

# DATA VISUALIZATION TOOL

**A**

## **MINOR PROJECT-I REPORT**

Submitted in partial fulfillment of the requirements

for the degree of

**BACHELOR OF TECHNOLOGY**

in

**COMPUTER SCIENCE & ENGINEERING**

By

**GROUP NO. 49**

**NAYAN SHRIVASTAVA**

**0187CS181091**

**PURVA JOSHI**

**0187CS181108**

**JUHI OJHA**

**0187CS181065**

**AASIM AKHTAR**

**0187CS181002**

Under the guidance of

**Rahul Dubey**

Professor



**Dec-2020**

**Department of COMPUTER SCIENCE & ENGINEERING**

**Sagar Institute of Science & Technology (SISTec)**

**Bhopal (M.P.)**

**Approved by AICTE, New Delhi & Govt. of M.P.**

**Affiliated to Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal (M.P.)**

*Sagar Institute of Science & Technology (SISTec), Bhopal*  
*Department of COMPUTER SCIENCE & ENGINEERING*  
*Bhopal (M.P.)*



*Dec-2020*

**CERTIFICATE**

I hereby certify that the work which is being presented in the B.Tech. Minor Project-I Report entitled **Data Visualization Tool**, in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science & Engineering** and submitted to the Department of Computer Science & Engineering, *Sagar Institute of Science & Technology (SISTec)*, Bhopal (M.P.) is an authentic record of my own work carried out during the period from Jul-2020 to Dec-2020 under the supervision of **Prof. Rahul Dubey**.

The content presented in this project has not been submitted by me for the award of any other degree elsewhere.

<i>NAME</i>	<i>ENROLLMENT NO.</i>	<i>SIGNATURE</i>
<i>Nayan Shrivastava</i>	<i>0187CS181091</i>	
<i>Purva Joshi</i>	<i>0187CS181108</i>	
<i>Juhi Ojha</i>	<i>0187CS181065</i>	
<i>Aasim Akhtar</i>	<i>0187CS181002</i>	

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

***Date:***

***Project Guide***

***HOD***

***Principal***

# **ABSTRACT**

Everything looks good when visualized. It is difficult to derive conclusions from huge chunks of data without proper representation. With the emerging technology it is easy to find tools on online platform with some specialization but dealing at small level needs a tool which is compatible and user friendly and not complicated with either big data or just few inputs to provide user with same output. A big need of data-driven decisions has now become a necessity by using visual elements like charts and graphs to provide an accessible way to see and understand trends, outliers and patterns in data without looking much into code and technology behind the automatically generated output. A web application with a fairly, simple, easy-to-use and attractive user interface which can be accessed through any device that supports Internet browsing. This application will provide support for multiple file formats such as Comma-separated values(CSV) and also adds an advantage of session so that user need not to upload file again and again. Delivering an output visual which can be downloaded in PDF or can be saved as JPG format on the device. Discovering emerging trends- DataVTool discover trends both in the business and in the market- can give businesses and edge over the competition, and ultimately affect the bottom line. Even extensive amount of complicated data starts making sense when presented graphically; businesses can recognize parameters that are highly correlated.

## **ACKNOWLEDGEMENT**

We take an opportunity to acknowledge and extend our heartfelt gratitude to our guide and the pivot of this project, **Prof. Rahul Dubey** who is most responsible for helping us to complete this work. He showed us different ways to approach the problems and the need to be persistent to accomplish the goal. His discernment in the choice of topic, his confidence when we doubted ourselves and his admirable guidance are some cogent reasons that make us over that without his support this project would be a chimera. Our deepest sense of gratitude towards, **Prof. Rajesh Rai** for guiding us to bring this project to its successful completion. We are highly indebted to him for encouragement, inspiration and guidance.

We are also thankful to **Prof. Ujjwal Nigam, Head of Department of Computer Science & Engineering** for cooperation and support to complete this work. We would also like to express thanks to **Dr. Keshavendra Choudhary(Principal, SISTec)** Gandhi Nagar Bhopal for their continuous support. Thanks are due to all the staff members and lab staff of Department of Computer Science and Engineering SISTec for providing all help and support.

## **TABLE OF CONTENTS**

<b>TITLE</b>	<b>PAGE NO.</b>
Abstract	iii
Acknowledgement	iv
List of tables	vi
List of figures	vii
List of abbreviations	viii
Chapter 1    Introduction	1-3
1.1    About Project	1-2
1.2    Project Objectives	2-3
Chapter 2    Software & Hardware Requirements	4-5
Chapter 3    Problem Description	6-7
Chapter 4    Literature Survey	8-10
Chapter 5    Software Requirements Specification	11-13
5.1    Functional Requirements	13
5.2    Non-Functional Requirements	13
Chapter 6    Software Design	14-16
6.1    ER Diagram	15-16
6.2    Table Structure	
Chapter 7    Output Screens	17-22
Chapter 8    Deployment	23-24
References	25-26
Project Summary	27-30
Appendix-1: Glossary of Terms	31-32

**LIST OF TABLES**

<b>TABLE NO.</b>	<b>TITLE OF TABLE</b>	<b>PAGE NO.</b>
1.1		
1.2		
2.1		
2.2		
2.3		
4.1		

## **LIST OF FIGURES**

<b>FIG. NO.</b>	<b>TITLE</b>	<b>PAGE NO.</b>
1.1	Basic Diagram	3
4.1	Flow Diagram of Data vis.	9
5.1	Incremental Model	11
7.1	Home Page	17
7.2	Upload Section	18
7.3	Graphs	19
7.4	Sample Graph	20
7.5	Beautiful Visuals	21

### **LIST OF ABBREVIATIONS**

<b>ACRONYM</b>	<b>FULL FORM</b>
CSS	Cascading Style Sheets
CSV	Comma Separated Values
HTML	Hyper Text Markup Language
VIZ	Visualization
GUI	Graphical User Interface



# Chapter 1

# Introduction

# CHAPTER 1

## INTRODUCTION

### 1.1 ABOUT PROJECT

We have developed a platform where non-specialized people can create beautiful and comprehensive graphs. This platform provides variety of graphs and plots for its users which are simple to design. We have added a lot of customizations like hue, saturation and color palettes that the user can choose from given options. This application is Open-source for developers to contribute and collaborate. It also helps in getting familiar with current trends of visualization using graphs and plots which is an emerging technology nowadays using graphs and plots.

