

CELL TOWER PLACEMENT SYSTEM

Presented By: Joy Abdelmasseh, Mark Nabil,
Mark Samir, Nermeen Haggag

Supervisor: Dr.Mohamed El-Gzzar
Teacher Assistant: Eng. Haytham Metawie

OUTLINE

Introduction

Objective

Overview Diagram

Block Diagram

Functional Requirements

Non-functional Requirements

Use-case Diagram

Class Diagram

UI

Accuracy

Dataset Sample

Demo

INTRODUCTION (1/2)

The main purpose of Cell Tower Placement System. Is to find the finest location for deploying a new cell tower.

02

INTRODUCTION (2/2)

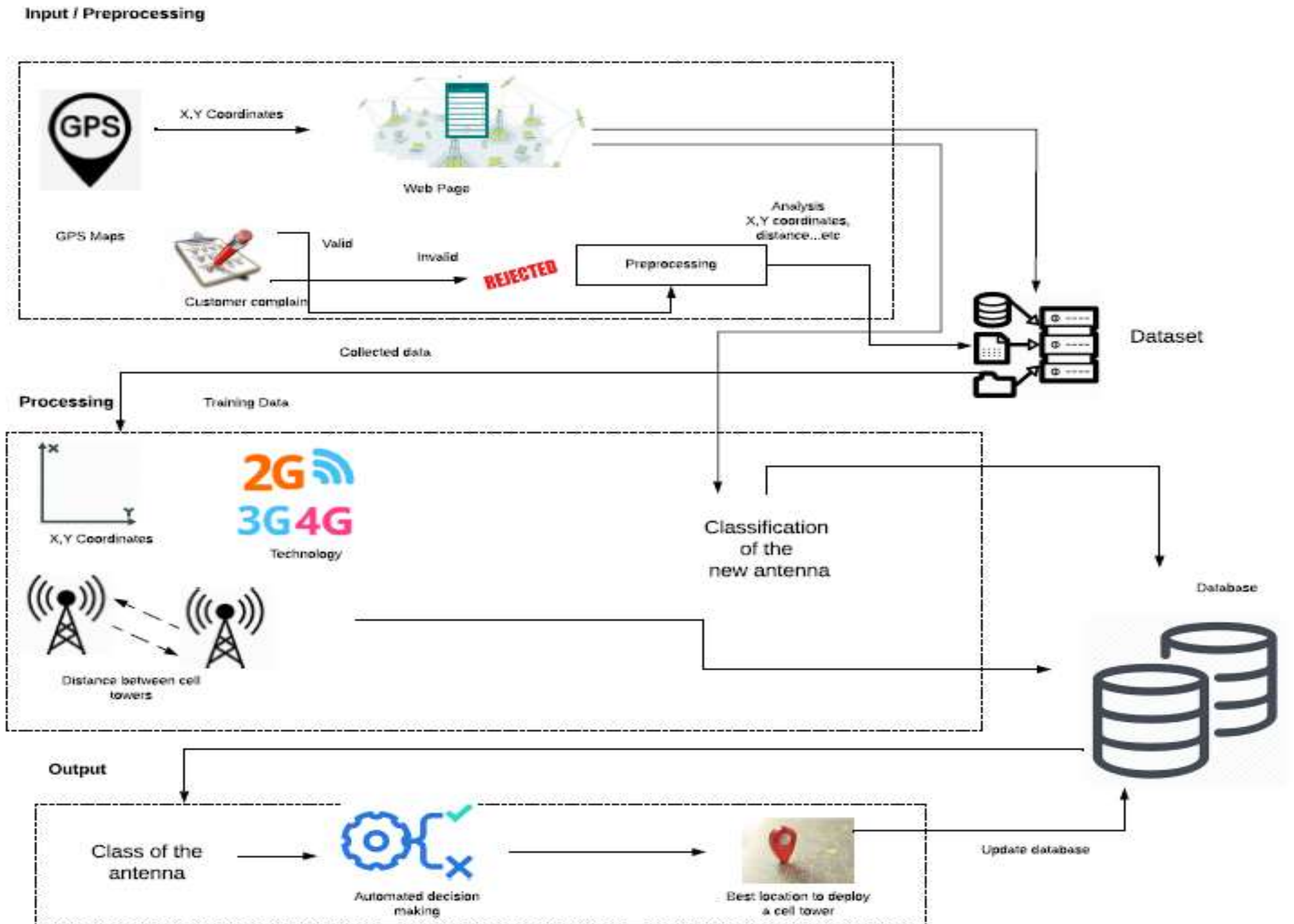
To avoid time consuming and help companies to decrease the cost of surveys.

