##### image1.jpg

**CSV FILE VALIDATION**

##### **A PROJECT REPORT**

**(Phase-I)**

###### ***Submitted by***

**MEENA S(1517106045)**

**MOUNISHA R(1517106048)**

**MUSKAN SARAF(1517106051)**

***in partial fulfillment for the award of the degree***

***of***

**BACHELOR OF TECHNOLOGY**

**I INFORMATION TECHNOLOGY**

**SONA COLLEGE OF TECHNOLOGY,** **SALEM-5**

**(Autonomous)**

**OCTOBER 2019**

**ANNA UNIVERSITY : CHENNAI 600 025**

**BONAFIDE CERTIFICATE**

Certified that this project report **“CSV FILE VALIDATION”** is the bonafide work of **“MEENA S(1517106048), MOUNISHA R(1517106048), MUSKAN SARAF(1517106051)”** who carried out the project work under my supervision.

**SIGNATURE SIGNATURE**

|  |  |
| --- | --- |
| Dr.J.Akilandeswari | Dr.J.Senthilkumar |
| **HEAD OF THE DEPARTMENT** | **SUPERVISOR** |
| Professor | Professor |
| Department of Information Technology | Department of Information Technology |
| Sona College of Technology, | Sona College of Technology, |
| Salem-636 005. | Salem-636 005. |

Submitted for mini Project viva voce examination held on .....................................

**INTERNAL EXAMINER EXTERNAL EXAMINER**

|  |
| --- |
|  |

**ABSTRACT**

CSV looks easy, but it can be hard to make a CSV file that other people can read easily. Invalid CSV files create challenges for those building data pipelines. Pipelines value consistency, predictability, and testability as they ensure uninterrupted operation from source to a target destination. Thus in this project we will get a csv file from the user with name, age, phone number, email id and create the records, validation of data, filter the duplicate records and check what has to be used in the date type and for null value. Through this the CSV file could become more definite

**ACKNOWLEDGEMENT**

First and foremost, we thank to **power of almighty** for showing us inner peace and for all blessings. Special gratitude to our parents, for showing their support and love always.

We express our sincere thanks to Chairman **Sri.C.Valliappa** andPrincipal  **Dr.S.R.R.Senthil kumar**  for providing adequate facilities to complete the project.

We are immensely grateful to Head of Information Technology, **Dr.J.Akilandeswari** for the continues encouragement to complete the project.

We express our heartfelt thanks to our project supervisor  ***Mr.S.David Samuel Azaiya*** for his valuable guidance and fruitful discussions throughout the course of the project work.

We feel proud in sharing this success with all our staff members and friends who helped directly or indirectly in completing this project successfully.

TABLE OF CONTENTS

**CHAPTER NO. TITLE PAGE NO.**

ABSTRACT iii LIST OF FIGURES vii

|  |  |  |
| --- | --- | --- |
| **1** | **Introduction** | 1 |
|  | 1.1 Existing System | 1 |
|  | 1.2 Proposed System | 1 |
|  | 1.3 Benefits  1.4 Features | 1  1 |
| **2** | **Requirements** | 2 |
|  | 2.1.1 Software Requirements | 2 |
|  | 2.1.2 Hardware Requirements | 3 |
| **3** | **System Design**  3.1 Description of the Program | 4  5 |
| **4** | **System Analysis** | 9 |
|  | 4.1 Various divisions in project | 13 |
|  |  |  |
|  | 4.2 User Characteristics | 13 |
|  | 4.3 Feasibility Study | 14 |
| **5** | **Implementation** | 16 |
|  | 5.1 Usability Aspect | 16 |
|  | 5.2 Technical Aspect | 16 |
|  | 5.2.1 Servers: Apache Tomcat to develop the product | 17 |
|  | 5.2.2 Database | 17 |
|  | 5.3 Module Implementation in Push Architecture | 18 |
|  | 5.3.1 HTML Codes | 19 |
| **6** | **Testing** | 21 |
|  | 6.1 Levels of Testing | 21 |
|  | 6.1.1 Unit Testing | 21 |
|  | 6.1.3 System Testing  6.2 Test Cases | 21  22 |
| **7** | **Snapshots** | 25 |
|  | 7.1 Snapshot 1 | 25 |
|  | 7.2 Snapshot 2 | 26 |
|  | 7.3 Snapshot 3 | 26 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

|  |  |
| --- | --- |
| **Conclusion And Future Work** |  |
| **References** |  |

