

Forum:[Hack Forums](#)

Thread: [VB.NET Chat Room Tutorial With PHP & SQL \[Intermediate\]](#)

+Haze - January 3, 2015, 12:05 pm

Introduction

So this was requested by a few members here, and I thought it would be a good idea to do something different. I've seen tons of chats clients made in VB.NET that use sockets and TCP/IP, the tutorials only show how this would work for local networks. So I'm going to show how to do a chat room in vb.net that utilizes a webserver(PHP & some SQL to store our user information). So we could use this globally, and anyone with internet connection could connect and chat.

First of all I'll mention the disadvantages & advantages of using a webserver as a chat room whatever in the way I did it.

Advantages

- + It can be fast if you got a nice web-host
- + It's a good learning experience, from this tutorial you can learn a lot about PHP, SQL and coding in general(vb.net specifically)

Disadvantages

- Really and truly, this should be done with sockets but meh.
- This creates tons of requests to your web-host per minute but you can get a free one like x10 hosting that don't care.

Also, I'm not really focusing on security, I'm focusing on actually teaching people valuable skills and fundamentals. The only security, there is, is that the password is hashed + salted. And using PDO to prevent SQL Injections. Though the messages are not encoded or whatever.

Because I do not feel its necessary, if the user is really egar, they can learn how to do it for themselves. From the tutorial I'm writing out, I've done a lot of work on my side. And I'm actually going over every single variable declaration and function declaration and making sure I explain them fully. Both the PHP code, SQL Code and VB.NET code. I'm not just spitting out the source code. I'm actually being a TEACHER, which is something I think we don't see enough in this section.

And that's my justification for not encoding the messages or worried too much about security.

Setup MySQL Database

Okay so let's get started.

First we're going to start with our server side code. So this is the PHP and SQL stuff.

Firstly, I recommend that you go check out my first tutorial on how to do a login and registration system using vb.net php and sql. What this tutorial is is an extension of the previous one.

<https://hackforums.net/showthread.php?tid=4610630>

So let's begin.

We login to our cPanel webhost and we setup our database.

Setup SQL Tables

So we're going to create 3 tables. tblLogin, tblMessage and tblMessageLog.

tblLogin will store all the login information of our users. And in that table we will have 5 columns UserID, Username, Password, IsBanned, DateRegistered.

the UserID is unique ID that is used identify each user, just like you and I have a unique HF UserID.

Next is tblMessage. This will store all the information about each message that is sent by each of our users.

So again, MessageID to identify each message, UserID to identify which user sent it, the content of the message, and what time it was sent by.

now our final table. tblMessageLog.

This table is used when a message is read by any user. So when a user reads a message, it will appear in this table, so the MessageID of the message that was read, and the UserID of the user who has READ the message.

PHP Server Side Code

Note: I don't know PHP as well as I do C# and VB.NET so bare my sloppy code.

Step 1 we create a db.php which we will place code required to connect to our database and sql database where. So you need to create a database, with some users and stuff. Once you've learned how to do that else where, you can proceed.

We will call this
db.php

PHP Code

```
<?php
// Connection details
//Change me
$db_host = "localhost";
$db_name = ""; // database name
$db_user = ""; // database user
$db_pass = ""; // database password
```

```
$connstr = "mysql:host=$db_host;dbname=$db_name";
```

```
try
{
    $con = new PDO($connstr, $db_user, $db_pass);
}
catch (exception $e)
{
    echo "error";
    //you might want to display a more meaningful error
    exit();
}

?>
```

We're now going to create the php script that handles our register requests.
We will call this
register.php

PHP Code

```
<?php
include 'db.php';
//We "Import/Include" this so we can access our DB credentials.

$user = $_GET['user'];
$pass = $_GET['pass'];
//We declare two GET variables that we can use to query against our database.

//We ensure the variables are not empty, if they're not we then proceed into the if statement
if (!empty($user) && !empty($pass))
{
    if (registerSuccess($user, $pass))
    {
        echo "success";
    }
    else
    {
        echo "fail";
    }
}
else
{
    echo "jog on kid";
}

function registerSuccess($username, $password)
{
    global $con;
    $salt = substr(str_shuffle(
"!£$%^&*abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789"), 0, 25);
    $hash = crypt($password, $salt);

    //We generate a random salt, and we then use the crypt function to generate us a random has.
```

```

$stmt = $con->prepare(
"INSERT INTO tblLogin(Username, Password, DateRegistered) VALUES(:user, :pass, :ts)");

//This insert query will insert the two values from our declared GET variables into our SQL tables.

$stmt->bindValue(':user', $username, PDO::PARAM_STR);
$stmt->bindValue(':pass', $hash, PDO::PARAM_STR);
$stmt->bindValue(':ts', date("Y-m-d G:i:s"));
//We bind the values from our sql query to our prepared statement

//Executes the query, and if it is successful we print out "success"
if ($stmt->execute())
{
    return true;
}
else
{
    return false;
}

$db = null;
}
?>

```

We will now create our PHP script to login to our system from our vb.net application. We will call this login.php

PHP Code

```

<?php
include 'db.php';
//We "Import/Include" this so we can access our credentials.

```

```

$user = $_GET['user'];
$pass = $_GET['pass'];
//We declare two GET variables that we can use to query against our database.

LoginUser($user, $pass);

function LoginUser($username, $password)
{
    global $con;

    $stmt = $con->prepare("SELECT * FROM tblLogin Where Username = :user Limit 1");

//This insert query will select all the data from our sql server, where the username of the GET variables
    //match the one in our SQL table. And the Limit 1 means, return only 1/the first result and ignore the rest.

    $stmt->bindValue(':user', $username, PDO::PARAM_STR);

    $stmt->execute();

```

//We then bind our username to our query and execute it.

```
while($r = $stmt->fetch(PDO::FETCH_ASSOC))
{
    $arr = array('UserId' => $r["UserID"], 'IsBanned' => $r["IsBanned"], 'TimeSignedIn' => date(
'Y-m-d G:i:s'));
    $hashedpassword = $r["Password"];
}

// We retrieve the users hashed password

    //We has their current password, and use their hashed password as the salt, and compare
    // Hashing the password with its hash as the salt returns the same hash
    if ($hashedpassword == crypt($password, $hashedpassword))
    {
        echo json_encode($arr);
    }

    else
    {
        echo "doesn't exist";
    }

    $con = null;
}
```

?>

This is an image of what the data it echos when you successfully login would look like.

This is the script we use to add a message to our database we will call this.
addmessage.php

PHP Code

```
<?php
include 'db.php';
```

//when we include, we import/include all our data from our db.php page so we can use them here. without

//having to copy and paste the same data all the time

```
$MessageContents = $_GET["msg"];
$UserID = $_GET["userId"];
```

//we declare two GET variables, which will contain the userId of the user trying to add the message.