### 1.2 PROJECT OBJECTIVES

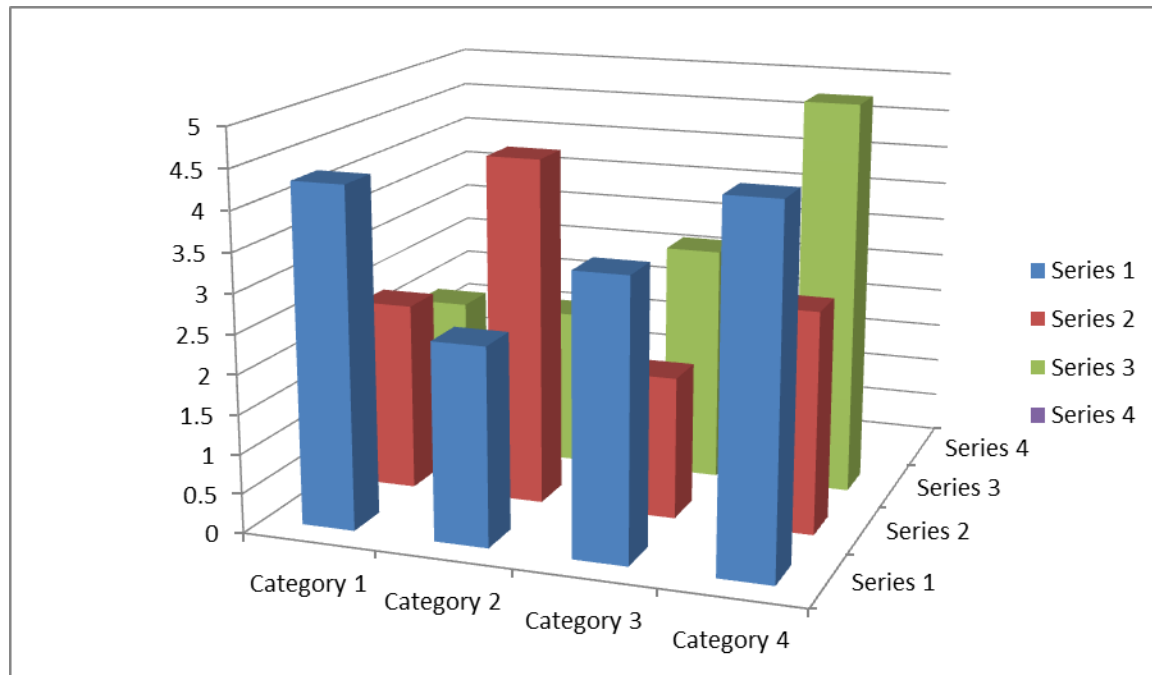
The project aim to deliver data in form of graphs and plots using Seaborn library. An easy to use and user-friendly GUI tool for data visualization. It is easy to create graphs with few and required inputs with delivering output in graphs without looking much about graphs or code. This application gives user an added advantage by using the same data for different graphs and also the session for the same remains unchanged until the user is working on our platform & gets logged out. User can easily access our platform from any device which supports Internet browsing.

#### 1.2.1 CHARACTERISTICS OF DATAVTOOL

- Minimalistic Interface
- Media visibility
- Easy to learn

#### 1.2.2 ADVANTAGES

- Open-source & Free of cost
- User Friendly
- Deployable without much knowledge



**Figure 1.1: Basic Diagram**

# Chapter 2

## Software and Hardware Requirements

## CHAPTER 2

# **SOFTWARE & HARDWARE REQUIREMENTS**

### **2.1 SOFTWARE REQUIREMENT**

- Frontend : HTML 5, CSS
- Technology : Python
- Library : Seaborn 0.11.1
- Database : Firebase
- Backend : Django, Panda, Numpy

### **2.2 HARDWARE REQUIREMENT**

- Processor : Minimum 1 GHz; Recommended 2GHz or more
- Hard Drive : Minimum 32 GB; Recommended 64 GB or more
- Memory : Minimum 1 GB; Recommended 4 GB or above
- Ethernet connection (LAN) OR a wireless adapter (Wi-Fi)

# **Chapter 3**

## **Problem Description**

## CHAPTER 3

### **PROBLEM DESCRIPTION**

As the “age of Big Data” kicks into high-gear, visualization is an increasingly key tool to make sense of the trillions of rows of data generated every day. Data visualization helps to tell stories by curating data into a form easier to understand, highlighting the trends and outliers. A good visualization tells a story, removing the noise from data and highlighting the useful information.

However, it’s not simply as easy as just dressing up a graph to make it look better or slapping on the “info” part of an infographic. Effective data visualization is a delicate balancing act between form and function. The plainest graph could be too boring to catch any notice or it make tell a powerful point; the most stunning visualization could utterly fail at conveying the right message or it could speak volumes. The data and the visuals need to work together, and there’s an art to combining great analysis with great storytelling.

