**A picture containing night sky

Description automatically generated****API BASED DATA VISUALIZER APP**

Specification Document

horizontal line

# MENTOR

# Baldeep Singh Chhabra

# Naman Goyal

# Team

# Amisha Verma

# 4 Reasons to Utilize Data Visualization SoftwareAnubhav Aryan

# Pratyush Mangal

# Udit Mimani

# 

# 

# 

# Introduction

* Data visualization is the graphical representation of information and data. By using visual elements like charts, graphs, and maps, data visualization tools provide an accessible way to see and understand trends, outliers, and patterns in data.
* In the world of Big Data, data visualization tools and technologies are essential to analyze massive amounts of information and make data-driven decisions.
* Our eyes are drawn to colors and patterns. We can quickly identify red from blue, square from circle. Our culture is visual, including everything from art and advertisements to TV and movies.
* Data visualization is another form of visual art that grabs our interest and keeps our eyes on the message. When we see a chart, we quickly see trends and outliers. If we can see something, we internalize it quickly. It’s storytelling with a purpose. If you’ve ever stared at a massive spreadsheet of data and couldn’t see a trend, you know how much more effective a visualization can be.

## 

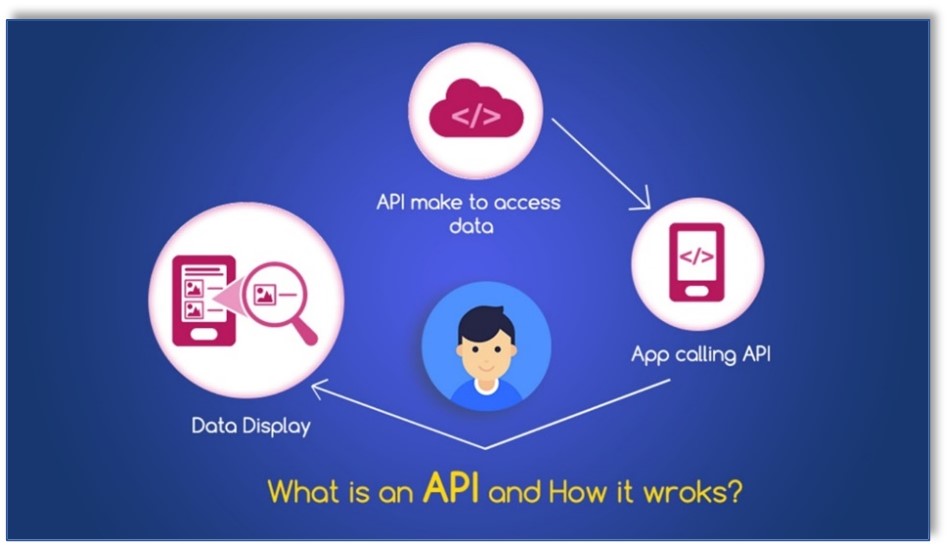
## Types of Visuals

The art of data visualization has also come a long way in the past 20+ years. Nowadays you also have the option to create more intricate visualization such as:

* Bubble clouds
* Bullet graphs
* Heat maps
* Radial trees
* And more

## 

## Application Programming Interface (API)



* An application programming interface (API) is an interface that defines interactions between multiple software applications or mixed hardware-software intermediaries.
* It defines the kinds of calls or requests that can be made, how to make them, the data formats that should be used, the conventions to follow, etc.
* It can also provide extension mechanisms so that users can extend existing functionality in various ways and to varying degrees.
* Data is typically stored in a database hosted on a physical server. To retrieve that data, you need to know how to talk to that database in order to get what you want. **That’s what an API does:** it helps by receiving requests and responding with the data needed to provide some functionality or service.

**Project Description**

We planned to make an Kaggle API based data visualizer which translates information into a visual context, and give us brief information such as:

* View upcoming contest list.
* View previous contest list.
* Analyses single user profile.
* No. of medals earned by a single user in all four categories:

1.Competitions

2.Dataset’s contribution

3. Notebook’s contribution

4. Discussion expertisation

* From contest list you can directly go to contest page

Graphical user interface, text, application, chat or text message

Description automatically generated Application

Description automatically generated with medium confidence

In the process of making API BASED VISUALIZER APP we are presenting an API BASED NEWS APP.

