

---

**Datalytics**

by Byte Peeps



# System Design document

1<sup>st</sup> July 2022

**Collective work by** Mohamed Tayeh (Moh), Bassel Ashi, Connie Lin, Karandeep Lubana (Giani), Jiaming Yang (Gloria), and Juan Camilo Corral

# Table of Contents

<b>Table of Contents</b>	<b>2</b>
<b>CRC Cards</b>	<b>3</b>
Database Model Classes	3
Backend CRC Classes	7
Frontend Auth Classes	10
Navigation Classes	11
System Interaction	<b>13</b>
<b>System Architecture</b>	<b>14</b>
<b>System Decomposition</b>	<b>15</b>

# CRC Cards

## Database Model Classes

\* Newly added components for this sprint are in purple

User		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Database model for application users</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>username: string</li> <li>password: string</li> <li>deletedAt: Date</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>FacebookApi</li> <li>InstagramApi</li> <li>YouTubeChannel</li> <li>TwitterUser</li> <li>RedditSubreddit</li> <li>YelpBusiness</li> <li>GoogleReviewsAccount</li> </ul>

FacebookApi		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Database model for Facebook APIs</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>token: string</li> <li>isActive: boolean</li> <li>userId: number</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>FacebookPost</li> </ul>

FacebookPost	
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Database model for Facebook posts</li> </ul>	<b>Fields</b> <ul style="list-style-type: none"> <li>message: string</li> <li>date: Date</li> <li>likes: number</li> <li>loves: number</li> <li>cares: number</li> <li>hahas: number</li> <li>wows: number</li> <li>sads: number</li> <li>angrys: number</li> </ul>

		<ul style="list-style-type: none"> <li>dataId: string</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>FacebookApi</li> <li>FacebookMedia</li> </ul>

FacebookComment		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Database model for Facebook comments</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>dataId: string</li> <li>username: string</li> <li>message: string</li> <li>date: Date</li> <li>likes: number</li> <li>sentimentAnalysis: string</li> <li>topicClassification: string</li> <li>subjectivityAnalysis: string</li> <li>postId: number</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>FacebookPost</li> </ul>

InstagramApi		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Database model for Instagram API</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>facebookApId: number</li> <li>userId: number</li> <li>noId: string</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>FacebookApi</li> <li>InstagramMedia</li> </ul>

InstagramMedia		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Database model for Instagram media</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>dataId: string</li> <li>caption: string</li> <li>date: Date</li> <li>likes: number</li> <li>numComments:</li> </ul>

		number <ul style="list-style-type: none"> <li>• apild: number</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>• InstagramApi</li> <li>• InstagramComment</li> </ul>

InstagramComment		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>• Database model for Instagram comments</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>• dataId: string</li> <li>• userName: string</li> <li>• message: string</li> <li>• date: Date</li> <li>• likes: number</li> <li>• sentimentAnalysis: string</li> <li>• topicClassification: string</li> <li>• subjectivityAnalysis: string</li> <li>• mediaId: number</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>• InstagramMedia</li> </ul>

InstagramTag		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>• Database model for Instagram tags, which are basically posts that the user is tagged in</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>• dataId: string</li> <li>• username: string</li> <li>• caption: string</li> <li>• date: Date</li> <li>• likes: number</li> <li>• numComments: number</li> <li>• sentimentAnalysis: string</li> <li>• topicClassification: string</li> <li>• subjectivityAnalysis: string</li> <li>• apild: number</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>• InstagramApi</li> </ul>

YouTubeChannel
----------------

<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Database model for YouTube channels</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>resourceId: string</li> <li>name: string</li> <li>userId: string</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>YouTubeVideo</li> </ul>

YouTubeVideo		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Database model for YouTube videos</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>resourceId: string</li> <li>date: number</li> <li>views: number</li> <li>likes: number</li> <li>channelId: number</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>YouTubeChannel</li> <li>YouTubeComment</li> </ul>

YouTubeComment		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Database model for YouTube videos</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>resourceId: string</li> <li>userName: stringr</li> <li>message: number</li> <li>likes: number</li> <li>sentimentAnalysis: string</li> <li>topicClassification: string</li> <li>subjectivityAnalysis: string</li> <li>videoid: number</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>YoutubeVideo</li> </ul>

YouTubeComment		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Database model for YouTube videos</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>resourceId: string</li> <li>userName: stringr</li> <li>message: number</li> <li>likes: number</li> <li>sentimentAnalysis: string</li> <li>topicClassification: string</li> <li>subjectivityAnalysis: string</li> <li>videoid: number</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>YoutubeVideo</li> </ul>

TwitterUser		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Database model for Twitter users</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>twitterId: string</li> <li>userName: string</li> <li>userId: string</li> <li>deletedAt: Date</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>User</li> <li>TwitterTweet</li> </ul>

