BULK MAILER

Mass Mail Dispatcher
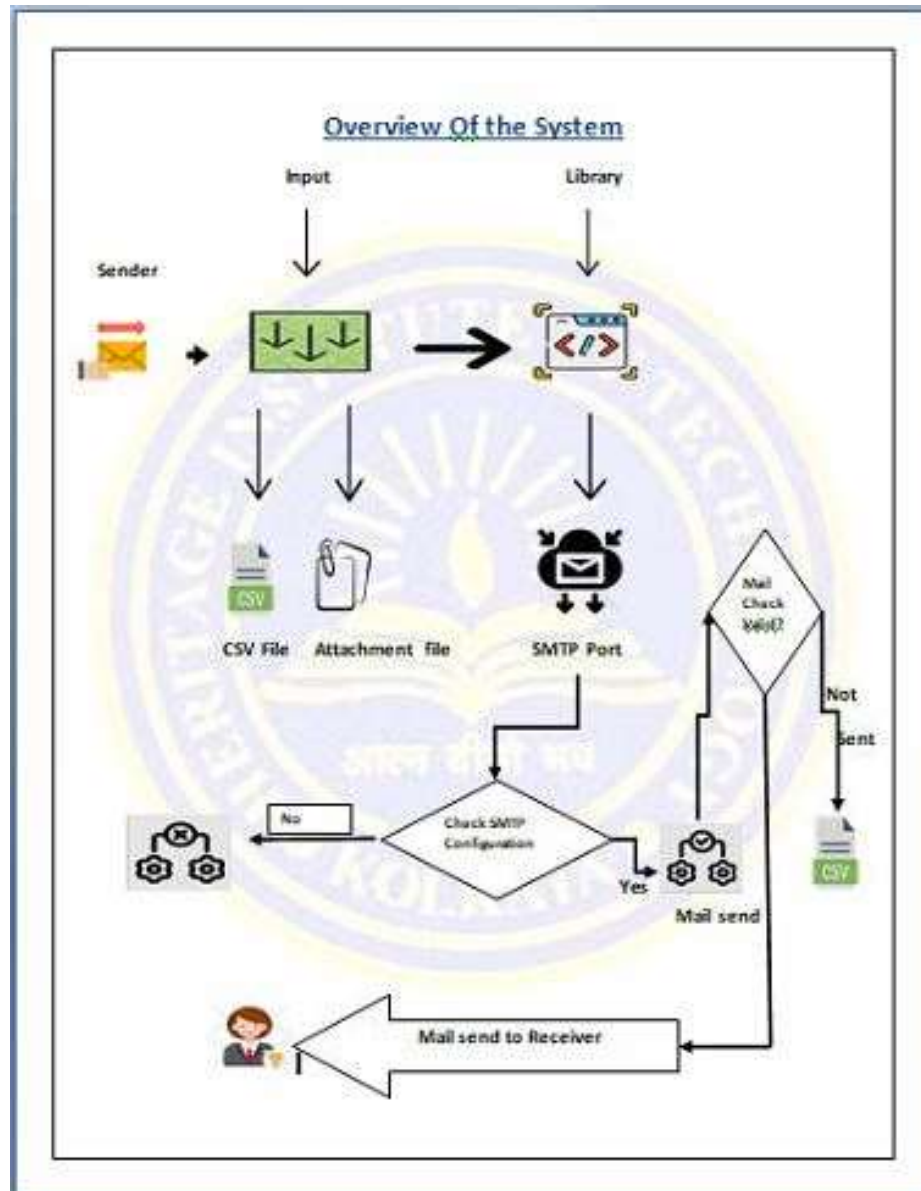
Monosij Chakraborty (12620010022)

