**A**

**INDUSTRIAL TRAINING REPORT**

**ON**

**Restaurant Review Analysis**



SHRI RAM MURTI SMARAK COLLEGE OF ENGINEERING, TECHNOLOGY & RESEARCH, BAREILLY

DR. A. P. J. ABDUL KALAM TECHNICAL UNIVERSITY, LUCKNOW

2020-2021

Department Name: Computer Science and engineering.

**SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR**

**COMPLETION OF DEGREE**

**BACHELOR OF COMPUTER SCIENCE**

Submitted To: Submitted by:

Naman Singhal (1845010039)

SHRI RAM MURTI SMARAK COLLEGE OF ENGINEERING, TECHNOLOGY &RESEARCH, BAREILLY



**CERTIFICATE**

This is to certify that the project report entitled “**Restaurant Review Analysis**”is a bonafide record of the project work done by the student:

**Naman Singhal (1845010039)**

In partial fulfilment of the requirement for the 5th semester of COMPUTER SCIENCE & ENGINEERINGin **SHRI RAM MURTI SMARAK COLLEGE OF ENGINEERING, TECHNOLOGY &RESEARCH, BAREILLY**

**(Mini Project Incharge) (Mini Project Guide)**

**----- -------**

**(Mini Project Incharge) (Mini Project Guide)**

**ACKNOWLEDGEMENT**

I am pleased to present this Project report entitled Restaurant Review Analysis. It is indeed a great pleasure and a moment of immense satisfaction for me to express my sense of profound gratitude and indebtedness towards my guide **Aditya sir** whose enthusiasm is the source of inspiration for me. I am extremely thankful for the guidance and untiring attention, which he bestowed on me right from the beginning. His valuable and timely suggestions at crucial stages and above all his constant encouragement have made it possible for me to achieve this work. Would also like to give our sincere thanks to **Er.Anubha Dhaka (Mini Project Incharge) Computer Sc.& Engg. Dept.SRMSCET&R, BAREILLY** for necessary help and providing me the required facilities for completion of this project report. I would like to thank the entire Teaching staﬀ who are directly or indirectly involved in the various data collection and software assistance to bring forward this seminar report. I express my deep sense of gratitude towards my parents for their sustained cooperation and wishes, which have been a prime source of inspiration to take this project work to its end without any hurdles. Last but not the least, I would like to thank all our B.Tech. Colleagues for their co-operation and useful suggestion and all those who have directly or indirectly helped me in completion of this project work.

**Naman Singhal (1845010039)**

**PREFACE**

Restaurants nowadays prefer taking online orders. It not only helps in getting effective customer feedback but also useful for managing orders easily. We are moving towards an automated and digital world. Having a significant online presence is necessary for any restaurant to be successful and prosperous.

Getting customer feedback and analyzing them in an effective manner makes the difference. This study analyses the restaurant reviews and presents useful information that the ratings do not consider or overlook. Combined research is done using two different datasets of restaurant reviews in this paper.

Machine learning algorithms like Naïve Bayes and Logistic regression is used for first classifying the reviews in proper aspects then performing sentiment analysis on them. Summarization is done using gensim and results are displayed using effective visualization techniques.

Future work is also discussed so that an efficient analysis system can be developed utilizing the potential of reviews.

Keywords- Support Vector Machine (SVM), Naive Bayes classifier, Sentiment Analysis, Topic Modelling, aspect classification.

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **S.N.** | **Title** | **Page No.** |
| **1.** | Chapter 1: Introduction About the Project | 1 |
| **2.** | Chapter 2: Profile of Organization | 2 |
| **3.** | Chapter 3: System Analysis | 3-4 |
| **4.** | Chapter 4: Module design | 5 |
| **5.** | Chapter 5: Data flow diagram | 6 |
| **6.** | Chapter 6: Software Environment | 7-11 |
| **7.** | Chapter 7: Testing Introduction | 12-15 |
| **8.** | Chapter 8: Implementation | 16-17 |
| **9.** | Chapter 9: Screen Shots | 18-19 |
| **10.** | Chapter 10: Conclusion | 20 |
| **11.** | References | 21 |

**Chapter 1**

**INTRODUCTION ABOUT THE PROJECT**

Goal of this project is to learn predicting whether restaurant review is positive or negative. A great restaurant review can point us toward our favorite spot. Some reviews are positive, but are so vague that you question their legitimacy. Some might have helpful information, but are so poorly written they’re unintelligible. And other submissions read more like a personal rant than a restaurant review.