TwitterTweet		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Database model for Twitter tweets</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>twitterId: string</li> <li>conversationId: string</li> <li>text: string</li> <li>date: Date</li> <li>retweets: number</li> <li>replies: number</li> <li>likes: number</li> <li>deletedAt: Date</li> <li>twitterUserId: number</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>TwitterConversation</li> <li>TwitterUser</li> </ul>

TwitterConversation		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Database model for Twitter conversations</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>twitterId: string</li> <li>text: string</li> <li>date: Date</li> <li>retweets: number</li> <li>replies: number</li> <li>likes: number</li> <li>sentimentAnalysis: number</li> <li>topicClassification: number</li> <li>subjectivityAnalysis: number</li> <li>deletedAt: Date</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>TwitterTweet</li> </ul>

RedditSubreddit		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Database model for Reddit subreddits</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>name: string</li> <li>userId: string</li> <li>deletedAt: Date</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>User</li> <li>RedditListing</li> </ul>

RedditListing		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Database model for Reddit listings</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>dataId: string</li> <li>title: string</li> <li>text: string</li> <li>date: Date</li> <li>score: number</li> <li>numComments: number</li> <li>permalink: string</li> <li>deletedAt: Date</li> <li>subredditId: number</li> </ul>
<b>Parent Class</b>	<b>Sub Class</b>	<b>Collaborations</b>



Model	None	<ul style="list-style-type: none"> <li>• RedditSubreddit</li> <li>• RedditComment</li> </ul>
-------	------	--

RedditComment		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>• Database model for Reddit comments</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>• dataId: string</li> <li>• text: string</li> <li>• date: Date</li> <li>• score: number</li> <li>• replies: number</li> <li>• sentimentAnalysis: number</li> <li>• topicClassification: number</li> <li>• subjectivityAnalysis: number</li> <li>• deletedAt: Date</li> <li>• listingId: number</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>• RedditListing</li> </ul>

GoogleReviewsAccount		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>• Database model for Google Reviews accounts</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>• token: string</li> <li>• accountId: string</li> <li>• isActive: boolean</li> <li>• userId: string</li> <li>• deletedAt: Date</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>• User</li> <li>• GoogleReviewsLocation</li> </ul>

GoogleReviewsLocation		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>• Database model for Google Reviews locations</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>• locationId: string</li> <li>• accountId: number</li> <li>• deletedAt: Date</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>• TwitterConversation</li> </ul>

		<ul style="list-style-type: none"> <li>GoogleReviewsAccount</li> <li>GoogleReviewsReview</li> </ul>
--	--	---

GoogleReviewsReview		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Database model for Google Reviews review</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>title: string</li> <li>review: string</li> <li>reviewer: string</li> <li>rating: number</li> <li>response: string</li> <li>date: Date</li> <li>reviewId: string</li> <li>sentimentAnalysis: number</li> <li>topicClassification: number</li> <li>subjectivityAnalysis: number</li> <li>deletedAt: Date</li> <li>locationId: number</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>GoogleReviewsLocation</li> </ul>

## Backend CRC Classes

facebook/reddit/setup/twitter/user/youtube

Controller/Instagram/Comment		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Has the API endpoint implementations for instagram comments</li> </ul>		<b>Endpoints</b> <ul style="list-style-type: none"> <li>getCommentById()</li> <li>getCommentsByMediaId()</li> <li>getCommentsSentimentAnalysis()</li> <li>getCommentsSubjectivityAnalysis()</li> <li>getWordCloudData()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Routes/Instagram/Comment</li> <li>DatumBox</li> <li>InstagramComment</li> <li>keywordExtraction</li> </ul>

Controller/Instagram/media		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Has the API endpoints for posts made by the user company</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getMediaById()</li> <li>getMediaByUserId()</li> <li>getSentimentAnalysisForTimeSeries()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Routes/Instagram/Comment</li> <li>InstagramMedia</li> </ul>

Controller/Instagram/Stats		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Provides the stats for the ui cards</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getInstagramStats()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>InstagramApi</li> <li>InstagramComment</li> <li>InstagramMedia</li> <li>InstagramTag</li> <li>User</li> </ul>

Controller/Instagram/tag		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Has the API endpoint implementations for instagram tags</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getTagById()</li> <li>getTagsByMediaId()</li> <li>getTagsSentimentAnalysis()</li> <li>getTagsSubjectivityAnalysis()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Routes/Instagram/Tag</li> <li>DatumBox</li> <li>InstagramTag</li> </ul>

Controller/Facebook/Comment		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Has the API endpoint implementations for Facebook comments</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getComments()</li> <li>getCommentsByPostId()</li> <li>getCommentsSubjectivityAnalysis()</li> <li>getWordCloudData()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>FacebookApi</li> <li>FacebookComment</li> <li>FacebookPost</li> <li>User</li> <li>keywordExtraction</li> </ul>