03

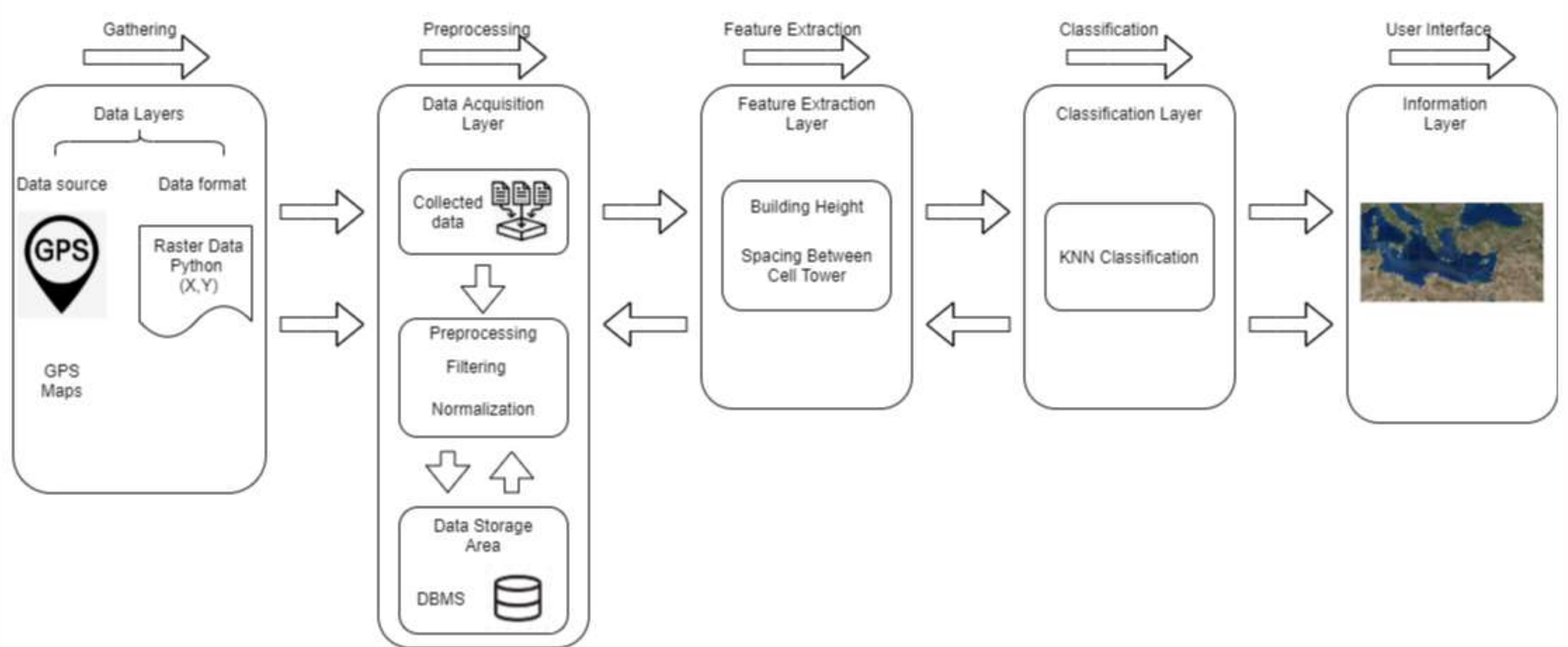
OBJECTIVE

Finding the **accurate** place to deploy a cell tower station costs mobile operators big chunks of money. Our target is to provide **automated** cell tower allocate system.