29

30

|  |  |
| --- | --- |
|  |  |
|  |  |

**LIST OF FIGURES**

**FIGURE NO. TITLE PG NO.**

|  |  |  |
| --- | --- | --- |
| 3.1  3.2  3.3  3.4  6.1  6.2  6.3 | Data Flow Diagram  Use case Diagram  ER-Diagram (Architecture)  ER-Diagram  Snapshot :1  Snapshot 2:  Snapshot 3: | 6  7  8  9  25  26  26 |

**CHAPTER 1**

**INTRODUCTION**

CSV is a simple file format used to store tabular data, such as a spreadsheet or database. They mostly use the A CSV is a comma-separated values file, which allows data to be saved in a tabular format.CSV look like a garden-variety spreadsheet but with a csv extension. CSV files can be used with most any spreadsheet program, such as Microsoft excel or google spreadsheet. They differ from other spreadsheet file types because you can not save cell, column, or row. Also you cannot save formulas in this format.

* 1. **Existing system**

It is possible to batch update existing Target process entities (User Stories, Tasks, Bugs, Features, Epics, Portfolio Epics, Requests) from CSV files. You can batch change states, business values, custom field values, add or change assigned tags, estimated effort, planned dates, assign users and teams. This article describes how to perform the import.

* 1. **proposed system**

Loading data through the web browser is recommended for smaller tables with simple relationships between them. This method is recommended for small, one time data loads. Using this method, the data schema is created for automatically.

* 1. **Benefits**
* CSV is human readable and easy to edit manually.
* CSV is simple to implement and parse.
* CSV is processed by almost all existing applications.
* CSV provides a straightforward information schema.
* CSV is considered to be standard format.
* CSV is compact.
  1. **Features**
* One line for each record.
* Comma separated fields.
* Space-characters adjacent to commas are ignored.
* Fields with in-built commas are separated by double quote characters.

**CHAPTER 2**

**SOFTWARE REQUIREMENT SPECIFICATION**

A software requirements specification (SRS) is a description of a software system to be developed. It lays out functional and non-functional requirements, and may include a set of use cases that describe user interactions that the software must provide.

**Characteristics of good SRS:**

* Complete.
* Consistent.
* Feasible.
* Modifiable.
* Unambiguous.
* Testable.

**2.1 Requirements**

**2.1.1 Software Requirements**:

**Software:**

Language: python

Libraries: nymph, pandas

Module: CSV

**2.2.2 Software Features:**

**Python:**

The so-called CSV (Comma Separated Values) format is the most common import and export format for spreadsheets and databases. CSV format was used for many years prior to attempts to describe the format in a standardized way in R1400. The lack of a well-defined standard means that subtle differences often exist in the data produced and consumed by different applications. These differences can make it annoying to process CSV files from multiple sources. Still, while the delimiters and quoting characters vary, the overall format is similar enough that it is possible to write a single module which can efficiently manipulate such data, hiding the details of reading and writing the data from the programmer.

The CSV module implements classes to read and write tabular data in CSV format. It allows programmers to say, “write this data in the format preferred by Excel,” or “read data from this file which was generated by Excel,” without knowing the precise details of the CSV format used by Excel. Programmers can also describe the CSV formats understood by other applications or define their own special-purpose CSV formats.

The CSV module’s reader and writer objects read and write sequences. Programmers can also read and write data in dictionary form using the and classes.

**Pandas:**

Pandas is one of the most popular Python libraries for Data Science and Analytics. I like to say it’s the “SQL of Python.” Why? Because pandas helps you to manage two-dimensional data tables in Python. Of course, it has many more features. In this pandas tutorial series, I’ll show you the most important (that is, the most often used) things that you have to know as an Analyst or a Data Scientist. This is the first episode and we will start from the basics.

**HTML:**

HTML stands for Hyper Text Markup Language, which is the most widely used language on Web to develop web pages. HTML was created by Berners-Lee in late 1991 but "HTML 2.0" was the first standard HTML specification which was published in 1995. HTML 4.01 was a major version of HTML and it was published in late 1999. Though HTML 4.01 version is widely used but currently we are having HTML-5 version.

**Hardware:**

Processer: Intel core i3

Memory: 4GB RAM

Storage: 194 GB

**CHAPTER 3**

**SYSTEM DESIGN**

A couple of the projects I’m involved with at the moment are at a stage where there’s some thinking going on around how to best provide CSV files for users. This has left me thinking about what options we actually have when it comes to designing a CSV file format.CSV is a very useful, but pretty mundane format. I suspect many of us don’t really think very much about how to organise our CSV files. It’s just a table, right? What decisions do we need to make?But there are actually quite a few different options we have that might make a specific CSV format more or less suited for specific audiences. So I thought I’d write down some of the options that occured to me. It might be useful input into both my current projects as well as future work on standard.

