VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

PROJECT REPORT

Topic: **Sentiment Analysis of Tourism Packages**
Data Mining (CSE3019)

Slot: G2

Prof. Lokesh Kumar R

Members:

Avik Mitra : 17BCE0208
Chinmay Goyal :17BCE0560

Acknowledgement

This project would not have been possible without the kind support and help of many individuals and our college, Vellore Institute of Technology, Vellore. We would like to extend our sincere thanks to all of them.

We are highly indebted to Prof. Lokesh Kumar R for his guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project.

We would like to express our gratitude towards our parents for their kind co-operation and encouragement which helped us in completion of this project.
We would like to express our special gratitude and thanks to industry persons for giving us such attention and time.
Our thanks and appreciations also go to our colleagues in developing the project and people who have willingly helped us out with their abilities.

Certificate

This is to certify that the project entitled, *Sentiment Analysis for Tourism Packages* submitted by Avik Mitra [17BCE0208] and Chinmay Goyal [17BCE0560] in partial fulfilment of the requirements for the course completion of course *Data Mining (CSE3019)* in Computer Science Engineering at the *Vellore Institute of Technology, Vellore* is an authentic work carried out by him under my supervision and guidance.

Date: 2 April 2019

Prof. Lokesh Kumar R

Vellore Institute of Technology, Vellore

TABLE OF CONTENTS

Abstract

Tourism industry introduces us to a variety of data about preferences of the travellers and gives us the better understanding of their preferences. Some prefer some packages over others. These preferences help the vendor to understand the pattern better and enable them to put emphasis on the most popular/ money-making schemes there are and improve their business and sales. This project analyses the data to find out the most popular tour packages and correlates it with the reason it is popular. We take a given dataset and realize the most efficient changes to a business model a vendor can make. We will be using different methods like correlation, data analysis, demand forecasting, sentiment analysis from controlled comments and find which package has been preferred by customers over others and why.

Data mining is the process of discovering patterns in large data sets involving methods at the intersection of machine learning, statistics, and database systems. Data mining is the analysis step of the "knowledge discovery in databases" process, or KDD. Aside from the raw analysis step, it also involves database and data management aspects, data pre-processing, model and inference considerations, interestingness metrics, complexity considerations, post-processing of discovered structures, visualization, and online updating. The difference between data analysis and data mining is that data analysis is to summarize the history such as analysing the effectiveness of a marketing campaign, in contrast, data mining focuses on using specific machine learning and statistical models to predict the future and discover the patterns among data.

In this project we will predict the best tour package by analysing the number of times it has been searched/looked for, what are the customer reviews (using sentiment analysis). We will be correlating the result of both the analysis to predict the best tour package which can help the vendor to increase its sale.

Sentiment analysis refers to the use of natural language processing, text analysis, computational linguistics, and biometrics to systematically identify, extract, quantify, and study affective states and subjective information. Sentiment analysis is widely applied to voice of the customer materials such as reviews and survey responses, online and social media, and healthcare materials for applications that range from marketing to customer service to clinical medicine.

List of Tables:

1. Introduction

1.1 System Overview

We use Vader sentiment analysis, a module in python, to analyse our text-based comments. The module provides us with data about the text line by line, about the positive reception, negative reception or the neutral reception of the statement. We then prioritise and impose constraints on the results with trained data of positive and negative nature and set the program such that the program gives near perfect detection of the positivity or the negativity of the training data and provides appropriate results.

1.2 Objective

Our objective in this project is to analyse sentiments based on comments and based on it give an overall idea of the popularity of a product in terms of rank to other consumers of the same product and give them a better understanding of popular demand.

1.3 Applications

Current Applications include use of this system in numerous e-commerce portals online for shopping as well as basic services. However the use of this system can be extended such that it gives a better understanding of popular opinion in surveys and other areas where it is immensely tedious to go through text based data that has been collected in the said survey.

1.4 Limitations

The limitations of the system is that it holistically takes into consideration the honesty of the customer while reviewing, as a constraint. Therefore additional

measures may be taken to prevent that from happening by pushing cross referencing measures from external as well as internal sources to check validity of a comment, either good or bad.

2. Literature Survey

Forecasting:

Forecasting is a subject that fascinates many people who are interested in the economics of tourism. Estimates of future demand at destination level are very important in managing and planning tourism development and the necessary investment. However, forecasting in the tourism sector is not an easy job. The tourism economists have developed forecasting models to predict travel demand for every major global market and particular traveller segments. More important, these models have established proven track records for accuracy. The reason is straight-forward and the models are firmly-rooted in the economic fundamentals of origin markets along with the changing dynamics of the tourism industry and traveller preferences.

In the process of planning, forecasting occurs in step "detailed research and analysis" and the stage of "synthetic" is expressed when the estimated amount that is intended to reach through the plan. Tourism forecasting methodology includes all methods used in the forecasting process. This methodology must be scientific because it can only obtain useful results. Despite its important role, the activity of forecasting tourism does not enjoy unanimous recognition of its importance in the planning process and in formulating macroeconomic policies. This is because first, most short-term forecasts are made and therefore cannot be a support for strategic planning and forecasting methodology on the other hand tourism is less developed than other economic sectors.

The Internet can also strengthen the marketing and communication functions of remote, peripheral and insular destinations, as well as small and medium-sized tourism enterprises, by enabling direct communication with prospective customers. Hence the Internet provides unprecedented and affordable opportunities for the global representation and marketing for both large and small tourism suppliers. Practical it provides unique opportunities for multimedia offer which is address the new consumer of tourism and resolved the follows issues:

1. increase security of transmissions;
2. ensure credibility and accountability of information;

3. secure intellectual property and copyright issues;
4. enhance bandwidth and reduce speed limitations;
5. reduce user confusion and dissatisfaction;
6. provide adequately trained specialists;
7. develop equal access for smaller and larger partners;
8. establish pricing structures for distribution of information and reservations;
9. enhance the standardization of information and reservation procedures.

Basis for Sentiment and intent analysis:

Sentiment is an attitude, thought, or judgment prompted by feeling. Sentiment analysis, which is also known as opinion mining, studies people's sentiments towards certain entities. Internet is a resourceful place with respect to sentiment information. From a user's perspective, people can post their own content through various social media, such as forums, micro-blogs, or online social networking sites. From a researcher's perspective, many social media sites release their application programming interfaces (APIs), prompting data collection and analysis by researchers and developers. For instance, Twitter currently has three different versions of APIs available, namely the REST API, the Search API, and the Streaming API.

Sentiment analysis is contextual mining of text which identifies and extracts subjective information in source material and helping a business to understand the social sentiment of their brand, product or service while monitoring online conversations.

It is important to classify incoming customer conversation about a brand based on following lines:

1. Key aspects of a brand's product and service that customers care about.
2. Users' underlying intentions and reactions concerning those aspects.

Sentiment Analysis is the most common text classification tool that analyses an incoming message and tells whether the underlying sentiment is positive, negative or neutral.

Intent analysis steps up the game by analysing the user's intention behind a message and identifying whether it relates an opinion, news, marketing, complaint, suggestion, appreciation or query.