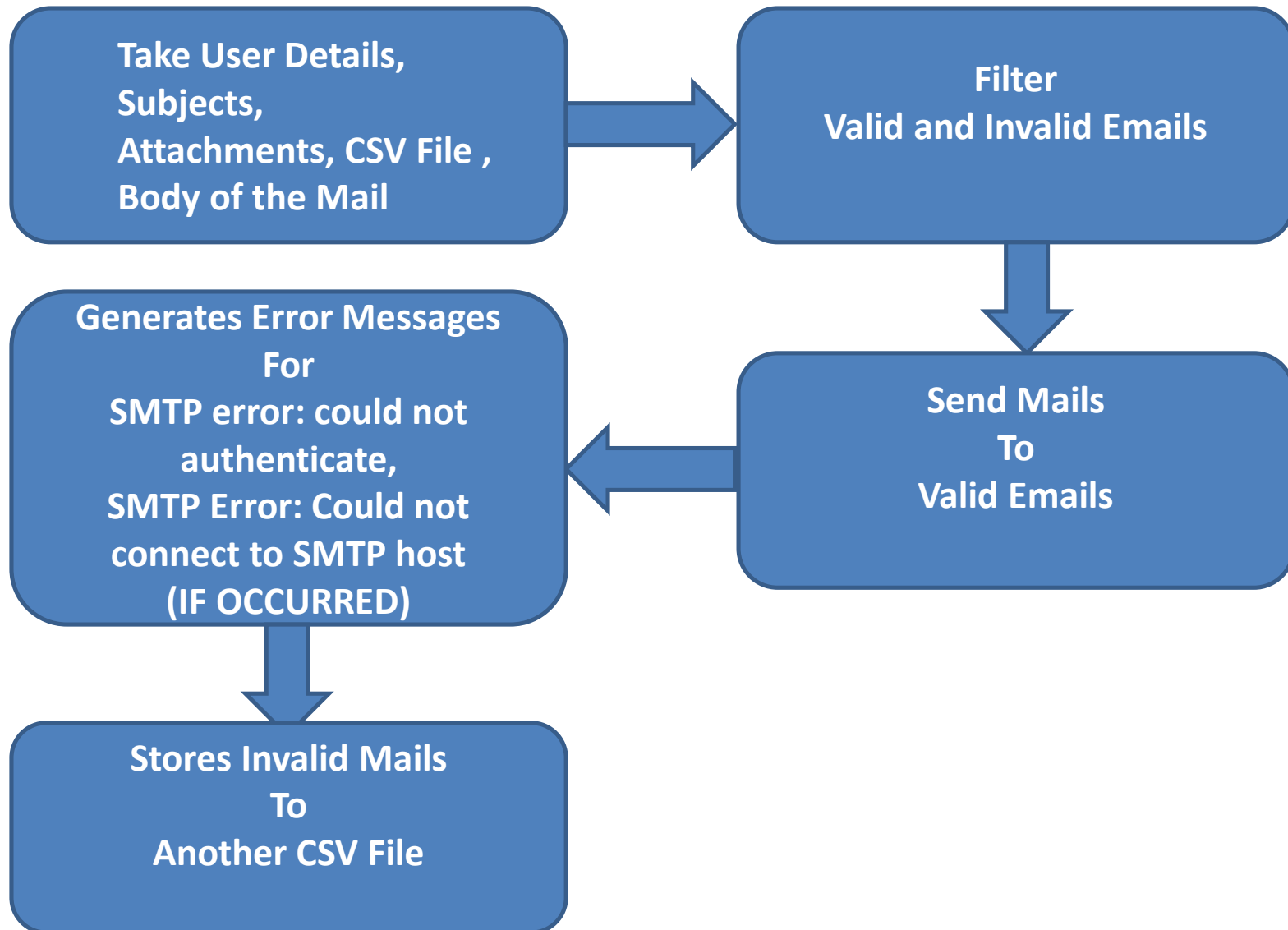**Anitra Guin (12620010002)**

Objective

The purpose of this project is to send huge bulk mail at a time. And also it can reduce having invalid mail. This process will save the time. We can perform some following task on this project:-