3.1 DESCRIPTION OF THE PROGRAM:

The employee switch diagram for employee payroll and leave management system is shown in the figure below. The input and output of this section is shown in the diagram, however number of details about the procedure of the employee payroll and leave management system is given here. Using this as a starting point, a logical DFD of the system is developed

**3.1 BLOCK DIAGRAM:**

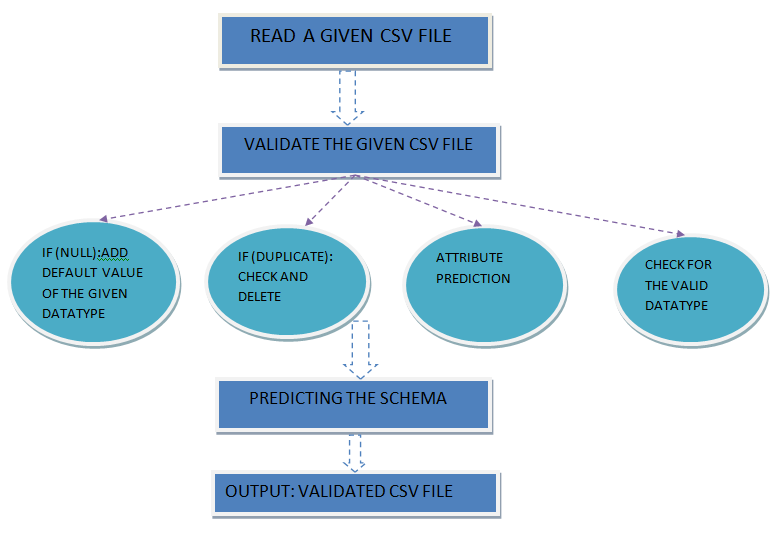
RAW DATA

APPLICATION UNDER TESTING

CSV FILE

VARIOUS USAGE OF DATASET

PROCESSED DATA



**CHAPTER 4**

**SYSTEM ANALYSIS**

Analysis is the process of breaking a complex topics or substance into smaller parts to gain a better understanding of it. Analysts in the field of engineering look at requirements, structures, mechanisms, and systems dimensions. Analysis is an exploratory activity.

Gathering requirements is the main attraction of the Analysis Phase. The process of gathering requirements is usually more than simply asking the users what they need and writing their answers down. Depending on the complexity of the application, the process for gathering requirements has a clearly defined process of its own. This process consists of a group of repeatable processes that utilize certain techniques to capture, document, communicate, and manage requirements.

**4.1 Various divisions in the project**

**4.2 User Characteristics**

**4.2.1 End Users**

* + No specific knowledge or skills are required from the end user.
* End user should have basic idea about computer operations and database.

**4.2.2 Administrator**

* Administrator must be having good knowledge of database management system.
* Administrator must be capable to manage user rights.
* Manual interfaces cannot be fully avoided. Documented proofs like data entry of employees etc. will have to be verified by the concerned management staff  before entering it into the computerized system.

**4.3 Feasibility Study**

Every project is feasible for given unlimited resources and infinitive time. Feasibility study is an evaluation of the proposed system regarding its workability, impact on the organization, ability to meet the user needs and effective use of resources. Thus when a new application is proposed it normally goes through a feasibility study before it is approved for development .Feasibility and risk analysis and related in many ways. If a project risk is great and feasibility of producing software is reduced. During the feasibility analysis in this project has been discussed below in the abovementioned topics.

* **Operational Feasibility:**

Feasibility of the working of the system after the installation in the organization as mentioned in the feasibility analysis.

* **Technical Feasibility:**

Technical feasibility is frequently the most difficult area to ensure this stage. It is essential that the process of analysis and definition to be conducted parallel to an assessment of the technical feasibility. The consideration that is normally associated with technical feasibility includes the resources availability of the Organization where the project is to be developed and implemented. By taking these facts into consideration before developing the resource availability at Retail Outlet of Hindustan Petroleum was observed. As very limited resources are required for this project hence this project is considered feasible for development.

* **Economic Feasibility:**

An evaluation of development cost is weighted against the ultimate income or benefits derived from the developed system. There was no need of extra hardware and software for development of this project. Hence this project has economically justified for development in this organization.

* **Motivational Feasibility:**

An evaluation of the probability that the organization is sufficient motivation to support the development and implementation of the application with necessary user participation, resources, training etc. The interest and support shown by the organization during the system study do not seem that the new system developed to have efficient support from the organization.