The purpose of this analysis is to build a prediction model to predict whether a review on the restaurant is positive or negative. To do so, we will work on Restaurant Review dataset, we will load it into predictive algorithms Multinomial Naive Bayes, Bernoulli Naive Bayes and Logistic Regression. In the end, we hope to find a "best" model for predicting the review's sentiment.

**Natural language processing (NLP)** is an area of computer science and artificial intelligence concerned with the interactions between computers and human (natural) languages, in particular how to program computers to process and analyze large amounts of natural language data. It is the branch of machine learning which is about analyzing any text and handling predictive analysis.

[Scikit-learn](https://tutorialspoint.dev/slugresolver/learning-model-building-scikit-learn-python-machine-learning-library/" \t "_blank) is a free software machine learning library for Python programming language. Scikit-learn is largely written in Python, with some core algorithms written in Cython to achieve performance. Cython is a superset of the Python programming language, designed to give C-like performance with code that is written mostly in Python.

**Ste[s invlolved:-**

**Step 1:** Import dataset with setting delimiter as ‘ ’ as columns are separated as tab space. Reviews and their category(0 or 1) are not separated by any other symbol but with tab space as most of the other symbols are is the review (like $ for price, ….!, etc) and the algorithm might use them as delimiter, which will lead to strange behavior (like errors, weird output) in output.

**Step 2:** Text Cleaning or Preprocessing

**Remove Punctuations, Numbers**: Punctuations, Numbers doesn’t help much in processing the given text, if included, they will just increase the size of bag of words that we will create as last step and decrease the efficiency of algorithm.

**Convert each word into its lower case**: For example, it useless to have same words in different cases (eg ‘good’ and ‘GOOD’).

1

**Chapter 2**

**Profile of organization**

Getting a job is as difficult as beating the crowd because being in the corporate world demands a lot from the applicant because of which the applicants are putting their best, which results in the increment of difficulty level. You can see each and every thing is connected but the solution of this problem is either spending years to reach to a desired position or come to Ducat. At Ducat we provide the entire necessary computer training which helps the newbie’s and also the experienced workers so that they can achieve better recognition in this competitive world.

Like other educational and training industry at Ducat you will be offered varieties programs but the instructors makes the difference and make Ducat stand out from others. Ducat contributes a lot to the knowledge of its trainees and we try our level hard to contribute the best to increase our trainee’s ability so that they stand out from others and whatever they contribute to the corporate world automatically becomes productive. Not only the fresher but also the corporates who are not able to deal with the rising technology and software are also helped here. Our motto is to deliver the best services to you and that is why we have taken the customized approach because we do not want you to compromise with your education.

It is not necessary that you have to leave your job in order to make-up with us. You can contact our experts and can get the best result. To serve you we are always at your service, you can contact us as and when you get time and clear your queries.

Ducat provides the best available programs which helps in enhancing the technical skills which seems to be beneficial for all the applicants.

**Software Development:** Ducat provide the best and latest IT software training which helps all the fresher and the cooperates to understand well and give them the knowledge to go hand in hand with the latest technologies. This does not only helps the companies but also increases the self-level to deal with all the necessary software.

**Instructor led campus:** Ducat helps all the new instructors to get the best exposure to show their talent in right way.

**Workshops and Placement Service:** At Ducat, workshops are held to increase the understanding level because theoretical values are always not enough and workshops helps in getting the practical knowledge which results in better understanding. As everything leads to the placement because if the institute does not provide placement services then it is ultimately bad for the applicants but we provide the best placement services and for that we give our best to give you the best.

2

**Chapter 3**

**System Analysis**

FEASIBILITY STUDY:

Feasibility study is made to see if the project on completion will serve the purpose of the organization for the amount of work, effort and the time that spend on it. Feasibility study lets the developer foresee the future of the project and the usefulness. Thus when a new application is proposed it normally goes through a feasibility study before it is approved for development. The document provide the feasibility of the project that is being designed and lists various areas that were considered very carefully during the feasibility study of this project such as Technical, Economic and Operational feasibilities. The following are its features:

1. **TECHNICAL FEASIBILITY:-**

The system must be evaluated from the technical point of view first. The assessment of this feasibility must be based on an outline design of the system requirement in the terms of input, output, programs and procedures. Having identified an outline system, the investigation must go on to suggest the type of equipment, required method developing the system, of running the system once it has been designed.

Technical issues raised during the investigation are:

* Does the existing technology sufficient for the suggested one?
* Can the system expand if developed?