OVERVIEW DIAGRAM



BLOCK DIAGRAM



FUNCTIONAL REQUIREMENTS(1/7)

07

Title	Finding best location
Description	Searching for the best location to place the antenna to enhance or cover new area according to the regulations
Input	X,Y-Coordinates
Action	Filtering locations and building to find the best location according to certain parameters and a well-trained data-set
Output	Multiple suggestions for most appropriate locations
Pre-condition	Normalized data set where all attributes are well defined
Post-condition	Marking the best locations on the map
Dependencies	FR1 FR2

FUNCTIONAL REQUIREMENTS(2/7)

08

Title	Find Antenna
Description	Some antennas may be available for renting and broadcasting through it and this is a cheaper and better option than finding place for new antenna
Input	X,Y-Coordinates
Action	Searching for antennas nearby the location that are available for renting
Output	Suitable antenna and its location
Pre-condition	Defining the antennas in the dataset whether it belongs to the company or not
Post-condition	Marking the antenna on the map
Dependencies	FR1 FR2

FUNCTIONAL REQUIREMENTS(3/7)

09

Title	Report location
Description	Cropping part of the map surrounding the report location and displaying it
Input	X, Y-Coordinates for the report location
Action	Taking the X,Y-Coordinates as an input and adding some values to them to crop the surrounding area
Output	Certain map for report location
Pre-condition	None
Post-condition	Certain map page
Dependencies	FR1 FR2

FUNCTIONAL REQUIREMENTS(4/7)

10

Title	View map
Description	Displaying the whole map for the country displaying a layer containing all the antennas
Input	None
Action	Retrieving the map associated with antennas dataset using GIS tool plug in
Output	Map showing the antennas
Pre-condition	None
Post-condition	Map page
Dependencies	FR1 FR2

FUNCTIONAL REQUIREMENTS(5/7)

11

Title	View report
Description	Users may report weakness of signal in certain locations so these reports are archives by the customers service to enhance the service
Input	None
Action	Retrieve reports from database
Output	Reports page containing all reports
Pre-condition	None
Post-condition	Reports page
Dependencies	FR1 FR2

FUNCTIONAL REQUIREMENTS(6/7)

12

Title	Log in
Description	For existing engineers to log into accounts
Input	Username and password
Action	Check if all fields are filled and compares data entered to the record in the database
Output	Home Page and Acceptance message or error message
Pre-condition	None
Post-condition	Redirect to the home page
Dependencies	FR1

FUNCTIONAL REQUIREMENTS(7/7)

13

Title	Sign up
Description	Registration function for the engineer in the company who are in charge for the antenna management and allocation
Input	Name , Email , Password , Telephone , Gender
Action	Check all fields are filled and submit the data
Output	Home Page and Acceptance message or error message
Pre-condition	None
Post-condition	Redirect to the home page and update the database with a new record
Dependencies	None

