

Documentation

A note to Professor Roy: based on the examples that you have given us, we have elected to use [this](#) example as our template/inspiration.

This document will address the several components found in **BTSverse/src/app**. Each component used in the BTS-Verse dashboard requires four different files:

- An HTML base (please be aware that HTML for Angular is somewhat different than typical HTML)
- A CSS sheet for the style and visual layout
- A TypeScript file that into which the HTML and CSS files are imported, converted into a piece usable by Angular with the constructor() method, and then exported
- A second TypeScript file that is automatically created and updated as you program. You do not need to edit this file.

The BTS-Verse dashboard was created to be a one-stop-shop to anyone interested in BTS. It condenses information from many different sources such as YouTube, Twitter, and Spotify and puts it all in one place. Another perk is that we have a data visualization of user interaction with the BTS twitter account over the period of a month. The product was written using Angular.js and hosted on Amazon Web Services. From Amazon Web services, we used RDS for the MySQL database services and S3 to host the website.

Wikipedia

The Wikipedia component displays the first paragraph of BTS' Wikipedia page, embedded straight from Wikipedia itself. The widget is updated with each refresh, and a timestamp of when the last update was can be seen at the bottom of the widget. This timestamp also contains a hyperlink to the full Wikipedia page.

Wikipedia API Tutorial: <https://codesnippet.io/wikipedia-api-tutorial/>

Twitter

The Twitter component is a holder for a list of 'cards' that contain parsed Twitter data. It is updated based on when the database was last updated with new Tweets. It formats and displays the Name of the Tweeter, their screen-name, tweet-text, like/retweet count, and date/time of creation. It contains quick embed links to the respective BTS twitter pages. Twitter API Tutorial:

<https://developer.twitter.com/en/docs/labs/tweets-and-users/quick-start/get-tweets>

Spotify

The Spotify component is a consistently-updated list of BTS' top ten most played tracks. The user can listen to a thirty-second preview of the song before they are instructed to

log into Spotify so they can listen to the whole song. This component can be edited by the programmer to also reflect an album or a playlist (which too will be updated with each refresh). It too contains an icon that will bring you to BTS' Spotify page.

Spotify API Documentation: <https://developer.spotify.com/documentation/web-api/>

YouTube

The YouTube component displays the most recent video uploaded to the Bangtan TV YouTube channel. The widget contains the typical features of YouTube video such as the ability to go full-screen, timeline scrubbing, and volume controls. All of the channel's videos have been put into a playlist in reverse chronological order, which can also be played from the widget. English subtitles are enabled by default, and the user can navigate to YouTube itself if they so desire.

YouTube API Documentation: <https://developers.google.com/youtube/v3/>

Twitter Statistics

The Twitter Data/Statistics component holds interaction over the past month from both twitter accounts '@bts-twt' and '@bts_bighit'. It is interactive and for each date, you can see the overall engagement of BTS verified accounts on twitter using combined likes/retweets. This data is also pulled from the database, and refreshes on update.

Chart.js Tutorial: <https://www.chartjs.org/docs/latest/>

Social Media Links

The social media links are href embeds that link to the respective social media platform of BTS. The icons are formatted for aesthetics, and included on the floating header, as well as within each available component. This also gives the user access to BTS' socials that weren't able to be implemented, such as Soundcloud and TikTok.

HRef Button Tutorial:

<https://www.w3docs.com/snippets/html/how-to-create-an-html-button-that-acts-like-a-link.html>