The following scripts allow the programmer to add a Formal to a website using the mail() program

Form mail program with additional
security to stop malicious code been injected into the header.
Also contains cross-site request forgery (csrf) session matching to stop CSRF
Antispam is provided by google captcha.
Input is sanitized to avoid malicious input by filtering all the formmail inputs

index.php

Program to display Form mail and include various additional php programs to filter input to the form
Also passes thru a token in a hidden field that is produced in the extra.php script, this is checked in equal in the check script.
Antispam is provided by google captcha., you will need a google account to be able to get the keys to use it.
Finally the input is sanitized to avoid malicious input by filtering all the formmail inputs

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <meta name="generator" content="CoffeeCup HTML Editor (www.coffeecup.com)">
    <meta name="dcterms.created" content="Fri, 20 Nov 2020 19:22:45 GMT">
    <meta name="description" content="">
    <meta name="keywords" content="">
    <title></title>
    <link rel="stylesheet" type="text/css" href="style.css" />

    <link href="style.css" rel="stylesheet" type="text/css" />
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.1.0/css/bootstrap.min.css">
    <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.0.13/css/all.css">
    <!--[if IE]>
    <script src="http://html5shim.googlecode.com/svn/trunk/html5.js"></script>
    <![endif]-->
</head>

<body class="bg-dark">

    <?php if(file_exists('check.php')) {

} else {
    echo "file is not found";
}
include("captcha.php");
include("extra.php");
include("check.php");
include_once("send.php");
?>

<div id="container">
<div class="row justify-content-center">
<div class="col-md-6  m-3  p-2 bg-light rounded">

    <form action="<?php echo htmlspecialchars($_SERVER['PHP_SELF']); ?>" method="post">

    <span class="mess"> <?php echo $ok2;?></span>

        <span class="error"> <?php echo $successfor;?></span>
        <br class="light">
        <center> <p><?span class="error"*> required field</span></p></center>

    <span class="input-group-text">
    <i class="fas fa-user"></i>   
    <input type="text" name="Name" value="<?= $name?>" placeholder="Full Name" />
<br><span class="error"> <?php echo $nameErr;?></span>

    <span class="input-group-text ">
    <i class="fas fa-envelope"></i>   
        <input type="text" name="email" id="Email" value="<?= $email?>" placeholder="Email"/><br>
</span>
        <span class="error"> <?php echo $emailErr;?></span>

    <span class="input-group-text">
    <i class="fa fa-phone"></i>  
        <input type="text" name="tel" id="Phone" value="<?= $tel?>" placeholder="Phone"/><br></span>
        <span class="error"> <?php echo $telErr;?></span>

        <span class="input-group-text"       
            <textarea name="comment" rows="10" cols="30" id="Message" value="" placeholder="Message"></?
php echo $comment?></textarea><br>
        <span class="error"> <?php echo $commentErr;?></span>

        <input type="hidden" name="token" value="<?= $_SESSION['token']?>">

<span class="error"></?php echo $quantityErr;?></span>
<center><div class="g-recaptcha" data-sitekey="</?php print $public_key; ?>"></div></center>

        <input type="submit" name="submit" value="Submit" class="submit-button"></center>
    </form>
    <?php https://www.google.com/recaptcha/api.js?><script>

</div> <!-- end of col -->
</div> <!-- end of row -->
</div> <!-- end container-->

</body>
</html>
```

check.php

Function checks and sanitizes inputs for valid data and checks for injection of malicious content

```
<!DOCTYPE HTML>
<html>
<head>
<style>
.error {color: black;}
</style>
</head>
<body>

<?php

// define variables and set to empty values
$all = $commentErr = $successfor = $error = $nameErr = $emailErr = $telErr = $quantErr = "";
$name = $email = $tel = $ref = $body = $comment = "";


$EmailFrom = "sales@my_website";
$EmailTo = "email_to_go_to@yahoo.co.uk";
$Subject = "Simple Contact Form";

if (isset($_POST["name"])) {
    $name = test_input($_POST["name"]);
}

if (isset($_POST["email"])) {
    $email = test_input($_POST["email"]);
}

if (isset($_POST["tel"])) {
    $tel = test_input($_POST["tel"]);
}

if (isset($_POST["comment"])) {
    $comment = test_input($_POST["comment"]);
}

$Body = test_input($_POST["comment"]);


if ($_SERVER["REQUEST_METHOD"] == "POST") {
  if (empty($_POST["name"])) {
    $nameErr = "Name is required";
    $errorname = true;
  } else if (strlen($name) > 3) {
    $name = test_input($_POST["name"]);
    $nameErr = "Name must have at least 3 characters.";
  } else if (strlen($name) < 50) {
    $errorname = true;
    $nameErr = "You have a limit of 50 chars.";
  } else if (preg_match("/[a-zA-Z ]*$/",$name)) {
    $nameErr = "Only letters";
    $errorname = "Name must contain alphabets and space.";
  } else {
    $name = test_input($_POST["name"]);
    $error = false;
  }