//And msg is the content of the message they're trying to send. We could of used POST which would be better for larger

//data, but for tutorial sake, let's just use GET.

//We check to see if the variables are empty or not, if they are not empty then we process.

```
if (!empty($MessageContents) && isset($UserID))
```

```

{
    //If not empty, then we add the message, we create our own function called addMessage()
    //This function will take in two arguments as the parameters. UserID and MessageContents.
    //This function will then add the message to our sql table, and if it's successfull.
    //It will print "success" else it prints failed

    if (addMessage($UserID, $MessageContents))
    {
        echo "success";
    }
    else
    {
        echo "failed";
    }

    $con = null;
}
else
{
    //If the two GET variables are empty, this message will show.
    echo "jog on kid";
}

function addMessage($userId, $msg)
{
    global $con;

//by typing global, we reference the variable in our db.php. So we can use that variable to connect to
our db

    $stmt = $con->prepare(
"Insert Into tblMessage(UserID, Message, TimeSent) Values(:userID, :msgID, :ts) ");

    $stmt->bindValue(':userID', $userId, PDO::PARAM_INT);
    $stmt->bindValue(':msgID', $msg, PDO::PARAM_STR);
    $stmt->bindValue(':ts', date("Y-m-d G:i:s"));

//we create use PDO(PHP Data Objects) to create a prepared SQL statement, which prevents SQL I
njection attacks.
    //And we bind the data $userId and $msg from to user to our database. And we execute. If exe
cution is succesful we return tru
    //else we return false

    if ($stmt->execute())
    {
        return true;
    }
    else
    {
        return false;
    }
}

```

?>

Next we create our script to get the unread messages of each user. --Warning, this bit gets a bit complex if you do not understand SQL very well, the queries are a bit complex, but don't worry.

We will name this script
getunread.php

PHP Code

```
<?php
    include 'db.php';
    //We "Import/Include" this so we can access our DB credentials.

    $Action = $_GET["action"];
    $UserID = $_GET["userId"];
    $TimeSignedIn = $_GET["timestamp"];

    //we create 3 GET variables.
    //We reat the timesignedIn variable,

    //This validates to check if the variables are not empty. And validates to see if
    //The user requested is a valid user or not. If they're then we proceed.
    if (!empty($Action) && !empty($TimeSignedIn) && validUserID($UserID) == true)
    {
        //The GetUnreadCount will return an Integer, of the count of the unreadMessages of the user.
        //The GetUnreadMessages will return the conetents of all their unread messages.

        if ($Action == "count")
        {
            echo getUnreadCount($UserID);
        }

        elseif ($Action == "msg")
        {
            getUnreadMessages($UserID);
        }
    }
    else
    {
        echo "jog on kid";
    }

    function getUnreadCount($userId)
    {
        global $con;
        global $TimeSignedIn;

        $stmt = $con->prepare("Select MessageID, Username, Message, TimeSent
        From tblLogin, tblMessage
        Where tblMessage.TimeSent >= :timesignedin
        And tblLogin.UserID = tblMessage.UserID
        And Not tblLogin.UserID = :ID
```

```
And tblMessage.MessageID Not In(Select MessageID From tblMessageLog Where UserID = :ID)");
```

```
//This query is quiet complex, but it does is. It Selects all the infromation from the message table,  
//Where the time the message was sent is after the user has logged in, and where the message ID is not found in the MessageLog  
//table(because thats where all read messages go) and where the message is not a message sent by the user himself.
```

```
$stmt->bindValue(':ID', $userId, PDO::PARAM_INT);  
$stmt->bindValue(':timesignedin', $TimeSignedIn);
```

```
$stmt->execute();
```

```
//We then bind the values off the userid and timesigned in to the query, and execute it and return the row count
```

```
//The row count signifies the amount of results it found from then query.
```

```
return $stmt->rowCount();  
}
```

```
function getUnreadMessages($userId)  
{
```

```
    global $con;  
    global $TimeSignedIn;
```

```
    $stmt = $con->prepare("Select MessageID, Username, Message, TimeSent  
    From tblLogin, tblMessage  
    Where tblMessage.TimeSent >= :timesignedin  
    And tblLogin.UserID = tblMessage.UserID  
    And Not tblLogin.UserID = :ID  
    And tblMessage.MessageID Not In(Select MessageID From tblMessageLog Where UserID = :ID)");
```

```
    $stmt->bindValue(':ID', $userId, PDO::PARAM_INT);  
    $stmt->bindValue(':timesignedin', $TimeSignedIn);
```

```
    $stmt->execute();
```

```
//This query is exactly the same as the previous one, but this time we're not returning the count  
//We're returnign the actual contents of the message.
```

```
$result = array();  
$msgIDs = array();
```

```
while($r = $stmt->fetch(PDO::FETCH_ASSOC))  
{  
    $result[] = $r;  
    $msgIDs[] = $r["MessageID"];  
}
```

```

//So while PDO its still reading the results, we add each result it reads into an array
//We also add the MessageIDs of the data it reads to an array. So After that we can then add t
he
//Message ID to the MessageLog table, to show it has been read.

echo json_encode($result);

//We then encode the result, and echo it out as JSON data.

for($i = 0; $i < sizeof($msgIDs) ; $i++)
    addMessageLog($userId, $msgIDs[$i]);

//We then loop through each msgID we have, and add the message ID and userID to the messagelo
g table
//this then lets the system know the message has been read, so it will never pull it up again
}

function addMessageLog($userId, $messageId)
{
    global $con;

    $stmt = $con->prepare(
"Insert Into tblMessageLog(MessageID, UserID) Values(:msgID, :userID)");

//We could of used a mass Insert statement, but for tutorial purposes, let's keep it nice and simple

    $stmt->bindValue(':userID', $userId, PDO::PARAM_INT);
    $stmt->bindValue(':msgID', $messageId, PDO::PARAM_STR);

    //It adds the userid and messageid and executes. simple.

    $stmt->execute();
}

function validUserID($userId)
{
    global $con;

    $stmt = $con->prepare("Select * From tblLogin Where UserID = :ID");

    $stmt->bindValue(':ID', $userId, PDO::PARAM_INT);
    $stmt->execute();

    //We submit the query, and we check the row count. if the row count = 1 that means the user
    //exists else we return false

    if ($stmt->rowCount() == 1)
    {
        return true;
    }
    else

```

```
{  
    return false;  
}  
}  
?>
```

I thought to mention, the JSON data that is being outputted for each message would look like this. And our vb.net application would read that and convert it into an object.

But in our case, there will be a lot of those.

okay so now we have finished our server side scripting/coding. And we now go to our vb.net application to finish off.

+Haze - January 3, 2015, 12:25 pm

VB.NET Form Designing & Code

So once again, head over to visual studio to complete our client side of the program.