The project should be developed such that the necessary functions and performance are achieved within the constraints. The project is developed within latest technology. Through the technology may become obsolete after some period of time, due to the fact that never version of same software supports older versions, the system may still be used. So there are minimal constraints involved with this project. The system has been developed using Java the project is techni-

3

cally feasible for development.

1. **ECONOMIC FEASIBILITY:-**

The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on project, which will give best, return at the earliest. One of the factors, which affect the development of a new system, is the cost it would require.

The following are some of the important financial questions asked during preliminary investigation:

* The costs conduct a full system investigation.
* The cost of the hardware and software.
* The benefits in the form of reduced costs or fewer costly errors.

Since the system is developed as part of project work, there is no manual cost to spend for the proposed system. Also all the resources are already available, it give an indication of the system is economically possible for development.

**3. BEHAVIORAL FEASIBILITY:-**

This includes the following questions:

* Is there sufficient support for the users?
* Will the proposed system cause harm?

The project would be beneficial because it satisfies the objectives when developed and installed. All behavioral aspects are considered carefully and conclude that the project is behaviorally feasible.

4

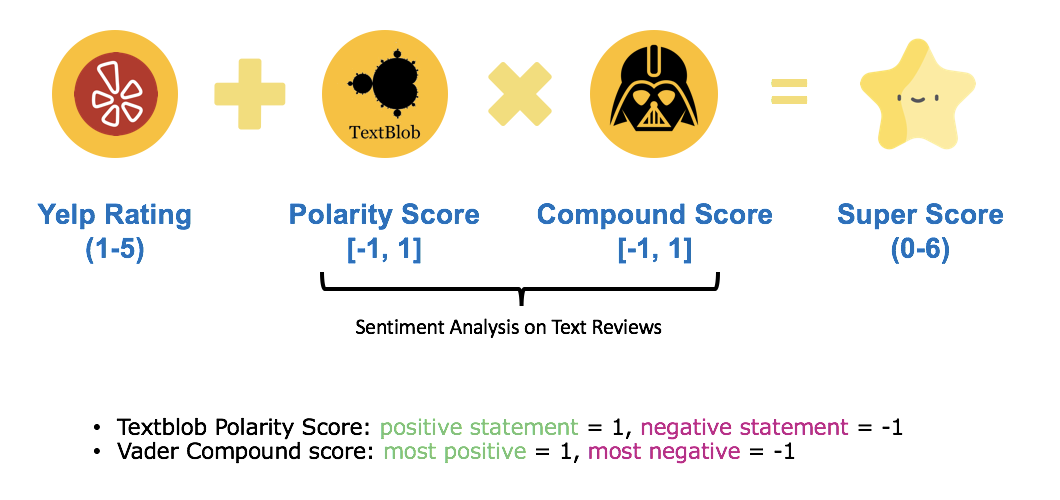
**Chapter 4**

**Module Design**

**Modules:**

1. **Login :** In this module HR enter the User id and password is checked andonly valid user id and password will get entry into search zone. This is a security feature to avoid entry of unauhorized users.
2. **Reset:** Through this HR can enter its start-up procedure as if you had turned the login wrong and then login again.
3. **Submit:** Through this HR can automatically submits a subject to a particular process or to the authority to present review consideration.
4. **Back:** Back button allows an end user to navigate to previously viewed web page.

**Restaurant Review Analysis:-**

****

5

**Chapter 5**

**Data Flow Diagram**

The Data flow Diagram shows the flow of data. It is generally made of symbols given below:

(1) A **square** shows the Entity: -

(2) A **Circle** shows the Process: -

(3) An **open Ended Rectangle** shows the data store: -

(4) An **arrow** shows the data flow:-

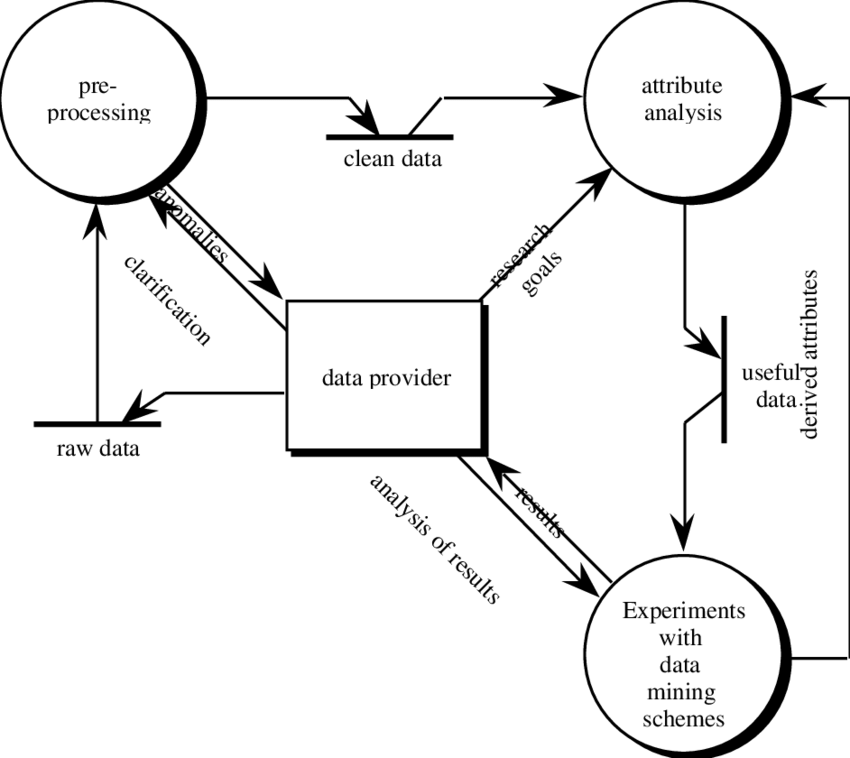
The DFD can be up to several levels.

The 0 level DFD states the flow of data in the system as seen from the outward in each module.

The first level DFD show more detail, about the single process of the 0 level DFD.

The second level DFD can show even more details and so on.

**Context Level DFD :-**



6

**Chapter -6**

**SOFTWARE ENVIRONMENT**

**Python:-**

Python is one of those rare languages which can claim to be both simple and powerful. You will find yourself pleasantly surprised to see how easy it is to concentrate on the solution to the problem rather than the syntax and structure of the language you are programming in.

The official introduction to Python is:

Python is an easy to learn, powerful programming language. It has efficient high-level data structures and a simple but effective approach to object-oriented programming. Python's elegant syntax and dynamic typing, together with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas on most platforms.

**Story behind the name:-**

## Guido van Rossum, the creator of the Python language, named the language after the BBC show "Monty Python's Flying Circus". He doesn't particularly like snakes that kill animals for food by winding their long bodies around them and crushing them.

## Features of Python:-

**Simple:-**

Python is a simple and minimalistic language. Reading a good Python program feels almost like reading English, although very strict English! This pseudo-code nature of Python is one of its greatest strengths. It allows you to concentrate on the solution to the problem rather than the language itself.

**Easy to learn:-**

As you will see, Python is extremely easy to get started with. Python has an extraordinarily simple syntax, as already mentioned.

**Free and Open Source:-**

Python is an example of a  FLOSS (Free/Libré and Open Source Software). In simple terms, you can freely distribute copies of this software, read its source code, make changes to it, and use pieces of it in new free programs. FLOSS is based on the concept of a community which shares knowledge. This is one of the reasons why Python is so good - it has been created and is constantly improved by a community who just want to see a better Python.

7

**High level language:-**

When you write programs in Python, you never need to bother about the low-level details such as managing the memory used by your program, etc.

**Object-oriented:-**

Python supports procedure-oriented programming as well as object-oriented programming. In procedure-oriented languages, the program is built around procedures or functions which are nothing but reusable pieces of programs. In object-oriented languages, the program is built around objects which combine data and functionality. Python has a very powerful but simplistic way of doing OOP, especially when compared to big languages like C++ or Java.

**2.SOFTWARE AND HARDWARE SPECIFICATION:-**

**Hardware Specification:-**

**Microprocessor : -**

Microprocessor is a controlling unit of a micro-computer, fabricated on a small chip capable of performing ALU (Arithmetic Logical Unit) operations and communicating with the other devices connected to it.

Microprocessor consists of an ALU, register array, and a control unit. ALU performs arithmetical and logical operations on the data received from the memory or an input device. Register array consists of registers identified by letters like B, C, D, E, H, L and accumulator. The control unit controls the flow of data and instructions within the computer.

**RAM :-**

RAM (Random Access Memory) is the internal memory of the CPU for storing data, program, and program result. It is a read/write memory which stores data until the machine is working. As soon as the machine is switched off, data is erased.

Access time in RAM is independent of the address, that is, each storage location inside the memory is as easy to reach as other locations and takes the same amount of time. Data in the RAM can be accessed randomly but it is very expensive.

**Hard Disk :-**

Hard disk drive is made up of a series of circular disks called **platters** arranged one over the other almost ½ inches apart around a **spindle**. Disks are made of non-magnetic material like aluminum alloy and coated with 10-20 nm of magnetic material.