Our application:

We use Vader sentiment analysis, a module in python, to analyse our text-based comments. The module provides us with data about the text line by line, about the positive reception, negative reception or the neutral reception of the statement. We then prioritise and impose constraints on the results with trained data of positive and negative nature and set the program such that the program gives near perfect detection of the positivity or the negativity of the training data and provides appropriate results.

How Is sentiment analysis done?

There are many ways that people analyse bodies of text for sentiment or opinions, but it usually boils down to two methods.

1. "Bag of Words" Model:

This model focuses completely on the words, or sometimes a string of words, but usually pays no attention to the "context" so-to-speak. The bag of words model usually has a large list, probably better thought of as a sort of "dictionary," which are considered to be words that carry sentiment. These words each have their own "value" when found in text. The values are typically all added up and the result is a sentiment valuation. The equation to add and derive a number can vary, but this model mainly focuses on the words, and makes no attempt to actually understand language fundamentals.

2. Using Natural Language Processing, and the attempt to truly "understand" the text:

This model attempts to have the machine actually understand the sentences structures, context, and is more focused on the succession of a string of words. Usually, this structure requires the machine to have understanding of grammar principles. To do this, Natural Language Processing (NLP) techniques are used to tag parts of speech, named entities, and more, in order to actually understand the "language" of the text, and not just look for target words.

Which form of sentiment analysis is better?

Both models can end up being very intense in their processes. Both models can also have similar accuracy in the end, so the choice between them usually depends on area of expertise or interest.

The "Bag of Words" models usually have massive amounts of machine learning that are built in and required. This often comes in the form of neural networks or support vector machines. The idea of these is to recognize patterns in data, in order to value words. Keep in mind though, the "Bag of Words" model is always seeing things as "objects" at best. There's never actually any attempt at understanding of grammar or sentence structure behind the text, besides pre-defined strings and words.

The attempts to truly understand the text, while possibly not requiring as much data to be machine learned against, is equally as difficult. Usually, the attempt to fully understand the text involves more "context" and word "succession" so to speak, without pre-built rules for exact structures. This model can indeed wind up using just as much machine learning as the past, but then, in my opinion, it winds up straying more to the side of "bag of words" as it does this

Code snippet: Python

```
with open("positive.txt","r") as f:
   for line in f.read().split('\n'):
      vs=analyzer.polarity_scores(line)

      if vs['compound'] >= threshold or vs['compound'] <= -threshold:
         if vs['compound']>0:
            pos_correct +=1
         pos_count +=1
```

The vs variable holds the values of polarity scores attribute of the analyser object, which is an object of the SentimentIntensityAnalyzer() class.
The vs variable produces values such as pos, neg and neu which stand for positive, negative and neutral. These refer as to whether the statement makes positive, negative or neutral sense to the compiler.
These values are then computed and a threshold value of 0.5 is set as a constraint. The compound feature used here, which is also present in the polarity scores attribute, gives a combined report of the positivity, negativity or neutrality of the text it has processed. Therefore, the range of -0.5 to 0.5 is set as the neutral zone and anything above and below those values are taken to be as positive or negative reception respectively.
The pos correct or neg correct variables count the number of lines thought of to be as positive or negative in nature and thus add to the counter. The pos count and neg

count variables keep track of the total number of new lines read by the program. Using these values, the percentage accuracy of positivity or negativity is calculated.

Vader sentiment analysis also recognises unconventional styles of typing or writing whereby keywords in a sentence that are capitalised are given more weight depending on their nature.

Advantages:

VADER has a lot of advantages over traditional methods of Sentiment Analysis, including:

- It works exceedingly well on social media type text, yet readily generalizes to multiple domains
- It **doesn't require any training data** but is constructed from a generalizable, valence-based, human-curated gold standard sentiment lexicon
- It is fast enough to be used online with streaming data, and
- It does not severely suffer from a speed-performance trade off.

Vader Sentiment:

Sentiment analysis is useful to a wide range of problems that are of interest to human-computer interaction practitioners and researchers, as well as those from fields such as sociology, marketing and advertising, psychology, economics, and political science. The inherent nature of microblog content - such as those observed on Twitter and Facebook - poses serious challenges to practical applications of sentiment analysis. Some of these challenges stem from the sheer rate and volume of user generated social content, combined with the contextual sparseness resulting from shortness of the text and a tendency to use abbreviated language conventions to express sentiments.

A comprehensive, high quality lexicon is often essential for fast, accurate sentiment analysis on such large scales. An example of such a lexicon that has been widely used in the social media domain is the Linguistic Inquiry and Word Count (LIWC).

Sociologists, psychologists, linguists, and computer scientists find LIWC appealing because it has been extensively validated. Also, its straightforward dictionary and simple word lists are easily inspected, understood, and extended if desired. Such attributes make LIWC an attractive option to researchers looking for a reliable lexicon to extract emotional or sentiment polarity from text. Despite their pervasive use for gaging sentiment in social media contexts, these lexicons are often used with little regard for their actual suitability to the domain.

VADER stands for Valence Aware Dictionary for sEntiment Reasoning.

A combination of qualitative and quantitative methods are used to produce, and then empirically validate, a gold-standard sentiment lexicon that is especially attuned to microblog-like contexts. We next combine these lexical features with consideration for five generalizable rules that embody grammatical and syntactical conventions that humans use when expressing or emphasizing sentiment intensity. We find that incorporating these heuristics improves the accuracy of the sentiment analysis engine across several domain contexts.

Sentiment Lexicons A substantial number of sentiment analysis approaches rely greatly on an underlying sentiment (or opinion) lexicon. A sentiment lexicon is a list of lexical features which are generally labelled according to their semantic orientation as either positive or negative. Manually creating and validating such lists of opinion-bearing features, while being among the most robust methods for generating reliable sentiment lexicons, is also one of the most time-consuming. For this reason, much of the applied research leveraging sentiment analysis relies heavily on pre-existing manually constructed lexicons. Because lexicons are so useful for sentiment analysis, we briefly provide an overview of several benchmarks. We first review three widely used lexicons in which words are categorized into binary classes (i.e., either positive or negative) according to their context free semantic orientation. We then describe three other lexicons in which words are associated with valence scores for sentiment intensity.

Because manually creating and validating a comprehensive sentiment lexicon is labour and time intensive, much work has explored automated means of identifying sentiment relevant features in text. Typical state of the art practices incorporate machine learning approaches to “learn” the sentiment-relevant features of text. The Naive Bayes (NB) classifier is a simple classifier that relies on Bayesian probability and the naive assumption that feature probabilities are independent of one another. Maximum Entropy (ME) is a general-purpose machine learning technique belonging to the class of exponential models using multinomial logistic regression. Unlike NB, ME makes no conditional independence assumption between features, and thereby accounts for information entropy (feature weightings). Support Vector Machines (SVMs) differ from both NB and ME models in that SVMs are non-probability classifiers which operate by separating data points in space using one or more hyperplanes (centre lines of the gaps separating different classes). We use the Python-based machine learning algorithms from scikit-learn.org for the NB, ME, SVM-Classification (SVM-C) and SVM-Regression (SVM-R) models.