Controller/Facebook/Post		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Has the API endpoint implementations for Facebook posts</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getPosts()</li> <li>getSentimentAnalysisForTimeSeries()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>FacebookApi</li> </ul>

		<ul style="list-style-type: none"> <li>FacebookComment</li> <li>FacebookPost</li> <li>User</li> </ul>
--	--	---

Controller/Facebook/Stats		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Provides the stats for the ui cards</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getFacebookStats()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>FacebookApi</li> <li>FacebookComment</li> <li>FacebookPost</li> <li>User</li> </ul>

Controller/Youtube/Comment		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Has the API endpoint implementations for youtube comments</li> </ul>		<b>Endpoints</b> <ul style="list-style-type: none"> <li>getCommentById()</li> <li>getAllComments()</li> <li>getWordCloudData()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Routes/YouTube/Comment</li> <li>YoutubeComment</li> <li>keywordExtraction</li> </ul>

Controller/Youtube/channel		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Has the API endpoint implementations for youtube channels</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getChannelById()</li> <li>getAllChannels()</li> <li>createChannel()</li> <li>deleteChannel()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Routes/YouTube/Comment</li> <li>YoutubeChannel</li> </ul>

Controller/Youtube/stats		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Provides the stats for the ui cards</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getYoutubeStats()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>YouTubeChannel</li> <li>YouTubeComment</li> <li>YouTubeVideo</li> <li>User</li> </ul>

Controller/Youtube/video		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Has the API endpoint implementations for youtube videos</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getVideoById()</li> <li>getAllVideos()</li> <li>deleteVideo()</li> <li>createVideo()</li> <li>getSentimentAnalysisForTimeSeries()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Routes/Youtube/Video</li> <li>YoutubeVideo</li> </ul>

Controller/Reddit/Comment		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Has the API endpoint implementations for Reddit comments</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getComments()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>RedditComment</li> <li>RedditListing</li> <li>RedditSubreddit</li> </ul>

Controller/Reddit/Listing		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Has the API endpoint implementations for Reddit Listing</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getSentimentAnalysisForTimeSeries()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>RedditComment</li> <li>RedditLlting</li> <li>RedditSubreddit</li> <li>User</li> </ul>

Controller/Reddit/Stats		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Provides the stats for the ui cards</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getRedditStats()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>RedditComment</li> <li>RedditLlting</li> <li>REdditSubreddit</li> </ul>

Controller/Twitter/Stats		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Provides the stats for the ui cards</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getTwitterStats()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>TwitterUser</li> <li>TwitterTweet</li> <li>TwitterConversation</li> </ul>

Controller/Setup/facebook		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Has the API endpoint implementations for setting up Facebook API</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getPages()</li> <li>connectPage()</li> <li>getCurrentPage()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Routes/Setup/facebook</li> <li>User</li> <li>FacebookApi</li> <li>FacebookPost</li> <li>FacebookComment</li> </ul>

Controller/Setup/instagram		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Has the API endpoint implementations for setting up Instagram API</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getSettings()</li> <li>connectPage()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Routes/Setup/instagram</li> <li>User</li> <li>FacebookApi</li> <li>InstagramApi</li> <li>InstagramMedia</li> <li>InstagramComment</li> </ul>

Controller/Setup/Reddit		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Has the API endpoint implementations for setting up Reddit API</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getSettings()</li> <li>connectPage()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li></li> </ul>



Controller/Setup/Twitter		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Has the API endpoint implementations for setting up Twitter API</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getSettings()</li> <li>connectPage()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li></li> </ul>

Controller/Setup/Yelp		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Has the API endpoint implementations for setting up Yelp API</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getSettings()</li> <li>connectPage()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li></li> </ul>

Controller/Setup/Youtube		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Has the API endpoint implementations for setting up Youtube API</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getSettings()</li> <li>connectPage()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li></li> </ul>

Routes/Instagram/Comment		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Maps API endpoints to their routes</li> </ul>		<b>Endpoints</b> <ul style="list-style-type: none"> <li>getComments()</li> <li>getCommentsByMediaId()</li> <li>getCommentsSentimentAnalysis()</li> <li>getCommentsSubjectivityAnalysis()</li> <li>getWordCloudData()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Controller/Instagram/Comment</li> <li>keywordExtraction</li> </ul>

Routes/Instagram/media		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Maps API endpoints to their routes</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getMediaById()</li> <li>getMediaByUserId()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Controller/Instagram/media</li> </ul>

Routes/Instagram/tag		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Maps API endpoints to their routes</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getTagById()</li> <li>getTagsByMediaId()</li> <li>getTagsSentimentAnalysis()</li> <li>getTagsSubjectivityAnalysis()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Controller/Instagram/tag</li> </ul>