8

Standard diameter of these disks is 14 inches and they rotate with speeds varying from 4200 rpm (rotations per minute) for personal computers to 15000 rpm for servers. Data is stored by magnetizing or demagnetizing the magnetic coating. A magnetic reader arm is used to read data from and write data to the disks. A typical modern HDD has capacity in terabytes (TB).

**Software Specification:-**

**Jupyter:-**

Project Jupyter is a comprehensive software suite for interactive computing, that includes various packages such as Jupyter Notebook, QtConsole, nbviewer, JupyterLab. This tutorial gives you an exhaustive knowledge on Project Jupyter. By the end of this tutorial, This tutorial is useful for everyone who wants to learn and practice data science libraries of Python/R etcThis is not a tutorial to teach Python programming.If you are beginner to Python and other related concepts, we suggest you to pick tutorials based on these, before you start your journey with Jupyter.

**Idle:-**

IDLE (Integrated Development and Learning Environment) is an integrated development environment (IDE) for Python. The Python installer for Windows contains the IDLE module by default.

IDLE is not available by default in Python distributions for Linux. It needs to be installed using the respective package managers. For example, in case of Ubuntu.

IDLE can be used to execute a single statement just like Python Shell and also to create, modify and execute Python scripts. IDLE provides a fully-featured text editor to create Python scripts that includes features like syntax highlighting, autocompletion and smart indent. It also has a debugger with stepping and breakpoints features.

**Anaconda prompt:-**

Anaconda command prompt is just like command prompt, but it makes sure that you are able to use anaconda and conda commands from the prompt, without having to change directories or your path.

When you start Anaconda command prompt, you'll notice that it adds/("prepends") a bunch of locations to your PATH. These locations contain commands and scripts that you can run. So as long as you're in the Anaconda command prompt, you know you can use these commands.

During the installation of Anaconda there is a choice to add these to the PATH by default, and if checked you can also use these commands on the regular command prompt. But the anaconda prompt will always work.

As far as updating conda, if it doesn't work in command prompt, you can do

9

# 3 Project Architecture and Platform:-

# Front End:-

**Tkinter:-**

**Tkinter** is actually an inbuilt **Python** module used to create simple **GUI** apps. It is the most commonly used module for **GUI** apps in the **Python**.Tkinter also offers access to the geometric configuration of the widgets which can organize the widgets in the parent windows.

There are a number of widgets which you can put in your tkinter application. Some of the major widgets are explained below:

**1.Button:-**

To add a button in your application, this widget is used. The general syntax is:-

W=Button(master, option=value)

**2. CheckButton: -**

To select any number of options by displaying a number of options to a user as toggle buttons. The general syntax is:

W= CheckButton(master, option=value)

**3.Label :** -

It refers to the display box where you can put any text or image which can be updated any time as per the code. The general syntax is:

W=Label(master, option=value)

**Back End:-**

**1:*CountVectorizer:*** -

The **CountVectorizer** provides a simple way to both tokenize a collection of text documents and build a vocabulary of known words, but also to encode new documents using that vocabulary. ... Call the transform() function on one or more documents as needed to encode each as a vector.

**2: *Naive Bayes:* -**

The calculations of the probabilities for each class are simplified to make their calculations tractable. Rather than attempting to calculate the probabilities of each attribute value, they are assumed to be conditionally independent given the class value.

1. **pyinstaller:-**

Abstracts details from the user by finding all their dependencies and bundling them together. Users won’t even know they’re running a Python project because the Python Interpreter itself is bundled into your application. Goodbye complicated installation instructions!PyInstaller performs this amazing feat by [introspecting](https://en.wikipedia.org/wiki/Type_introspection) given Python code, detecting dependencies, and then packaging them into a suitable format depending on users Operating System.

10

In addition, PyInstaller can create executables for Windows, Linux, or macOS. This means Windows users will get a .exe, Linux users get a regular executable, and macOS users get a .app bundle. There are some caveats to this.

**4:*Stop-words:* -**

A **stop word** is a commonly used **word** (such as “the”, “a”, “an”, “in”) that a search engine has been programmed to ignore, both when indexing entries for searching and when retrieving them as the result of a search query.

**5:*MultinomailNB:-***

The multinomial Naive Bayes classifier is suitable for classification with discrete features (e.g., word counts for text classification). The multinomial distribution normally requires integer feature counts. However, in practice, fractional counts such as tf-idf may also work.

**6:*todense*: -**

Simply returns dense data representation of NDFrame.