**News App**

****

We are going to create a simple News App on Android. Users would be able to read live news articles from The Guardian in this APP.

**About the Guardian**

The Guardian is a British daily newspaper. It was founded in 1821 as The Manchester Guardian and changed its name in 1959. Along with its sister papers The observer and The Guardian Weekly, The Guardian is part of the Guardian Media Group, owned by the Scott Trust. The trust was created in 1936 the “secure the financial and editorial independence of The Guardian in perpetuity and to safeguard the journalistic freedom and liberal values of The Guardian free from commercial or political interference.

**Steps:**

1. Collecting a data from an API

* Our first step is to collect news data through News APIs. As you can see in the above flowchart, we are displaying news title, author name, publish date, and on screen click (that will redirect you in the default browser as shown above).
* For this, we have to collect the data from the News API. And we will display this data on the UI. As you can see, we have created a list containing news articles. In this we have used the guardian API.

**url =** [**https://content.guardianapis.com/search?show-tags=contributor&api-key=test**](https://content.guardianapis.com/search?show-tags=contributor&api-key=test)

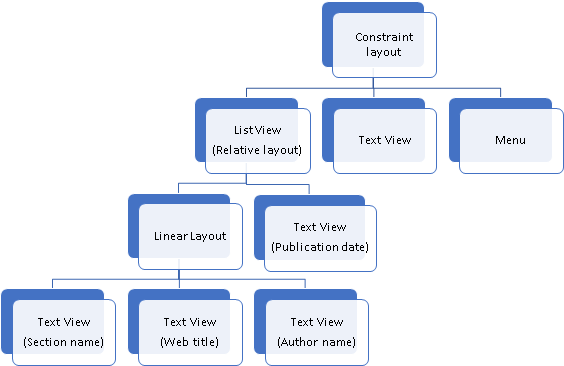
Diagram

Description automatically generated

2.Creating activity\_main.xml and MainActivity.java file

* Firstly, we need to give the look to the opening page (i.e., main activity) of an app, for this we created an activity\_main.xml file.
* Now we create a mainActivity.java file to give a functionality to the main activity of the app.
* **Activity:** This class is a crucial component of an Android app, and the way activities are launched and put together is a fundamental part of the platform's application model.

**The view hierarchy for the activity page is shown below:**



Now, we will create other activities to make our app more functionable, such as,

* What will happen when a list item clicked?
* How to give permissions to the app such as internet connectivity?
* What will happen if it will not be able to collect data?
* How to bind two or more activities?

**AndroidManifest File**

The **AndroidManifest**. **xml file** contains information about the package, including components of the application such as activities, services, broadcast receivers, content providers etc. …

It is responsible to protect the application to access any protected parts by providing the permissions. The permissions that the app needs in order to access protected parts of the system or other apps. It also declares any permissions that other apps must have if they want to access content from this app.

For example:

* <uses-permission android:name="android. permission. INTERNET" />

It gives permission to the app to access an internet.

* <uses-permission android:name="android. permission. ACCESS\_NETWORK\_STATE" />

It allows apps to access information about the network.

* <activity android:name= “. MainActivity">

<intent-filter>

<action android:name="android. intent. action. MAIN" />

<category android:name="android. intent. category. LAUNCHER" />

</intent-filter>

</activity>

It contains all the activities of an app.

**Build an Intent**

* An Intent is an object that provides runtime binding between separate components, such as two activities. The Intent represents an app’s intent to do something. We can use intents for a variety of tasks.
* In this case we are using an Implicit Intent (Implicit intent do not name a specific component, but instead declare a general action to perform, which allows a component from another app to handle it. For example, if you want to show the user a location on a map, you can use an implicit intent to request that another capable app show a specified location on a map)

public void onItemClick(AdapterView<?> adapterView, View view, int position, long l) {  
 News currentNews = mAdapter.getItem(position);  
 Uri newsUri = Uri.*parse*(currentNews.getUrl());  
 Intent websiteIntent = new Intent (Intent.*ACTION\_VIEW*, newsUri);  
 startActivity(websiteIntent);

**Further Modifications**

* The Menu button, as shown in the prototype, is not yet functional, it will redirect the user to the categorised list of different sections of the news article.
* Further we were planning to attach a few APIs to our app so as to make our app versatile and provide the user with a variety of choices to pick among the news handles.