Machine learning approaches are not without drawbacks. First, they require (often extensive) training data which are, as with validated sentiment lexicons, sometimes troublesome to acquire. Second, they depend on the training set to represent as many features as possible (which often, they do not – especially in the case of the

short, sparse text of social media). Third, they are often more computationally expensive in terms of CPU processing, memory requirements, and training/classification time (which restricts the ability to assess sentiment on streaming data). Fourth, they often derive features “behind the scenes” inside of a black box that is not (easily) human interpretable and are therefore more difficult to either generalize, modify, or extend (e.g., to other domains).

Overall methodology:

The above technique would help to create a point-based system, assigned to each product quantitatively and help product placement based on customer response through words.

The snippet of percentage would be integrated with an overall system of internalised upvotes or downvotes over the system to continually rank packages, while updating the database of packages at the same time. The ranking system would help understand demands for a particular package and help us understand what is being preferred by the consumer and what is not.

This ranking/prioritising system will be based on both purchase frequencies and customer response through text comments.

3. System Analysis

3.1 Existing Systems

Present systems include complex algorithms that assess the viability of the product through its online review. This is not solely based on or dependent on the honesty of the consumers but also other factors that bring the truthfulness of the review or rating into consideration before factoring its value into the recommender systems present in the current e-commerce websites and portals.

3.2 Proposed System

Our system depends on user honesty and therefore we factor out the system/ machine errors that are bound to happen once a comment is pushed. It is imperative that the data pushed may be in reality positive but then perceived as negative or neutral by the failsafe systems in place due to the nature or placement of the letters and even spelling errors.

4. Requirement Specification

4.1 Hardware Requirements

This project is software based and therefore in reality only the computer system on which it is written is used as a hardware support.

Dell Inspiron 13

- Processor: Intel Core i7-8550U.
- Graphics: Intel UHD Graphics 620.
- RAM: 16GB, DDR4, 2133MHz.
- Storage: 512GB Solid State Drive.
- Display: 13.3-inch FHD (**1920 x 1080**) IPS Truelife LED-Backlit Narrow Border Touch Display.

4.2 Software Specification

Python Programming for basic backend

Django module to link programming of backend to the Html pages

Hypertext Markup Language(HTML) for frontend display

5. System Design Specification

5.1 System Architecture

Data flow architecture: In data flow architecture, the whole software system is seen as a series of transformations on consecutive pieces or set of input data, where data and operations are independent of each other. In this approach, the data enters into the system and then flows through the modules one at a time until they are assigned to some final destination (output or a data store).

The connections between the components or modules may be implemented as I/O stream, I/O buffers, piped, or other types of connections. The data can be flown in the graph topology with cycles, in a linear structure without cycles, or in a tree type structure.

The main objective of this approach is to achieve the qualities of reuse and modifiability. It is suitable for applications that involve a well-defined series of independent data transformations or computations on orderly defined input and output such as compilers and business data processing applications. There are three types of execution sequences between modules−

- Batch sequential
- Pipe and filter or non-sequential pipeline mode
- Process control

5.2 Module Description

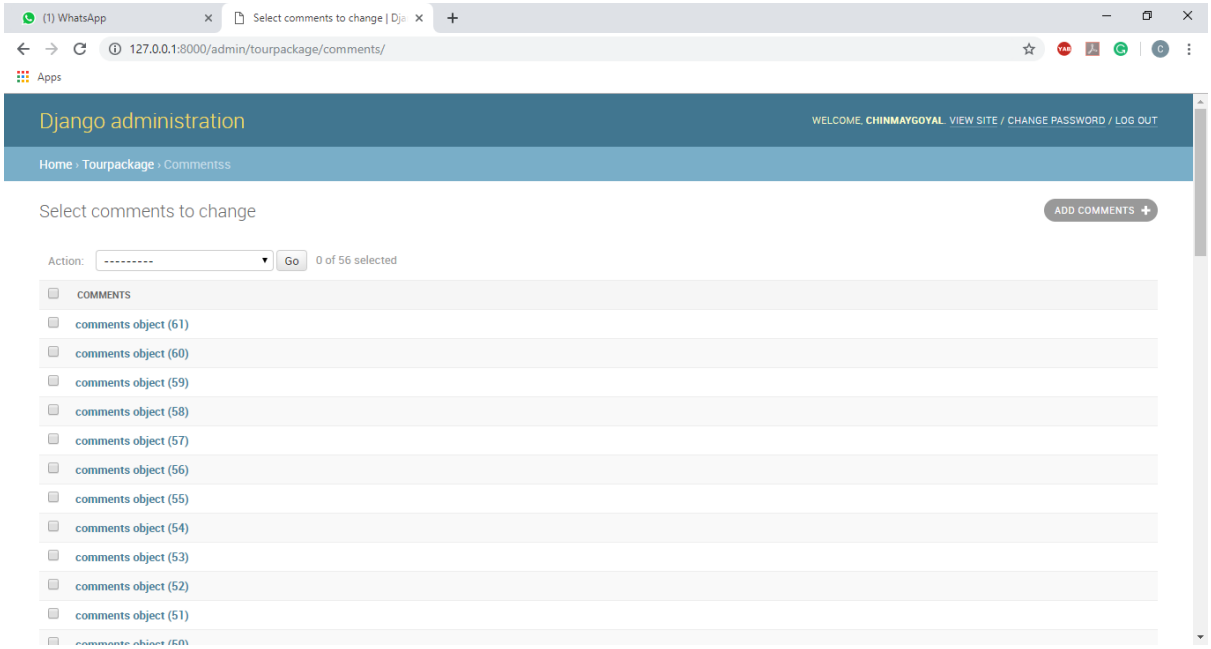
The project comprises of:

1. Package list page: Enlists the different available packages for sale and review of the product.

2. Package description page: Consists of information about the package where it is possible to post comments about the package that in turn affects the ranking and priority of the said package.

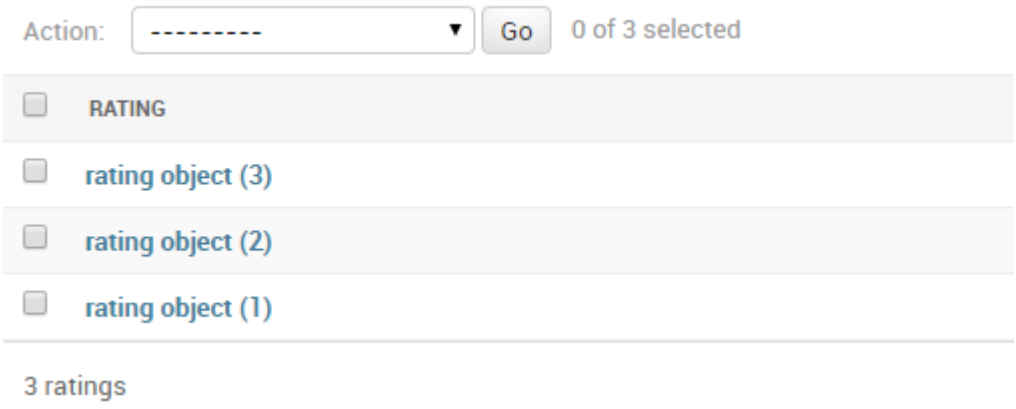
5.3 Database Design

A view of Comments table where all the comments are stored in new tuples




```
class comments(models.Model):
    package = models.IntegerField()
    comment = models.TextField()
```

A view of rating table where there are tuples for each package and rating is updated in those tuples itself



```
class rating(models.Model):
    rank = models.IntegerField()
```

6. System Implementation

We use Vader sentiment analysis, a module in python, to analyse our text-based comments. The module provides us with data about the text line by line, about the positive reception, negative reception or the neutral reception of the statement. We then prioritise and impose constraints on the results with trained data of positive and negative nature and set the program such that the program gives near perfect detection of the positivity or the negativity of the training data and provides appropriate results.