Routes/Instagram/Stats		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Maps API endpoints to their routes</li> </ul>		<b>Endpoints</b> <ul style="list-style-type: none"> <li>getInstagramStats()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Controller/Instagram/Stats</li> </ul>

Routes/Youtube/Comment		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Maps API endpoints to their routes</li> </ul>		<b>Endpoints</b> <ul style="list-style-type: none"> <li>getComments</li> <li>getWordCloudData()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Controller/Youtube/Comment</li> </ul>

Routes/Youtube/channel		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Maps API endpoints to their routes</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getChannelById()</li> <li>getAllChannels()</li> <li>createChannel()</li> <li>deleteChannel()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Controller/Youtube/channel</li> </ul>

Routes/Youtube/video		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Maps API endpoints to their routes</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getVideoById()</li> <li>getAllVideos()</li> <li>createVideo()</li> <li>deleteVideo()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Controller/Youtube/video</li> </ul>

Routes/Facebook/Comment		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Maps API endpoints to their routes</li> </ul>		<b>Endpoints</b> <ul style="list-style-type: none"> <li>getComments()</li> <li>getCommentsByPostId()</li> <li>getCommentsSentimentAnalysis()</li> <li>getCommentsSubjectivityAnalysis()</li> <li>getWordCloudData()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Controller/Facebook/Comment</li> </ul>

Routes/Facebook/Post		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Maps API endpoints to their routes</li> </ul>		<b>Endpoints</b> <ul style="list-style-type: none"> <li>getSentimentAnalysisForTimeSeries()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Controller/Facebook/Post</li> </ul>

Routes/Facebook/Stats		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Maps API endpoints to their routes</li> </ul>		<b>Endpoints</b> <ul style="list-style-type: none"> <li>getFacebookStats()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Controller/Facebook/Stats</li> </ul>

Routes/Reddit/Comment		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Maps API endpoints to their routes</li> </ul>		<b>Endpoints</b> <ul style="list-style-type: none"> <li>getComments()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Controller/Reddit/Comment</li> </ul>

Routes/Twitter/Comment		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Maps API endpoints to their routes</li> </ul>		<b>Endpoints</b> <ul style="list-style-type: none"> <li>getComments()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Controller/Twitter/Comment</li> </ul>

Routes/Twitter/Tweet		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Maps API endpoints to their routes</li> </ul>		<b>Endpoints</b> <ul style="list-style-type: none"> <li>getSentimentAnalysisForTimeSeries()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Controller/Twitter/Tweet</li> </ul>

Routes/Twitter/Stats		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Maps API endpoints to their routes</li> </ul>		<b>Endpoints</b> <ul style="list-style-type: none"> <li>getTwitterStats()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Controller/Twitter/Stats</li> </ul>

Routes/Reddit/Listing		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Maps API endpoints to their routes</li> </ul>		<b>Endpoints</b> <ul style="list-style-type: none"> <li>getSentimentAnalysisForTimeSeries()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Controller/Reddit/Listing</li> </ul>

Routes/Reddit/Stats		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Maps API endpoints to their routes</li> </ul>		<b>Endpoints</b> <ul style="list-style-type: none"> <li>getRedditStats()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Controller/Reddit/Stats</li> </ul>

Routes/YouTube/Video		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Maps API endpoints to their routes</li> </ul>		<b>Endpoints</b> <ul style="list-style-type: none"> <li>getSentimentAnalysisForTimeSeries()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Controller/YouTube/Video</li> </ul>

Routes/YouTube/Stats		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Maps API endpoints to their routes</li> </ul>		<b>Endpoints</b> <ul style="list-style-type: none"> <li>getYouTubeStats()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Controller/YouTube/Stats</li> </ul>

Routes/Setup/facebook		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Maps API endpoints to their routes</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getPages()</li> <li>connectPage()</li> <li>getCurrentPage()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Controller/Setup/facebook</li> </ul>

Routes/Setup/instagram		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Maps API endpoints to their routes</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>getSettings()</li> <li>connectPage()</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Controller/Setup/instagram</li> </ul>

DatumBox		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Creates a wrapper for calling the datumbox external machine learning api</li> </ul>		<b>Endpoints</b> <ul style="list-style-type: none"> <li>DatumBoxAPICall</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> None

Keyword Extraction		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Creates a wrapper to extract the latest keywords from a string text</li> </ul>		<b>Endpoints</b> <ul style="list-style-type: none"> <li>KeywordExtraction</li> </ul>
<b>Parent Class</b> Model	<b>Sub Class</b> None	<b>Collaborations</b> None

