

Srikanth Gude – sgcnm@umsystem.edu

Git Hub Link - <https://github.com/gudesrikanth/webcourse/tree/main/Webpart/ICP4>

Gopi Nelluri – gng75@umsystem.edu

Git Hub Link -

https://github.com/gopinelluri9/demo_remote_repository/tree/main/WebPart/ICP4

ICP – 4(jQuery, JavaScript)

Introduction:

jQuery is a free JavaScript library that can use to communications between an HTML/CSS document, or especially the Document Object Model (DOM), and JavaScript. Explaining the terms, it clarifies HTML document traversing and manipulation, browser event handling, DOM animations, Ajax interactions, and cross-browser JavaScript development. By using jQuery, we can control HTML, CSS and DOM elements. Developers can build plug-ins on top of the JavaScript library using a wide selection of plug-ins.

JavaScript is commonly used to develop the web pages and it is a client – side scripting language to add dynamic and interactive elements to websites.

Task

GitHub User Finder:

I have integrated the github api using *XMLHttpRequest* and implemented github user finder, where we input the username and fetch the details based on the username and display the details in the html.

Webpage Description:

We have created a webpage GitHub User Finder using various tags. The Website looked like below.

Github User Finder

Enter a username to search..



Name	gudesnikanth
ID	22870866
GitHub URL	https://github.com/users/gudesnikanth
Twitter ID	
Public Repositories	2
Public Gists	0
Followers	0
Following	0
Bio	Graduate student at University of Missouri Kansas city
Created at	2016-10-16T15:27:03Z
Updated at	2022-02-09T21:40:27Z

Code:

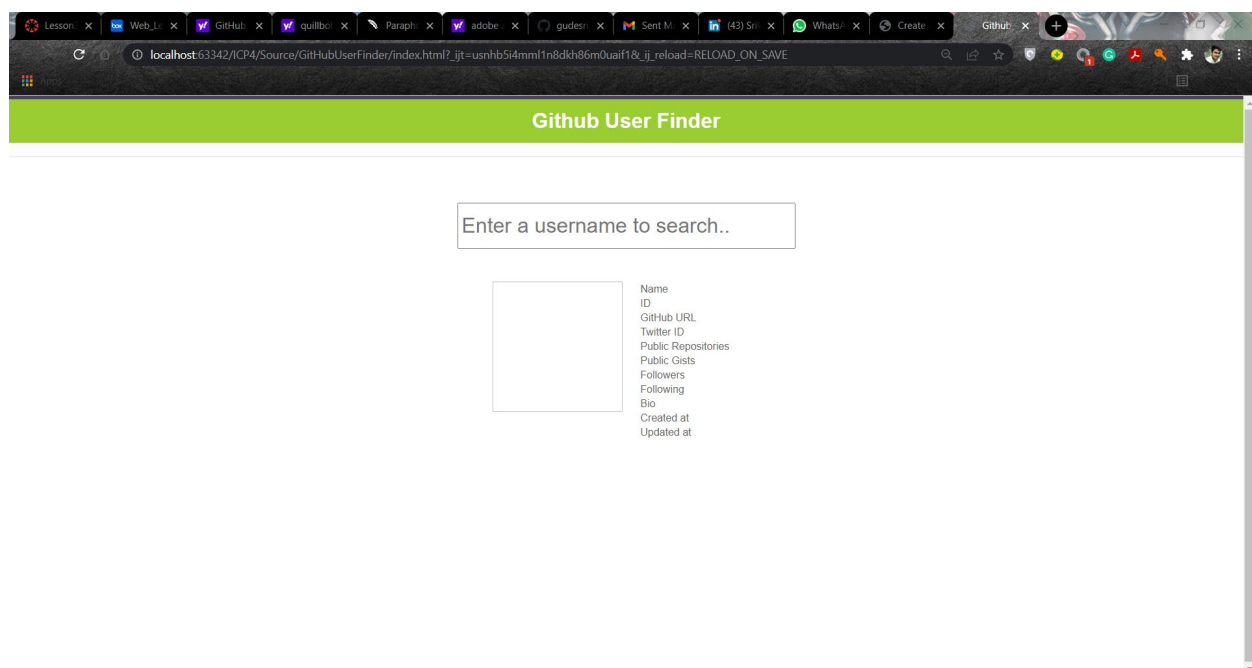
```
<html xmlns:background-color="http://www.w3.org/1999/xhtml">
<head>
  <title>Github User Finder</title>
  <link href="style.css" rel="stylesheet">
  <script src="jquery.js" type="text/javascript"></script>
  <script src="script.js" type="text/javascript"></script>
</head>
<body>
  <header>
    <h1>Github User Finder</h1>
  </header>
  <div id="container">
    <div class="search" style="padding-top: 20px; border-radius: 50px">
      <i class="fa fa-github fa-2x" style="position: relative"></i>
      <input id="username" placeholder="Enter a username to search.." type="text"></input>
    </div>
    <div class="search">
      <div id="profile">
        <div class="avatar">
          <img id="gitimage" style="height: 200px; width: 200px">
        </div>
        <!--Acquiring details from JavaScript based on Element by id-->
        <div class="information" style="padding-left: 20px">
          <table>
            <tr><td>Name</td> <td><span id="gitname" style="padding-left: 40px"></span></td></tr>
            <tr><td>ID</td> <td><span id="gitid" style="padding-left: 40px"></span></td></tr>
            <tr><td>GitHub URL</td> <td><a id="giturl" style="padding-left: 40px"></a></td></tr>
            <tr><td>Twitter ID</td> <td><span id="twitter" style="padding-left: 40px"></span></td></tr>
            <tr><td>Public Repositories</td> <td><span id="gitrepo" style="padding-left: 40px"></span></td></tr>
            <tr><td>Public Gists</td> <td><span id="gitgist" style="padding-left: 40px"></span></td></tr>
```