* **Schedule Feasibility:**

An evaluation of the time needed for the development of this project. The time schedule required for the development of this project is very important, since more development time effects machine time, costs and delays in the development of the other systems. So the project should be complete within affixed schedule time as far as the organization is concerned.

**CHAPTER 5**

**IMPLEMENTATION**

Implementation is the realization of an application, or execution of a plan, idea, model, design, specification, standard, algorithm, or policy. In other words, an implementation is a realization of a technical specification or algorithm as a program, software component, or other computer system through programming and deployment. Many implementations may exist for a given specification or standard.

**5.1 Usability Aspect**

**Python:**

The so-called CSV (Comma Separated Values) format is the most common import and export format for spreadsheets and databases. CSV format was used for many years prior to attempts to describe the format in a standardized way in R1400. The lack of a well-defined standard means that subtle differences often exist in the data produced and consumed by different applications. These differences can make it annoying to process CSV files from multiple sources. Still, while the delimiters and quoting characters vary, the overall format is similar enough that it is possible to write a single module which can efficiently manipulate such data, hiding the details of reading and writing the data from the programmer.

The CSV module implements classes to read and write tabular data in CSV format. It allows programmers to say, “write this data in the format preferred by Excel,” or “read data from this file which was generated by Excel,” without knowing the precise details of the CSV format used by Excel. Programmers can also describe the CSV formats understood by other applications or define their own special-purpose CSV formats.

The CSV module’s reader and writer objects read and write sequences. Programmers can also read and write data in dictionary form using the and classes.

**Pandas:**

Pandas is one of the most popular Python libraries for Data Science and Analytics. I like to say it’s the “SQL of Python.” Why? Because pandas helps you to manage two-dimensional data tables in Python. Of course, it has many more features. In this pandas tutorial series, I’ll show you the most important (that is, the most often used) things that you have to know as an Analyst or a Data Scientist. This is the first episode and we will start from the basics.

**HTML:**

HTML stands for Hyper Text Markup Language, which is the most widely used language on Web to develop web pages. HTML was created by Berners-Lee in late 1991 but "HTML 2.0" was the first standard HTML specification which was published in 1995. HTML 4.01 was a major version of HTML and it was published in late 1999. Though HTML 4.01 version is widely used but currently we are having HTML-5 version.

**PROGRAM:**

**HTML CODE:**

<!doctype html>

<html>

<style>

h1 {text-align: center;}

h1 {border-style: double;}

</style>

<body style="background-color:LightPink;">

<title>VALIDATION OF A CSV</title>

<h1>VALIDATION OF CSV FILE</h1>

<h2> You can upolad a csv file which you want to evaluate and use for your

project..</h2><h3> Here we will check for

<ul> <li> Null values</li>

<li> Fill null values</li>

<li>Data type pediction</li>

<li> Label encoding </li>

<li>Predict duplicate value</li>

<li>Remove duplicate values</li>

<li> OneHot encoding (if needed)</li>

<li> Feature scaling (if needed)</li>

and after all these is done u can get the csv which is

ready to use for your project</ul></h3>

<form action = "/login" method=post enctype=multipart/form-data>

<input id="csvFileInput" type="file" name="f1" accept=".csv"/>

<h3> Do you want to do OneHot Encoding and Feature Scaling?</h3>

<lable for ="s"> CHOOSE YOUR OPTION </lable>

<select name="s">

<option value ="Yes">YES </option>

<option value ="No">NO </option>

<input type=submit value=Upload>

<p><b>{{ans}}</b></p>

</form>

<p> FOR REFFRENCE</P>

<p><h3>Null Value :</h3> A NULL value in a table is a value in a

field that appears to be blank. It is very important to understand

that a NULL value is different than a zero value or a field that

contains spaces. And here we will try to fill the null value with

MODE value. The mode of a set of data values is the value that appears

most often.</p>

<p> <h3>Data Type Pediction</h3>

A data type, in programming, is a classification that specifies

which type of value a variable has and what type of mathematical,

relational or logical operations can be applied to it without causing

an error. A string, for example, is a data type that is used to

classify text and an integer is a data type used to classify

whole numbers. So here we also predict the data type for each column heading.</p>

<P><h3> Label Encoding</h3>

Label encoding is to convert the textual data form to numerical form.

