**Description:**

World hunger got you down? Worried about the sorry state of our economy? No matter who you are, everyone has a reason to be upset. There is, however, at least one thing to be excited about. The best website in the world finally has a mobile app.

t3k.no is an user-driven, electronic music community. Like Dubstep? House? Trance? Curious about electronic music? Then this is the app for you. With t3k.no, you can discover new songs in a variety of genres or browse through our archives for old anthems. Still can’t find the song you are looking for? No problem. t3k.no makes it easy to add additional songs to the database directly from YouTube. If the community likes your submission you will earn reputation based of its score.   
  
So what are you waiting for? It’s time to download t3k.no and stay up-to-date with the latest tracks. Impress friends with your awesome taste in music and help contribute to the largest database of electronic music in the world\*.  
  
\* - Blatant lie. Based off no evidence whatsoever.

**Overview**

I want to make an app for my website <http://t3k.no>. The website is a user-generated, social music player focusing on electronic music. Electronic music (EDM) is becoming more and more popular, and with so many new tracks being released everyday, it is difficult to keep up with the latest popular tracks. The website allows users to submit links to YouTube songs they like, and other users have the ability to vote on songs. Users can filter songs by specific genres, upload date, artist, or score. However, the website uses an embedded flash player, making it inaccessible via iOS devices.   
  
I want the app to have the same basic functionality of the site. Users should be able to start the app, hit play, and continuously listen to songs. They should also be able to use the specified filters, search the database, and vote on songs.

**Intended Audience**

Fans of EDM or anyone curious about that type of music.

**Challenges**

* Getting around the flash restrictions on Apple
* Detecting when one song has finished playing so the app can automatically start the next song
* Cleverly fitting everything into one view (fairly sure that if you switch views, it is impossible to keep the youtube video playing)

**Resources**

* UIWebView
* TableView
* NSURLConnection + JSON

**Mock Screens**

**Goals – Prototype**

* All songs accessible and filters working
* Voting functionality
* Search functionality

**Goals – Production**

The website itself if still very much under construction so I expect the app the grow along with the site.

* Usernames
* Social Media Integration
* Commenting