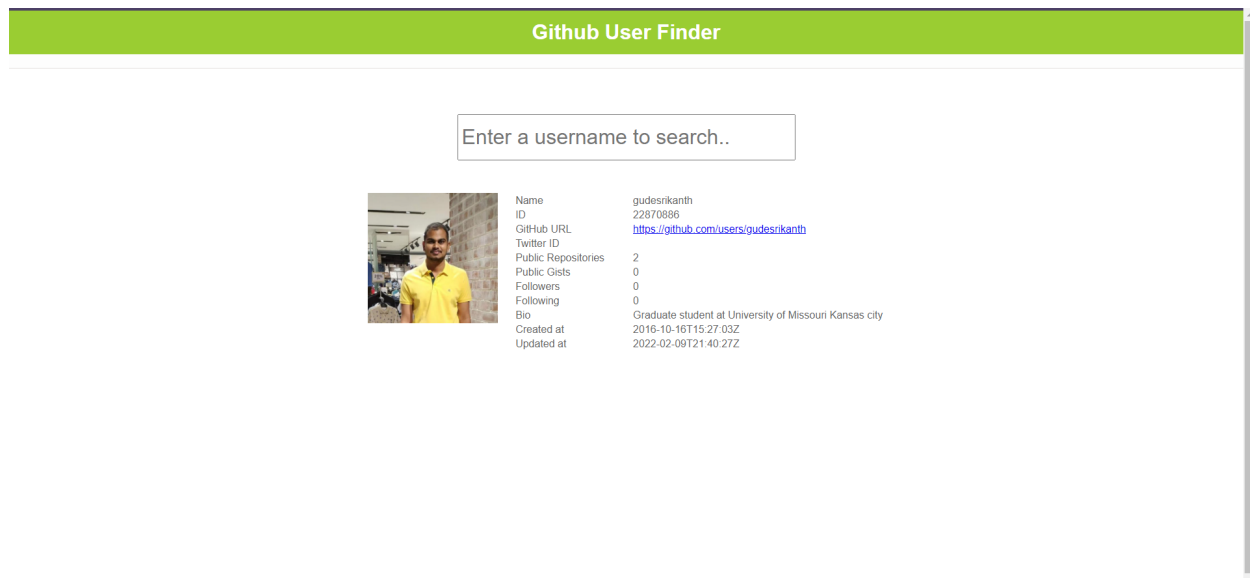
```

        <tr><td>Public Gists</td> <td><span id="gitgist" style="padding-left: 40px"></span></td></tr>
        <tr><td>Followers</td> <td><span id="gitfollowers" style="padding-left: 40px"></span></td></tr>
        <tr><td>Following</td> <td><span id="gitfollowing" style="padding-left: 40px"></span></td></tr>
        <tr><td>Bio</td> <td><span id="gitbio" style="padding-left: 40px"></span></td></tr>
        <tr><td>Created at</td> <td><span id="gitcreated" style="padding-left: 40px"></span></td></tr>
        <tr><td>Updated at</td> <td><span id="gitupdated" style="padding-left: 40px"></span></td></tr>
    </table>
</div>
</div>
</div>
</body>
</html>