- Read the CSV file which contain huge no of mail ids.
- This system can send text, audio, video file using attachment facility.
- Check whether these CSV exists in system's file manager or not, if it doesn't exist then an error message will be generated.
- Read the CSV file and detect the valid mail id and send mail to those mail id's successfully.
- The system also detects the invalid mails and stores them into an another CSV file for user's benefit.
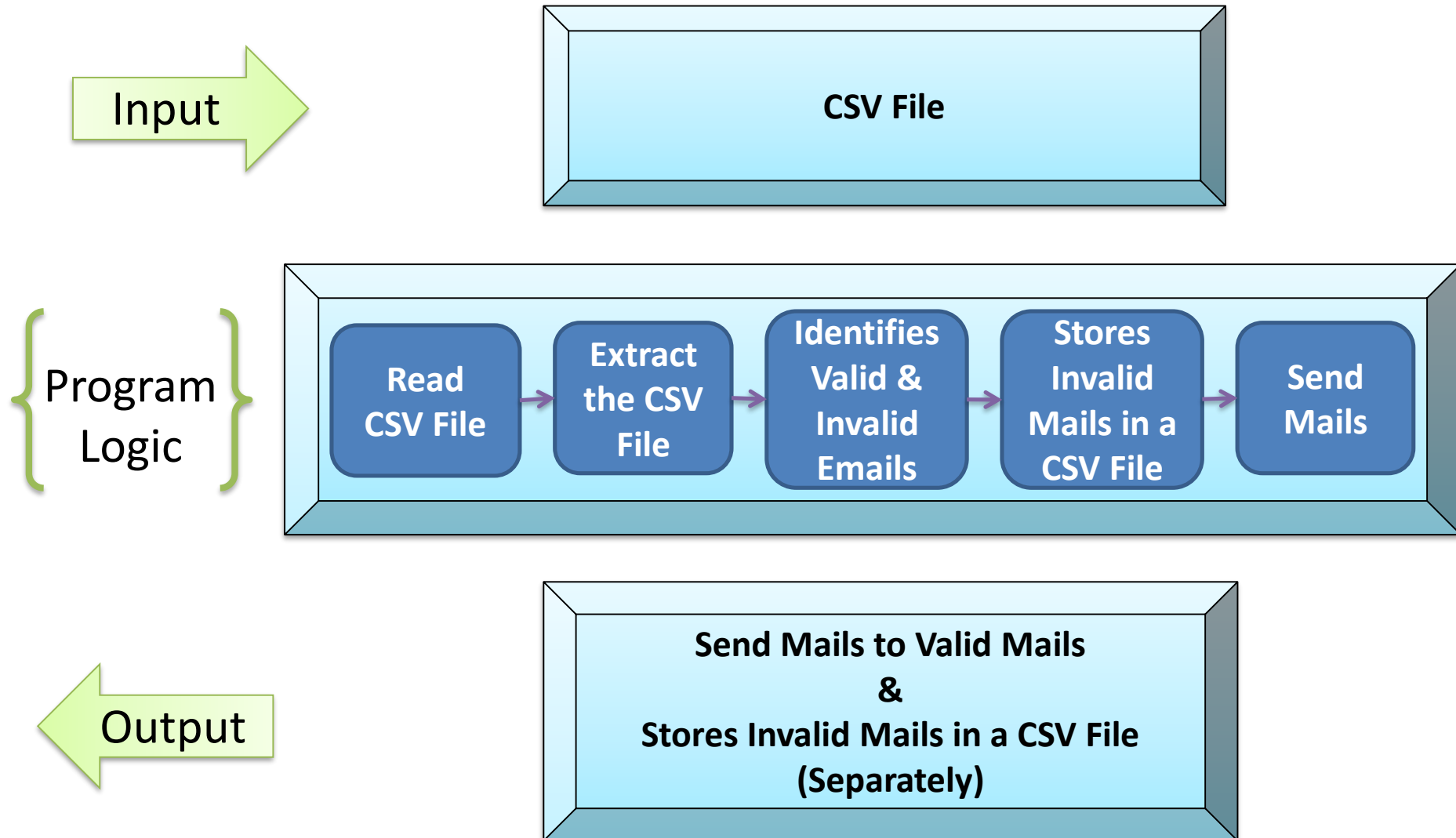
Overview Of The System