7. Conclusion and Future Enhancements

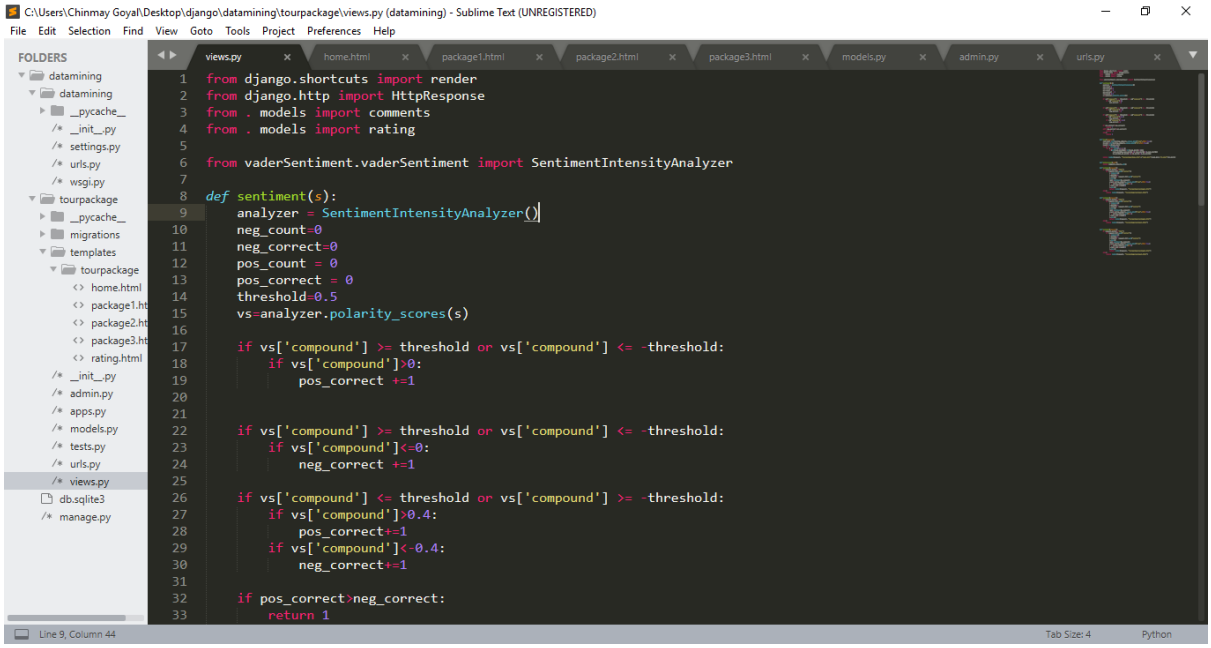
This project demonstrates a simple use of the bag of words system to classify comments as positive, negative or neutral. This helps us in assessing the quality of a said package that is being offered and helps provide an insight to the popularity of the item.

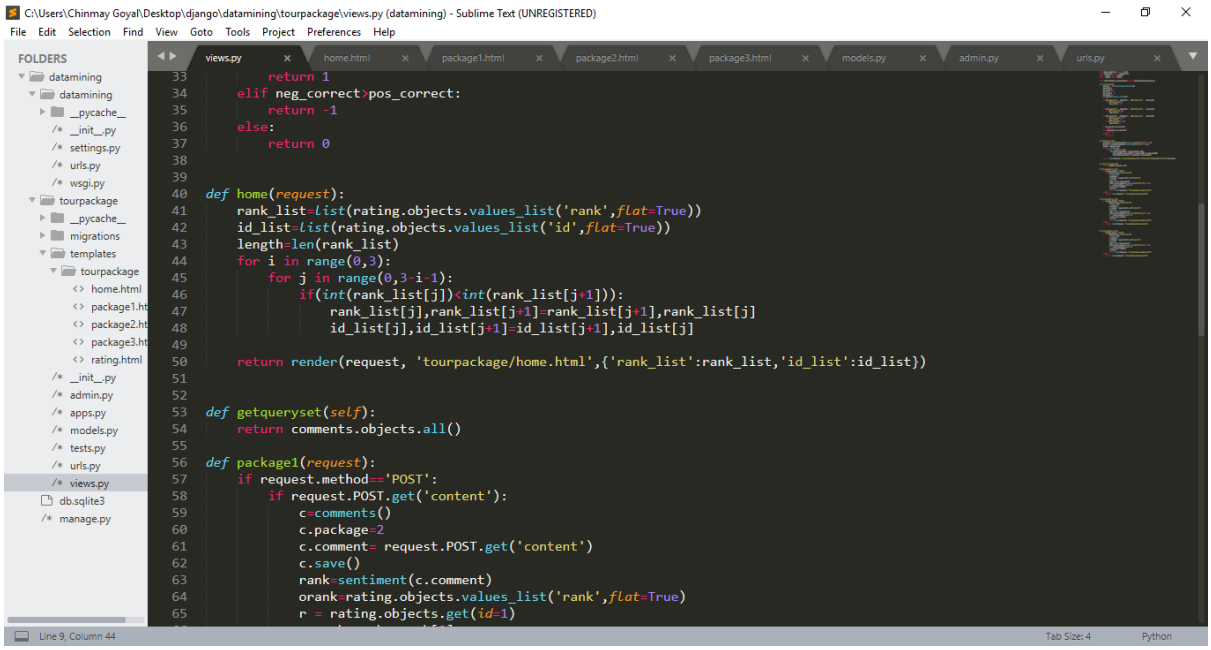
Though effective the dependence on user honesty and the lack of protection against false smear campaigns is a problem