```

Output:





In above screenshot output is displayed for various tags explained below and we have used some css styles, jquery, javascript to make it interactive with web elements to get user data from github server.

<h>: It is used for main headings content, we were used 'h1' tag to show the main header content "GitHub User Finder".

<link rel="stylesheet" href=".css"/>: We must use an external style sheet to maintain the style for HTML pages, we can use any text editor can be used to create the external style sheet. The file should be saved with .css extension, then only we can use the file for styles.

<script>: By using this tag we can call the external java script file in html.

<table>: We will maintain a table order for the content retrieving from the web application.

<input>: Actually, it will give a text box editor for html page. Here, we have used it for enter the username for searching.

Code:

```
var url = `https://api.github.com/users/${user}`;  
//creat a XMLHttpRequest  
var xhttp = new XMLHttpRequest();  
xhttp.open( method: "GET", url, async: true);
```

XMLHttpRequest: Here, we have initialized a variable called **url** for GitHub URL. After that we have created an object called xhttp for XMLHttpRequest () method. We have created a connection between web application and GitHub server by using the Get request.

```
//call showUser or noSuchUser methods based on the status of xhttp upon loading  
xhttp.onload = () => {  
    if ( xhttp.status == 200) {  
        showUser(JSON.parse(xhttp.response));  
    }  
    else  
    {  
        noSuchUser(username);  
    }  
}  
xhttp.send();
```

In the above screenshot, it is clearly showing that call showUser or noSuchUser depends upon the xhttp status. If the status is equal to 200 then it will call the showUser method otherwise, will call the noSuchUser method.

```

function showUser(user) {
    // set the contents div elements in the div '#profile' with the user content
    var url = `https://github.com/users/${user.login}`;
    document.getElementById( elementId: 'gitimage').src=user.avatar_url;
    document.getElementById( elementId: 'gitname').innerText=user.login;
    document.getElementById( elementId: 'gitid').innerText=user.id;
    document.getElementById( elementId: 'giturl').href=user.url;
    document.getElementById( elementId: 'giturl').innerText=url;
    document.getElementById( elementId: 'twitter').innerText=user.twitter_username;
    document.getElementById( elementId: 'gitrepo').innerText=user.public_repos;
    document.getElementById( elementId: 'gitgist').innerText=user.public_gists;
    document.getElementById( elementId: 'gitfollowers').innerText=user.followers;
    document.getElementById( elementId: 'gitfollowing').innerText=user.following;
    document.getElementById( elementId: 'gitbio').innerText=user.bio;
    document.getElementById( elementId: 'gitcreated').innerText=user.created_at;
    document.getElementById( elementId: 'gitupdated').innerText=user.updated_at;
}

```

If any information is found it will call showUser method. After calling this method variable url is passing value and object to jquery as `${user.login}` to github server, we must use `getElementById` to get inner text of user information and setting the contents of html tags with the user content.

```

function noSuchUser(username) {
    window.alert("User Details " + username + " Not Found");
    window.location.reload();
}

```

In above screenshot we have used window alert to display popup alert if any information is not found and it will call noSuchUser method.