**7:*Feature vector*: -**

A **feature vector** is just a **vector** that contains information describing an object's important characteristics.

11

Software Testing is the process of executing software in a controlled manner, in order to answer the question - Does the software behave as specified?. Software testing is often used in association with the terms verification and validation. Validation is the checking or testing of items, includes software, for conformance and consistency with an associated specification. Software testing is just one kind of verification, which also uses techniques such as reviews, analysis, inspections, and walkthroughs. Validation is the process of checking that what has been specified is what the user actually wanted.

**Validation : Are we doing the right job?**

**Verification : Are we doing the job right?**

Software testing should not be confused with debugging. Debugging is the process of analyzing and localizing bugs when software does not behave as expected. Although the identification of some bugs will be obvious from playing with the software, a methodical approach to software testing is a much more thorough means for identifying bugs. Debugging is therefore an activity which supports testing, but cannot replace testing.

Other activities which are often associated with software testing are static analysis and dynamic analysis. Static analysis investigates the source code of software, looking for problems and gathering metrics without actually executing the code. Dynamic analysis looks at the behavior of software while it is executing, to provide information such as execution traces, timing profiles, and test coverage information.

Testing is a set of activity that can be planned in advanced and conducted systematically. Testing begins at the module level and work towards the integration of entire computers based system.

**Chapter -7**

**TESTING INTRODUCTION**

12

They are-

* Testing is a process of executing a program with the intend of findingan error.
* A good test case is one that has high possibility of finding an undiscovered error.
* A successful test is one that uncovers an undiscovered error.

 If a testing is conducted successfully according to the objectives as stated above, it would uncovered errors in the software also testing demonstrate that the software function appear to be working according to the specification, that performance requirement appear to have been met.

There are three ways to test program.

* For correctness
* For implementation efficiency
* For computational complexity

Test for correctness are supposed to verify that a program does exactly what it was designed to do. This is much more difficult than it may at first appear, especially for large programs.

**1. TEST PLAN:-**

A test plan implies a series of desired course of action to be followed in accomplishing various testing methods. The Test Plan acts as a blue print for the action that is to be followed. The software engineers create a computer program, its documentation and related data structures. The software developers is always responsible for testing the individual units of the programs, ensuring that each performs the function for which it was designed. There is an independent test group (ITG) which is to remove the inherent problems associated with letting the builder to test the thing that has been built. The specific objectives of testing should be stated in measurable terms. So that the mean time to failure, the cost to find and fix the defects, remaining defect density or frequency of occurrence and test work-hours per regression test all should be stated within the test plan.

The levels of testing include:

13

* Unit testing
* Integration Testing
* Data validation Testing
* Output Testing

**A. UNIT TESTING:-**

Unit testing focuses verification effort on the smallest unit of software design – the software component or module. Using the component level design description as a guide, important control paths are tested to uncover errors within the boundary of the module. The relative complexity of tests and uncovered scope established for unit testing. The unit testing is white-box oriented, and step can be conducted in parallel for multiple components. The modular interface is tested to ensure that information properly flows into and out of the program unit under test. The local data structure is examined to ensure that data stored temporarily maintains its integrity during all steps in an algorithm’s execution. Boundary conditions are tested to ensure that all statements in a module have been executed at least once. Finally, all error handling paths are tested.

Tests of data flow across a module interface are required before any other test is initiated. If data do not enter and exit properly, all other tests are moot. Selective testing of execution paths is an essential task during the unit test. Good design dictates that error conditions be anticipated and error handling paths set up to reroute or cleanly terminate processing when an error does occur. Boundary testing is the last task of unit testing step. Software often fails at its boundaries.

Unit testing was done in Sell-Soft System by treating each module as separate entity and testing each one of them with a wide spectrum of test inputs. Some flaws in the internal logic of the modules were found and were rectified

**B. INTEGRATION TESTING:-**

Integration testing is systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing. The objective is to take unit tested components and build a program structure that has been dictated by design. The entire program is tested as whole. Correction is difficult because isolation of causes is complicated by vast expanse of entire program.

14

After unit testing in Sell-Soft System all the modules were integrated to test for any inconsistencies in the interfaces. Moreover differences in program structures were removed and a unique program structure was evolved.

**C. VALIDATION TESTING OR SYSTEM TESTING:-**

This is the final step in testing. In this the entire system was tested as a whole with all forms, code, modules and class modules.

Black Box testing method focuses on the functional requirements of the software. That is, Black Box testing enables the software engineer to derive sets of input conditions that will fully exercise all functional requirements for a program.