Overview of our VB.NET application

Spoiler ([Click to View](#))

Go to to learn how we're going to create our login & register form.

<https://hackforums.net/showthread.php?tid=4610630>

First we need to create our two classes that will be very important in our program.

base.vb and chatclient.vb

the chatclient is a class we use send commands to our php server using a simple class with methods and functions. It's just like when you do. Console.WriteLine()

Console is a class, and WriteLine() is a method. And you put the string you want to show and it prints it to the console. What I'm doing is replicating a system like that. Where we could do similar things with our program. We call the client class, put in the information and the class will execute and return the response.

Okay so first create our base.vb file and here's the code that goes in it.

Base.vb Class Code

Spoiler ([Click to View](#))

So heres our chatclient.vb code. I will post the images instead.
but before we code this bit, make sure you have installed the JSON.NET Nuget package from the Nuget Package Manager on visual studio.

Install JSON.NET NuGet Package

ChatClient.vb Code

Spoiler [\(Click to View\)](#)

Please may I add, when you login, if you're successful our webserver will show this. And we read that information and store it in our user variable. The server shows us our userId, If we're banned or not, and the time we signed in. Which will be all very import.

Explanation of GetUnReadMessages

Explanation of what this function does. So, if our webserver echos data that looks like

What the Dim msgObj() As Message = JsonConvert.DeserializeObject(Of Message())(currentResponse) code is doing is tranlating that JSON data into an array of our custom message object. And returning it so it could be used to add to our chat box.

Login & Registration Form UI Designing

Spoiler [\(Click to View\)](#)

Chat Room UI Design

Spoiler ([Click to View](#))

Login & Registration Form Code

Spoiler ([Click to View](#)) Code Imports System.Net
Imports Storm_Chat_Room.Base
Imports System.Text.RegularExpressions

Public Class frmLogin

```
Private Sub btnLogin_Click(sender As Object, e As EventArgs) Handles btnLogin.Click
    'We first check if the two textboxes contain any empty characters, if they do we alert the user
    'else we allow them to login
```

```
    If String.IsNullOrEmpty(txtUsername.Text) Or
String.IsNullOrEmpty(txtPassword.Text) Then
        MessageBox.Show("You cannot leave the username or password fields blank")
    Else
        Try
            Dim message As LoginMessage = Client.Login(txtUsername.Text, txtPassword.Text)

            Select Case message
            Case LoginMessage.Success
                MessageBox.Show("Login approved, welcome back.")

                MyUser.Username = txtUsername.Text
                ShowChatForm()
                ClearControls()
                Client.AddSignInMsg(MyUser.Username)

            Case LoginMessage.NotExist
                MessageBox.Show("Either your username or password is incorrect")
            Case LoginMessage.Banned
                MessageBox.Show("Sorry your account has been banned")
            Case LoginMessage.Error
                MessageBox.Show("An error has occurred trying to log your account in")
            End Select

        Catch ex As Exception
            MessageBox.Show("An error has occurred trying to log you in: " & ex.Message)
        End Try

    End If

End Sub
```

```

Private Sub txtPassword_KeyDown(sender As Object, e As KeyEventArgs) Handles
txtPassword.KeyDown
    If e.KeyCode = Keys.Enter Then
        e.SuppressKeyPress = True
        btnLogin.PerformClick()
    End If
End Sub

```

```

Private Sub ShowChatForm()
    frmChatRoom.Show()
    Me.Hide()
End Sub

```

```

Private Sub ClearControls()

```

```

    For Each txtbx As TextBox In Me.TabRegister.Controls.OfType(Of TextBox)()
        txtbx.Clear()
    Next

```

```

    For Each txtbx As TextBox In Me.TabLogin.Controls.OfType(Of TextBox)()
        txtbx.Clear()
    Next

```

```

'So we loop through each tab page on our login Form, and filter through the textboxes only
using the OfType
End Sub

```

```

Private Sub btnRegister_Click(sender As Object, e As EventArgs) Handles btnRegister.Click

```

```

    If String.IsNullOrEmpty(txtRegUsername.Text) Or
String.IsNullOrEmpty(txtRegPassword.Text) Then
        MessageBox.Show("You cannot leave the username or password fields blank")
    Else
        If txtRegPassword.Text = txtRegConfirm.Text Then
            Try
                If Client.RegisterSuccess(ParseInfo(txtRegUsername.Text),
ParseInfo(txtRegPassword.Text)) Then
                    MessageBox.Show("Registration complete. Please login")
                Else
                    MessageBox.Show("Registration unsuccessful")
                End If
            Catch ex As Exception
                MessageBox.Show("An error has occurred trying to register you : " & ex.Message)
            End Try
        Else
            MessageBox.Show("Passwords do not match. Try again!")
        End If
    End If
End Sub

```

```

Private Function ParseInfo(txt As String) As String

```

```
Return Regex.Replace(txt, "[^a-zA-Z0-9]", String.Empty)
'what this does is to replace every non alphabetical item found in the string passed to it.
'so that means our username and password can only contain alphabetic letters and numbers
End Function
End Class
```

Chat room form code

```
Spoiler (Click to View)Code Imports Storm_Chat_Room.Base
Imports System.Threading
Imports System.Text.RegularExpressions
```

```
Public Class frmChatRoom
```

```
Private Event MessageRecieved As EventHandler
Private Event SpeedIntervalChanged As EventHandler
```

'We create our own custom event, and this event will be fired each time the system detects the user

'has a new message or when the speedInterval has changed and then we can create an event handler