# **Chapter 4**

## **Literature Survey**



## **CHAPTER 4**

# **LITERATURE SURVEY**

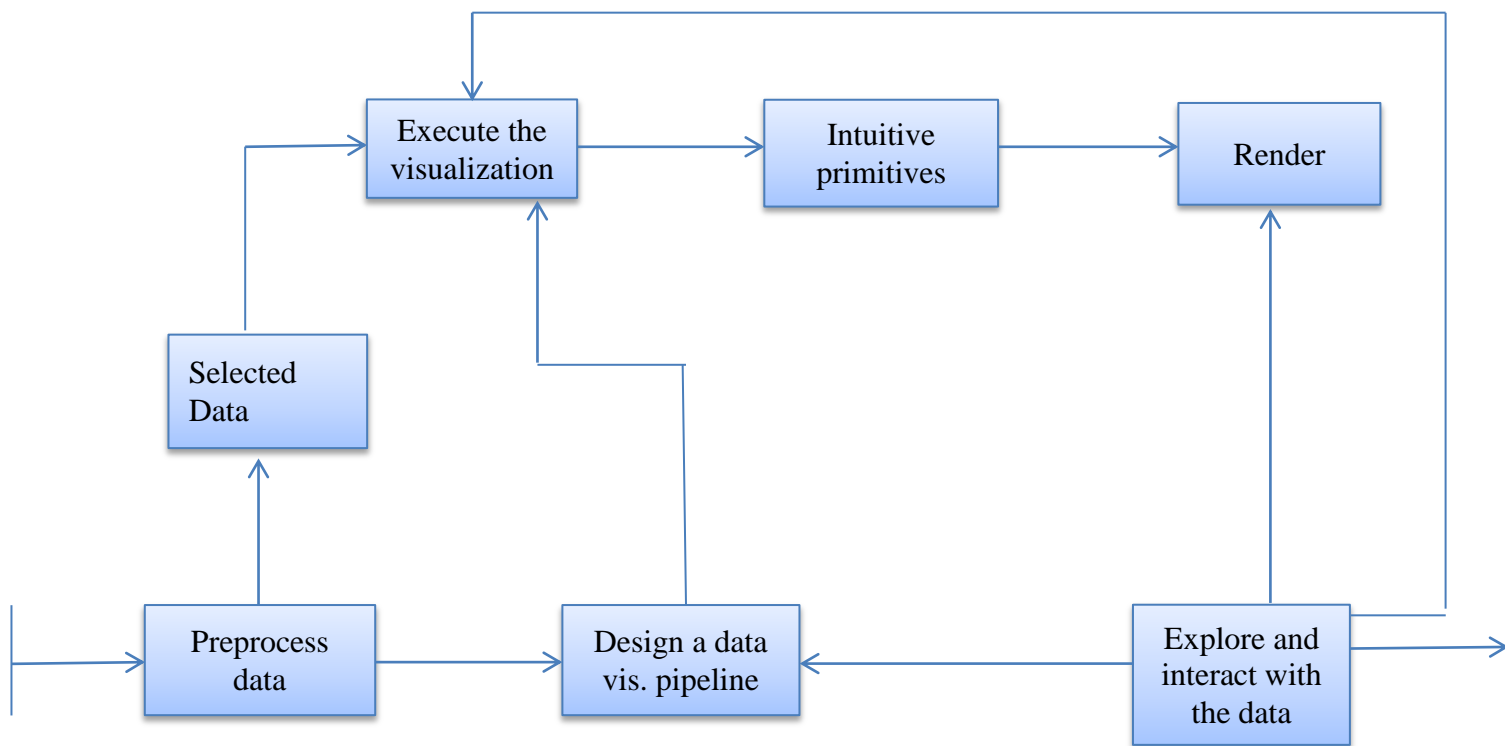
Data has been a big topic for a few years now, and it's only going to grow bigger as we get our hands on more sophisticated forms of technology and new applications in which to use them. The problem now is beginning to shift; originally, tech developers and researchers were all about gathering greater quantities of data. Now, with all this data in tow, consumers and developers are both eager for new ways to condense, interpret, and take action on this data.

One of the newest and most talked-about methods for this is data visualization, a system of reducing or illustrating data in simplified, visual ways. The buzz around data visualization is strong and growing, but is the trend all it's cracked up to be?

### **4.1 The Need for Data Visualization**

There's no question that data visualization can be a good thing, and it's already helped thousands of marketers and analysts do their jobs more efficiently. Human abilities for pattern recognition tend to revolve around sensory inputs—for obvious reasons. We're hard-wired to recognize visual patterns at a glance, but not to crunch complex numbers and associate those numbers with abstract concepts. Accordingly, representing complex numbers as integrated visual patterns would allow us to tap into our natural analytic abilities.

Seaborn allows us to make attractive and informative statistical graphics. Although matplotlib makes it possible to visualize essentially anything, it is often difficult and tedious to make the plots visually attractive. At real time working we visually analyze relationships between numerical variables with scatter plots, joint plots. However, current visualization tools and visual analytical systems fall short of providing a seamless user experience and several improvements could be made to current commercially available visualization tools. By conducting a systematic literature review, requirements of visualization tools were identified and categorized into six groups: dimensionality reduction, data reduction, scalability and readability, interactivity, fast retrieval of results, and user assistance. The most common themes found in the literature were dimensionality reduction and interactive data exploration.



