'Fennec' Project Backlog

John Du, Mitch Holm, Shawn Nirappil, George Lo, Joe Coy,
Michael Crabill, Timothy Vincent

1. Problem Statement

There is no current solution for a community driven synchronous youtube videos and sound clip playback web app with a social aspect between participating users. Others existing systems with synchronous video don't have a way to see future videos that have been enqueued. Also these systems lack both malicious user removal and a blended of queue for music and videos.

2. Background Information

Multiple versions of synchronized listening have been attempted. A previous solution called Plug.dj relied too heavily on revenue from membership and consequently, shut down. Another solution called Sync Video offered the basic functionality of letting people join rooms and watch synchronized videos, although it lacked in two key areas: only the owner could queue up songs, and searching for youtube videos had to be done outside the website. Our website will allow multiple users to enter media into the queue, and allow searching YouTube and SoundCloud within the website.

3. Environment

Our development environment will include Python with the Flask library for our RESTful API on the backend. Node.js will be used for the parsing of YouTube and SoundCloud into JSON format. The frontend will be written in HTML, CSS and JavaScript with use of the JQuery library to communicate to the backend.

4. Functional Requirements

Backlog ID	Functional Requirements	Hours
1	As a user, I would like to view YouTube videos SoundCloud media in a synchronized manner with my friends and strangers.	5
2	As a user, I would like to be able to communicate with my friends and strangers through a text based medium.	5
3	As a user, I would like to be identified by a unique username of my choosing.	3
4	As a user, I would like to hear a sound when a message is posted.	2
5	As a user, I would like to be able to search available media through Fennec.	8
6	As a user, I would like to be able to search YouTube's results for a query.	5

7	As a user, I would like to be able to search Soundcloud's results for a query.	5
8	As a user, I would like to be able to preview items returned in my search by their thumbnail images.	6
9	As a user, I would like to add media to the queue to be played.	5
10	As a user, I would like to vote to remove content from the queue.	6
11	As a user, I would like to vote to remove a user's ability to add to the queue.	3
12	As a user, I would like to vote to kick malicious users from a station.	3
13	As a user, I would like to create 'stations' that can be joined by others.	10
14	As a user, I would like to be able to set a color for my station to differentiate it from others.	2
15	As a user, I would like to be able to give a sharable link to join a 'station'.	4
16	As a user, I would like to browse joinable 'stations' through the web client.	9
17	As a user, I would like to decide if my 'station' is visible in the station browser.	2
18	As a user, I would like to be able to name my station.	2
19	As a user, I would like to see stations sorted by viewer count.	6
20	As a user, I would like to be able to see the nicknames and total count of listeners in each station.	5
21	As a user, I would like to circumnavigate YouTube ads (ads may create issues with synchronized playback).	10
22	As a user, I would like to be able to view and copy the direct link to the media.	2
	Total:	120

5. Non-Functional Requirements

- ■ Responsiveness: Searches for media should not require a loading screen, and appear reasonably quickly after the user searches. The UI should not stutter or lag when used.

- ■ Reliability: Stations should be persistent until everyone in the station leaves. The user should always be able to join or create a station without bugs.
- ■ Efficiency: Minimize requests to the server, and minimal strain on the user's device.
- ■ Synchronized: Users should be synced to other users within a few seconds, to enable a unified listening and watching experience.
- ■ Aesthetics: Clean, modern design that is both minimal and simple, and uses a clear and consistent color scheme that reflects user preferences.
- ■ Intuitive: The site is immediately understandable to the average user.