Black Box testing attempts to find errors in the following categories; incorrect or missing functions, interface errors, errors in data structures or external data access, performance errors and initialization errors and termination errors.

**D. OUTPUT TESTING OR USER ACCEPTANCE TESTING:-**

The system considered is tested for user acceptance; here it should satisfy the firm’s need. The software should keep in touch with perspective system; user at the time of developing and making changes whenever required. This done with respect to the following points

* Input Screen Designs,
* Output Screen Designs,
* Online message to guide the user and the like.

The above testing is done taking various kinds of test data. Preparation of test data plays a vital role in the system testing. After preparing the test data, the system under study is tested using that test data. While testing the system by which test data errors are again uncovered and corrected by using above testing steps and corrections are also noted for future use.

**Validation Checking:-**

At the culmination of integration testing, software is completely assembled as a package; interfacing errors have been uncovered and corrected, and a final series of software test-validation checks may begin. Validation can be defined in many ways, but a simple definition (Albeit Harsh) is that validation succeeds a test procedure defines specific test cases that will be used in attempt to uncover errors in conformative

15

**Chapter-8**

**IMPLEMENTATION**

Implementation is the stage of the project where the theoretical design is turned into a working system. It can be considered to be the most crucial stage in achieving a successful new system gaining the users confidence that the new system will work and will be effective and accurate. It is primarily concerned with user training and documentation. Conversion usually takes place about the same time the user is being trained or later. Implementation simply means convening a new system design into operation, which is the process of converting a new revised system design into an operational one.

Implementation is the stage of the project where the theoretical design is tuned into a working system. At this stage the main work load, the greatest upheaval and the major impact on the existing system shifts to the user department. If the implementation is not carefully planned and controlled it can create chaos and confusion.

Implementation includes all those activities that take place to convert from the existing system to the new system. The new system may be a totally new, replacing an existing manual or automated system or it may be a modification to an existing system. Proper implementation is essential to provide a reliable system to meet organization requirements. The process of putting the developed system in actual use is called system implementation. This includes all those activities that take place to convert from the old system to the new system. The system can be implemented only after through testing is done and if it is found to be working according to the specifications. The system personnel check the feasibility of the system. The more complex the system being implemented, the more involved will be the system analysis and design effort required to implement the three main aspects: education and training, system testing and changeover. The implementation state involves the following tasks:-

16

* Careful planning.
* Investigation of system and constraints.
* Design of methods to achieve the changeover.
* Training of the staff in the changeover phase.

1. **Implementation procedure:-**

Implementation of software refers to the final installation of the package in its real environment, to the satisfaction of the intended uses and the operation of the system. In many organizations someone who will not be operating it, will commission the software development project. In the initial stage people doubt about the software but we have to ensure that the resistance does not build up, as one has to make sure that

* The active user must be aware of the benefits of using the new system.
* Their confidence in the software is built up.
* Proper guidance is imparted to the user so that he is comfortable in using the application.

Before going ahead and viewing the system, the user must know that for viewing the result, the server program should be running in the server. If the server object is not up running on the server, the actual process won’t take place.

**2.User Training:-**

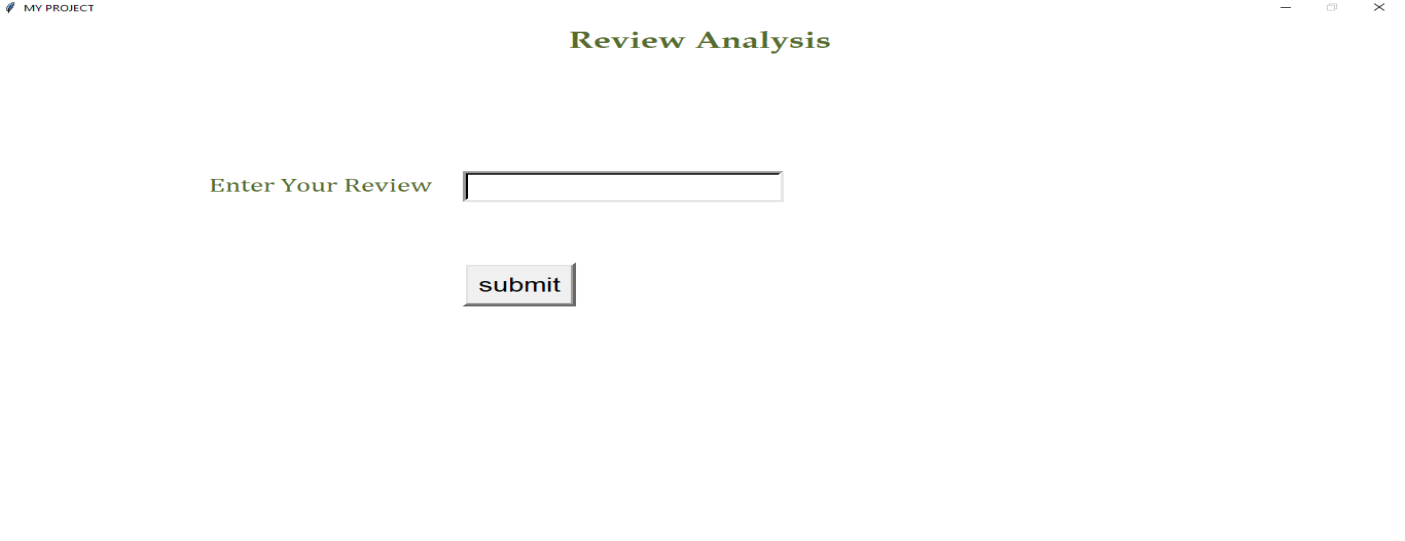
User training is designed to prepare the user for testing and converting the system. To achieve the objective and benefits expected from computer based system, it is essential for the people who will be involved to be confident of their role in the new system. As system becomes more complex, the need for training is more important. By user training the user comes to know how to enter data, respond to error messages, interrogate the database and call up routine that will produce reports and perform other necessary functions.