Flow of the Project

```
Take User Details,          →          Filter
Subjects,                              Valid and Invalid Emails
Attachments, CSV File ,
Body of the Mail
```

```
Generates Error Messages    ←          Send Mails
For                                    To
SMTP error: could not                  Valid Emails
authenticate,
SMTP Error: Could not
connect to SMTP host
(IF OCCURRED)
```

```
Stores Invalid Mails
To
Another CSV File
```

Use Cases of the Project

- **<u>Send Mails Quickly</u>:** This system can send mail fast.

- **<u>Decrease In payload</u>:** Removing the invalid mails and stores into another CSV file which saves lot of time.

- **<u>Usability</u>:** The PHP mailer library can be used by multiple users in order to send bulk mail with attachment.

TASK 1: Identify Mails From CSV File

Input →

CSV File

Program Logic {

Read CSV File → Extract the CSV File → Identifies Valid & Invalid Emails → Stores Invalid Mails in a CSV File → Send Mails

← Output

Send Mails to Valid Mails
&
Stores Invalid Mails in a CSV File
(Separately)

Input: CSV File

This CSV File
Contains a mixture
Of Valid & Invalid
Mails.
Our system takes
this as input.

Here we took an
Example of 20 mails,
But in real we can
Deal with thousand
of mails.

Detecting Invalid Mails

anitraguin.11111@gmail.com

anitra.guin.mca22@heritageit.edu.in

monosij.chakraborty.mca22@heritageit.edu.in

chakramanas1410@gmail.com

nitaikrishnaguin@gmail.com

anitaguin66@gmail.com

chakrabortychitra20@gmail.com

monodeadpool28@gmail. ---->>>>This is an Invalid Email
contact@odersin ---->>>>This is an Invalid Email
info.mp09gmail.com ---->>>>This is an Invalid Email

anitraofficial@gmail.com

jhipoffice@gmail.com

mcscet@.com ---->>>>This is an Invalid Email

cite_rkl@rediffmail.com

ksertcoe.com ---->>>>This is an Invalid Email

mc.marvel28@gmail.com

dccomics98@gmail.com

monosijc5@gmail.com

itsmc98@gmail.com

dataminingmca22gmail.com ---->>>>This is an Invalid Email

Output: Invalid.CSV

Invalid mails get
Stores in another CSV
File.

Rest of the valid mails
Will receive the mail.

Screenshots of The Project

Registration Page

Personel Details

Name Anitra Guin

Email anitra.guin.mca22@heritag

Contact number 9038744720

Choose your Gender:- ● Male ○ Female

Create Password: ••••

Confirm Password: ••••

Date of Birth 25-08-1998

Submit   Reset

Already have an Account?

Log-in

Information saved in Database

Redirecting To Log-in Page

localhost/internship_project3/user_login.php

... Index (20) WhatsApp Jadavpur University... Gmail YouTube Maps Is Covid-19 changin... Classes