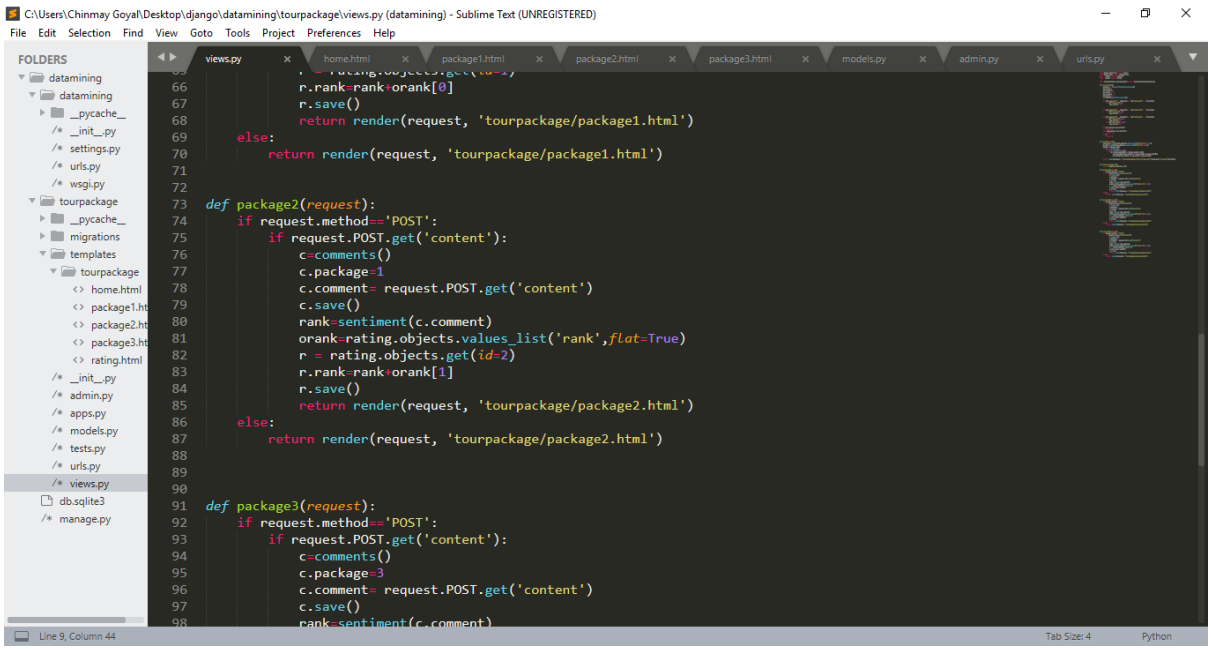
These issues can be solved with the help of other factors that prevent spamming as well as check validity of review data by cross referencing it with other reputed resources. This will make the system more resistant to manipulation and help in giving it a better chance at being more authentic and serve the right consumers.