6. Use Cases

Each is defined by

Name: Short descriptive name and the backlog ID

Steps: 2 column to describe user actions and system responses

Case: Sync Media (1)	System Responses
1. User tries to start watching video	2. User's device request time stamp from server
	3. Receives timestamp
	4. Plays video at given time, accounting for delays

Case: Communicate through Text (2)	System Responses
1. User clicks to join a station	2. The users is added to the list of connected users in that station
3. User types text and presses enter	4. User's text is parsed, then relayed through a server to other connected users in that station
	5. The user's message appears

Case: Create Username (3)	System Responses
1. User creates a name for the website	2. The page loads and requests username
3. User provides a username	4. The user is notified if the provided name is available

5. The user confirms their username.	6. The username is sent to the server

Case: Message Sound (4)	System Responses
1. User joins a station and is listening on IRC	2. A message is received from the server
	3. A sound is played.

Case: Search Fennec (5)	System Responses
1. User enters search query	2. System sends queries to server
	3. System returns parsed results in a nice format
4. User may click items on list	

Case: Search YouTube (6)	System Responses
1. User enters a search query text into the search box	
2. User presses the search button	3. System queries YouTube
	4. System returns parsed results
5. User may click items on list	

Case: Search SoundCloud (7)	System Responses
1. User enters a search query text into the search box	
2. User presses the search button	3. System queries Soundcloud
	4. System returns parsed results
5. User may click items on list	

Case: Thumbnail Images in Search (8)	System Responses
1. User enters search query	2. System queries Youtube and Soundcloud

	3. System displays thumbnail images for these images as they are found from their parent sites

Case: Add Media to Queue (9)	System Responses
1. Users enters a station then submits a search query in the search bar, clicking on the desired result they want to add	2. System requests query from server
	3. Returns the search results in correct format
5. User scrolls through the search query and clicks item they would like to add	4.The user's privilege is checked, and a request is sent with the media name, URI and user who added it.

Case: Remove Queue Content (10)	System Responses
1.User requests content to be removed with either clicking an item in queue or via IRC message	2. System sends request to server to remove content
	3. When vote requirement is met, content is removed

Case: Remove User Privilege (11)	System Responses
1. User sends a request to remove a user's ability to add content to the queue.	2. When specified threshold is reached, the server ignores all future requests to add content to the queue.

Case: Remove User from Station (12)	System Responses
1. User requests another user to be removed from the station via IRC message.	2. Server removes user from the station when majority is reached

Case: Create Station (13)	System Responses
1. User presses 'Create Station' button	2. Server creates station box and returns a

	box with the options
3. User fills in desired title	
4. User fills in desired color	
5. User says whether it will be public or not	
6. User presses 'Create' button	7. System redirects to new station if there are 100 or less stations currently active

Case: Set Station Color (14)	System Responses
1. The station creator clicks the color box	2. A color picker appears
3. The user may choose a color	
4. The user confirms their color	5. The color is shown in the color button
	6. The color will be shown in the station upon station creation

Case: Create Shareable Link (15)	System Responses
1. Inside a station, the user clicks on the share button	2. The web client displays the link on the web client
3. The user can copy and paste the text link to send others to their station	

Case: Browse Joinable Stations (16)	System Responses
1. User loads the fennec homepage	2. System returns available stations
3. User sees various station titles in their given color.	

Case: Change Station Visibility (17)	System Responses
1. The user attempts to create a station	
2. The user checks station visibility to false	
3. The user fills in other information correctly	

4. The user creates the station	5. The station is created
	6. Other users cannot see the video on the homepage

Case: Station Name (18)	**System Responses**
1. User attempts to create a station	
2. Station creator provides a station name.	
3.User fills out privacy and color	
4. User presses 'Create'	5. Station is created in the given name

Case: Sort by Viewer Count(19)	**System Responses**
1. User clicks to view station list	2. Sort and list stations to display to user

Case: Names & Count (20)	**System Responses**
1. User clicks button to display users in station	2. Sort and list users in station to display to user

Case: Skip Ads (21)	**System Responses**
1. User joins station	2. System uses a specific content link to prevent an ad from playing

Case: Direct Media Link (22)	**System Responses**
1. User clicks button for direct link to current playing media	2. Display pop-up with direct link to copy