14

NON-FUNCTIONAL REQUIREMENTS



Security



Reliability



Performane
and Speed

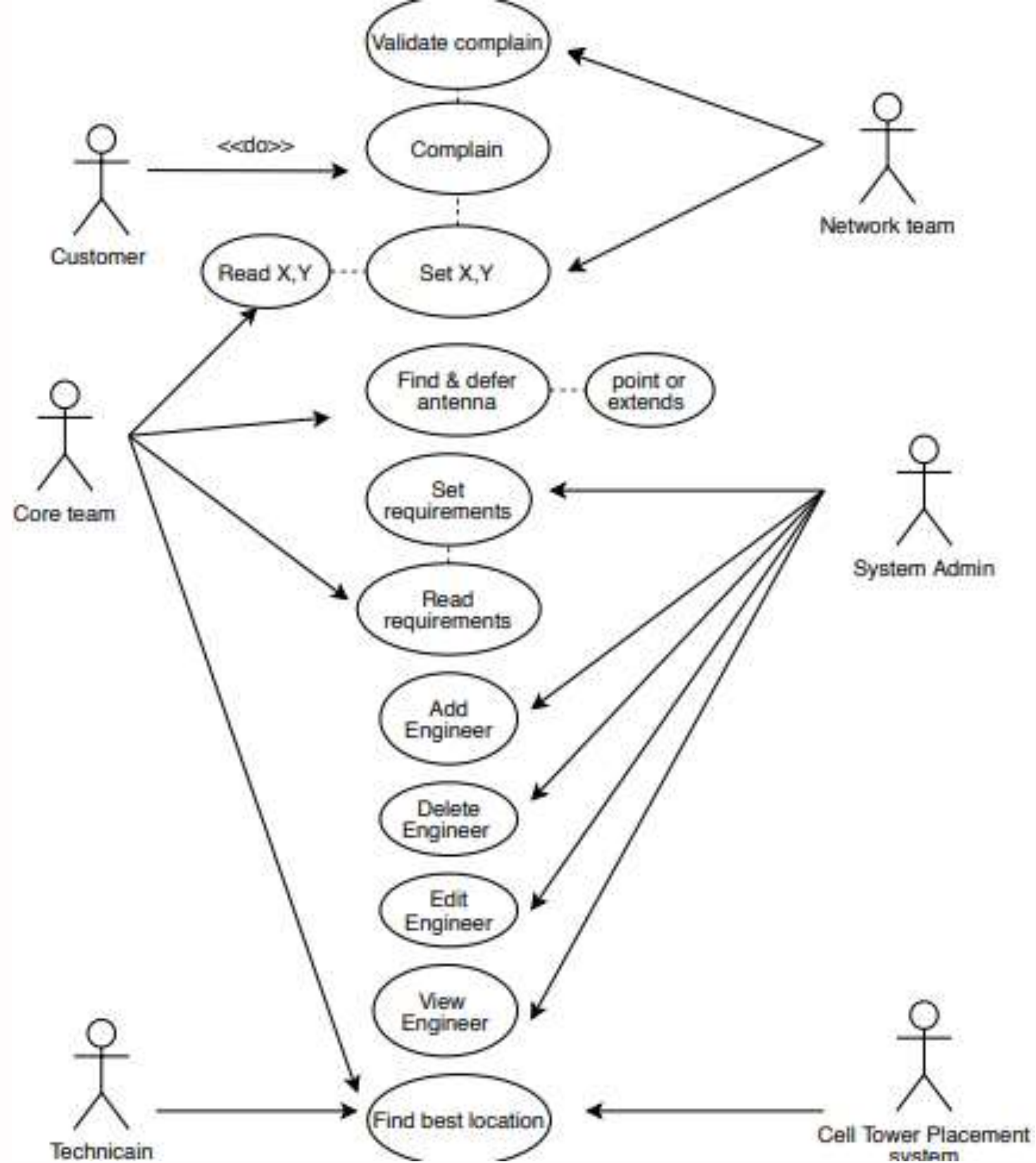


Scalabilty



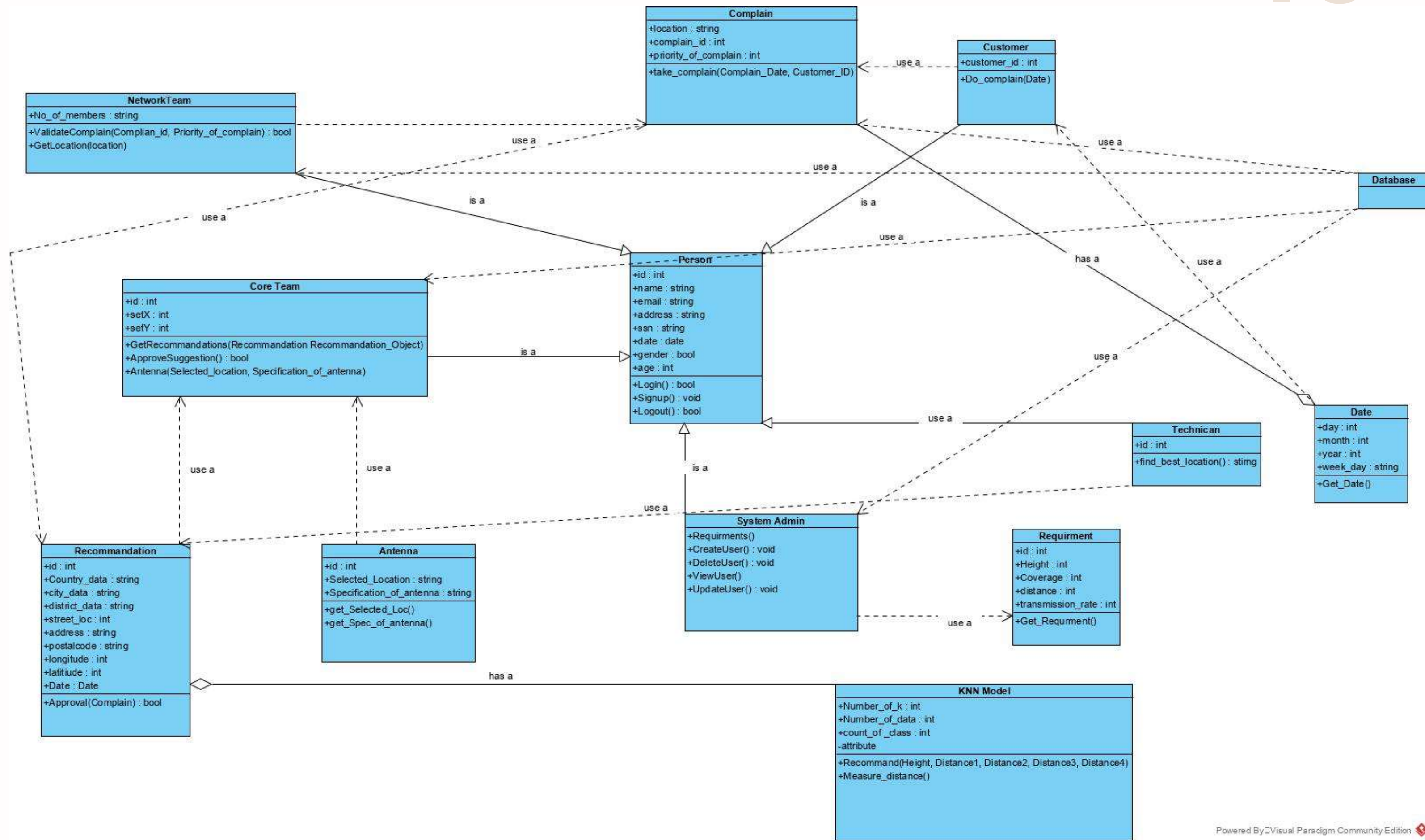