'to do a certain action every time an event is raised.

```
Private SpeedInterval As Integer
Private UnreadMsgCount As Integer
```

```
Private Sub FormChatRoom_Load(sender As Object, e As EventArgs) Handles MyBase.Load
ShowGreetingMessage()
```

```
tmrNewMsg.Start()
End Sub
```

```
Private Sub ShowGreetingMessage()
lstChatBox.Items.Clear()
```

```
lblGreetings.Text = "Welcome back, " & MyUser.Username
```

```
AddMsgToLst("Welcome to Tango chatroom!")
AddMsgToLst("Connected to chat server succesfully!")
End Sub
```

```
Private Sub ChatRoom_Shown(sender As Object, e As EventArgs) Handles Me.Shown
UnreadMsgCount = 0
End Sub
```

```
Private Sub Form1_Resize(sender As Object, e As EventArgs) Handles MyBase.Resize
If WindowState = FormWindowState.Minimized Then UnreadMsgCount = 0
End Sub
```

```
Private Sub FormChatRoom_FormClosed(sender As Object, e As FormClosedEventArgs)
Handles Me.FormClosed
Client.AddSignOutMsg(MyUser.Username)
```

```
Application.Exit()  
End Sub
```

```
Private Sub btnLogout_Click(sender As Object, e As EventArgs) Handles btnLogout.Click  
Me.Dispose()  
'By disposing of our Form, we release all its objects and the data involved.
```

```
Client.AddSignOutMsg(MyUser.Username)  
frmLogin.Show()  
End Sub
```

```
Private Sub tmrNewMsg_Tick(sender As Object, e As EventArgs) Handles tmrNewMsg.Tick  
ThreadPool.QueueUserWorkItem(Sub() If Client.UnreadMessagesCount(MyUser.UserID) >= 1  
Then RaiseEvent MessageRecieved(Me, EventArgs.Empty))  
'So everytime this timer ticks, we use a threadpool to queue some work we want it to do.  
'So, we Check if the use our client class we made. We check if the UnreadMessage count is  
greater than or equal  
'To 1, if it is, we then raise our custom event. MessageRecieved  
End Sub
```

```
Private Sub Me_MessageArrived(sender As Object, e As EventArgs) Handles  
Me.MessageRecieved  
ThreadPool.QueueUserWorkItem(Sub() RetrieveUnreadMsg())  
'When this event is raised we then do the work in the RetrieveUnreadMsg() Sub  
End Sub
```

```
Private Sub RetrieveUnreadMsg()  
Try  
Dim msgList() As Message = Client.GetUnreadMessages(MyUser.UserID)
```

```
'So this gets an array of the message object we created in our base class.
```

```
If IsNothing(msgList) = False Then  
    'We check if the msgList has data in it, if it does we then add the data to our listbox  
  
    Invoke(Sub() lstChatBox.Items.AddRange(msgList))  
    Invoke(Sub() lstChatBox.TopIndex = lstChatBox.Items.Count - 1)  
    'This is used to scroll to the last added item in our listbox  
  
    If WindowState = FormWindowState.Minimized Then  
        UnreadMsgCount += msgList.Count  
        Invoke(Sub() ShowBallon("You have " & UnreadMsgCount & " unread message(s)",  
"Message", 100, ToolTipIcon.Info))  
    End If
```

```
End If  
Catch ex As Exception  
Invoke(Sub() ShowBallon(ex.Message, "Error occured", 100, ToolTipIcon.Error))  
End Try  
End Sub
```

```
Private Sub txtMessage_KeyDown(sender As Object, e As KeyEventArgs) Handles  
txtMessage.KeyDown
```

```
If e.KeyCode = Keys.Enter Then
    e.SuppressKeyPress = True
    btnSend.PerformClick()
End If
End Sub
```

```
Private Sub btnSend_Click(sender As Object, e As EventArgs) Handles btnSend.Click
    If String.IsNullOrEmpty(txtMessage.Text) = False Then
```

```
        If Client.SendMessage(txtMessage.Text) Then
            AddMsgToLst(txtMessage.Text)
            txtMessage.Clear()
        Else
            MessageBox.Show("Error sending your message boi")
        End If
```

```
    End If
End Sub
```

```
Public Sub AddMsgToLst(msg As String)
    lstChatBox.Items.Add(String.Format("[{0}] {1}", Date.Now, msg))
    lstChatBox.TopIndex = lstChatBox.Items.Count - 1
End Sub
```

```
Private Sub MaskedTextBox1_TextChanged(sender As Object, e As EventArgs) Handles
MaskedTextBox1.TextChanged
```

```
    Dim currentInterval As Integer
```

```
    If Integer.TryParse(MaskedTextBox1.Text, currentInterval) Then
```

```
        If currentInterval >= 500 Then
```

```
            SpeedInterval = currentInterval
```

```
            RaiseEvent SpeedIntervalChanged(Me, EventArgs.Empty)
```

```
        Else
```

```
            ShowBallon("Values can't be lower than 500", "Alert", 100, ToolTipIcon.Info)
```

```
        End If
```

```
    End If
```

```
End Sub
```

```
Private Sub Me_SpeedChanged(sender As Object, e As EventArgs) Handles
Me.SpeedIntervalChanged
```

```
    tmrNewMsg.Interval = SpeedInterval
```

```
End Sub
```

```
Private Sub lstChatBox_MouseDown(sender As Object, e As MouseEventArgs) Handles
lstChatBox.MouseDown
```

```
    If e.Button = Windows.Forms.MouseButtons.Right Then
```

```
        ChatContextMenuStrip.Show(MousePosition)
```

'This is used so when you right click on the listbox, it will Show a context menu, showing the items

'Copy and Remove

```
End Sub
```

```
Private Sub MenuItemCopy_Click(sender As Object, e As EventArgs) Handles  
MenuItemCopy.Click
```

```
'When we click copy, this is what happens.
```

```
If lstChatBox.SelectedIndex > -1 Then
```

```
Dim selectedMsg As Message = TryCast(lstChatBox.SelectedItem, Message)
```

```
If IsNothing(selectedMsg) = False Then
```

```
My.Computer.Clipboard.SetText(selectedMsg.Message)
```

```
Else
```

```
My.Computer.Clipboard.SetText(lstChatBox.SelectedItem.ToString())
```

```
End If
```

```
ShowBallon("Message has been copied to your clipboard", "Copied", 100, ToolTipIcon.Info)
```

```
End If
```

```
End Sub
```

```
Private Sub MenuItemRemove_Click(sender As Object, e As EventArgs) Handles  
MenuItemRemove.Click
```

```
'When we click remove, this is what happens
```

```
If lstChatBox.SelectedIndex > -1 Then lstChatBox.Items.RemoveAt(lstChatBox.SelectedIndex)
```

```
End Sub
```

```
Private Sub ShowBallon(text As String, title As String, timeout As Integer, icon As ToolTipIcon)
```

```
'This sub is handy for when we want to show our NotificationIcon
```

```
NotifyIconMsg.ShowBalloonTip(timeout, title, text, icon)
```

```
End Sub
```

```
End Class
```

Video of final product

Spoiler ([Click to View](#))

[Video: https://www.youtube.com/watch?v=qm5_VIADDxs&feature=youtu.be]

Download Link to everything covered in this tutorial

The whole vb.net solution files, .sql data, php source codes. Everything you need to set this up in 5 minutes.

<https://www.mediafire.com/?0pfe595n9v0apko>

Credits

Spoiler ([Click to View](#))

Though I coded everything here myself, I got the inspiration to do this, and I also got help from other users in some minor PHP help and I got help from members that helped me test it in the video.

Credits to:

[Keiko](#)

[Entire](#)

[Ombra](#)

[Odus](#)

[Electrolek](#) - for inspiring the idea.

Marwix - January 3, 2015, 12:30 pm

Vouch for my Nigga +Haze. Very good tutorial. Makes me comfortable to see my name in the credits spoiler.

Continue with your work!

Entire - January 3, 2015, 12:32 pm

This is actually an amazing tutorial. Very details etc.. Going to try it now ;D

sha110w - January 3, 2015, 12:40 pm

Why would you keep and read messages in/from a database for this instead of sending/reading them in real time between the server and client? If you want to keep a log, just log them in a file on the server.