It is an important pre-processing step for the structured dataset

in supervised learning. It helps us in giving different value for

each and every value. label encoding is important because our spyder

or jupyter notebook will not accept textual data so if the data is converted

then it will be easy for the user to get the output as required.</P>

<p><h3>Predict Duplicate Value</h3>

Predicting duplicate value is also important so that our dataset is so

clear with all needed or important information. Sometimes it happens

that we enter some data twice or many times so it is not nessaceary to

run the epochs all the time so by predicting and deleting it we are

making the csv with all short and important data.</p>

<p><h3>OneHot Encoding</h3>

One hot encoding is a process by which categorical variables are

converted into a form that could be provided to ML algorithms to do

a better job in prediction. The categorical value represents the

numerical value of the entry in the dataset. As the number of unique

entries increases, the categorical values also proportionally

increases. the categorical values start from 0 goes all the way

up to N-1 categories. 0 indicates non existent while 1 indicates

existent. </P>

<p><h3>Feature Scaling</h3>

Feature Scaling is a technique to standardize the independent

features present in the data in a fixed range. It is performed

during the data pre-processing. In data processing, it is

also known as data normalization and is generally performed

during the data preprocessing step. In this all the value of the

colunm are ranged between some values.</p>

</body>

</html>

**app.py code:**

from flask import Flask

import pandas as pd

from flask import request,render\_template

from werkzeug.utils import secure\_filename

import os

app=Flask(\_name\_)

@app.route('/', methods=['GET', 'POST'])

def upload\_file():

return render\_template("index.html")

@app.route('/login', methods=['GET', 'POST'])

def upload\_file1():

if request.method =="POST":

f=request.files["f1"]

option=request.form["s"]

basepath=os.path.dirname(\_file\_)

file\_path=os.path.join(basepath,"uploads",secure\_filename(f.filename))

f.save(file\_path)

data=pd.read\_csv(file\_path)

shape=data.shape

coloum\_null=list(data.isnull().any())

coloum\_name=list(data)

for i in range(len(coloum\_null)):

if(coloum\_null[i]):

data[coloum\_name[i]].fillna(data[coloum\_name[i]].mode()[0],inplace = True)

dl=[]

for j in range(len(coloum\_name)):

dl.append((type(data[coloum\_name[j]][1])))

datatype={coloum\_name[k]:dl[k] for k in range(0,len(dl))}

dataarray=data.to\_numpy()

droplist=[]

for k in range(len(dataarray)):

for j in range(k+1,len(dataarray)):

if(list(dataarray[k])==list(dataarray[j])):

droplist.append(j)

data.drop(index=droplist,inplace=True)

from sklearn.preprocessing import LabelEncoder

le=LabelEncoder()

for i in range(len(dl)):

if dl[i]==str:

data[coloum\_name[i]]= le.fit\_transform(data[coloum\_name[i]])

file\_path\_validated=os.path.join(basepath,"validated",secure\_filename(f.filename))

data.to\_csv(file\_path\_validated)

if(option=="Yes"):

dataarray1=data.to\_numpy()

import numpy as np

from sklearn.preprocessing import OneHotEncoder

one=OneHotEncoder()

z=[]

index=[]

for i in range(len(dl)):

if dl[i]==str:

z.append(one.fit\_transform(dataarray1[:,i:i+1]).toarray())

index.append(i)

dataarray1=np.delete(dataarray1,index,axis=1)

for j in range(len(z)):

dataarray1=np.concatenate((z[j],dataarray1),axis=1)

from sklearn.preprocessing import StandardScaler

sc=StandardScaler()

dataarray1=sc.fit\_transform(dataarray1)

fpav=os.path.join(basepath,"validated","data",secure\_filename(f.filename))

np.savetxt(fpav,dataarray1,delimiter=",")

return render\_template("index.html",ans="Schema of the provided csv file:No of row:"+str(shape[0])+"No of coloum:"+str(shape[1])+"Data type:"+str(datatype)+

"Your file is sucessfully validated and saved!")

if \_name=='main\_':

app.run(debug=True)

**CHAPTER 6**

**TESTING**

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the software system meets its requirements and user expectations and does not fail in an unacceptable manner.

**6.1 Levels of Testing**

**6.1.1 Unit testing:**

Unit testing focuses verification effort on the smallest unit of software design- the software component or module. The unit test is white-box oriented. The unit testing implemented in every module of student attendance management System. by giving correct manual input to the system ,the data’s are stored in database and retrieved. If you want required module to access input or get the output from the End user. Any error will accrued the time will provide handler to show what type of error will accrued.

**6.1.2 System testing:**

System testing is actually a series of different tests whose primary purpose is to fully exercise the computer-based system. Below we have described the two types of testing which have been taken for this project. it is to check all modules worked on input basis .if you want change any values or inputs will change all information so specified input is must.