Usability

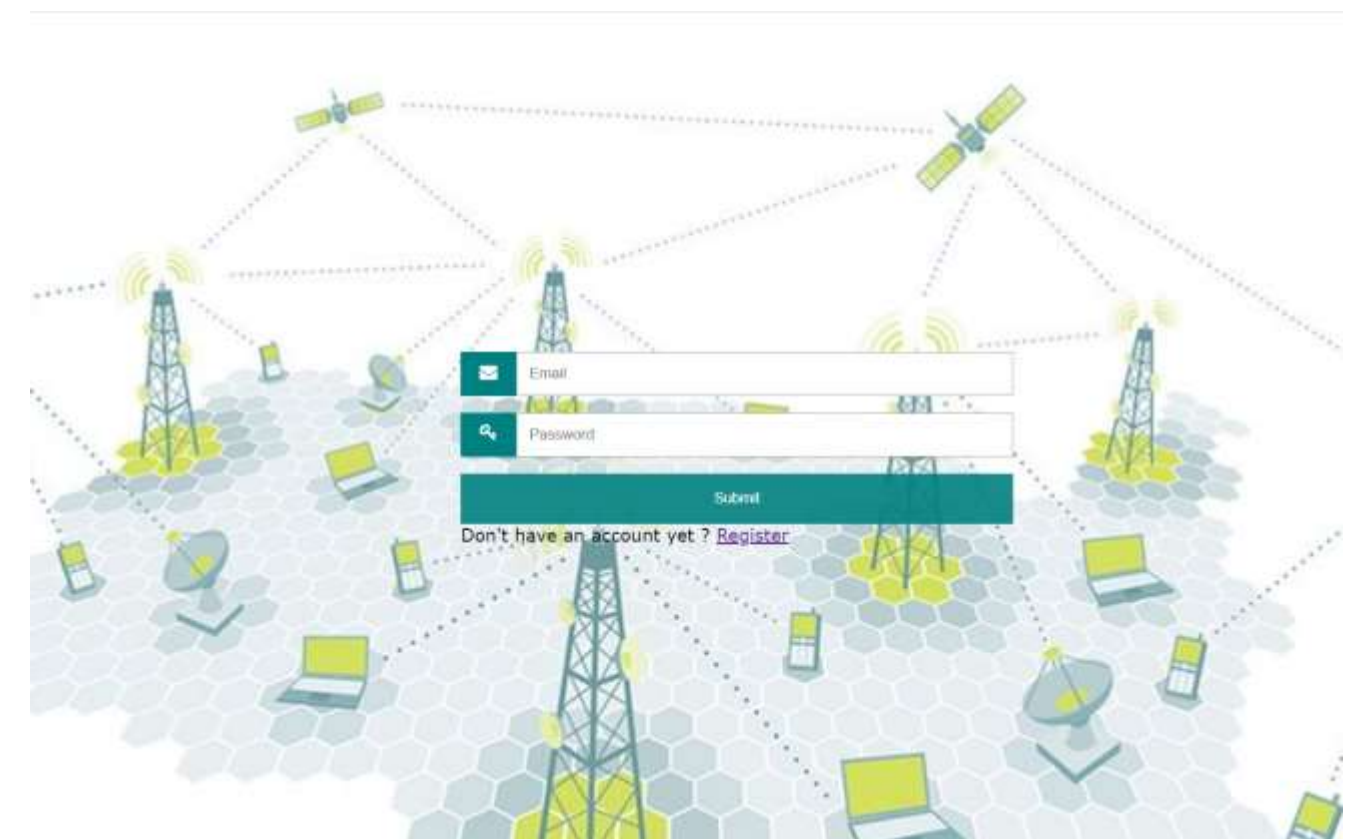