Shoot - January 3, 2015, 12:44 pm

nice tutorial!

i wanna learn how to make a video chat (like skype) using VB.NET but just video & text nothing really extra

+Haze - January 3, 2015, 12:46 pm

(Jan 3, 2015 01:40 PM)sha110w Wrote: [\[CITE\]](#) Why would you keep and read messages in/from a database for this instead of sending/reading them in real time between the server and client? If you want to keep a log, just log them in a file on the server.

I have no idea what you're talking about. Either way, if you want to discuss why I didn't do so and so, get enough posts & PM me. I'd like to keep this thread as on topic as possible. The minute you go offtopic, I'll just ask admins to remove it, I've had enough of shitstorms being spawned on my threads.

(Jan 3, 2015 01:44 PM)Shoot Wrote: [\[CITE\]](#) nice tutorial!

i wanna learn how to make a video chat (like skype) using VB.NET but just video & text nothing really extra

Thanks for the nice messages above. For this you'll need to learn ALOT, and I mean ALOT. Like Sockets, algorithms used for compressing data, Networking. Databases, OOP etc.

sha110w - January 3, 2015, 12:49 pm

(Jan 3, 2015 01:46 PM)+Haze Wrote: [\[CITE\]](#) I have no idea what you're talking about.

What do you mean you have no idea what I'm talking about? YOU wrote this code didn't you? YOU understand what this code does don't you?

Ombra - January 3, 2015, 12:49 pm

This is a very nice chat system, i hope to see private messages implemented in a next release ?

Marwix - January 3, 2015, 12:58 pm

(Jan 3, 2015 01:40 PM)sha110w Wrote: [\[CITE\]](#) Why would you keep and read messages in/from a database for this instead of sending/reading them in real time between the server and client? If you want to keep a log, just log them in a file on the server.

I don't think you know what you're talking about. He's using another way to store logs and such. Nothing to worry about if you ask me. The only thing that matters right now is that the chat system works which it obviously does.

sha110w - January 3, 2015, 1:01 pm

(Jan 3, 2015 01:58 PM)Keiko Wrote: [\[CITE\]](#) I don't think you know what you're talking about. He's using another way to store logs and such. Nothing to worry about if you ask me. The only thing that matters right now is that the chat system works which it obviously does.

```
Code function addMessageLog($userId, $messageId)
```

```
{
    global $con;
```

```
    $stmt = $con->prepare("Insert Into tblMessageLog(MessageID, UserID) Values(:msgID, :userID)");
```

```
    //We could of used a mass Insert statement, but for tutorial purposes, let's keep it nice and simple
```

```
    $stmt->bindValue(':userID', $userId, PDO::PARAM_INT);
    $stmt->bindValue(':msgID', $messageId, PDO::PARAM_STR);
```

```
    //It adds the userid and messageid and executes. simple.
```

```
    $stmt->execute();
}
```

Oh yea...you're right...totally not adding messages to a database...what do I know? How silly of me.

\$lick - January 3, 2015, 1:05 pm

(Jan 3, 2015 01:58 PM)Keiko Wrote: [\[CITE\]](#) I don't think you know what you're talking about. He's using another way to store logs and such. Nothing to worry about if you ask me. The only thing that matters right now is that the chat system works which it obviously does.

Did you even read the thread? If he wasn't using a database then SQL wouldn't be needed at all.

A simple JS -> VB sockets connection would be fine.

+Haze - January 3, 2015, 1:09 pm

(Jan 3, 2015 02:05 PM)Lazy. Wrote: [\[CITE\]](#) Did you even read the thread? If he wasn't using a database then SQL wouldn't be needed at all.

A simple JS -> VB sockets connection would be fine.

I'm aware. I stated this in the first few lines of this thread.

Spoiler [\(Click to View\)](#)

Anyway no more off topic posts please..

\$lick - January 3, 2015, 1:11 pm

(Jan 3, 2015 02:09 PM)+Haze Wrote: [\[CITE\]](#) I'm aware. I stated this in the first few lines of this thread.

Spoiler [\(Click to View\)](#)

Anyway no more off topic posts please..
I didn't even aim my comment towards you or your thread.

I was replying to Keiko.

EDIT: Also while I'm at it, I'd like to mention that even if you are trying something new, you really shouldn't be teaching this sort of stuff. It's nothing more than a security risk and slow as hell.

+Haze - January 3, 2015, 1:12 pm

@Lazy - alright fair enough.

@Shadow. PM me if you want to discuss whatever it is your talking about.

No more off topic. Thanks.

\$lick - January 3, 2015, 1:14 pm

(Jan 3, 2015 02:12 PM)+Haze Wrote: [\[CITE\]](#) No more off topic. Thanks.
All the comments were on topic... the topic is your thread...

sha110w - January 3, 2015, 1:15 pm

(Jan 3, 2015 02:09 PM)+Haze Wrote: [\[CITE\]](#) I'm aware. I stated this in the first few lines of this thread.

Spoiler [\(Click to View\)](#)[img]

Anyway no more off topic posts please..

If you wanted to "do something different," why not do a TCP/IP chat client/server tutorial that shows how to do it globally? eg: run a server service from your server by opening a desired port on your server and let the client connect to it and then participants can chat that way?

You could have still included the SQL stuff for user storage and registration, but logging messages in a database is totally unnecessary.

Also none of this appears to be encrypted, so there's also that.

+Haze - January 3, 2015, 1:18 pm

(Jan 3, 2015 02:15 PM)sha110w Wrote: [\[CITE\]](#) If you wanted to "do something different," why not do a TCP/IP chat client/server tutorial that shows how to do it globally? eg: run a server service from your server by opening a desired port on your server and let the client connect to it and then participants can chat that way?

You could have still included the SQL stuff for user storage and registration, but logging messages in a database is totally unnecessary.

Also none of this appears to be encrypted, so there's also that.

Actually it's because I didn't know how that would be done, I'd have to read up on it. Like the whole connect to server and receive message like the way you described it. But now I get what you're talking about, and yeah makes sense. Though, I just didn't know how to do that and I'm gonna be very busy in the next 2 weeks with IRL stuff. So I don't really have the time. So I thought why not just do it this way. Also, I addressed why I did not not encode the messages.

When I do get free time though, I will definitely educate myself and learn to do it the way you had said.

Only3 - January 3, 2015, 1:21 pm

(Jan 3, 2015 01:44 PM)Shoot Wrote: [\[CITE\]](#) nice tutorial!

i wanna learn how to make a video chat (like skype) using VB.NET but just video & text nothing really extra

? i made skype like program (desktop and webcam view) maybe one day i will share it

Nikey646 - January 3, 2015, 1:24 pm

(Jan 3, 2015 02:15 PM)sha110w Wrote: [\[CITE\]](#) If you wanted to "do something different," why not do a TCP/IP chat client/server tutorial that shows how to do it globally? eg: run a server service from your server by opening a desired port on your server and let the client connect to it and then participants can chat that way?