**Figure 4.1 Flow Diagram of Data vis.**

# **Chapter 5**

# **Software Requirements**

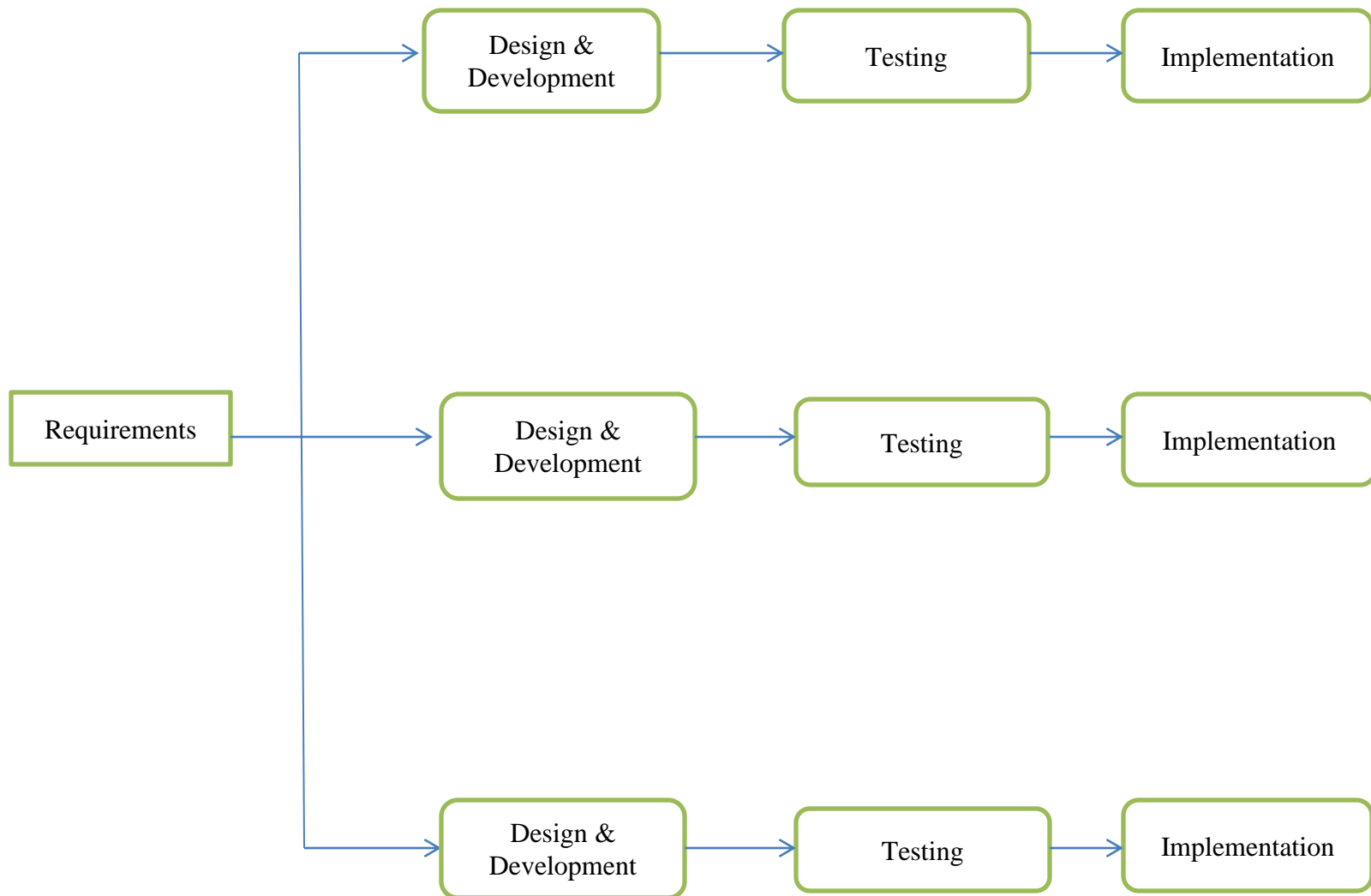
# **Specification**

## CHAPTER 5

# SOFTWARE REQUIREMENTS SPECIFICATION

### 5.1 INCREMENTAL MODEL

Incremental Model is a process of software development where requirements divided into multiple standalone modules of the software development cycle. In this model, each module goes through the requirements, design, implementation and testing phases. Every subsequent release of the module adds function to the previous release. The process continues until the complete system achieved.



**Fig 5.1 : Incremental Model**

Phase 1: Requirement analysis

Phase 2: Design & Development

Phase 3: Testing

Phase 4: Implementation

### **5.1 FUNCTIONAL REQUIREMENTS**

- User must Input in the Formatting Options.
- User must upload the file in .csv.
- While selecting “create” user should upload file for any graph/plots.