## Pipeline CRC Classes

dataPipelines/facebook		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Runs daily jobs to synchronize the database data with the facebook platform data</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>startPipeline()</li> <li>updateAccountPost()</li> <li>updatePostComments()</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>FacebookApi</li> <li>FacebookPost</li> <li>FacebookComment</li> <li>DatumBoxAPICall</li> <li>CronJob</li> </ul>

dataPipelines/twitter		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Runs daily jobs to synchronize the database data with the twitter platform data</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>startPipeline()</li> <li>updateUserTweets()</li> <li>updateTweetConversation()</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>TwitterUser</li> <li>TwitterTweet</li> <li>TwitterConversation</li> <li>DatumBoxAPICall</li> <li>CronJob</li> </ul>

dataPipelines/instagram		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Runs daily jobs to synchronize the database data with the instagram platform data</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>startPipeline()</li> <li>updateAccountMedia()</li> <li>updateAccountTags()</li> <li>updateMediaComments()</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>InstagramApi</li> <li>InstagramMedia</li> <li>InstagramTag</li> <li>InstagramComment</li> <li>DatumBoxAPICall</li> <li>CronJob</li> </ul>

dataPipelines/reddit		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Runs daily jobs to synchronize the database data with the reddit platform data</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>startPipeline()</li> <li>updateListings()</li> <li>updateComment()</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>RedditSubreddit</li> <li>RedditListing</li> <li>RedditComment</li> <li>DatumBoxAPICall</li> <li>CronJob</li> </ul>

dataPipelines/youtube		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Runs daily jobs to synchronize the database data with the Youtube platform data</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>startPipeline()</li> <li>updateVideos()</li> <li>updateVideoStatistics()</li> <li>updateComments()</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>YoutubeChannel</li> <li>YoutubeVideo</li> </ul>



		<ul style="list-style-type: none"> <li>• Youtube Comment</li> <li>• GoogleReviewsLocation</li> <li>• GoogleReviewsReview</li> <li>• DatumBoxAPICall</li> <li>• CronJob</li> </ul>
--	--	---

dataPipelines/googleReviews		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>• Runs daily jobs to synchronize the database data with the Google Reviews platform data</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>• startPipeline()</li> <li>• updateLocationReviews()</li> <li>• googleReviewsScheduledJob</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>• GoogleReviewsAccount</li> <li>• GoogleReviewsLocation</li> <li>• GoogleReviewsReview</li> <li>• DatumBoxAPICall</li> <li>• apis/googleReviews</li> <li>• CronJob</li> </ul>

## Frontend Auth Classes

Sign In		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>• Allow users to login using a username and a password</li> <li>• Users can choose to register an account if they don't have one, and they will be redirected to a sign up page</li> <li>• Users will be redirected to the home page after successful login</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>• Auth Storage</li> <li>• Notification</li> <li>• User Slice</li> </ul>

Sign Up		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>• Allow users to sign up an account</li> <li>• After sign up, the user can choose to go back to the login page</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>• N/A</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>• Auth Storage</li> <li>• Notification</li> <li>• User Slice</li> </ul>

Sign Out Button		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>• Contains a styled version of the sign out button and handles the sign out logic</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>• None</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>• Web App Slice4</li> </ul>

Is Signed In		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Checks that the user is signed in before displaying the component that is given to it as a field</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>children</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>User Slice</li> </ul>

Auth Storage		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Handles session retrieval and storage</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>None</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>None</li> </ul>

User Slice		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Redux slice that contains the user information</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>Username</li> <li>signInStatus</li> <li>signUpStatus</li> <li>signOutStatus</li> <li>errorMessage</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>None</li> </ul>

User Info		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Fetches the username from the User Slice</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>None</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>User Slice</li> </ul>

## Pages Classes

App		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Stores the top level components for the frontend</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>N/A</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Router Component</li> <li>MUI Theme</li> </ul>

Dashboard		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Provide a summary of reported statistics</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>N/A</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> None

Surveys		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Lets users create surveys to collect data from participants</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>N/A</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> None

ReviewApps		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Provides statistics gathered from review apps</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>N/A</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> None

SocialMedia		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Provides statistics gathered from social media</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>N/A</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>FacebookTab</li> <li>Youtube Tab</li> <li>Reddit Tab</li> <li>Instagram Tab</li> <li>Twitter Tab</li> </ul>

Settings		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Interface to access different tabs about different platform setups</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>N/A</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Facebook Setup Tab</li> <li>Youtube Setup Tab</li> <li>Reddit Setup Tab</li> <li>Instagram Setup Tab</li> <li>Twitter Setup Tab</li> <li>Yelp Setup Tab</li> </ul>

Facebook Tab		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Organizes all the components that will be displayed on the facebook tab page</li> </ul>		<b>Fields</b> None
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Time Series Chart</li> <li>Sentiment Analysis Chart</li> <li>Subjectivity Analysis Chart</li> <li>Card Header</li> <li>Comments Table</li> <li>Word Cloud</li> </ul>