Log-in

User-name Anitra Guin

Password: ••••

Login

If User Authentication doesn't Match

localhost/internship_project3/user_login.php

... Index (20) WhatsApp Jadavpur Univer... Classes

localhost says

Login not Successful

OK

Home Page
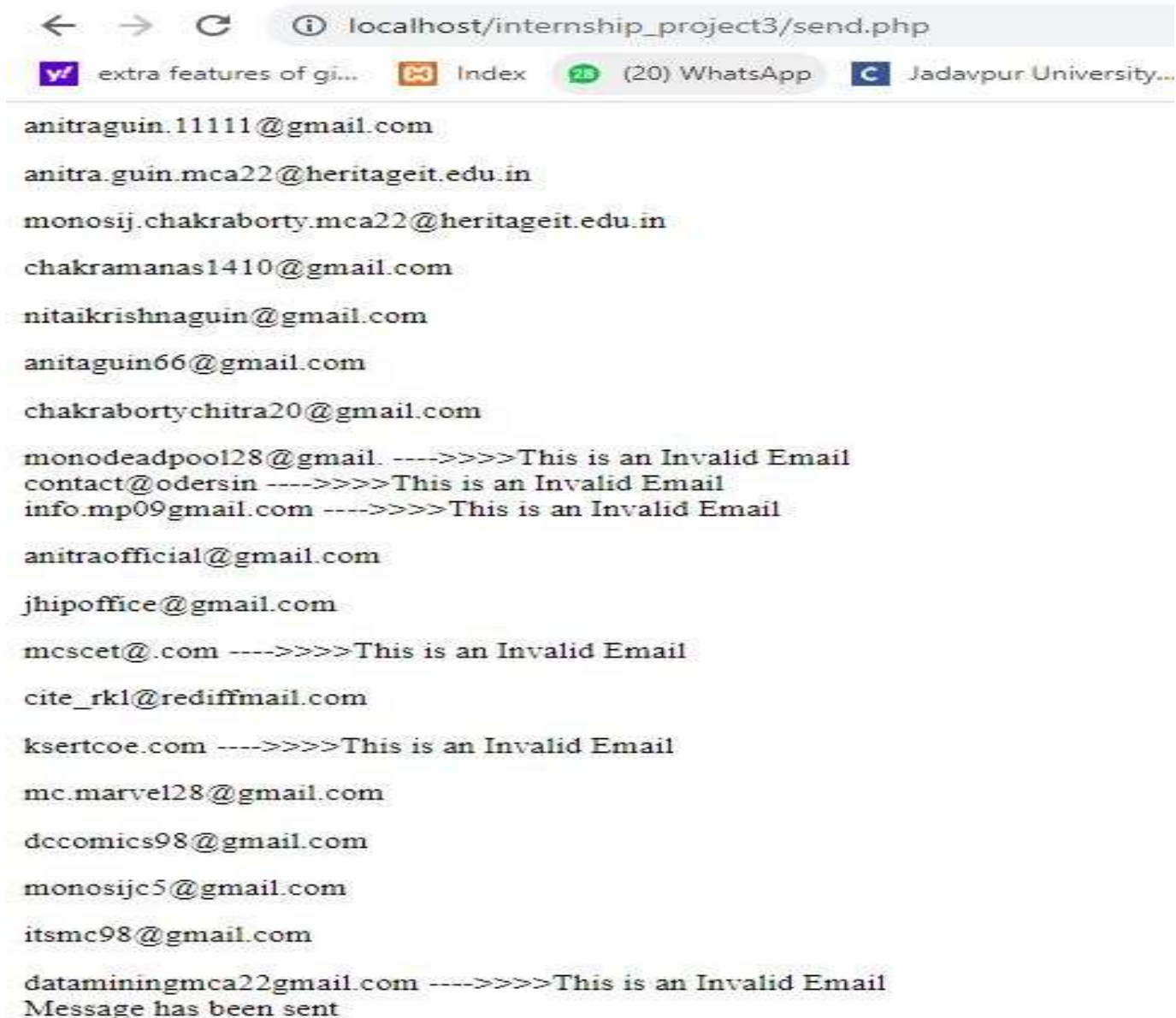
Mail has been Sent

← → C ⟳ ⓘ localhost/internship_project3/send.php

y! extra features of gi...   🔲 Index   28 (20) WhatsApp   C Jadavpur University...

anitraguin.11111@gmail.com

anitra.guin.mca22@heritageit.edu.in

monosij.chakraborty.mca22@heritageit.edu.in

chakramanas1410@gmail.com

nitaikrishnaguin@gmail.com

anitaguin66@gmail.com

chakrabortychitra20@gmail.com

monodeadpool28@gmail. ---->>>>This is an Invalid Email
contact@odersin ---->>>>This is an Invalid Email
info.mp09gmail.com ---->>>>This is an Invalid Email

