

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <meta http-equiv="X-UA-Compatible" content="IE=edge">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <link rel="stylesheet" href="/css/login.css">

  <title>Sign Up</title>

  <script>

    if (window.location.hostname !== "localhost") {

      if (location.protocol !== "https:") {

        location.replace(

          `https:${location.href.substring(

            location.protocol.length

          )}`

        )

      }

    }

  </script>

  <script src="./localforage.js"></script>

</head>

<body>

  <div class="wrapper">

    <div class="loginContainer">

      <span>Login to Continue</span>

    </div>

  </div>

</body>

</html>
```

```

<div class="traditionalLoginContainer">

  <form class="signupForm" action="/" method="post">

    <input type="text" name="firstName" placeholder="First Name" id="firstName">

    <input type="text" name="lastName" placeholder="Last Name" id="lastName">

    <input type="text" name="username" placeholder="User Name" id="username">

    <input type="email" name="email" placeholder="Email" id="email">

    <input type="password" name="password" placeholder="Password" id="password">

    <input class="loginButton" type="submit" value="Sign Up">

  </form>

</div>

<div class="loginWithFireContainer">

  <button type="button" class="fire" title="Login with Fire" id="fire">Login with Fire
</button>

</div>

<a class="hyperLink" href="/login">Already have an Account? Login </a>

</div>

</div>

<script>

  // Necessary for Fire OAuth to Function

  const fireBroadcastingChannel = new BroadcastChannel('fireOAuthChannel');

  fireBroadcastingChannel.addEventListener('message', async event => {

    let data = event.data

    /**

    * @typedef {Object<string, any>} Data

    * @property {boolean} success - Whether the login was successful

```

* @property {string} token - The data returned from the login i.e. Fire Token

*/

// data.token is the message sent from the fireOAuthChannel after verification

// data.success is a boolean that indicates whether the verification was successful

// data.token is the fire token

// What to do with the Fire Token?

// * Fire Token is an unique token which uniquely identifies the user who authorized your login attempt with Fire

// * You can use this token ONLY ONCE as it will be destroyed after the first use

// 1. Send the fire token to the Fire Server to verify the user

// - You can do that client sided or server sided

// - You need to send a POST Request to the Fire Server with the fire token

// at the URL: <http://localhost:3003/api/tokens/verify>

// - The Fire Server will verify the fire token and return a response

// - If the verification was successful - CODE (200), the Fire Server will return a response with the user's data

// - If the verification was unsuccessful - CODE (400) or CODE (401), the Fire Server will return a response with an error 'message'

// - You can use the data returned from the Fire Server to create a new user in your database

// This example will send the token to Fire Servers and console.log the response

console.log("%c" + `Fire Token: \${data.token}`, `color: #f1c40f; font-weight: bold;`);

```
const response = await fetch('https://fire.adaptable.app/api/tokens/verify', {  
  method: 'POST',  
  headers: {  
    'Content-Type': 'application/json'  
  },  
  body: JSON.stringify({  
    token: data.token  
  })  
})  
  
// get the response  
const responseData = await response.json()  
  
// console.log the response  
console.log(responseData)  
  
await localforage.setItem('userData', {...responseData, isFire: true})  
  
// Adding the user data to the user Database  
let database = await localforage.getItem("userDatabase")  
  
if (database == null) {  
  database = []  
}  
  
database.push(responseData)  
  
await localforage.setItem("userDatabase", database)  
  
// redirect to the home page  
window.location.href = '/'  
  
})
```

```

function popupwindow(url, title, w, h) {

    var left = (screen.width/2)-(w/2);

    var top = (screen.height/2)-(h/2);

    return window.open(url, title, 'toolbar=no, location=no, directories=no, status=no, menubar=no,
scrollbars=no, resizable=no, copyhistory=no, width='+w+', height='+h+', top='+top+', left='+left);

}

document.getElementById("fire").addEventListener("click", function() {

    popupwindow("/fireoauth.html", "Fire OAuth", 450, 600)

})

</script>

```

```

<script>

// this.Website's Scripts / App Logic

document.querySelector(".signupForm").addEventListener("submit" , async (e) => {

    e.preventDefault()

    let firstName = document.getElementById("firstName").value

    let lastName = document.getElementById("lastName").value

    let username = document.getElementById("username").value

    let email = document.getElementById("email").value

    let password = document.getElementById("password").value

    let profilePic = `https://avatars.dicebear.com/api/adventurer-
neutral/${firstName}${lastName}.svg?backgroundColor=variant03`

    let data = { firstName, lastName, username, email, password, profilePic }

```

```
    await localforage.setItem("userData", data)

    let database = await localforage.getItem("userDatabase")

    if (database == null) {

        database = []

    }

    database.push(data)

    await localforage.setItem("userDatabase", database)

    window.location.href = "/"

  })
</script>
</body>
</html>
```