**Challenges:**

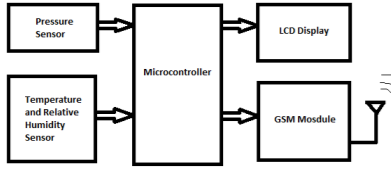
● The first challenge was to decide whether we wanted to build an android application or a website application.

● The other challenge was to figure out the language and the platform on which we were going to build the application.

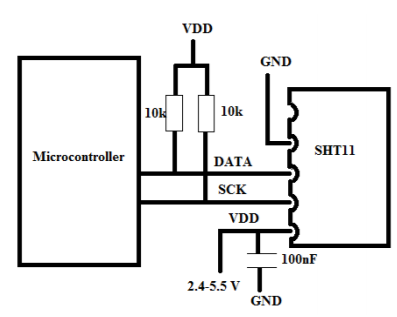
* Another challenge was to find a trustworthy source from where we could get the API.
* We did not pick web development because of the time constraint, none of us were proficient in it and learning it from the beginning was not feasible in the time available. Due to this reason we decided to make an android application as suggested by our mentor, as it was easier than web dev to learn.
* Now, we picked several API, they were Kaggle, Hackerrank, Facebook, News, Google news etc. After our research we decided to use kaggle API for data visualisation. In case of the other APIs, we were unable to fetch the data, or it required subscription.
* For creating an app, we decided to learn app development first, so we started learning it from udacity. The link is provided in the resources for the course that we chose for our work. Due to the time constraints, we were not able to make the required app, but we would like to continue the work ahead.
* Due to time constraints, we paused the idea of making a kaggle API. As it needed a bit of python experience, but since we were working on java it would be another challenge for us to learn python. Considering all of them we picked the News API. This was considered because it was easily available and the source, we used to be The Guardian.

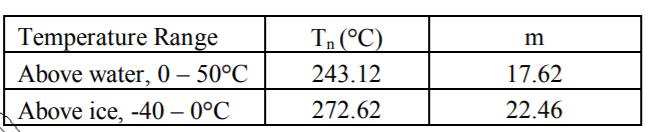
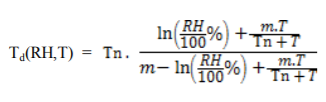
# Previous Project:

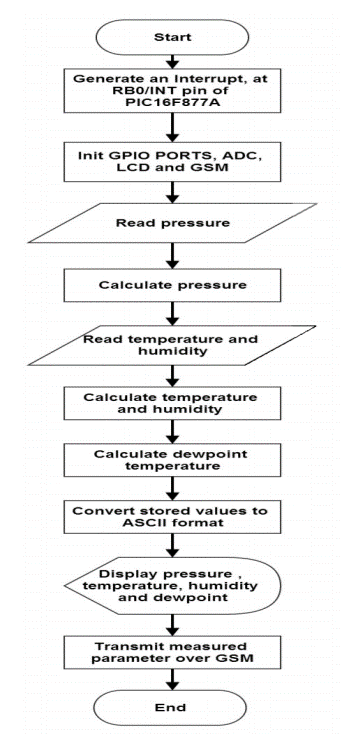
* **Weather monitoring device, transmitting section-** 
  + We basically receive dummy values from pressure sensors and temperature and relative humidity sensor through Microcontrollers.
  + GSM module is then used to send the data received by microcontrollers to GSM network.
  + LCD displays are then used to display the original temperature, pressure etc.