### **5.2 NON-FUNCTIONAL REQUIREMENTS**

- Portability
- Reliability
- Performance
- Reusability
- Flexibility

# **Chapter 6**

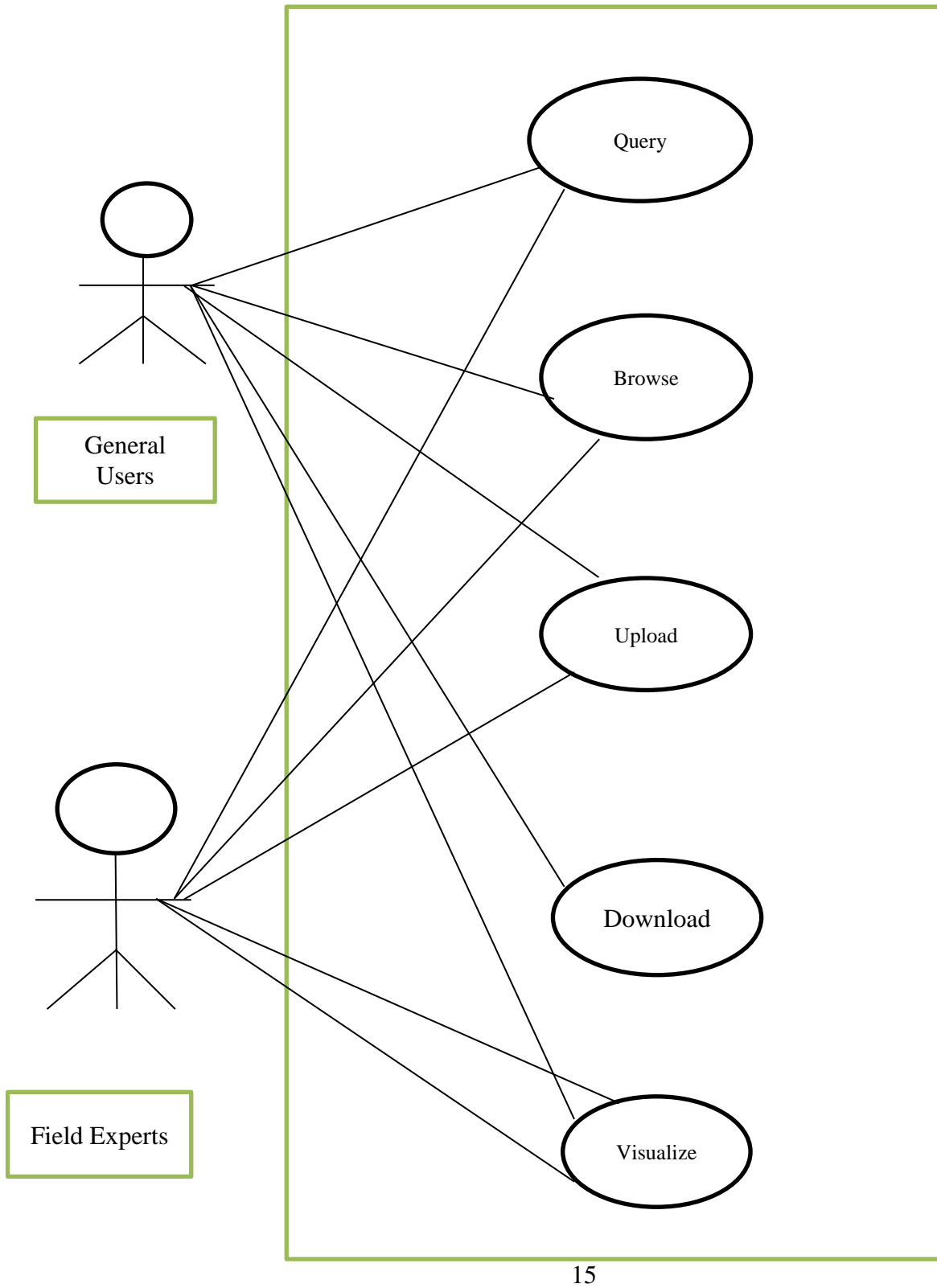
# **Software Design**

# CHAPTER 6

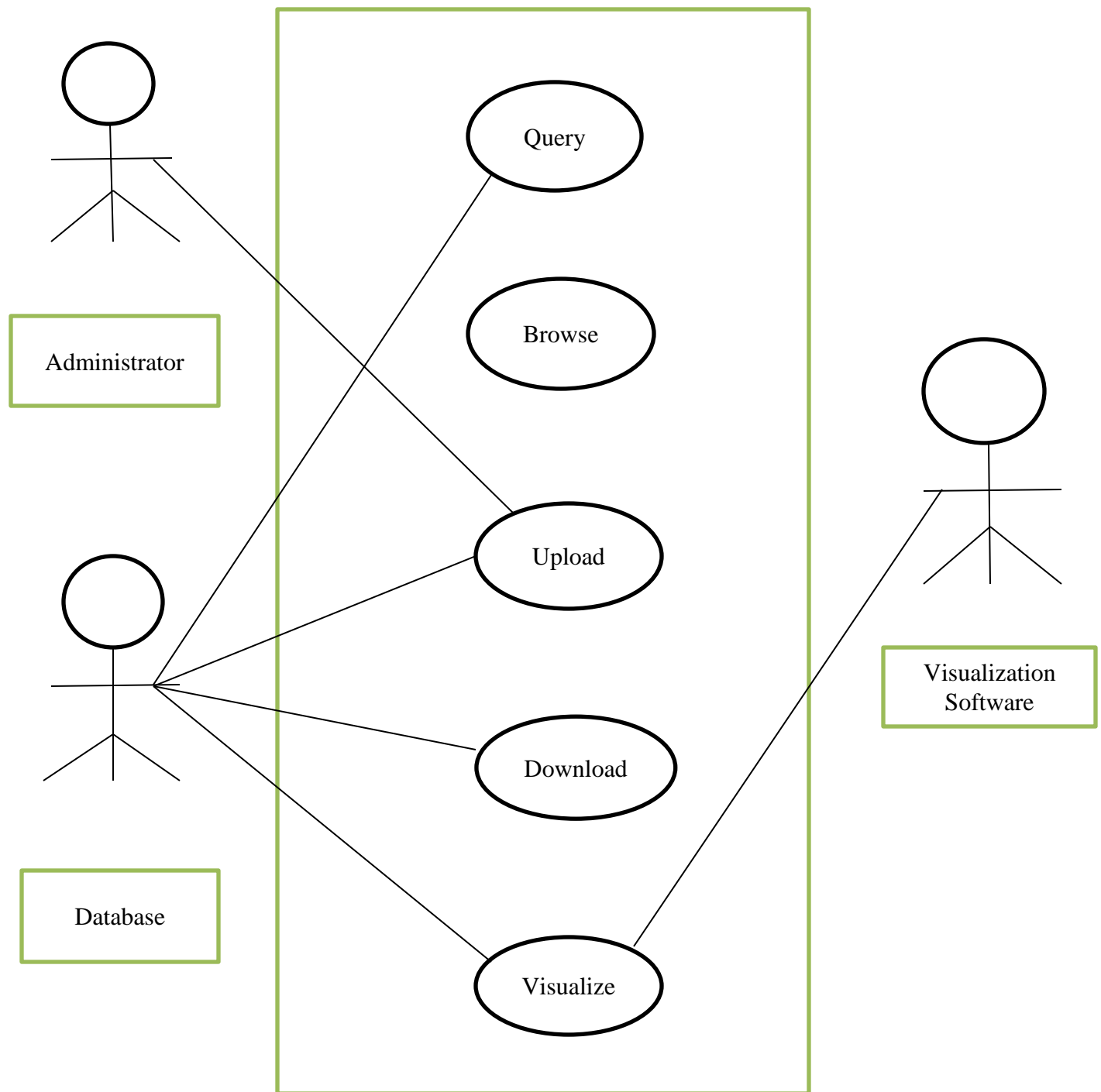
## SOFTWARE DESIGN

### 6.1 USE CASE DIAGRAM

#### 6.1.1 USE CASE DIAGRAM FOR SYSTEM



### 6.1.2 USE CASE DIAGRAM FOR ADMIN





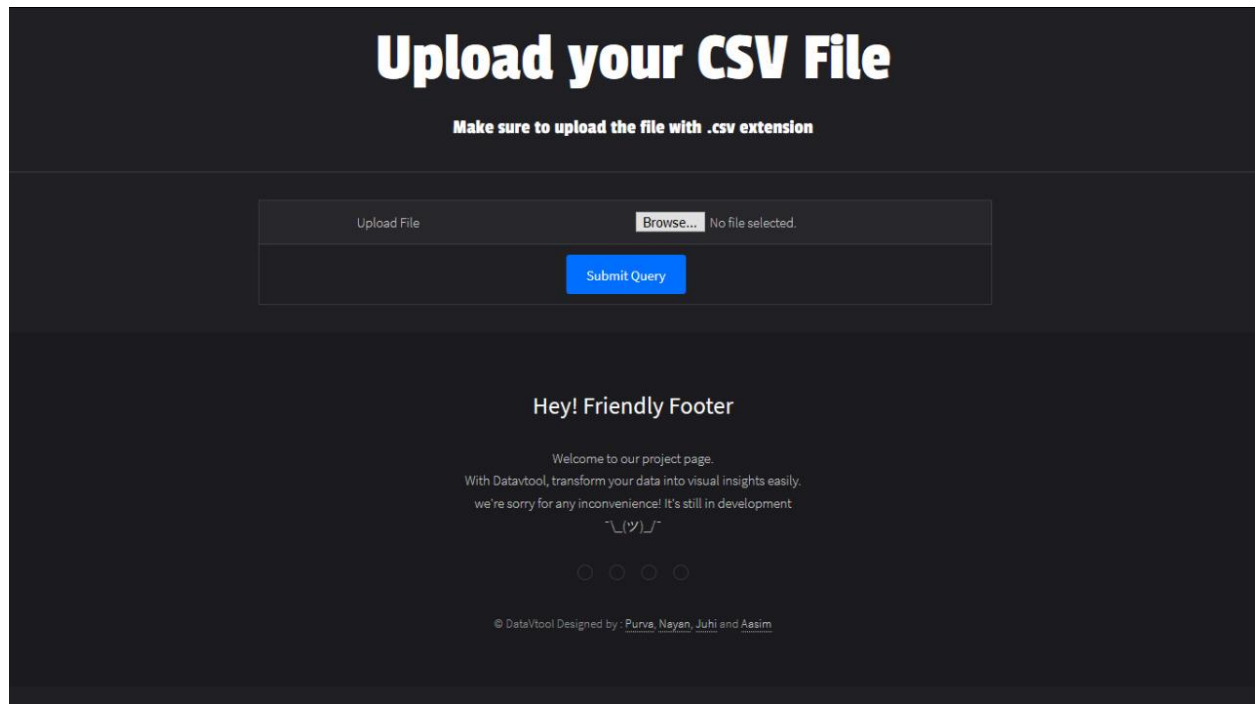
# **Chapter 7**

## **Output Screens**



**Figure 7.1 : Home Page**

This figure shows the Home Page, where user needs to upload file in .csv