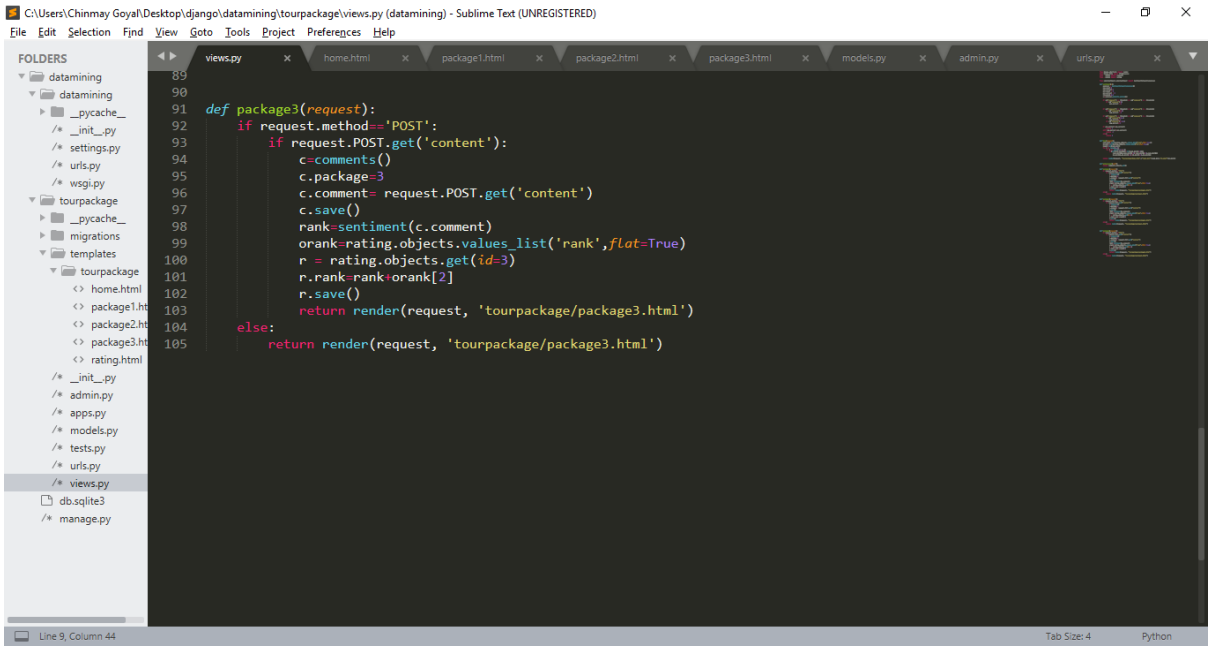
8. Appendices

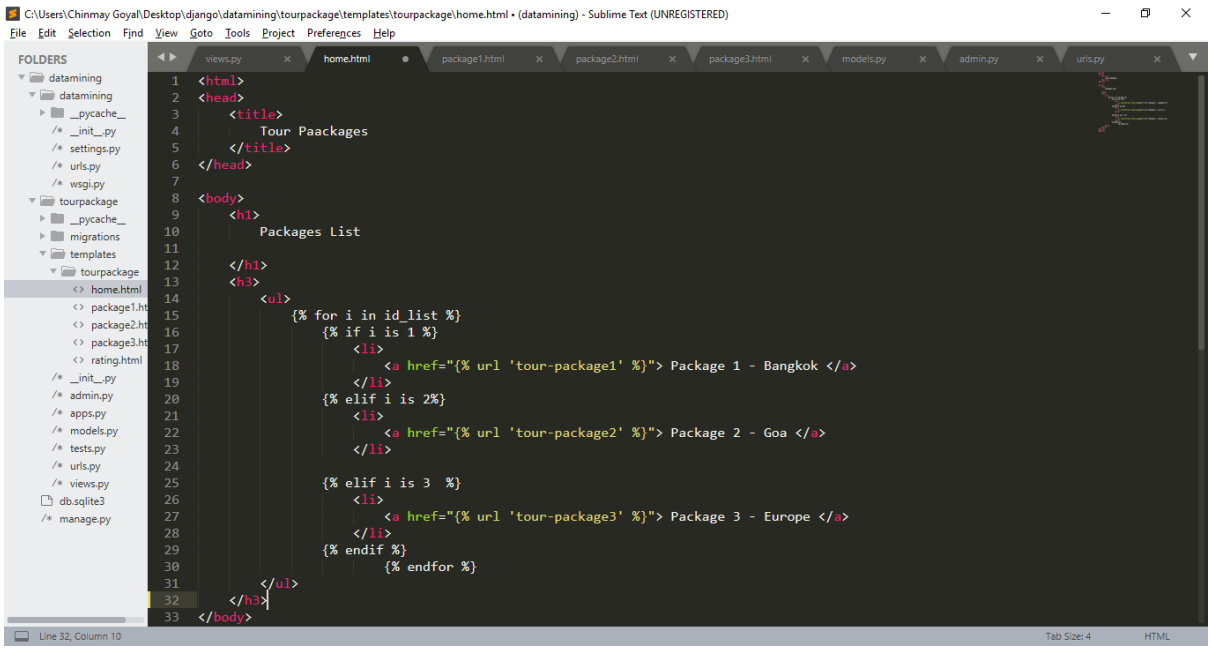
8.1 APPENDIX 1 - SAMPLE SOURCE CODE


C:\Users\Chinmay Goyal\Desktop\django\datamining\tourpackage\views.py (datamining) - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
FOLDERS
views.py
home.html
package1.html
package2.html
package3.html
models.py
admin.py
urls.py
from django.shortcuts import render
from django.http import HttpResponse
from . models import comments
from . models import rating
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer
def sentiment(s):
analyzer = SentimentIntensityAnalyzer()
neg_count=0
neg_correct=0
pos_count = 0
pos_correct = 0
threshold=0.5
vs=analyzer.polarity_scores(s)
if vs['compound'] >= threshold or vs['compound'] <= -threshold:
if vs['compound']>0:
pos_correct +=1
if vs['compound'] >= threshold or vs['compound'] <= -threshold:
if vs['compound']<=0:
neg_correct +=1
if vs['compound'] <= threshold or vs['compound'] >= -threshold:
if vs['compound']>0.4:
pos_correct+=1
if vs['compound']<-0.4:
neg_correct+=1
if pos_correct>neg_correct:
return 1
Line 9, Column 44
Tab Size: 4
Python


C:\Users\Chinmay Goyal\Desktop\django\datamining\tourpackage\views.py (datamining) - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
FOLDERS
views.py
return 1
elif neg_correct>pos_correct:
return -1
else:
return 0
def home(request):
rank_list=list(rating.objects.values_list('rank',flat=True))
id_list=list(rating.objects.values_list('id',flat=True))
length=len(rank_list)
for i in range(0,3):
for j in range(0,3-i-1):
if(int(rank_list[j])<int(rank_list[j+1])):
rank_list[j],rank_list[j+1]=rank_list[j+1],rank_list[j]
id_list[j],id_list[j+1]=id_list[j+1],id_list[j]
return render(request, 'tourpackage/home.html',{'rank_list':rank_list,'id_list':id_list})
def getqueryset(self):
return comments.objects.all()
def package1(request):
if request.method=='POST':
if request.POST.get('content'):
c=comments()
c.package=2
c.comment= request.POST.get('content')
c.save()
rank=sentiment(c.comment)
orank=rating.objects.values_list('rank',flat=True)
r = rating.objects.get(id=1)
Line 9, Column 44
Tab Size: 4
Python

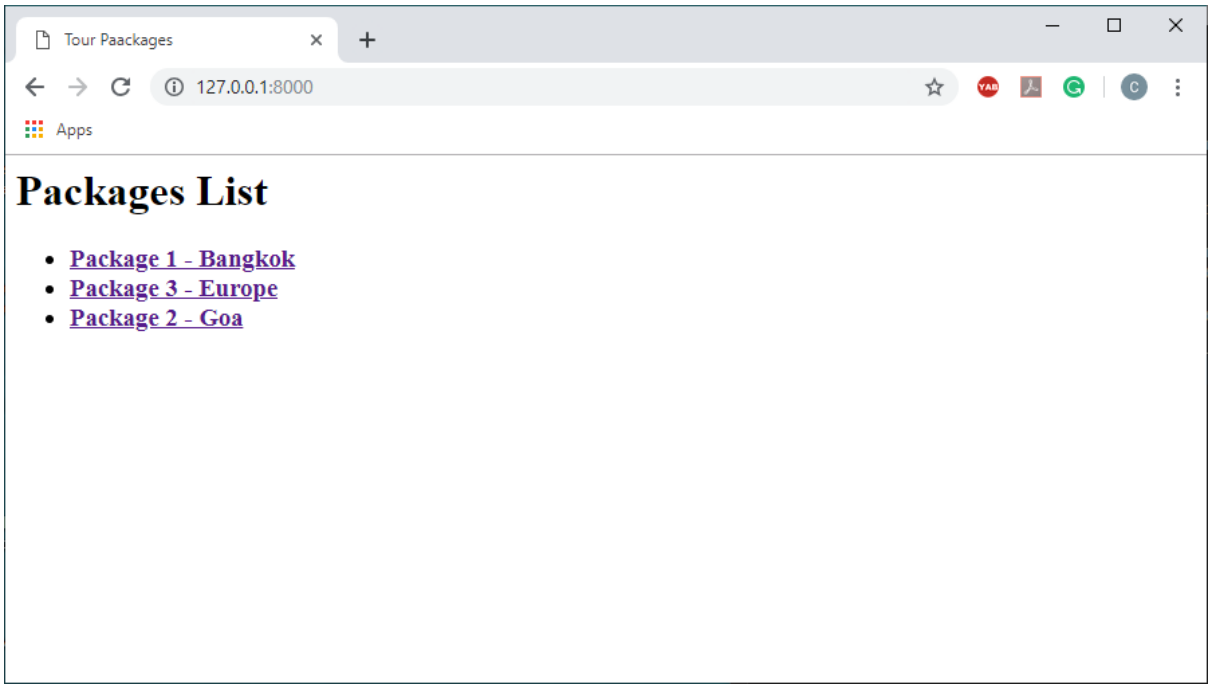

C:\Users\Chinmay Goyal\Desktop\django\datamining\tourpackage\views.py (datamining) - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
FOLDERS
views.py
home.html
package1.html
package2.html
package3.html
models.py
admin.py
urls.py
r.rank=rank+orank[0]
r.save()
return render(request, 'tourpackage/package1.html')
else:
return render(request, 'tourpackage/package1.html')
def package2(request):
if request.method=='POST':
if request.POST.get('content'):
c=comments()
c.package=1
c.comment= request.POST.get('content')
c.save()
rank=sentiment(c.comment)
orank=rating.objects.values_list('rank',flat=True)
r = rating.objects.get(id=2)
r.rank=rank+orank[1]
r.save()
return render(request, 'tourpackage/package2.html')
else:
return render(request, 'tourpackage/package2.html')
def package3(request):
if request.method=='POST':
if request.POST.get('content'):
c=comments()
c.package=3
c.comment= request.POST.get('content')
c.save()
Line 9, Column 44
Tab Size: 4
Python