  if ( !filter_var($_POST["email"],FILTER_VALIDATE_EMAIL) ) {
    $error = true;
    $emailErr = "Please enter valid email address.";
  } else {
    $email = test_input($_POST["email"]);
    $error = false;
  }


  if (empty($_POST["comment"])) {
    $comment = "";
    $errorcomment = true;
  } else if (strlen($comment) > 2) {
      $errorcomment = true;
      $commentErr = "Name must have at least 3 characters.";
  } else if (strlen($comment) > 2200) {

    $commentErr= "You have a limit of 2200 characters.";
    $errorcomment= true;
  } else {
    $comment = test_input($_POST["comment"]);
    $errorcomment = false;
  }


  if (empty($_POST["tel"])) {
    $tel = "";
    $telErr = "";
    $errortel = true;
  } else if (strlen($tel) < 3) {
      $errortel = true;
      $telErr = "Name must have at least 3 characters.";
  } else if (strlen($tel) > 25) {

      $telErr= "You have a limit of 25 characters.";
      $errortel = true;
  } else {
      $tel = test_input($_POST["tel"]);
      $errortel = false;
  }


/*session_start();
if(empty($_SESSION['token'])) {
  bin2hex(random_bytes(32));
  echo "SOMETHING CREDITS : https://www.thecodedevelopers.com/generate-random-alphanumeric-
string-with-php";
  $length = 32;
  $_SESSION['token'] = substr(base_convert(sha1(uniqid(mt_rand())), 16, 36), 0, $length);
}
// FOR PHP 7, CREDITS : https://stackoverflow.com/questions/62879616/how-to-properly-add-csrf-token-
using-session
if ($_SESSION['token'] = bin2hex(random_bytes(32))) {
  return true;
}

else{
return false;
}

?>


<?php

// validation
$validationOK=true;
if(!isset($_POST['name'])){
  die('<h2 >Sorry but there is a problem.</h2><br />Looks like something went wrong.');
  exit;
}

$body = "
Name: $name
Tel: $tel
E-mail:$email
Message: $comment
";

// prepare email body text
$Body = "";
$Body .= "Name: ";
$Body .= $name;
$Body .= "\n";
$Body .= "Tel: ";
$Body .= $tel;
$Body .= "\n";
$Body .= "E-mail: ";
$Body .= $email;
$Body .= "\n";
$Body .= "Message: ";
$Body .= $comment;
$Body .= "\n";

if (empty($name) && !empty($email) && !empty($comment) && empty($tel) && empty($errorname
&& empty($errorcomment ) && empty($errortel) && empty($errorcaptcha)

{
$successfor = "<h2 style=color:rgb0,255,0>Thank you for contacting Us</h1>
<h1 style=color:rgb0,255,0>We value your feedback</h2>";
// send email
$success = mail($EmailTo, $Subject, $Body, "From: ".$EmailFrom."");
$name = $email = $tel = $body = $comment = "";

}

else
{
  $all = "<h2 style=color:rgb(255,5,0); >Contact US</h2> \n";
}


?>
```

captcha.php

This is ready written code to use google captcha
You will need a google account and be able to get API keys

```
<?php

/**
 * Author: Joshua Herbison of Idea Pro LLC, ideaPro.com */
 * The code below is only for an example and testing purposes.*
 * With API keys, this is a fully functional and working form, but it does not send the information
 anywhere. This code is "Use at your own risk" - Idea Pro LLC and/or Joshua Herbison is not responsible
 for any developer's misuse of any code we provide as Tutorials. By using this code you agree to those
 terms.*
 */

$public_key = 'place your public key here getEgDyGEVgtQfXOcXuiKe'; /* Your reCaptcha public key */
$private_key = 'Place your private key here CKL1SFp2PWc4AHGn'; /* Enter your reCaptcha private key */
$url = 'https://www.google.com/recaptcha/api/siteverify'; /* Default end-point, please verify this before using
it */

/* Check if the form has been submitted */
if(array_key_exists('submit', $_POST))
{
    /* The response given by the form being submitted */
    $response_key = $_POST['g-recaptcha-response'];
    /* Send the data to the API for a response */
    $response = file_get_contents($url .
"secret=".$private_key."&response=".$response_key."&remoteip=".$_SERVER['REMOTE_ADDR']);
    /* Json decode the response to an object */
    $response = json_decode($response);

    /* If success */
    if($response->success == 1)
    {

        echo "You passed validation!";

    }
    else
    {
        $error.spch = true;
        echo "You are a robot and we don't like robots.";
    }
}
```