anitraofficial@gmail.com

jhipoffice@gmail.com

mcscet@.com ---->>>>This is an Invalid Email

cite_rk1@rediffmail.com

ksertcoe.com ---->>>>This is an Invalid Email

mc.marvel28@gmail.com

dccomics98@gmail.com

monosijc5@gmail.com

itsmc98@gmail.com

dataminingmca22gmail.com ---->>>>This is an Invalid Email
Message has been sent

Snapshots of Some Codes

Server Settings

```
24    try {
25        //Server settings
26        $mail->SMTPDebug = 0;                        //Enable verbose debug output
27        $mail->isSMTP();                             //Send using SMTP
28        $mail->Host       = 'smtp.gmail.com'; //Set the SMTP server to send through
29        $mail->SMTPAuth   = true;                    //Enable SMTP authentication
30        $mail->Username   = $sender;                 //SMTP username
31        $mail->Password   = $sender_password;    //SMTP password
32        $mail->SMTPSecure = PHPMailer::ENCRYPTION_SMTPS; //Enable implicit TLS encryption
33        $mail->Port       = 587;                     //TCP port to connect to; use 587
34        $mail->Port       = 465;                     //SSL port
35
```

//Sending mails using PHPMailer Instance and separating valid and invalid mails

```php
51        //Sending mails using PHPMailer Instance and seperating valid and invalid mails
52     $csv = $fileName;
53     $fh = fopen($csv, 'r');
54     $invalid = fopen("invalid.csv", 'a');
55          while(list($email) = fgetcsv($fh, 1024, ',')){
56             $sntz_email = filter_var($email, FILTER_SANITIZE_EMAIL);
57             if(filter_var($sntz_email, FILTER_VALIDATE_EMAIL)){
58               printf("<p> %s </p>", $sntz_email);
59               $mail->addAddress($sntz_email);
60
61             }else{
62             print($sntz_email." ---->>>>This is an Invalid Email <br>");
63
64             $form_data = array(
65                 'email' => $sntz_email
66             );
67             fputcsv($invalid, $form_data);
68
69             }
70
71        }
72
```

Index.php

```html
<h1 class="text-center fw-bold text-dark mb-3">Welcome <?php echo $_SESSION['username']; ?></h1>
<h2 class="text-center fw-bold text-dark mb-3">Email Sender</h2>
<form action="send.php" method="post" enctype="multipart/form-data">
    <div class="row">
        <div class="col-6 mb-3">
            <label for="sender_name" class="form-label">Sender Name</label>
            <input type="text" class="form-control" id="sender_name" name="sender_name" value="<?php echo $_SESSION['username']; ?>" placeholder="John Doe" readonly>
        </div>
        <div class="col-6 mb-3">
            <label for="sender" class="form-label">Sender Email</label>
            <input type="email" class="form-control" id="sender" name="sender" placeholder="name@example.com" required>
        </div>
        <div class="col-6 mb-3">
            <label for="sender_password" class="form-label">Sender Password</label>
            <input type="password" class="form-control" id="sender_password" name="sender_password" placeholder="Enter your email password" required>
        </div>
        <div class="col-6 mb-3">
            <label for="subject" class="form-label">Subject</label>
            <input type="text" class="form-control" id="subject" name="subject" placeholder="How are you?" required>
        </div>
        <div class="col-6">
            <label for="attachments" class="form-label">Attachments (Multiple)</label>
            <input type="file" class="form-control" multiple id="attachments" name="attachments[]" placeholder="name@example.com">
        </div>
    </div>
    <div class="mb-3">
        <label for="recipient" class="form-label">Recipient Emails</label>
        <input type="file" class="form-control" multiple id="recipient" name="recipient[]">

    </div>
    <div class="mb-3">
        <label for="body" class="form-label">Body</label>
        <textarea class="form-control" id="body" name="body" placeholder="Hi, How are you?" rows="5" required></textarea>
    </div>
    <div>
        <button class="btn btn-primary me-2" name="send" type="submit">Send Email</button>
        <button class="btn btn-danger" type="reset">Reset Form</button>
        <input type="button" onclick="location.href='user_login.php'" value="Log-out" style="position:relative; left:62%">
    </div>
</form>
```