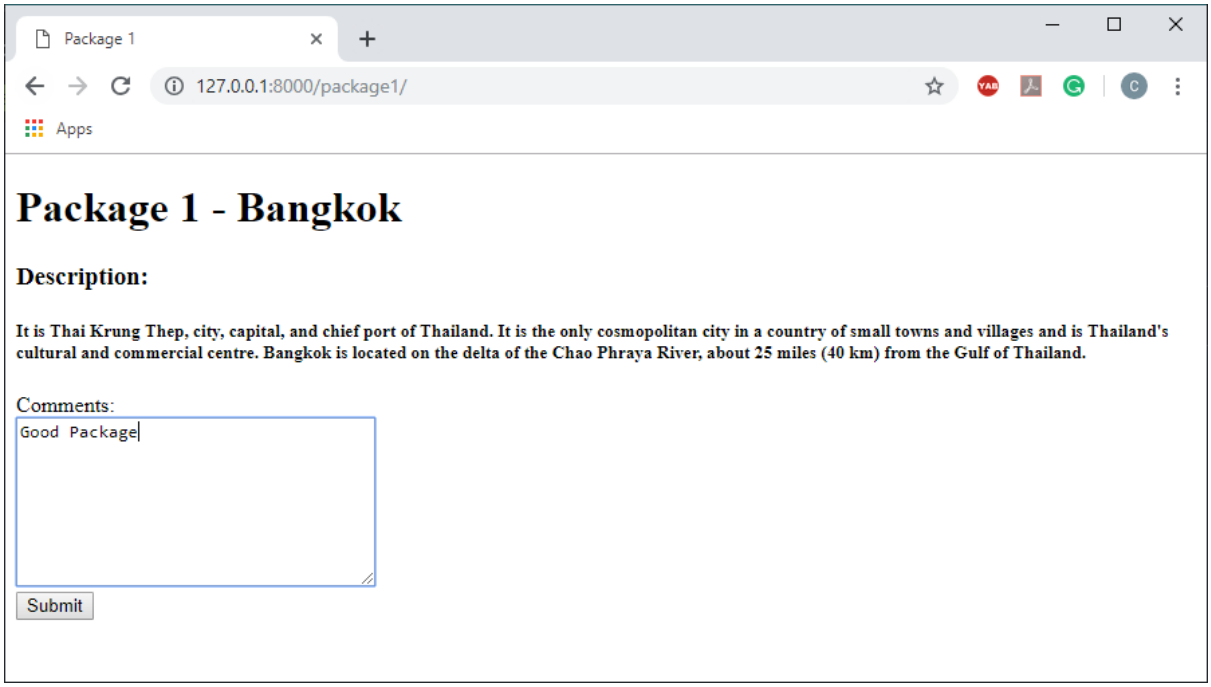

C:\Users\Chinmay Goyal\Desktop\django\datamining\tourpackage\views.py (datamining) - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
FOLDERS
views.py
home.html
package1.html
package2.html
package3.html
models.py
admin.py
urls.py
def package3(request):
if request.method=='POST':
if request.POST.get('content'):
c=comments()
c.package=3
c.comment= request.POST.get('content')
c.save()
rank=sentiment(c.comment)
orank=rating.objects.values_list('rank',flat=True)
r = rating.objects.get(id=3)
r.rank=rank+orank[2]
r.save()
return render(request, 'tourpackage/package3.html')
else:
return render(request, 'tourpackage/package3.html')
Line 9, Column 44
Tab Size: 4
Python