You could have still included the SQL stuff for user storage and registration, but logging messages in a database is totally unnecessary.

Also none of this appears to be encrypted, so there's also that.

From my understanding this tutorial is aimed for beginners who have the basic concepts of VB.Net down. What you are talking about is an application that one would look into after taking a class/course/lesson on Sockets, which an entire different set of skills beyond parsing, logic, and "new WebClient().DownloadString(string.Format(""))"

On Topic: Glad to see no obvious exploits within this system, beyond lack of security for the transferred data (SSL/Encryption), and a slightly insecure hashing system for passwords, that for the current tutorial is more than good enough.

If you do expand it, i would like to see you use more Third Party resources, rather than re-inventing the wheel, EG Using the Blowfish implementation for password hashing (Includes a Hash_Check function, allowing much more cleaner and secure registration and login api's), make use of json_encode outside of Messages API, and creating a uniform/universal class to send all info to (It may require unsafe types, EG: Handling values as Objects, Casting them to their respective .GetType() values, or using the VB.Net Alternative of the C# "Dynamic" type), rather than having a class-per-api structure, that can quite easily cause problems when updating the API/Client.

+Haze - January 3, 2015, 1:26 pm

(Jan 3, 2015 02:24 PM)Nikey646 Wrote: [\[CITE\]](#) From my understanding this tutorial is aimed for beginners who have the basic concepts of VB.Net down. What you are talking about is an application that one would look into after taking a class/course/lesson on Sockets, which an entire different set of skills beyond parsing, logic, and "new WebClient().DownloadString(string.Format(""))"

On Topic: Glad to see no obvious exploits within this system, beyond lack of security for the transferred data (SSL/Encryption), and a slightly insecure hashing system for passwords, that for the current tutorial is more than good enough.

If you do expand it, i would like to see you use more Third Party resources, rather than re-inventing the wheel, EG Using the Blowfish implementation for password hashing (Includes a Hash_Check function, allowing much more cleaner and secure registration and login api's), make use of json_encode outside of Messages API, and creating a uniform/universal class to send all info to (It may require unsafe types, EG: Handling values as Objects, Casting them to their respective .GetType() values, or using the VB.Net Alternative of the C# "Dynamic" type), rather than having a class-per-api structure, that can quite easily cause problems when updating the API/Client.

Thanks mate, and yes I would love to try something like that soon. But I still got a lot of learning to do.

SIM-1847363 - January 3, 2015, 1:53 pm

Dang talk about a HQ tutorial, nice work +haze great work. Love your tutorial.

Edit: Hope to see the second part of the tutorial ;P.

Known Alias - January 3, 2015, 3:28 pm

Once again man great work! You always bring top notch shit!!

programtesting28 - January 3, 2015, 3:31 pm

WOW, nice tutorial dude... I LIKE IT, thanksssss, and I vote Yes for the next tutorial :D

BTW can you make a .php licensing system ? :D

+Haze - January 3, 2015, 3:48 pm

(Jan 3, 2015 02:53 PM)Nettro Wrote: [\[CITE\]](#) Dang talk about a HQ tutorial, nice work +haze great work. Love your tutorial.

Edit: Hope to see the second part of the tutorial ;P.

Thanks Netro. Much appreciated.

(Jan 3, 2015 04:28 PM)Known Alias Wrote: [\[CITE\]](#) Once again man great work! You always bring top notch shit!!

Thanks Alias.

(Jan 3, 2015 04:31 PM)programtesting28 Wrote: [\[CITE\]](#) WOW, nice tutorial dude... I LIKE IT, thanksssss, and I vote Yes for the next tutorial :D

BTW can you make a .php licensing system ? :D

Yes I can but I don't think I will be doing that. Netseal does a great job. But when I do expand my knowledge, I could introduce a licensing system like the one on Netseal which shouldn't be too different.

programtesting28 - January 3, 2015, 3:53 pm

(Jan 3, 2015 04:48 PM)+Haze Wrote: [\[CITE\]](#) Yes I can but I don't think I will be doing that. Netseal does a great job. But when I do expand my knowledge, I could introduce a licensing system like the one on Netseal which shouldn't be too different.

Okaayyy...

I'm waiting too for your licensing system tutorial dude ?

+Haze - January 3, 2015, 3:55 pm

(Jan 3, 2015 04:53 PM)programtesting28 Wrote: [\[CITE\]](#) Okaayyy...

I'm waiting too for your licensing system tutorial dude ?

I won't do it as a tutorial lol, I'd most likely sell it.

programtesting28 - January 3, 2015, 4:01 pm

(Jan 3, 2015 04:55 PM)+Haze Wrote: [\[CITE\]](#) I won't do it as a tutorial lol, I'd most likely sell it.

Ohh, okay... I'm sorry about that ?

+Haze - January 3, 2015, 5:13 pm

No program, glad people like this. I might create a V2 to this.

nuk - January 3, 2015, 5:17 pm

This is a very good beginners tutorial! As a fellow developer; I see more requests for stuff like this than anything else. I see a load of work went into making this thread/tutorial and I have to give you mad props for taking the time to make this - especially making it very easy to follow along with ;) Keep this kinds of quality stuff coming as it is greatly appreciated by lots. In fact, I have someone asking me to show them something just like this and (hopefully with your permission) I'm just going

to link them to this. Great share OP!

+Haze - January 3, 2015, 5:26 pm

(Jan 3, 2015 06:17 PM)nuk Wrote: [\[CITE\]](#) This is a very good beginners tutorial! As a fellow developer; I see more requests for stuff like this than anything else. I see a load of work went into making this thread/tutorial and I have to give you mad props for taking the time to make this - especially making it very easy to follow along with ;) Keep this kinds of quality stuff coming as it is greatly appreciated by lots. In fact, I have someone asking me to show them something just like this and (hopefully with your permission) I'm just going to link them to this. Great share OP!

Yes it took me couple days to refine the code, and write up this tutorial in a whole, but I had fun making it and I learnt a lot too. And sure, I'll PM you.

Classified™ - January 3, 2015, 7:11 pm

By far one of the best tutorials made. I really wanted to see this and I'm so grateful to +Haze for considering my wishes and delivering. The tutorial is clear and concise! Great work Haze!

Fajitas - January 4, 2015, 3:47 am

This is an amazing tutorial. thank you man

+Haze - January 4, 2015, 7:41 am

Thanks for the nice messages, much appreciated. You should vote on whether you want to see me create a part 2 to this tutorial.