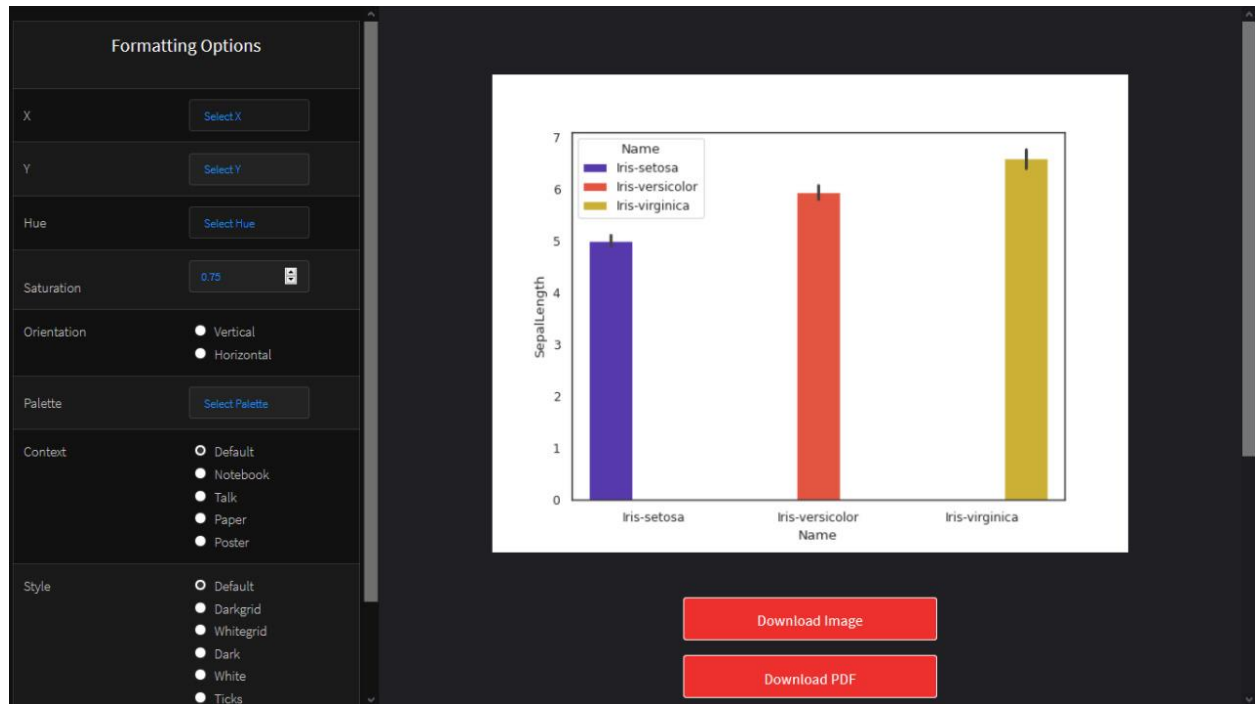
**Figure 7.2 : Upload Section**

Here user will get the option for uploading the file, in case not uploaded.



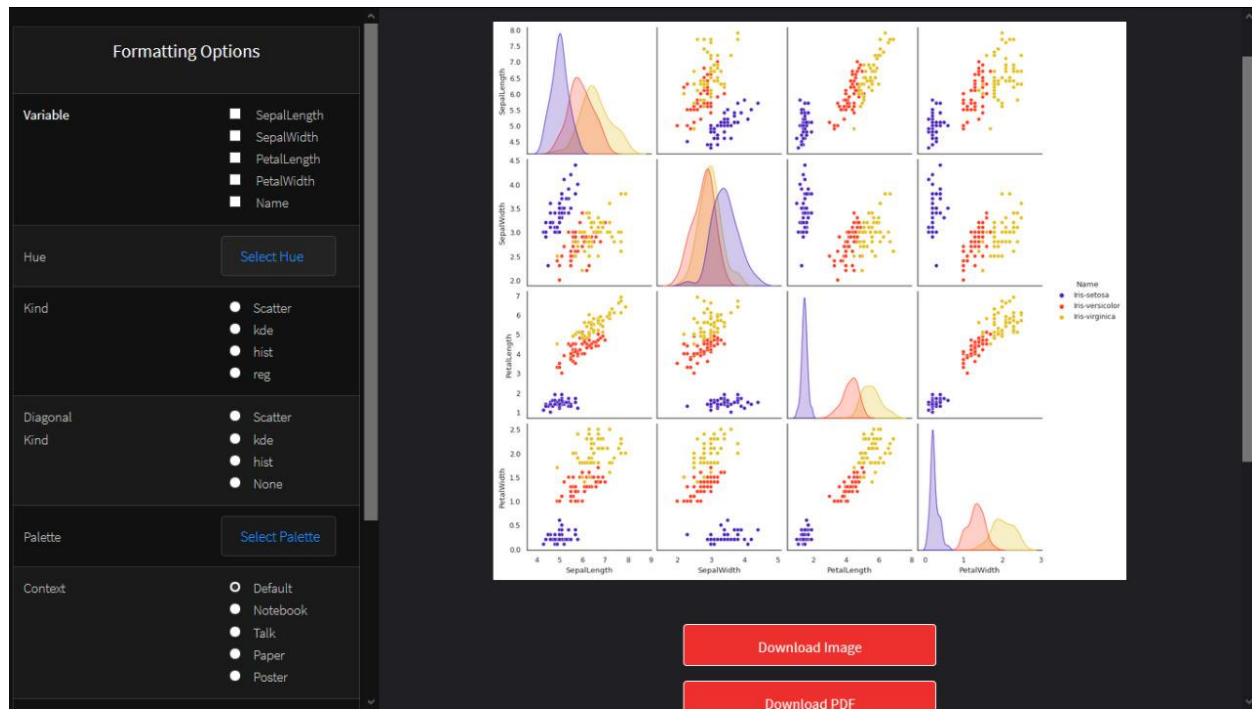
**Figure 7.3 : Graphs**

This page will provide user various options of graph after user uploads the csv file.



**Figure 7.5 : Sample Graph**

Now the user is able to Download the file in respective local system in form of PDF or JPEG.



**Figure 7.5 : Beautiful Visuals**

Another sample where user can customize graphs accordingly.

# **Chapter 8**

# **Deployment**

The process of deploying an application to pythonanywhere seems to be the most user friendly way possible. Below is a list of steps initially undertaken to deploy the application to pythonanywhere.

1. Compress project folder.
2. Upload project folder to pythonanywhere.
3. Open bash console and unzip compressed project folder.
4. Create virtual environment and install all the dependencies of the project as specified in requirements.txt.
5. Change paths and references for accessing files to actual path in pythonanywhere.
6. Modify WSGI file on pythonanywhere by changing app to the application name.



