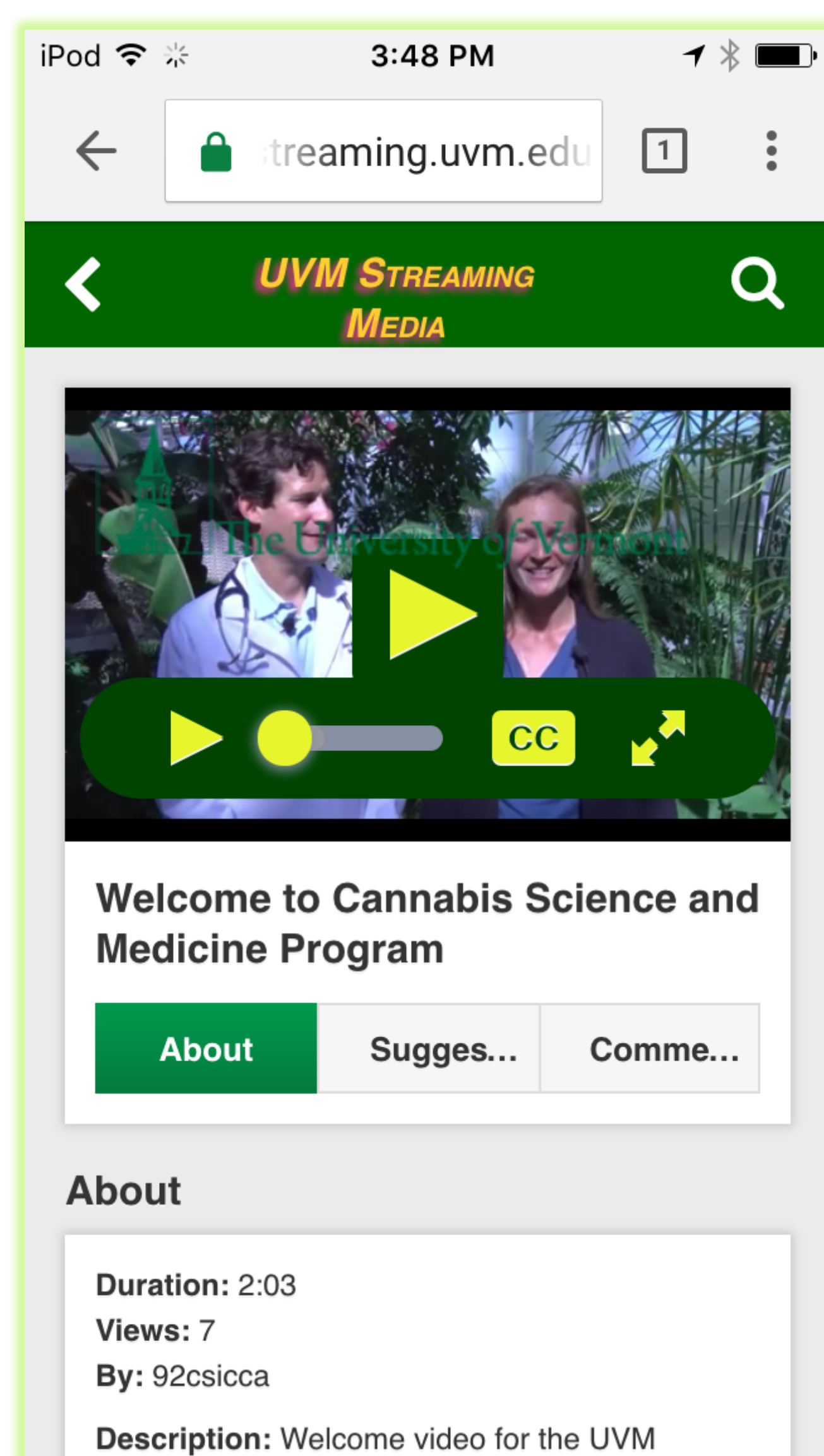
Embed in WordPress (oEmbed)
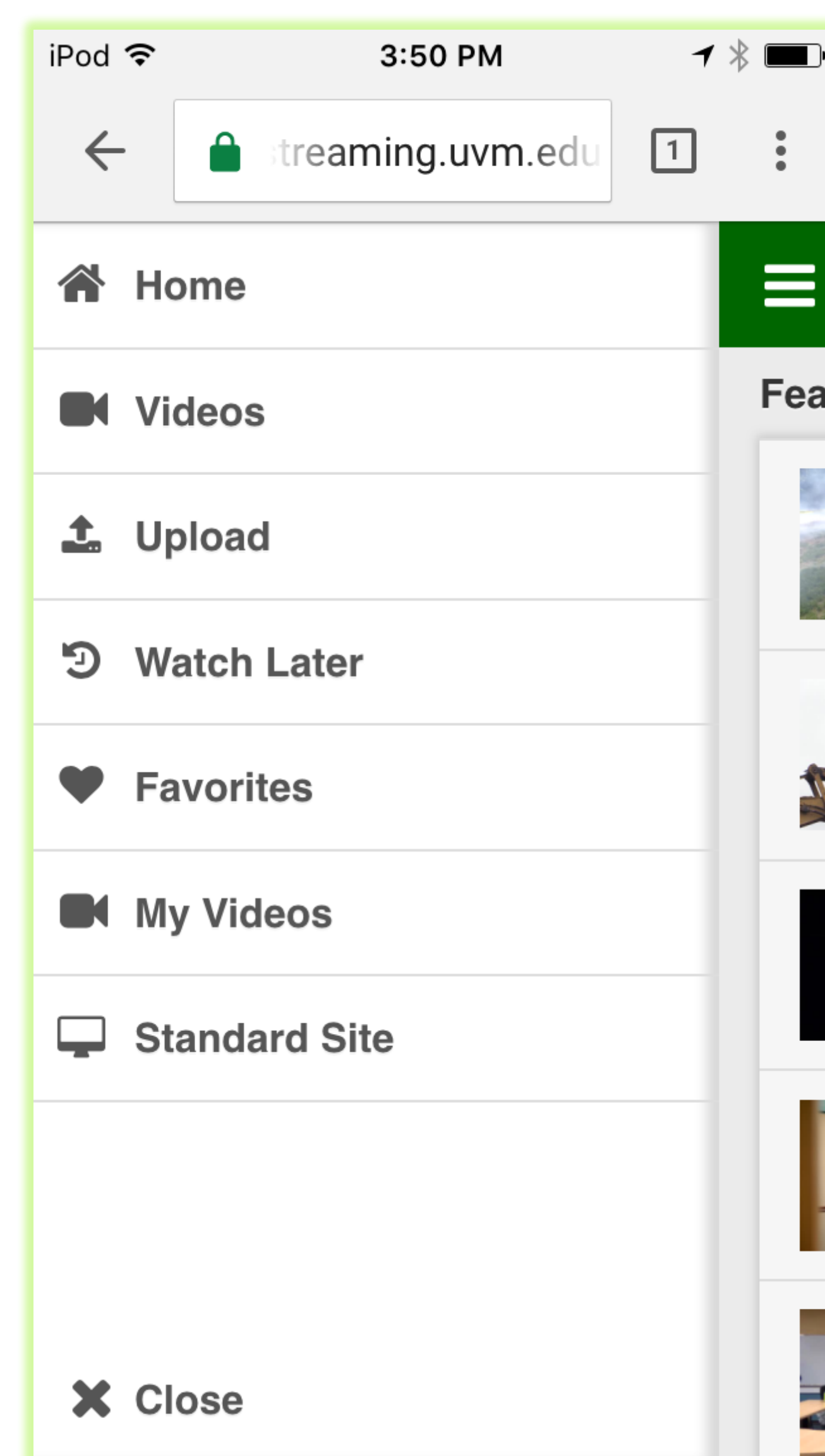
Share Videos (Blackboard)

Share Videos (WordPress)

Mobile Browse

Mobile Watch

Mobile Menu

Challenges, Successes, Lessons

- As always, browsers are a moving target.
- Goal was 100% MPEG-DASH with HLS fallback rendered in HTML5, no Flash (achieved 2016-10-10).
- ffmpeg settings arcane, akin to black magic. Lots of trial and error.
- On fast campus network, users can upload much faster than transcoding can complete, resulting in dozens of competing jobs. Need a queuing system.
- Local code development branch needs to merge back with open source Cumulusclips.org code base.
- Is this an LTA?
  - Cost was under $2000, plus initial staff development time.
  - Guarding against feature creep and bloat.
  - Support requests have been few and far between => easy to learn and non-intimidating.
  - Growing community of users and quantity of media assets => easy to learn and non-intimidating.
  - About 12% of media is student submissions.