Code:

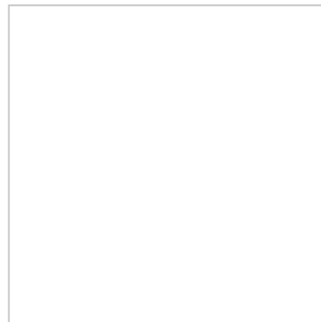
```
$(document).ready(function () {
    $(document).on( e: 'keypress', t: '#username', n: function (e) {
        //check if the enter(i.e return) key is pressed
        if (e.which == 13) {
            //get what the user enters
            username = $(this).val();
            //reset the text typed in the input
            $(this).val( e: "");
            //get the user's information and store the response
            response = getGithubInfo(username);
            //if the response is successful show the user's details
            if (response.status == 200) {
                showUser(JSON.parse(response.responseText));
                //else display suitable message
            } else {
                noSuchUser(username);
            }
        }
    })
});
```

In above screenshot it will validate the given input and checks the if condition with enter button keycode 13, it retrieves the information and stores the details in variable response using getGithubInfo function. After getting the user information successfully again it will validate another if condition with response status 200 and calls showUser function to parse the response information to text and it will display the user details else it will return else with no such username.

Output:

Github User Finder

GudeSrikanth



Name
ID
GitHub URL
Twitter ID
Public Repositories
Public Gists
Followers
Following
Bio
Created at
Updated at

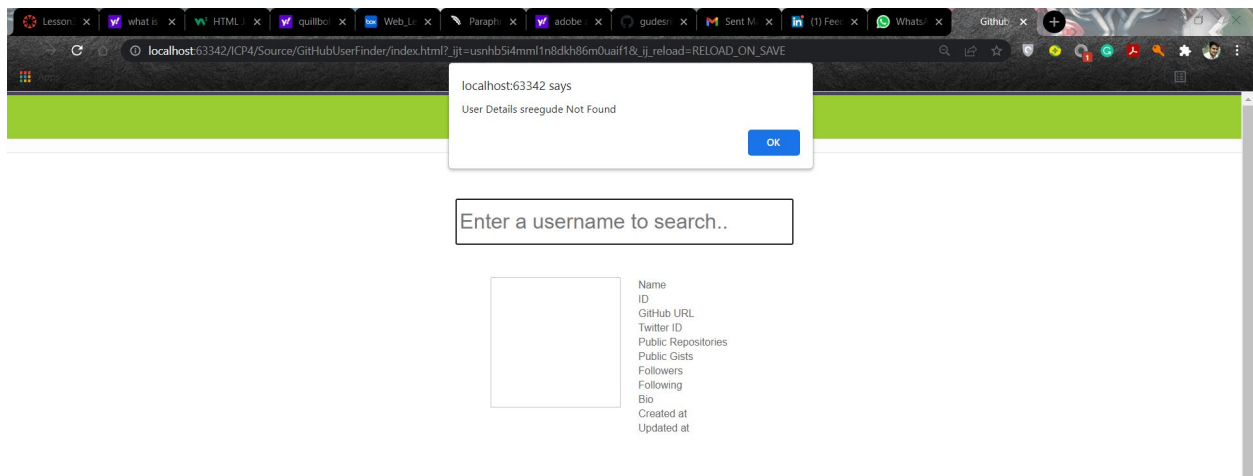
Github User Finder

Enter a username to search..



Name	gudesrikanth
ID	22870886
GitHub URL	https://github.com/users/gudesrikanth
Twitter ID	
Public Repositories	2
Public Gists	0
Followers	0
Following	0
Bio	Graduate student at University of Missouri Kansas city
Created at	2016-10-16T15:27:03Z
Updated at	2022-02-09T21:40:27Z

In above screenshot output is displayed for username given in search box, after clicking enter above details are retrieved and displayed with respected fields.



In above screenshot we have given invalid user as a input it will shows output as user details are not found in window popup.

Conclusion:

In this ICP we learned about how to develop a web page by using the javascript and jquery and at last we have designed and developed web pages successfully. We didn't face any major issues while doing this ICP.