Instagram Tab		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Organizes all the components that will be displayed on the instagram tab page</li> </ul>		<b>Fields</b> None
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Time Series Chart</li> <li>Sentiment Analysis Chart</li> <li>Subjectivity Analysis Chart</li> <li>Card Header</li> <li>Comments Table</li> <li>Word Cloud</li> </ul>

Twitter Tab		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Organizes all the components that will be displayed on the Twitter tab page</li> </ul>		<b>Fields</b> None
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Time Series Chart</li> <li>Sentiment Analysis Chart</li> <li>Subjectivity Analysis Chart</li> <li>Card Header</li> <li>Comments Table</li> <li>Word Cloud</li> </ul>

Reddit Tab		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Organizes all the components that will be displayed on the reddit tab page</li> </ul>		<b>Fields</b> None
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Time Series Chart</li> <li>Sentiment Analysis Chart</li> <li>Subjectivity Analysis Chart</li> <li>Card Header</li> <li>Comments Table</li> <li>Word Cloud</li> </ul>

YouTube Tab		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Organizes all the components that will be displayed on the youtube tab page</li> </ul>		<b>Fields</b> None
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Time Series Chart</li> <li>Sentiment Analysis Chart</li> <li>Subjectivity Analysis Chart</li> <li>Card Header</li> <li>Comments Table</li> <li>Word Cloud</li> </ul>

Facebook Setup Tab		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Interface for the user to set up their facebook connection</li> </ul>		<b>Fields</b> None
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Notification</li> <li>Facebook Setup Slice</li> </ul>

Instagram Setup Tab		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Interface for the user to set up their instagram connection</li> </ul>		<b>Fields</b> None
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Notification</li> <li>Instagram Setup Slice</li> </ul>

Twitter Setup Tab		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Interface for the user to set up their twitter connection</li> </ul>		<b>Fields</b> None
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Notification</li> <li>Twitter Setup Slice</li> </ul>

Reddit Setup Tab		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Interface for the user to set up their reddit connection</li> </ul>		<b>Fields</b> None
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Notification</li> <li>RedditSetup Slice</li> </ul>

Youtube Setup Tab		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Interface for the user to set up their youtube connection</li> </ul>		<b>Fields</b> None
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Notification</li> <li>YouTube Setup Slice</li> </ul>

Yelp Setup Tab		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Interface for the user to set up their yelp connection</li> </ul>		<b>Fields</b> None
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Notification</li> <li>Yelp Setup Slice</li> </ul>



## Navigation Classes

Protected Route		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Creates a Route and only allows navigation to that route if the user is signed in</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>None</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Auth Storage</li> </ul>

Router Component		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Creates the routes of the different pages of the platform</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>None</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Protected Route</li> <li>Header</li> </ul>

Side Nav		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Fetches the username from the User Slice</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>None</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>User Slice</li> <li>Web App Slice</li> </ul>

## Chart Classes

TimeSeries Chart		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Show sentiment analysis of posts over a time range</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>N/A</li> </ul>
<b>Parent Class</b> Dashboard	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Routes/Instagram/media</li> <li>Routes/Facebook/post</li> <li>Routes/Twitter/tweet</li> <li>Routes/Reddit/listing</li> <li>Routes/YouTube/video</li> </ul>

Pie Chart		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Show sentiment analysis of a post</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>N/A</li> </ul>
<b>Parent Class</b> Dashboard	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Routes/Instagram/comments</li> <li>Routes/Facebook/comments</li> </ul>

Comments Table		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Show comments of posts</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>N/A</li> </ul>
<b>Parent Class</b> Dashboard	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Routes/Instagram/comments</li> <li>Routes/Facebook/comments</li> <li>Routes/Twitter/comments</li> <li>Routes/Reddit/comments</li> <li>Routes/YouTube/comments</li> </ul>

Card Info		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Shows metrics that go on UI cards at the top of each page</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>Data</li> <li>isLoading</li> <li>error</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>None</li> </ul>

Cards Generator		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Receives data in a list and generates as many Card Info as necessary to display each metric in the data as a card</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>cardData</li> <li>cardDataLength</li> <li>isLoading</li> <li>error</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Card Info</li> </ul>

Card Charts		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Shows the charts in a consistently formatted manner</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>name</li> <li>children</li> <li>error</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>None</li> </ul>

Card Header		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Fetches the data for the header component of the page depending on the platform being viewed by the user and passes it to the Cards Generator</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>None</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Cards Generator</li> <li>Web App Slice</li> <li>Date Selector Slice</li> <li>Routes/Instagram/stats/cards</li> <li>Routes/Facebook/stats/cards</li> <li>Routes/Twitter/stats/cards</li> <li>Routes/Reddit/stats/cards</li> <li>Routes/YouTube/stats/cards</li> </ul>

