

```

1  <!DOCTYPE html>
2  <html Lang="en">
3
4  <head>
5      <meta charset="UTF-8">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/codemirror/6.65.7/codemirror.min.css">
8      <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/codemirror/6.65.7/theme/panda-syntax.min.css">
9      <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/codemirror/6.65.7/addon/hint/show-hint.min.css">
10     <link rel="stylesheet" href="style.css">
11     <title>Live Code Editor </title>
12 </head>
13
14 <body>
15
16     <!-- Login Form -->
17     <div id="loginForm">
18         <input type="email" id="email" placeholder="Email">
19         <input type="password" id="password" placeholder="Password">
20         <button id="loginButton">Login</button>
21     </div>
22
23     <!-- Code Editors (Initially hidden) -->
24     <div id="codeEditors" style="display: none;">
25         <div class="code-box">
26             <div class="editor" id="html"></div>
27             <div class="editor" id="css"></div>
28             <div class="editor" id="js"></div>
29         </div>
30         <div class="preview">
31             <iframe id="live-preview"></iframe>
32         </div>
33     </div>
34
35     <script src="https://www.gstatic.com/firebasejs/9.0.2/firebase-app.js"></script>
36     <script src="https://www.gstatic.com/firebasejs/9.0.2/firebase-auth.js"></script>
37     <script src="https://www.gstatic.com/firebasejs/9.0.2/firebase-firestore.js"></script>
38     <script src="https://cdnjs.cloudflare.com/ajax/libs/codemirror/6.65.7/codemirror.min.js"></script>
39     <script src="https://cdnjs.cloudflare.com/ajax/libs/codemirror/6.65.7/mode/htmlmixed/htmlmixed.min.js"></script>
40     <script src="https://cdnjs.cloudflare.com/ajax/libs/codemirror/6.65.7/mode/css/css.min.js"></script>
41     <script src="https://cdnjs.cloudflare.com/ajax/libs/codemirror/6.65.7/mode/javascript/javascript.min.js"></script>
42     <script src="https://cdnjs.cloudflare.com/ajax/libs/codemirror/6.65.7/addon/edit/closebrackets.min.js"></script>
43     <script src="https://cdnjs.cloudflare.com/ajax/libs/codemirror/6.65.7/addon/edit/closebrackets.min.js"></script>
44     <script src="https://cdnjs.cloudflare.com/ajax/libs/codemirror/6.65.7/addon/hint/show-hint.min.js"></script>
45     <script src="https://cdnjs.cloudflare.com/ajax/libs/codemirror/6.65.7/mode/xml/xml.min.js"></script>
46     <script type="module">
47         // Import the functions you need from the SDKs you need
48         import { initializeApp } from "https://www.gstatic.com/firebasejs/10.11.1/firebase-app.js";
49         import { getAnalytics } from "https://www.gstatic.com/firebasejs/10.11.1/firebase-analytics.js";
50
51         import { getAuth, createUserWithEmailAndPassword } from "firebase/auth";
52         // TODO: Add SDKs for Firebase products that you want to use
53         // https://firebase.google.com/docs/web/setup#available-libraries
54
55         // Your web app's Firebase configuration
56         // For Firebase JS SDK v7.20.0 and later, measurementId is optional
57         const firebaseConfig = {
58             apiKey: "AIzaSyBv0oI1f3kg1YyR35UQMEGrTEl1t4ZS9Mo",
59             authDomain: "livecodeeditor-808ed.firebaseio.com",
60             projectId: "livecodeeditor-808ed",
61             storageBucket: "livecodeeditor-808ed.appspot.com",
62             messagingSenderId: "818373088803",
63             appId: "1:818373088803:web:d477fabf98bd9924834396",
64             measurementId: "G-1NG57GRQ2M"
65         };
66
67         // Initialize Firebase
68         const app = initializeApp(firebaseConfig);
69         const analytics = getAnalytics(app);
70     </script>
71 </script>
72
73
74
75     // Function to handle Login
76     function login() {
77         const email = document.getElementById("email").value;
78         const password = document.getElementById("password").value;
79
80         firebase.auth().signInWithEmailAndPassword(email, password)
81             .then((userCredential) => {
82                 // Signed in
83                 const user = userCredential.user;
84                 console.log(user);
85                 // Hide Login form and show code editors