Connecting to Database

```php
<?php
    $db = mysqli_connect('localhost', 'root', '') or
        die ('Unable to connect. Check your connection parameters.');
        mysqli_select_db($db, 'internship-project2' ) or die(mysqli_error($db));

?>
```

Registration Form

```html
                <legend class="text-center fw-bold text-dark mb-3">Personel Details</legend>
                <div class="col-6 mb-4">
                    <label for="sname">Name</label>
                    <input type = "text" name = "name" id="sname" placeholder="Enter your Name here"><br>
                </div>
                <div class="col-6 mb-4">
                    <label for="semail">Email</label>
                    <input type="text" name="email" id="semail" placeholder="Enter your Email here"><br>
                </div>
                <div class="col-6 mb-4">
                    <label for="spnumber">Contact number</label>
                    <input type="number" name="ph_number" id="spnumber" placeholder="Enter your phone number here"><br>
                </div>
                <div class="col-6 mb-4">
                    <p1>Choose your Gender:- </p1>
                    <input type="radio" name="gender" id="male" value="Male">
                    <label for="male">Male</label>
                    <input type="radio" name="gender" id="female" value="Female">
                    <label for="female">Female</label><br>
                </div>
                <div class="col-6 mb-4">
                    <label for="passwrd">Create Password:</label>
                    <input type="password" name="password" id="passwrd" placeholder="Enter your password here"><br>
                </div>
```

Storing User Information From Registration Form to Database

```php
6        if($_SERVER['REQUEST_METHOD']=== 'POST'){
7
8            $sname = $_POST['name'];
9
10           $semail = $_POST['email'];
11           $snumber = $_POST['ph_number'];
12           $sgender = $_POST['gender'];
13           $spassword = $_POST['password'];
14           $scpassword = $_POST['c_password'];
15           $sdob = $_POST['DOB'];
16
17
18           $stmnt = $conn->prepare("INSERT INTO user_info (name,email_id,contact,gender,password,c_password,
19                                    DOB) VALUES
20                                    (:a,:b ,:c ,:d , :e, :f, :g)");
21
22           $stmnt->bindvalue(':a',$sname);
23           $stmnt->bindvalue(':b',$semail);
24           $stmnt->bindvalue(':c',$snumber);
25           $stmnt->bindvalue(':d',$sgender);
26           $stmnt->bindvalue(':e',md5($spassword));
27           $stmnt->bindvalue(':f',md5($scpassword));
28           $stmnt->bindvalue(':g',$sdob);
29
30
31           $stmnt->execute();
32
33           echo "<script>alert('Record Created')</script>";
```

Tools Used

VS-Code (Visual Studio Code) For Web-Developers-
Visual Studio Code is "a free-editor that helps the programmer write code helps in debugging and corrects the code using the intelli-sense method". In normal terms, it facilitates users to write the code in an easy manner. Many people say that it is half of an IDE and an editor, but the decision is up to the coders. Any program/software that we see or use works on the code that runs in the background. Traditionally coding was used to do in the traditional editors or even in the basic editors like notepad! These editors used to provide basic support to the coders.

PHP (Hypertext Preprocessor)- PHP is a programming language that allows web developers to write scripts that generate HTML pages dynamically.PHP was originally developed as a server-side scripting language, meaning that it runs on the server before being sent to the browser. However, with the advent of PHP 5, PHP has become more popular as a client-side scripting language. Client-side scripting languages run on the browser, so the code is executed directly to browser.

HTML (Hypertext Markup Language)- HTML is an acronym which stands for Hyper Text Markup Language which is used for creating web pages and web applications. HTML is a markup language which is used for creating attractive web pages with the help of styling, and which looks in a nice format on a web browser. An HTML document is made of many HTML tags and each HTML tag contains different content.

CSS(Cascading Style Sheet): Cascading Style Sheets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable. CSS handles the look and feel part of a web page. Using CSS, you can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, layout designs, and variations in display for different devices and screen sizes as well as a variety of other effects.

Thank You