Post Analysis		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li></li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>None</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> None

Word Cloud		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Shows most frequent words and phrases found in comments</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>None</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Routes/Instagram/comments</li> <li>Routes/Facebook/comments</li> <li>Routes/Twitter/comments</li> <li>Routes/Reddit/comments</li> <li>Routes/YouTube/comments</li> </ul>

## Reusable Classes

Error Message		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Displays the given error in a consistently formatted fashion</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>error</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>None</li> </ul>

Notification		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Renders a default template for notifications</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>state</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>NotificationState</li> </ul>

Logo		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Contains a styled version of our logo</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>None</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>None</li> </ul>

PageName		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Contains a styled version of the current page that the user is viewing</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>None</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>Web App Slice</li> </ul>

Loader		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Displays a loading icon in a consistently formatted fashion</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>None</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>None</li> </ul>

No Data		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Displays a no data message in a consistently formatted fashion</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>None</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>None</li> </ul>

Tab Panel		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>Takes a page to render as a tab page when used with React MUI to ensure each tab page has consistent styling</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>className</li> <li>Index</li> <li>value</li> <li>children</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> None

## Other

Date Selector Slice		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>• Redux slice that contains the current date range selected by the user</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>• startDate</li> <li>• endDate</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>• None</li> </ul>

Date Selector		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>• The frontend component that the user interacts with to set the date</li> </ul>		<b>Fields</b> None
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>• Data Selector Slice</li> </ul>

Web App Slice		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>• Redux slice that contains information about the general states of the application, such as which page and subpage the user is currently viewing</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>• pageName</li> <li>• appName</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>• None</li> </ul>

Facebook Setup Slice		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>• Redux slice that contains information about the facebook setup page</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>• Stage</li> <li>• pages</li> <li>• currentPage</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> None

Instagram Setup Slice		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>• Redux slice that contains information about the instagram setup page</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>• status</li> <li>• pages</li> <li>• connectedPageId</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> None

Yelp Setup Slice		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>• Redux slice that contains information about the yelp setup page</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>• status</li> <li>• business</li> <li>• newBusiness</li> <li>• searchObject</li> <li>• searchResults</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> None

Twitter Setup Slice		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>• Redux slice that contains information about the twitter setup page</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>• status</li> <li>• username</li> <li>• newUsername</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> None



YouTube Setup Slice		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>• Redux slice that contains information about the youtube setup page</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>• status</li> <li>• channel</li> <li>• newChannel</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> None

Reddit Setup Slice		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>• Redux slice that contains information about the reddit setup page</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>• status</li> <li>• subreddit</li> <li>• newSubreddit</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> None

Header		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>• Displays the header of the applications and all the classes that make up the header</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>• None</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>• Is Signed In</li> <li>• Page Name</li> <li>• Logo</li> <li>• Data Selector</li> <li>• Sign Out Button</li> <li>• User Info</li> </ul>

Mui Theme		
<b>Responsibilities</b> <ul style="list-style-type: none"> <li>• Creates the React MUI theme for the platform</li> </ul>		<b>Fields</b> <ul style="list-style-type: none"> <li>• None</li> </ul>
<b>Parent Class</b> None	<b>Sub Class</b> None	<b>Collaborations</b> <ul style="list-style-type: none"> <li>• None</li> </ul>

# System Interaction

## Client:

The frontend is designed to run on Chrome, a chromium engine based website, and Firefox.