86                 document.getElementById("loginForm").style.display = "none";
87                 document.getElementById("codeEditors").style.display = "block";
88                 // Initialize code editors and set up live preview studio
89                 setupLivePreviewStudio();
90             })
91             .catch((error) => {
92                 const errorCode = error.code;
93                 const errorMessage = error.message;
94                 alert(errorMessage);
95             });
96     }
97

```

```

97 // Event Listener for Login button
98 document.getElementById("loginButton").addEventListener("click", login);
99
100
101 // Catching commonly used elements to minimize dom queries
102 const livePreviewFrame = document.getElementById('live-preview');
103 const htmlEditor = document.getElementById('html');
104 const cssEditor = document.getElementById('css');
105 const jsEditor = document.getElementById('js');
106
107 // Function to set up the Live preview iframe and include necessary scripts
108 function initializeLivePreview() {
109     livePreviewFrame.contentWindow.document.body.innerHTML = '';
110     const styleElement = document.createElement('style');
111     styleElement.setAttribute('id', 'live-preview-style');
112     livePreviewFrame.contentWindow.document.head.appendChild(styleElement);
113
114     const pagedJsScript = document.createElement('script');
115     pagedJsScript.src = 'https://unpkg.com/pagedjs/dist/paged.legacy.polyfill.js';
116     livePreviewFrame.contentWindow.document.head.appendChild(pagedJsScript);
117 }
118
119 // Function to update the Live preview iframe with the html code from editor
120 function updateLiveHTMLPreview(codeEditors) {
121     livePreviewFrame.contentWindow.document.body.innerHTML = codeEditors.html.getValue();
122 }
123
124 // Function to update the Live preview iframe with the css code from editor
125 function updateLiveCSSPreview(codeEditors) {
126     const styleElement = livePreviewFrame.contentWindow.document.getElementById('live-preview-style');
127     styleElement.innerHTML = codeEditors.css.getValue();
128 }
129
130 // Function to update the Live preview iframe with the js code from editor
131 function updateLiveJSPreview(codeEditors) {
132     const scriptElement = document.createElement('script');
133     scriptElement.innerHTML = codeEditors.js.getValue();
134     livePreviewFrame.contentWindow.document.body.appendChild(scriptElement);
135 }
136
137 // Function to initialize CodeMirror editors for html, css and javascript
138 function initializeCodeEditors() {
139     function getDefaultOptions(object) {
140         const defaultOptions = {
141             lineNumbers: true,
142             autoCloseTags: true,
143             autoCloseBrackets: true,
144             theme: 'panda-syntax'
145         };
146         if (object) {
147             const keys = Object.keys(object);
148             for (const key of keys) {
149                 defaultOptions[key] = object[key];
150             }
151         }
152         return defaultOptions;
153     }
154
155     const codeEditors = {
156         html: CodeMirror(htmlEditor, getDefaultOptions({
157             mode: 'text/html',
158             value: '',
159         })),
160         css: CodeMirror(cssEditor, getDefaultOptions({
161             mode: 'css',
162             value: '',
163             extraKeys: { 'Ctrl-Space': 'autocomplete' },
164             hintOptions: {
165                 completeSingle: false,
166                 closeOnUnfocus: false
167             }
168         })),
169         js: CodeMirror(jsEditor, getDefaultOptions({
170             mode: 'javascript',
171             value: ''
172         })),
173     };
174     return codeEditors;
175 }
176
177 // Function to set up the Live preview studio with CodeMirror editors and event listeners
178 function setupLivePreviewStudio() {
179     const codeEditors = initializeCodeEditors();
180
181     // Event Listener for changes in HTML editor
182     CodeMirror.on(codeEditors.html, 'change', () => {
183         updateLiveHTMLPreview(codeEditors);
184     });
185
186     // Event Listener for changes in CSS editor