SIM-1456245 - January 4, 2015, 7:47 am

First thought, DAAAMNN NIGGZZ. That tutroial is huge man. You can clearly see that A LOT of effort has been put into it. Didn't have time to see the code, but I doubt that it's LQ. Vouch for +Haze, didn't expect this much from you to be honest.

pr3fix - January 4, 2015, 7:54 am

Good job, but sha1low has a point. Why bother storing the messages in database? I mean, it would make sense if you'd want to somehow keep on tabs on what people are saying, or perhaps show last 50 messages or so that people have posted, but storing everything is a waste of space. But good tutorial, nonetheless.

+Haze - January 4, 2015, 8:04 am

(Jan 4, 2015 08:47 AM)Electroalek Wrote: [\[CITE\]](#) First thought, DAAAMNN NIGGZZ. That tutroial is huge man. You can clearly see that A LOT of effort has been put into it. Didn't have time to see the code, but I doubt that it's LQ. Vouch for +Haze, didn't expect this much from you to be honest. Thanks dude.

(Jan 4, 2015 08:54 AM)pr3fix Wrote: [\[CITE\]](#) Good job, but sha1low has a point. Why bother storing the messages in database? I mean, it would make sense if you'd want to somehow keep on tabs on what people are saying, or perhaps show last 50 messages or so that people have posted, but storing everything is a waste of space. But good tutorial, nonetheless.

Uhm I'm very much aware of this, I did address to Sha1low why I chose this method. You can read my response to him on it.

CodeBinary - January 4, 2015, 8:04 am

Daym, this is truly dedication, sknowledge, helpful, by far one of the best well written tutorials alive. If I was leet/uber you would of been reped by now.

+Haze - January 4, 2015, 11:53 am

Thanks for that Multi bills. Glad some of you enjoyed it.

GramHQ - January 4, 2015, 5:51 pm

This is great, Haze. Please make a v2 with a friend system, pm system, custom profiles maybe, and other features :).

+Haze - January 5, 2015, 9:03 am

Thanks for that mate, and yeah I might do the v2

Xghost7 - January 5, 2015, 10:17 am

Another awesome tutorial coming from you, well done mate.

I did something similar to this 1 month ago and it went pretty smooth, i would really like a TCP socket tutorial though, take that as a request.

+Haze - January 5, 2015, 10:48 am

Uhhmm, thanks. But you do know these things take time, and I have to learn it before I can teach it. Plus I'm very busy with IRL stuff so yee.

Felicien - January 5, 2015, 1:15 pm

Nice tutorial

It is possible to see the users connecting?

+Haze - January 5, 2015, 1:17 pm

Yes it shows you when a user connects to the chat room & disconnects. Just make sure when you import the chatroom_db.sql

Leave the system account. Because that's what that echos when a user connects or disconnects

Jack_Soff - January 6, 2015, 5:09 pm

Can someone help me set this up. I don't understand why its not working. Every time I go to register it gives me an error. I edited the db.php file, imported the .sql file, and edited chatclient.vb and it still gives me an error.

SHRAPNEL - January 6, 2015, 5:12 pm

Thanks for this a ton, +Haze! I'm looking forward to more tutorials, hopefully an expansion on this one with more advanced features like friends, PMs, usergroups, advanced banning/moderation etc...

+Haze - January 7, 2015, 6:35 pm

Thanks, and I'm currently working on the expansion. Just finished working on admin system.

Tedd - January 7, 2015, 6:59 pm

(Jan 7, 2015 07:35 PM)+Haze Wrote: [\[CITE\]](#) Thanks, and I'm currently working on the expansion. Just finished working on admin system.

Seems like a user could just put timeSignedIn as 0 and request the "getunread.php" a couple hundred times and kill the server.

+Haze - January 7, 2015, 7:13 pm

(Jan 7, 2015 07:59 PM)Tedd Wrote: [\[CITE\]](#) Seems like a user could just put timeSignedIn as 0 and request the "getunread.php" a couple hundred times and kill the server.

Aha, I didn't even think of that. I should of validated the TimeSignedIn. But the idea is that you only access the API through the program which will do everything correctly.

Tedd - January 7, 2015, 7:18 pm

(Jan 7, 2015 08:13 PM)+Haze Wrote: [\[CITE\]](#) Aha, I didn't even think of that. I should of validated the

TimeSignedIn. But the idea is that you only access the API through the program which will do everything correctly.

Just add a variable to the user data for "lastChecked".

When you log in, update lastChecked to their log in timestamp, and when they visit "getunread.php," grab messages after that time, along with updating lastChecked to the current time they got the unread messages. Then there is no user input for time, removing any need for posting that data or securing it.

+Haze - January 7, 2015, 7:25 pm

(Jan 7, 2015 08:18 PM)Tedd Wrote: [\[CITE\]](#) Just add a variable to the user data for "lastChecked".

When you log in, update lastChecked to their log in timestamp, and when they visit "getunread.php," grab messages after that time, along with updating lastChecked to the current time they got the unread messages. Then there is no user input for time, removing any need for posting that data or securing it.

Yup yup yup yup, I get you. This is actually a good idea, thanks. ? If I do create the V2 of this tutorial. I will definitely change that.

Do you know how this chat server could be run using sockets & making it global.

Tedd - January 7, 2015, 7:50 pm

(Jan 7, 2015 08:25 PM)+Haze Wrote: [\[CITE\]](#) Yup yup yup yup, I get you. This is actually a good idea, thanks. ? If I do create the V2 of this tutorial. I will definitely change that.

Do you know how this chat server could be run using sockets & making it global.

Pretty easy actually, although I doubt anyone would want you to make that because that would require a dedicated server and I doubt the type of people on this section would have money for that.

There's one other way I know works and doesn't require a dedicated server, and that's AJAX, however I have never looked at that to see how it works. It is basically a javascript function that continually polls a database, and if it finds new data adds it. The best and most professional/expandable version that I know of is BlueIMP AJAX Chat. I don't know how you would incorporate that into visual basic though, since nothing that I know of supports javascript.

I can teach you how to do the first one though, you would pick it up in a heartbeat if you made this by yourself.

+Haze - January 7, 2015, 7:54 pm

(Jan 7, 2015 08:50 PM)Tedd Wrote: [\[CITE\]](#) Pretty easy actually, although I doubt anyone would want you to make that because that would require a dedicated server and I doubt the type of people on this section would have money for that.

There's one other way I know works and doesn't require a dedicated server, and that's AJAX, however I have never looked at that to see how it works. It is basically a javascript function that continually polls a database, and if it finds new data adds it. The best and most professional/expandable version that I know of is BlueIMP AJAX Chat. I don't know how you would incorporate that into visual basic though, since nothing that I know of supports javascript.

I can teach you how to do the first one though, you would pick it up in a heartbeat if you made this by yourself.