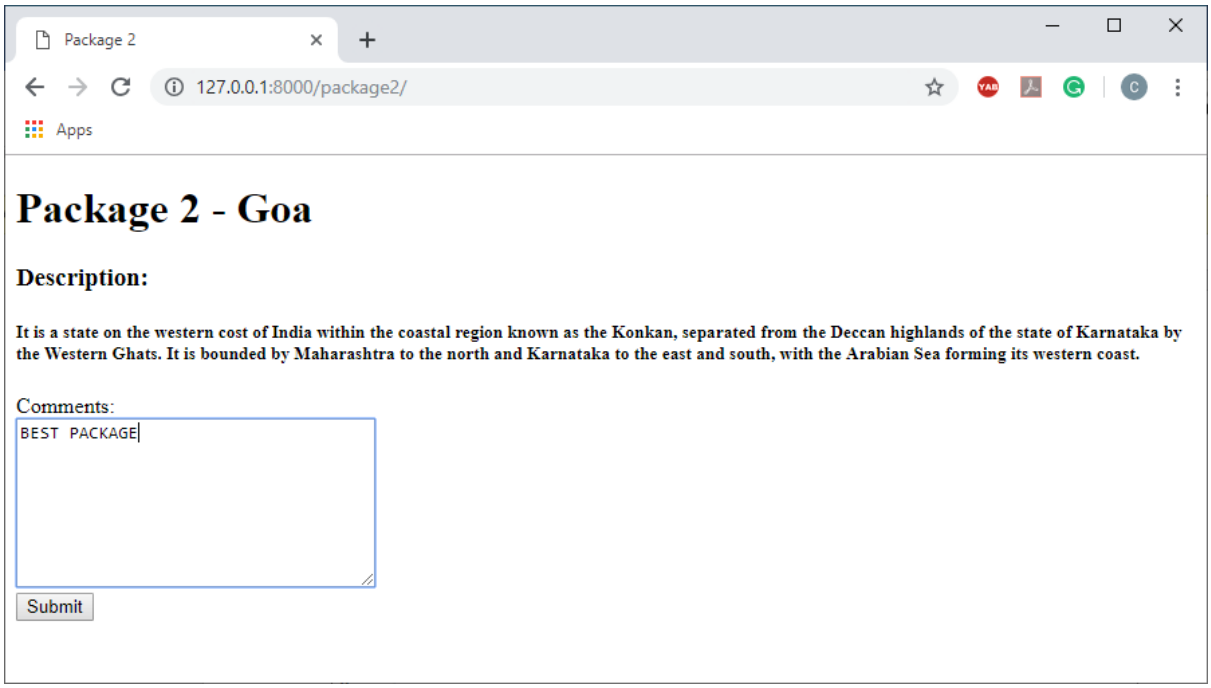


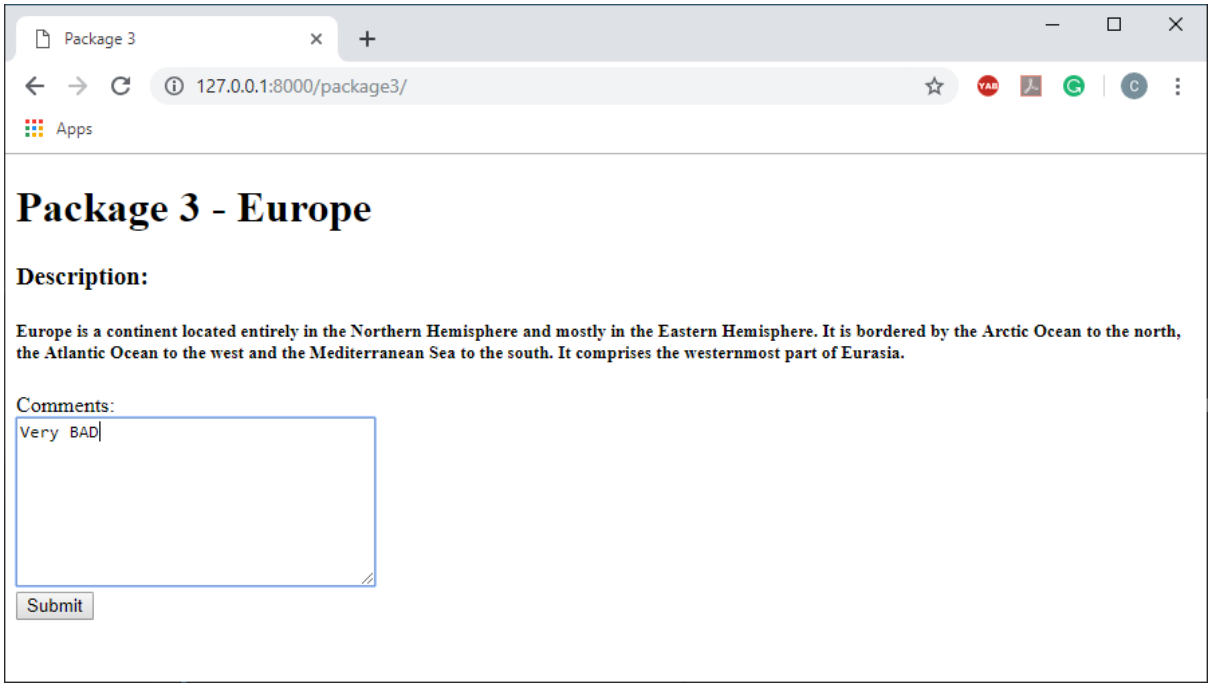
8.2 APPENDIX 1 – SCREENSHOTs/OUTPUTs

First Page

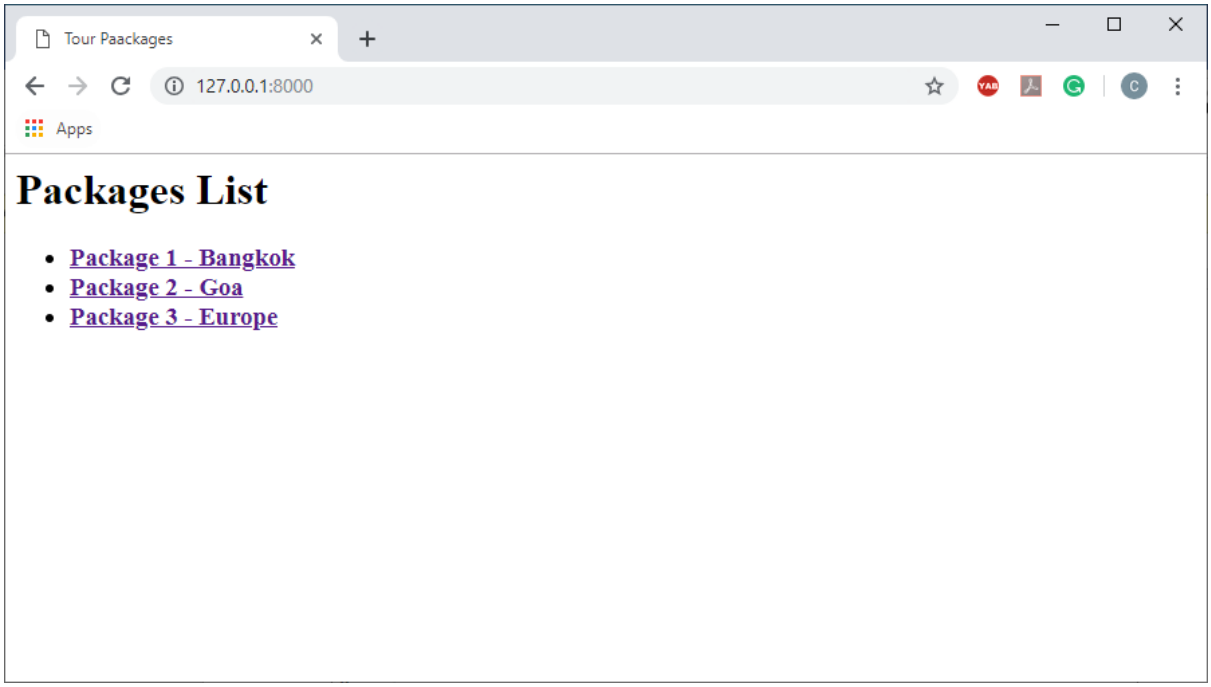



Package 1
127.0.0.1:8000/package1/
Apps
Package 1 - Bangkok
Description:
It is Thai Krung Thep, city, capital, and chief port of Thailand. It is the only cosmopolitan city in a country of small towns and villages and is Thailand's cultural and commercial centre. Bangkok is located on the delta of the Chao Phraya River, about 25 miles (40 km) from the Gulf of Thailand.
Comments:
Good Package
Submit


Package 2
127.0.0.1:8000/package2/
Apps
Package 2 - Goa
Description:
It is a state on the western cost of India within the coastal region known as the Konkan, separated from the Deccan highlands of the state of Karnataka by the Western Ghats. It is bounded by Maharashtra to the north and Karnataka to the east and south, with the Arabian Sea forming its western coast.
Comments:
BEST PACKAGE
Submit



Page after adding all comments and when sorting is done



9. References

9.1 List of Journals

https://www.google.com/url?sa=t&source=web&rct=j&url=https://pdfs.semanticscholar.org/beec/0b7ed38a6a9408724e87b142608ec0c8e586.pdf&ved=2ahUKEwjq5oOMy7DhAhVLfysKHY01B9IQFjAAegQIBhAC&usg=AOvVaw3kfckpP9kuJrdVSElcnAAD

https://www.google.com/url?sa=t&source=web&rct=j&url=https://www.sciencedirect.com/science/article/abs/pii/S0148296317300231&ved=2ahUKEwjq5oOMy7DhAhVLfysKHY01B9IQFjADegQIARAB&usg=AOvVaw1wjegJnYpZezOkjgZtYb98

https://www.google.com/url?sa=t&source=web&rct=j&url=http://cs229.stanford.edu/proj2012/ZarghamNassirpourNasiri-ElectronicDevicesSalesPredictionUsingSocialMediaSentimentAnalysis.pdf&ved=2ahUKEwjq5oOMy7DhAhVLfysKHY01B9IQFjAIegQIAxAB&usg=AOvVaw1GR0cshueycZtZyl3QHws8&cshid=1554180025051

https://www.google.com/url?sa=t&source=web&rct=j&url=https://www.ripublication.com/ijaer18/ijaerv13n14_25.pdf&ved=2ahUKEwjq5oOMy7DhAhVLfysKHY01B9IQFjAJegQIBBAB&usg=AOvVaw0EDJDzcctcyxekWBDIDzUC&cshid=1554180025051

9.2 List of Websites

https://www.google.com/url?sa=t&source=web&rct=j&url=%23&ved=2ahUKEwjY9r7TzLDhAhWFfisKHSHWCogQwqsBMAB6BAgDEAU&usg=AOvVaw0VTJzWxrN8ZFOD4xbU2nov

https://www.stackoverflow.com

https://www.google.com/url?sa=t&source=web&rct=j&url=https://www.tutorialspoint.com/django/django_template_system.htm&ved=2ahUKEwjGvpW-zLDhAhXJfSsKHWnUAYwQFjACegQIAxAB&usg=AOvVaw2jex-b5Gm5Cb2xDYvHHtvQ