AudStrum specification

The idea of Audstum is to create a sound file search engine with similar mechanics to googles ‘I’m feeling lucky’ with the option to bookmark and save found clips along with your own uploaded files.

The website should use an API to search for sound clips on ‘freeSound’ that will – if found – display both the file and its information for the viewer to see. Secondly, the user will be able to create an account where they can save their searches. Finally, users will have the option to upload their own audio clips which other users of the site can search for.

we will need 4 main pages –

* A search home page which will use the API to search freesound parallel to uploaded files. This page will include a save option the append the file to the database
* A create account/login page which will append the user’s information to a database or validate a pre-existing one.
* An account page which will be available after logging in which will display the files the user would have saved.
* An upload page for users to upload their own clips.

If we were to make money from this – we might consider doing so via royalty fees from the audio files users have uploaded to us – giving them a percentage of capital from their clips we give out to customers. If this was the case – we might consider a fifth page for use of our audio clips. This will also mean users will need to sign an agreement and optionally supplying a secure risk-free method of being funded/buying rights i.e. PayPal. Otherwise, we will need to store such information in the database. Note: saved clips got from freeSound no not apply.

We will need to use different tools in order to create Audstrum –

* 4 HTML pages
* A single CSS stylesheet
* A database that makes use of NoSQL
* Use of Jquery for website dynamics
* Smaller Additional techniques to link elements together.

The final site (excluding the optional rights page) should consist of 6 components – 4 HTML, 1 CSS and 1 database which we can split up between the 4 of us so we have equal amounts to work on.

We could assign 1 HTML page per person and look at the database and stylesheet as a group (or split into teams of 2)

Due to the nature of the website, we should work on the database first, followed by the HTML and then come together with our ideas for style.