* + **SHT11 connection with microcontroller-**



* + **Dew Point-** Sensor not measuring dew point directly, however dew point can be derived from humidity and temperature readings. Since humidity and temperature are both measured on the same monolithic chip, the sensor allows superb dew point measurements in Degree Celsius (°C). Wikipedia indicates that Relative humidity of 100% indicates the dew point is equal to the current temperature and that the air is maximally saturated with water. However, dew point is calculated through formula relating temperature and humidity, given by: 
  + **Devices used:**
    - **Microcontroller:** It is the heart of the whole system. It uses sensors for taking input. It receives signals (analog and digital signals) equivalent to the quantities of the weather variables to be measured. From sensors connected to it and conversion and processing through pre- programmed instructions written in C language to ensure that corresponding measurements made by these sensors are available in forms that are meaningful and useful for human analysis, interpretation and record.
    - **Sensors:** They are used to take measurements and pass it to microcontrollers.
    - **LCD Units:** LCD display is capable of displaying different characters and symbols. It is used to display the measured parameters such as pressure, temperature, relative humidity and dew point temperature.
    - **GSM Module** is then used to send the data received by microcontrollers to GSM network.
* Receiver Station: -
  + Flow chart-



# Resources:

* Android development courses
  + User Interface -<https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqa3lUY2ZwUDB6eGJuOGNuLTVUZUhKc2dGRG9aQXxBQ3Jtc0tsSk5OWHJSUW95M0Q4ZzJhQzBlSDJjeDgtVjcwUkhxeHM0S2ZybWw1TEo3cmdPcUZQOVBQRDF3YjByS1hoODZFSUJfc1RZdWdyRDdacFFJZ1ZITVB4blp2dTM0OVFVZURjVW92cXRyb2NycFZVcktabw&q=https%3A%2F%2Fwww.udacity.com%2Fcourse%2Fandroid-basics-user-interface--ud834>
  + User Input -<https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbl80TEVNeE1IeWJuYVItajlZUnVMNEh3YldtUXxBQ3Jtc0trMnFpVUd0dkdmRFI3elpCOU9FUjhqRnk5WC0yNzRQZGFfNHV0dXFGTk1yRFJKai1URHN1SFFMVS1WTEFKQkRBRlJ2VlFWYXRSRUZMeU5ZMVRSbkhyME5Wdlh1RFJXV29ieEFfa3UwZ0E5MHhlbkNPSQ&q=https%3A%2F%2Fwww.udacity.com%2Fcourse%2Fandroid-basics-user-input--ud836>
  + Multiscreen Apps - <https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbUd3Nk83Y1IxSFRDSVYzLVB4ZTdIY09kZ3U4UXxBQ3Jtc0tuSVVfMUltRHh5YU1mWlV5VUQ2SGh6ZWZvT2tpeDl4OVFDWDlBVDVyR0tUS09Ld0UzZmtCc2t5Y0E0dS1DbU1kRUpNZXZqZ0dGZnFQX2RkNmMtR001MDJTcjdyRmFPdGVhRVBhQmFaTHQzX3cxRHFVYw&q=https%3A%2F%2Fwww.udacity.com%2Fcourse%2Fandroid-basics-multiscreen-apps--ud839>
  + Networking course -<https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbHl2S0h1OEJmQkdwdnhXd0x4YkxGUklCaUFIQXxBQ3Jtc0tscW5TSFFRSnNxWnRmTDRHZWVzM0NxWVlGTV9TYUNGMWlMalR4Zy01R1A2VWhpR3Y1bTRtYkJlUFZCV1J0SXZDWXlTWUM0YlFLVnJmMGRXbElzSTE2MXJZZ1B4RF9pSkd4X0NlbHYwYUlnRTlaeEl3aw&q=https%3A%2F%2Fwww.udacity.com%2Fcourse%2Fandroid-basics-networking--ud843>
* Kaggle API: <https://github.com/Kaggle/kaggle-api>
* News APIs: <https://rapidapi.com/blog/rapidapi-featured-news-apis/?utm_source=google&utm_medium=cpc&utm_campaign=Beta&utm_term=%2Bapi%20%2Bnews_b&gclid=Cj0KCQjwytOEBhD5ARIsANnRjVgXQrWGtmG0kJkPB1uaZOPg3lHgUNRrV11NI0jvtW8P7iSKz0d_CyMaAou3EALw_wcB>
* Android Developer site: <https://developer.android.com/docs>
* Stack overflow: <https://stackoverflow.com/>
* Java Tutorials: <https://www.tutorialspoint.com/java/index.htm>