**6.1.3 Performance Testing**

Performance testing is designed to test the run-time performance of software within the context of an integrated system. Performance testing occurs throughout all steps in the testing process. Even at the unit level, the performance of an individual module may be assessed as white-box tests are conducted.

**6.1.4 Alpha/ Beta Testing**

### **Alpha testing**

Alpha testing is simulated or actual operational testing by potential users/customers or an independent test team at the developers' site. Alpha testing is often employed for off-the-shelf software as a form of internal acceptance testing, before the software goes to beta testing.

**Beta testing**

Beta testing comes after alpha testing and can be considered a form of external user acceptance testing. Versions of the software, known as beta versions, are released to a limited audience outside of the programming team. The software is released to groups of people so that further testing can ensure the product has few faults or bugs. Sometimes, beta versions are made available to the open public to increase the feedback field to a maximal number of future users.

**6.2 Test cases**

Test case is an object for execution for other modules in the architecture does not represent any interaction by itself. A test case is a set of sequential steps to execute a test operating on a set of predefined inputs to produce certain expected outputs. There are two types of test cases:-manualand automated*.* A manual test case is executed manually while an automated test case is executed using automation.

In system testing, test data should cover the possible values of each parameter based on the requirements. Since testing every value is impractical, a few values should be chosen from each equivalence class. An equivalence class is a set of values that should all be treated the same.

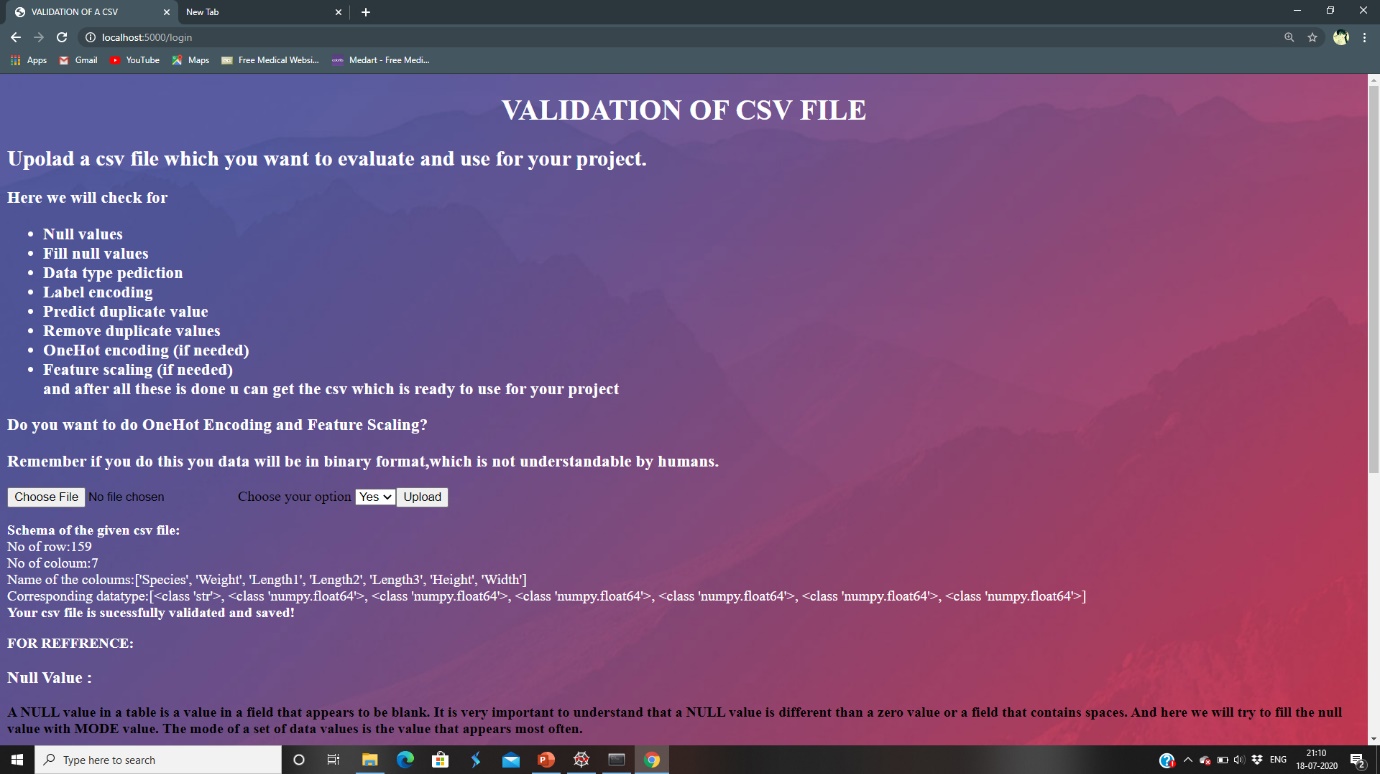
Ideally, test cases that check error conditions are written separately from the functional test cases and should have steps to verify the error messages and logs. Realistically, if functional test cases are not yet written, it is ok for testers to check for error conditions when performing normal functional test cases. It should be clear which test data, if any is expected to trigger errors.

