

Standard Analytics of Datasets

FRONTEND APP => frontend1	1
PAGES	1
COMPONENTS	1
BACKEND APP => datasets	1
Models	1
Serializers	2
Views	2
FRONTEND API CALLS	2
BACKEND VIEWS	3

FRONTEND APP => frontend1

- PAGES

- Home
 - Home.css
 - Home.jsx
- Analysis
 - Analysis.css
 - Analysis.jsx

- COMPONENTS

- Chart
 - Chart.css
 - Chart.jsx
- FeaturedInfo
 - FeaturedInfo.css
 - FeaturedInfo.jsx

BACKEND APP => datasets

- Models

- Metadata
- Download_tracker
- Data_1
- Data_2
- Data_3
- Data_4

- Serializers

- Views

- DatasetView
- MetadataView
- GetMetadataView
- DownloadTrackerView
- UpdateMetaDataView
- AnalyticsView

FRONTEND API CALLS

S.No	Function name	Description
1	get_metadata()	Given the dataset name in the URL this function will make a GET call to the backend django app to get the metadata,model name, charts data(to display download count) and columns data of the given dataset
2	handleCitation()	This function makes POST call when the user clicks on the button in the UI if he/she has cited or not
3	handleDownload()	This function makes POST call when the user clicks on download button which increases the download count in the backed which reflect even after re-rendering the page.

BACKEND VIEWS

S.No	Class name	Description
1	GetMetaDataView	This class has two functions GET() and POST(). When a get request is sent it will extract the dataset name and with the use of metadata model as mentioned above will return the metadata of the dataset request When a post call is made this will take the object from the frontend and updates the metadata table
2	DownloadTrackerView	This class has a function for GET() which on request calculates the number of downloads per each month and send the month vs count data

3	AnalyticsView	This class has a function for GET() which on request send the analytics (per column) to the frontend
---	---------------	--