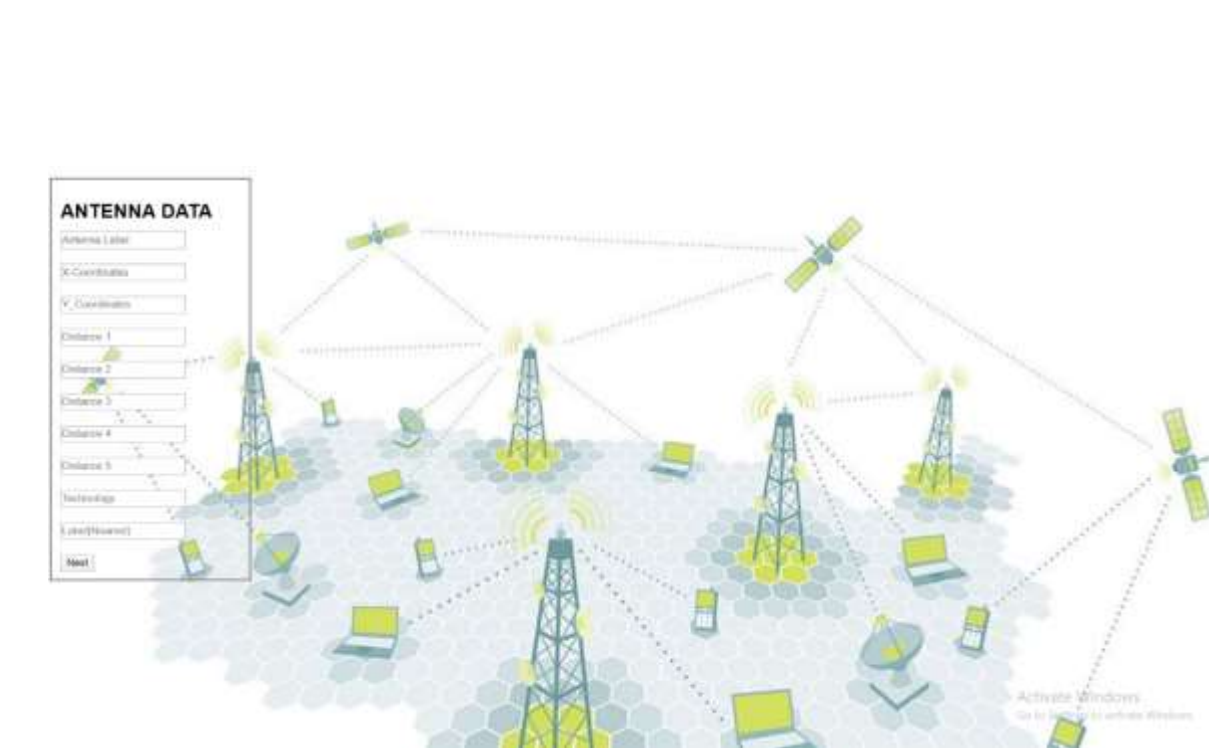
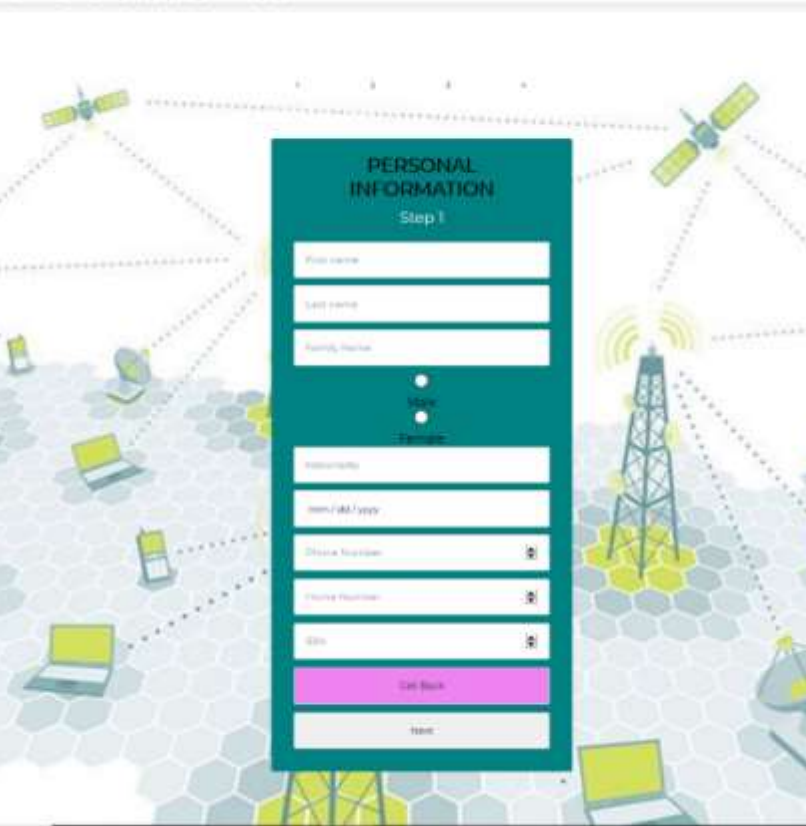
USE-CASE DIAGRAM