# References

# REFERENCES

---

## WEBSITES

1. <https://www.w3schools.com/>
2. <https://seaborn.pydata.org/>
3. <https://stackoverflow.com/>
4. <https://www.geeksforgeeks.org/>
5. <https://tutorial.djangogirls.org/en/deploy/>

## **PROJECT SUMMARY**

### ***About Project***

<b>Title of the project</b>	Data Visualization Tool
<b>Semester</b>	5
<b>Members</b>	4
<b>Team Leader</b>	Purva Joshi
<b>Describe role of every member in the project</b>	Backend & Connectivity handled by Purva Joshi and Nayan Shrivastava. Frontend part, Documentation & PPT work done by Juhi Ojha and Aasim Akhtar.
<b>What is the motivation for selecting this project?</b>	Because of the way the human brain processes information, using charts or graphs to visualize large amounts of complex data is easier than poring over spreadsheets or reports. Data visualization can also: Identify areas that need attention or improvement.
<b>Project Type</b> (Desktop Application, Web Application, Mobile App, Web)	Web Application

### ***Tools & Technologies***

<b>Programming language used</b>	Python
<b>Compiler used</b> (with version)	Python 3.8
<b>IDE used</b> (with version)	VS Code 1.2, Sublime Text 3
<b>Front End Technologies</b> (with version, wherever Applicable)	HTML5, CSS
<b>Back End Technologies</b> (with version, wherever applicable)	Django 3.1.4
<b>Database used</b> (with version)	-

### ***Software Design & Coding***

<b>Is prototype of the software developed?</b>	Yes
--	-----

<b>SDLC model followed</b> (Waterfall, Agile, Spiral etc.)	Incremental
<b>Why above SDLC model is followed?</b>	Generates working software quickly and early during the software life cycle.
<b>Justify that the SDLC model mentioned above is followed in the project.</b>	Throughout the project code has been analysed, designed & mandatory changes were made in the code and tested to find out the bugs.
<b>Software Design approach followed</b> (Functional or Object Oriented)	Functional
<b>Name the diagrams developed</b> (according to the Design approach followed)	In the analysis phase a rough diagram was made to get through the requirements and during the other phases all the team members were engaged to take care of the steps been followed.
<b>In case Object Oriented approach is followed, which of the OOPS principles are covered in design?</b>	-
<b>No. of Tiers</b> (example 3-tier)	-
<b>Total no. of front end pages</b>	11
<b>Total no. of tables in database</b>	-
<b>Database is in which Normal Form?</b>	-
<b>Are the entries in database encrypted?</b>	-
<b>Front end validations applied</b> (Yes / No)	Yes
<b>Session management done</b> (in case of web applications)	Yes
<b>Is application browser compatible</b> (in case of web applications)	Yes
<b>Exception handling done</b> (Yes / No)	Yes
<b>Commenting done in code</b> (Yes / No)	Yes
<b>Naming convention followed</b> (Yes / No)	Yes

<b>What difficulties faced during deployment of project?</b>	Bugs during deployment and had to change filepath according to the server
<b>Total no. of Use-cases</b>	2
<b>Give titles of Use-cases</b>	<ol style="list-style-type: none"> <li>1. Use case Diagram for User</li> <li>2. Use case Diagram for Admin</li> </ol>

### **Project Requirements**

<b>MVC architecture followed</b> (Yes / No)	<b>No</b>
<b>If yes, write the name of MVC architecture followed</b> (MVC-1, MVC-2)	-
<b>Design Pattern used</b> (Yes / No)	<b>No</b>
<b>If yes, write the name of Design Pattern used</b>	-
<b>Interface type</b> (CLI / GUI)	<b>GUI</b>
<b>No. of Actors</b>	<b>User</b>
<b>Name of Actors</b>	-
<b>Total no. of Functional Requirements</b>	<b>3</b>
<b>List few important non-Functional Requirements</b>	<b>Portability, Flexibility</b>

### **Testing**

<b>Which testing is performed?</b> (Manual or Automation)	Manual
<b>Is Beta testing done for this project?</b>	No

**Write project narrative covering above mentioned points**

Data Visualization Tool is an web application developed by our team consisting of 4 members using Incremental model which is easy to follow and our tool provides user compatible, customized and friendly environment to go through graphs which can be seen when user uploads a file in .csv format. There were some difficulties while going through all the bugs when some new functionalities were added. Django framework can be implemented at the backend to get customizable options and for frontend CSS is compatible. To modify and taking this tool at another level we will we using database and some more framework like Ajax, JavaScript.

Nayan Shrivastava  
0187CS181091

Purva Joshi  
0187CS181108

Juhi Ojha  
0187CS181065

Aasim Akhtar  
0187CS181002

Guide Signature

Prof. Rahul Dubey

## APPENDIX-1

## GLOSSARY OF TERMS

---

### C

#### CSS

Cascading Style Sheets (CSS) is used to format the layout of a webpage. With CSS, you can control the color, font, the size of text, the spacing between elements, how elements are positioned and laid out, what background images or background colors are to be used, different displays for different devices and screen sizes, and much more!.

### D

#### Django

Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of Web development, so you can focus on writing your app without needing to reinvent the wheel. It's free and open source.

### G

#### GUI

The graphical user interface is a type of user interface that allows users to interact with electronic devices through graphical icons and visual indicators such as secondary notation, instead of text-based user interfaces, typed command labels or text navigation. GUIs were introduced in reaction to the perceived steep learning curve of command-line interfaces (CLIs), which

require commands to be typed on a computer keyboard.

## S

**SEABORN** Seaborn is a library for making statistical graphics in Python. It builds on top of matplotlib and integrates closely with Pandas Data Structures. Seaborn helps you explore and understand your data. Its plotting functions operate on dataframes and arrays containing whole datasets and internally perform the necessary semantic mapping and statistical aggregation to produce informative plots. Its dataset-oriented, declarative API lets you focus on what the different elements of your plots mean, rather than on the details of how to draw them.