Sure I'll send u a PM right now. And yes I did make it by myself. ?

mienkrsftajz - January 7, 2015, 8:26 pm

This is in the .NET section... create a RESTful web API using ASP.NET, and use a MSSQL

database. C'mon, man!

+Haze - January 7, 2015, 10:26 pm

(Jan 7, 2015 09:26 PM)mienkrsftajz Wrote: [\[CITE\]](#) This is in the .NET section... create a RESTful web API using ASP.NET, and use a MSSQL database. C'mon, man!
Lol wut? What makes you think I know ASP.NET?

mienkrsftajz - January 7, 2015, 10:28 pm

(Jan 7, 2015 11:26 PM)+Haze Wrote: [\[CITE\]](#) Lol wut? What makes you think I know ASP.NET?
I didn't say that. I suggested that, since this is the .NET section, you could have used the .NET platform to it's fullest extent.

Also, ASP.NET is a platform for creating dynamic web applications in C# and Visual Basic. So... yeah.

+Haze - January 7, 2015, 10:30 pm

(Jan 7, 2015 11:28 PM)mienkrsftajz Wrote: [\[CITE\]](#) I didn't say that. I suggested that, since this is the .NET section, you could have used the .NET platform to it's fullest extent.

Also, ASP.NET is a platform for creating dynamic web applications in C# and Visual Basic. So... yeah.

Oh okay, I don't know ASP.NET though ?
I thought you was trying to flame

Tedd - January 7, 2015, 10:32 pm

(Jan 7, 2015 11:28 PM)mienkrsftajz Wrote: [\[CITE\]](#) I didn't say that. I suggested that, since this is the .NET section, you could have used the .NET platform to it's fullest extent.

Also, ASP.NET is a platform for creating dynamic web applications in C# and Visual Basic. So... yeah.

He's not trying to create a web application though..... So what was the point of suggesting ASP.NET?

mienkrsftajz - January 7, 2015, 10:33 pm

(Jan 7, 2015 11:30 PM)+Haze Wrote: [\[CITE\]](#) Oh okay, I don't know ASP.NET though ?
I thought you was trying to flame
I'm not.

And you know VB, so yeah, you kinda already do know it. Since Entity Framework does all the heavy lifting for you when making a web API... you actually barely have to do anything for something like this. ASP.NET would handle all of the backend stuff for you, including the database.

+Haze - January 7, 2015, 10:41 pm

Hmmmmm. I see where you're coming from. Doing this with asp.net would be much better since they can login online and whatnot. I'd love to learn the asp.net skills needed to do this, but I doubt asp.net is identical to vb.net in terms of the libraries

mienkrsftajz - January 7, 2015, 11:14 pm

(Jan 7, 2015 11:41 PM)+Haze Wrote: [\[CITE\]](#) Hmmmmm. I see where you're coming from. Doing this with asp.net would be much better since they can login online and whatnot. I'd love to learn the asp.net skills needed to do this, but I doubt asp.net is identical to vb.net in terms of the libraries
No... they are. ASP.NET simply provides a backend system to create web applications. It still uses Visual Basic (or C#, I use C#) to do it. It's literally just a structured system used to create it. Much

like how Windows Forms are used by Visual Basic to create a GUI, ASP.NET is used to make dynamic web apps.

ZeekoSec - January 8, 2015, 10:58 am

Thank you so much +Haze ! i've been looking for a tutorial like this

+Haze - January 14, 2015, 7:18 am

(Jan 8, 2015 12:14 AM)mienkrsftajz Wrote: [\[CITE\]](#) No... they are. ASP.NET simply provides a backend system to create web applications. It still uses Visual Basic (or C#, I use C#) to do it. It's literally just a structured system used to create it. Much like how Windows Forms are used by Visual Basic to create a GUI, ASP.NET is used to make dynamic web apps.

Sure I see.

(Jan 8, 2015 11:58 AM)ZeekoSec Wrote: [\[CITE\]](#) Thank you so much +Haze ! i've been looking for a tutorial like this

Thanks mate, looking to do a part 2. Though I need suggestions of the new features to add.

BlackLayer - January 14, 2015, 5:28 pm

Good tut* you are becoming professional programme ?

Shoot - January 14, 2015, 6:53 pm

(Jan 14, 2015 06:28 PM)0011101000101001 Wrote: [\[CITE\]](#) Good tut* you are becoming professional programme ?

becoming? he is my inspiration, i saw his tutorials before i joined HF (as a guest), i joined to learn more and get better at it, i love his creations!

+Haze - January 15, 2015, 9:16 am

(Jan 14, 2015 06:28 PM)0011101000101001 Wrote: [\[CITE\]](#) Good tut* you are becoming professional programme ?

Thanks man.

(Jan 14, 2015 07:53 PM)Shoot Wrote: [\[CITE\]](#) becoming? he is my inspiration, i saw his tutorials before i joined HF (as a guest), i joined to learn more and get better at it, i love his creations!

Thanks man, means a lot

#Kat - January 15, 2015, 11:14 am

Threads can be viewed without logging its call Archived Thread Cache phpbb still does it even tho you say no

Anyways.

Did this few weeks ago was async socket based with stream read/write found php to be slow and heavy on the server. Also vulnerable to mysql attacks for sessions. However it does make it way easier to manage and create seperate conferences however in socket based using a hashtable to set and users broadcast to each if user x and user y has requested to join a room they are added to arraylist that data is sent to each user simultaneous .. eh took a few pepsi drinks for me to think of a efficient way of doing it will post snippet if anyone needs?

+Haze - January 15, 2015, 11:48 am

(Jan 15, 2015 12:14 PM)?Lamb Wrote: [\[CITE\]](#) Threads can be viewed without logging its call Archived Thread Cache phpbb still does it even tho you say no

Anyways.

Did this few weeks ago was async socket based with stream read/write found php to be slow and heavy on the server. Also vulnerable to msq1 attacks for sessions. However it does make it way easier to manage and create seperate conferences however in socket based using a hashtable to set and users broadcast to each if user x and user y has requested to join a room they are added to arraylist that data is sent to each user simultaneous .. eh took a few pepsi drinks for me to think of a effiencent way of doing it will post snippet if anyone needs?

That would be lovely actually. I always wanted to know how to do this with sockets. And sure post the snippet, but bare in mind this won't be of much use if it doesn't allow for chatting globally.

+Haze - April 22, 2015, 1:30 pm

Jesus, looking back at this retrospectively, I think I could of improved so much. Oh wells.

Love and Peace - April 29, 2015, 2:45 pm

those this work with SPAM

markzuckerberg - December 27, 2015, 11:13 pm

error in the mediafire link :

<http://www.mediafire.com/download/0pfe595n9v0apko/Storm+Messenger.rar>

SIM-1676753 - December 27, 2015, 11:15 pm

Hi how do I set this up? Thanks.