CLASS DIAGRAM

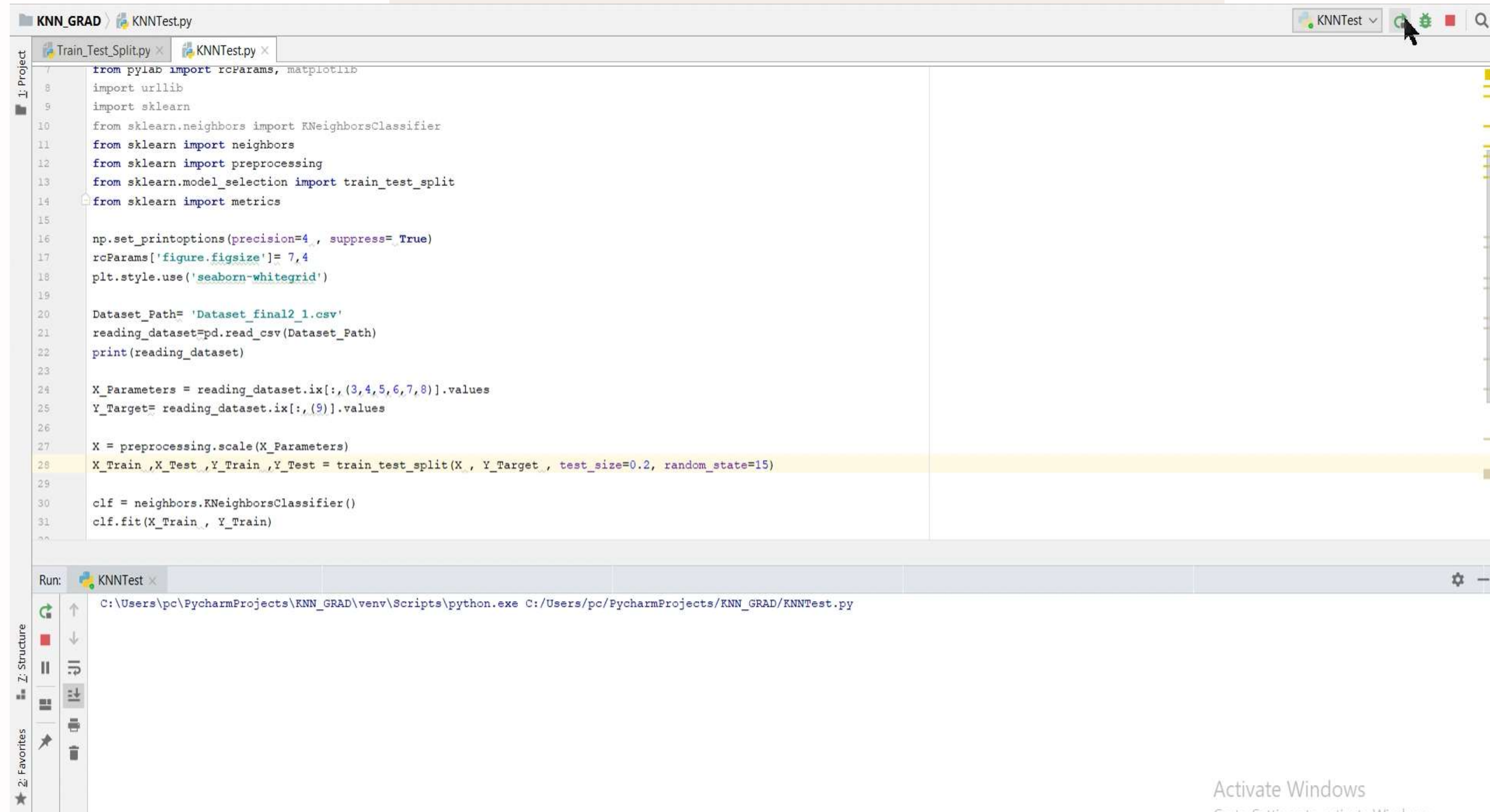
16





DEMO

18



```
1 from pylab import rcParams, matplotlib
2
3 import urllib
4
5 import sklearn
6
7 from sklearn.neighbors import KNeighborsClassifier
8
9 from sklearn import neighbors
10
11 from sklearn import preprocessing
12
13 from sklearn.model_selection import train_test_split
14
15 from sklearn import metrics
16
17 np.set_printoptions(precision=4, suppress=True)
18 rcParams['figure.figsize']= 7,4
19 plt.style.use('seaborn-whitegrid')
20
21 Dataset_Path= 'Dataset_final2_1.csv'
22 reading_dataset=pd.read_csv(Dataset_Path)
23 print(reading_dataset)
24
25 X_Parameters = reading_dataset.ix[:,(3,4,5,6,7,8)].values
26 Y_Target= reading_dataset.ix[:,(9)].values
27
28 X = preprocessing.scale(X_Parameters)
29
30 X_Train_,X_Test_,Y_Train_,Y_Test = train_test_split(X , Y_Target , test_size=0.2, random_state=15)
31
32 clf = neighbors.KNeighborsClassifier()
33 clf.fit(X_Train_ , Y_Train_)
```

Run: KNNTest x

C:\Users\pc\PycharmProjects\KNN_GRAD\venv\Scripts\python.exe C:/Users/pc/PycharmProjects/KNN_GRAD/KNNTest.py

Activate Windows
Go to Settings to activate Windows.

Accuracy

Train %	Test %	Accuracy %
60%	40%	90%
70%	30%	92%
80%	20%	92%
90%	10%	85%

Dataset Sample :

20

LABEL	X_Coordinates	Y_Coordinates	Dist 1	Dist 2	Dist 3	Dist 4	Dist 5	Technology	Class
4359	29.98156	31.33976	106.985	2240	5640	5120	5250	3	0
2902	29.97944	31.36271	3400	3350	3240	2230	2330	3	2
5723	29.97861	31.39806	2130	1990	3370	3240	4060	2	2
180	29.99665	31.39132	2100	3400	1660	210.16	2480	2	0
6489	29.99697	31.38967	2170	3190	210.16	1690	2420	3	0
180	30.00853	31.4028	1750	1900	796.25	833.28	1860	3	1

ANY QUESTION ?

THANK YOU

DATABASE SCHEMA

23