**CHAPTER 7**

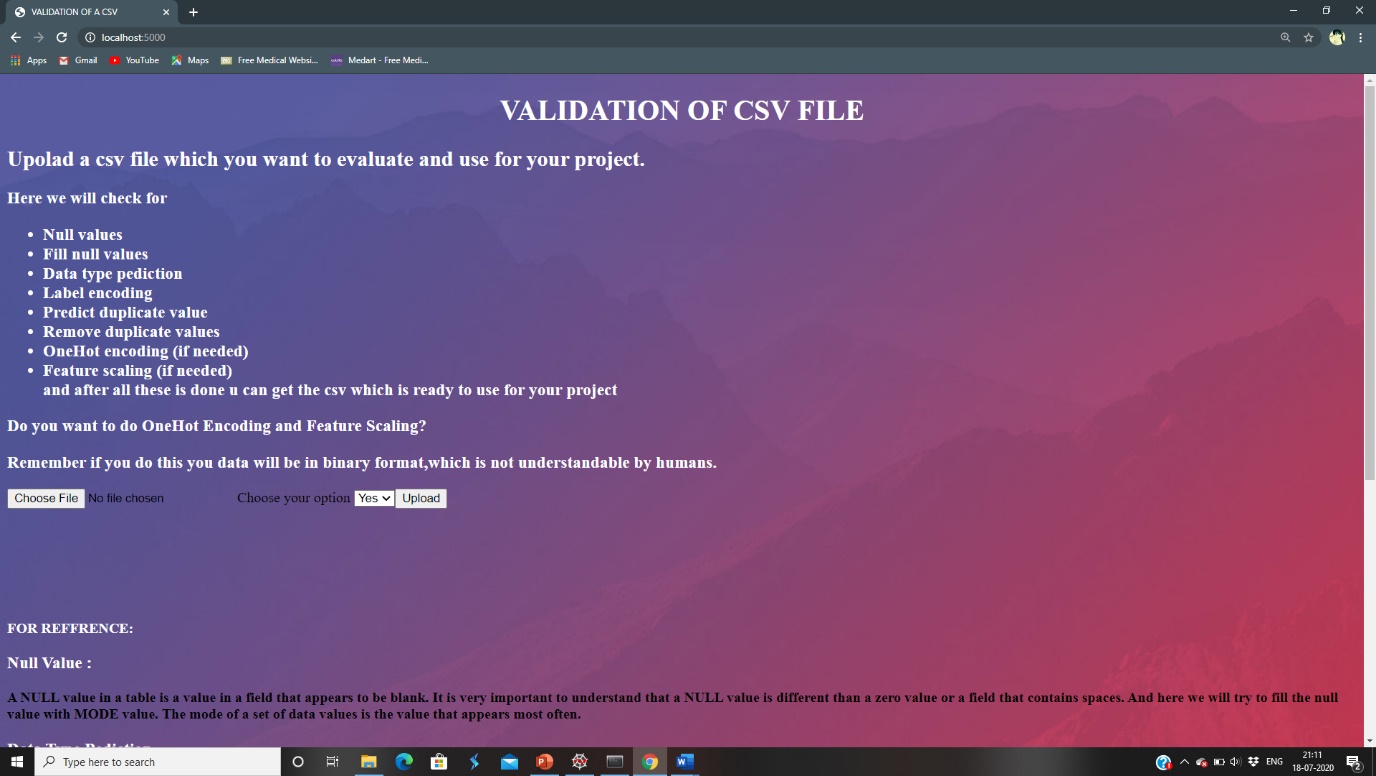
**SNAPSHOTS AND RESULT DISCUSSION**

In this chapter we will present the output of this project

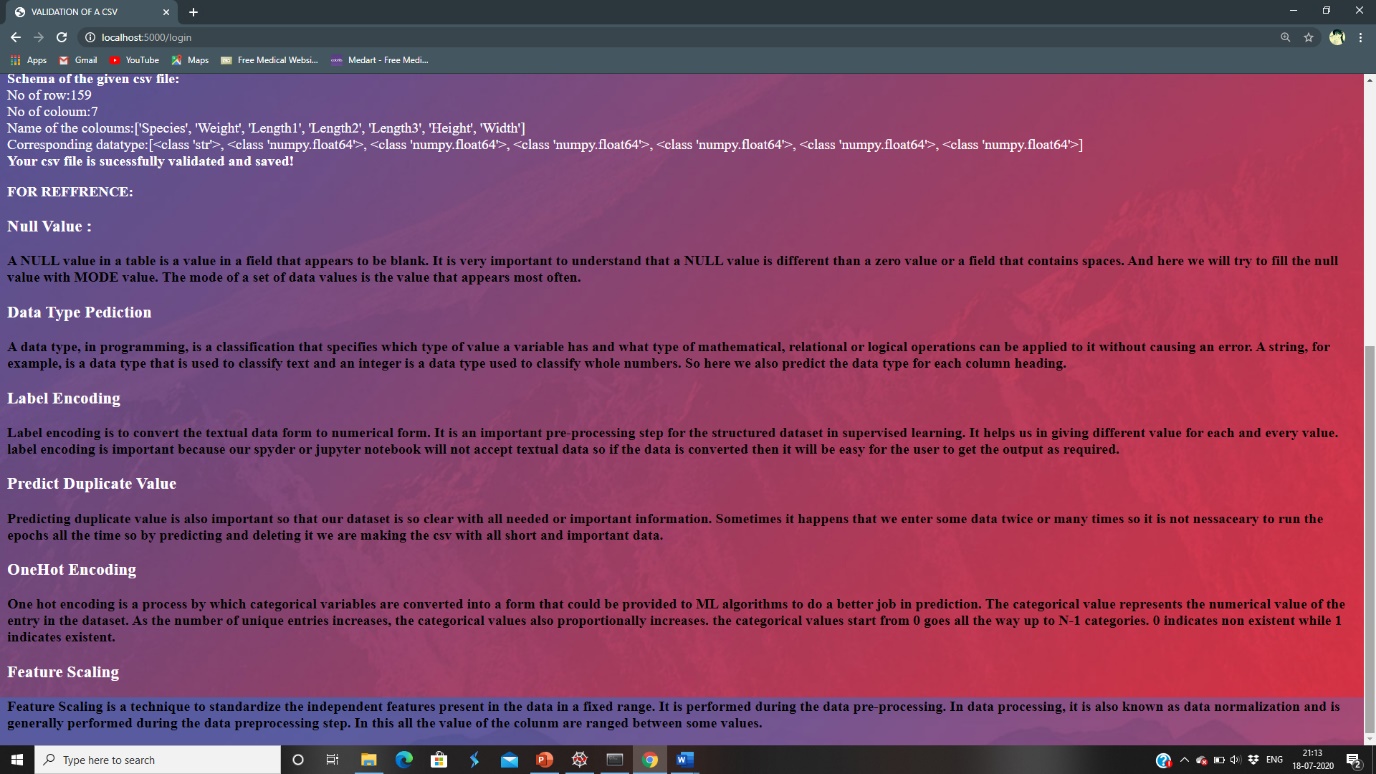
**Output 1:**



**Output 2:**



**Output3:**



**CONCLUSION AND FUTURE WORK**

With the theoretical inclination of our syllabus it becomes very essential to take the utmost advantage of any opportunity of gaining practical experience that comes along. The construction of this Mini Project- “validation of csv file” was one of these opportunities. It gave us the requisite practical knowledge to supplement the already taught theoretical concepts thus making us more competent as a computer engineer. The project from a personal point of view also helped us in understanding the following aspects of project development: The planning that goes into implementing a project. The importance of proper planning and an organized methodology. The project also provided the opportunity of interacting with our teachers and to gain from their vast experience.

There is always room for improvement, and the software I created can also be improved. This is especially because we had to create it within a limited time. With more time, the software can be improved to include security and different types of users. This would be the first step in making the software network-enabled, and eventually web-enabled. This was mine original after-thought to programming the software, and I had chosen HTML. In addition, the software can also be improved.

**REFERENCES**

[1] www.google.com – search engine which was used to search for the modified codes.

[2] [www.pandas.net](http://www.pandas.net) – is a website where you can get some of the common idea for handling csv file.

[3] [www.numpy.com](http://www.numpy.com) – is a website referred to get the idea for saving file.

[4] [www.w3schools.com](http://www.w3schools.com) – is a website where one can learn and understand the concepts of different coding languages.

[5] https://github.com/Mounisha3232/CSVFileValidation