17

**Chapter-9**

**SCREEN SHOTS**

**Home Page**





****

18

****

****

19

**Chapter- 10**

**Conclusion**

The *conclusion* of review should establish a sense of closure or completeness that leaves the customer with lingering thoughts about the restaurant. It is formal examination by people in authority. This is usually done in order to see whether it can be improved or corrected. Reviews are the foremost thing, which visitors consider while finalizing the restaurant. The better the review, the higher the chances they will book a seat in that restaurant. Reviews helps us to improve the food quality, arrangements, service etc

20

**References**

1. Basant, A., M. Namita, B. Pooja, and Sonal Garg. "Sentiment analysis using common-sense and context information." (2015).
2. Bykh, Serhiy, and Detmar Meurers. "Native Language Identification using Recurring n-grams– Investigating Abstraction and Domain Dependence-grams– Investigating Abstraction and Domain Dependence." In Proceedings of COLING 2012, pp. 425-440 (2012).
3. Westerski, Adam, Carlos Angel Iglesias Fernandez, and Fernando Tapia Rico. "Linked opinions:Describing sentiments on the structured web of data." (2011).
4. Yi, Jeonghee, and Neel Sundaresan. "A classifier for semi-structured documents." In KDD, pp. 340-344. (2000).
5. Moghaddam, Samaneh, and Martin Ester. "Opinion digger: an unsupervised opinion miner from unstructured product reviews." In Proceedings of the 19th ACM international conference on Information and knowledge management, pp. 1825-1828. ACM, (2010).
6. Wiegand, Michael, Alexandra Balahur, Benjamin Roth, Dietrich Klakow, and Andrés Montoyo. "A survey on the role of negation in sentiment analysis." In Proceedings of the workshop on negation and speculation in natural language processing, pp. 60-68. (2010).
7. Flekova, Lucie, Daniel PreoţiucPietro, and Eugen Ruppert. "Analysing domain suitability of a sentiment lexicon by identifying distributionally bipolar words." In Proceedings of the 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, pp. 77-84. (2015).
8. Asghar, Muhammad Zubair, Aurangzeb Khan, Shakeel Ahmad, and Fazal Masud Kundi. "A review of feature extraction in sentiment analysis." Journal of Basic and Applied Scientific Research 4, no. 3 (2014): 181-186.
9. Lin, Yuming, Tao Zhu, Hao Wu, Jingwei Zhang, Xiaoling Wang, and Aoying Zhou. "Towards online antiopinion spam: Spotting fake reviews from the review sequence." In 2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014), pp. 261-264. IEEE, (2014).
10. Gimpel, Kevin, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A. Smith. Part-of-speech tagging for twitter:Annotation, features, and experiments. Carnegie-Mellon Univ Pittsburgh Pa School of Computer Science, (2010).
11. Cavnar, William B., and John M. Trenkle. "N-gram-based text categorization." In Proceedings of SDAIR-94, 3rd annual symposium on document analysis and information retrieval, vol. 161175. (1994).
12. Aggarwal, Charu C. "Text Summarization." In Machine Learning for Text, pp. 361-380. Springer, Cham, (2018).

21