## Local Development:

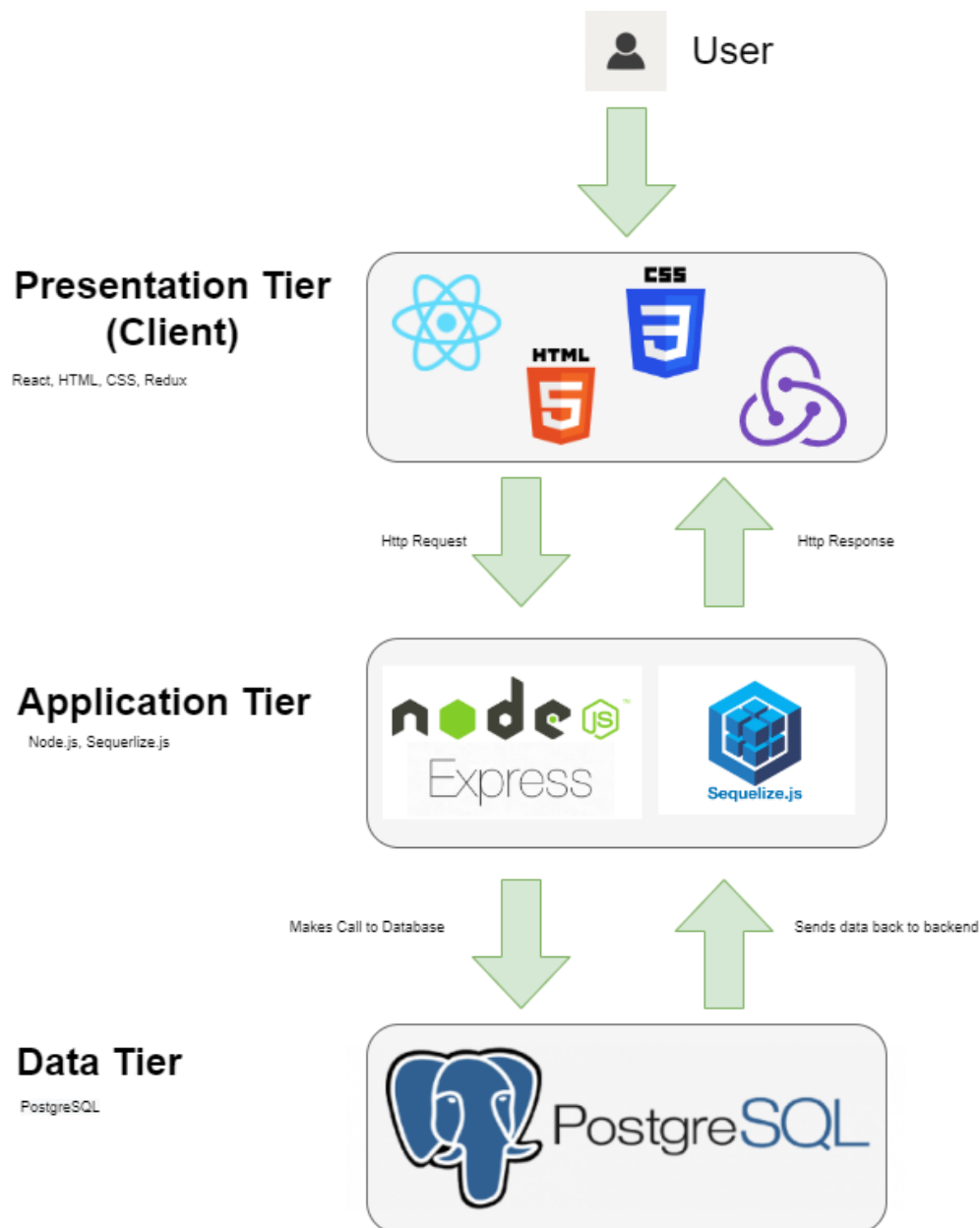
There are no assumptions based on the operating system, the README.md instructions are provided for Windows and Mac/Linux. There are no dependencies on the DB's and network configuration.

## Deployed System:

We are using docker containers to deploy each component of our application, hence inherently there are no dependencies on the environment of the virtual machine. The network will be configured in the future with CORS so that it is only allowed to communicate with the frontend that is hosted on our domain.

# System Architecture

The application utilizes the **Three-Tier Architecture**, consisting of three layers, the presentation layer, application layer and data layer. Our presentation tier uses the React framework for building graphic user interfaces. The presentation tier also consists of tools such as Redux that allow us to write clean code. The presentation tier uses the http transfer protocol to communicate to the application tier which consists of a Node/Express server. The Express server utilizes the sequelize.js library to connect to the data tier, which consists of a postgresSQL database.



Reference for this article: <https://www.ibm.com/cloud/learn/three-tier-architecture>

# System Decomposition

## Client Decomposition:

The client is using a web browser to access the website. This means that the client's browser makes a GET request to the server and fetches the frontend. The client interacts with the frontend to make API requests with the backend servers using predefined URLs. These requests are routed in EXPRESS (running on NodeJS) to the correct API endpoint and if the user request is legal, it is allowed. These endpoints use sequelize to communicate with our Postgres database.

## Local Development System Decomposition:

Assuming the developer has followed the README.md and completed the installation correctly, they should be able to simply start a frontend server that tracks their continuous changes in the code. The developer should also have a simultaneously running backend server that also tracks their continuous code changes while developing. The local frontend communicates directly with the backend through the local URL. Furthermore, the database is also running the local machine using the built-in Postgres server program. The backend communicates with the database server locally using a URL as well. In case of any errors, these will be

## Deployed System Decomposition:

The deployed system works based on docker containers running on the same virtual machine rented from digital ocean. Each of the frontend, backend and database have their own URL and associated docker container. The communication between them is done using TLS to ensure the security and integrity of communication. All servers/docker containers are running at the same time. In addition, there is another container running NGINX to act as a reverse proxy to the requests that are coming into the server and handle the TLS certificate.