187     CodeMirror.on(codeEditors.css, 'change', () => {
188         updateLiveCSSPreview(codeEditors);
189     });
190
191     // Event Listener for changes in HTML editor
192     CodeMirror.on(codeEditors.js, 'change', () => {
193         updateLiveJSPreview(codeEditors);
194     });
195 }
196
197 // Event Listener to set up the Live preview studio after the dom is fully loaded
198 document.addEventListener('DOMContentLoaded', () => {
199     initializeLivePreview();
200 });
201 </script>
202 </body>
203
204 </html>
205

```

```
1 @import url('https://fonts.googleapis.com/css2?family=Source+Sans+3:wght@300;400;500;600;700&display=swap');
2
3 body{
4     margin: 0;
5 }
6
7 .container{
8     font-family: "Source Sans 3", sans-serif;
9 }
10
11 .header{
12     background: #000;
13     color: #fff;
14     height: 60px;
15     border-bottom: 5px solid #333;
16 }
17
18 .header .title{
19     padding: 0.5rem;
20 }
21
22 .header .title .main-title{
23     font-size: 18px;
24     font-weight: 600;
25 }
26
27 .header .title .sub-title{
28     font-size: 14px;
29 }
30
31 .header .title .sub-title span{
32     color: #999;
33 }
34
35 .editor{
36     width: 33.33%;
37     height: 100%;
38     float: left;
39     box-sizing: border-box;
40     position: relative;
41 }
42
43 .editor #html{
44     border-right: 1px solid #777;
45 }
46
47 .editor #css{
48     border-left: 20px solid #333;
49     border-right: 1px solid #777;
50 }
51
52 .editor #js{
53     border-left: 20px solid #333;
54 }
55
56 .code-box{
57     position: relative;
58     height: calc(50vh - 4.6rem);
59     border-top: 1px solid #000;
60     border-bottom: 10px solid #333;
61     overflow: hidden;
62 }
63
64 .preview{
65     position: relative;
66     height: calc(90vh - 30px);
67     background: #fff;
68     border-top: 10px ridge #333;
69 }
70
71 .preview iframe{
72     width: 100%;
73     height: 100%;
74 }
75
76 ::-webkit-scrollbar{
77     width: 5px;
78     height: 5px;
79 }
80
81 ::-webkit-scrollbar-track{
82     background: transparent;
83 }
84
85 ::-webkit-scrollbar-thumb{
86     background: #555;
87     border-radius: 20px;
88 }
```

```

1 // Catching commonly used elements to minimize dom queries
2 const livePreviewFrame = document.getElementById('live-preview');
3 const htmlEditor = document.getElementById('html');
4 const cssEditor = document.getElementById('css');
5 const jsEditor = document.getElementById('js');
6
7 // Function to set up the Live preview iframe and include necessary scripts
8 function initializeLivePreview() {
9     livePreviewFrame.contentWindow.document.body.innerHTML = '';
10    const styleElement = document.createElement('style');
11    styleElement.setAttribute('id', 'live-preview-style');
12    livePreviewFrame.contentWindow.document.head.appendChild(styleElement);
13
14    const pagedJsScript = document.createElement('script');
15    pagedJsScript.src = 'https://unpkg.com/pagedjs/dist/paged.legacy.polyfill.js';
16    livePreviewFrame.contentWindow.document.head.appendChild(pagedJsScript);
17 }
18
19 // Function to update the Live preview iframe with the html code from editor
20 function updateLiveHTMLPreview(codeEditors) {
21     livePreviewFrame.contentWindow.document.body.innerHTML = codeEditors.html.getValue();
22 }
23
24 // Function to update the Live preview iframe with the css code from editor
25 function updateLiveCSSPreview(codeEditors) {
26     const styleElement = livePreviewFrame.contentWindow.document.getElementById('live-preview-style');
27     styleElement.innerHTML = codeEditors.css.getValue();
28 }
29
30 // Function to update the Live preview iframe with the js code from editor
31 function updateLiveJSPreview(codeEditors) {
32     const scriptElement = document.createElement('script');
33     scriptElement.innerHTML = codeEditors.js.getValue();
34     livePreviewFrame.contentWindow.document.body.appendChild(scriptElement);
35 }
36
37 // Function to initialize CodeMirror editors for html, css and javascript
38 function initializeCodeEditors() {
39     function getDefaultOptions(object) {
40         const defaultOptions = {
41             lineNumbers: true,
42             autoCloseTags: true,
43             autoCloseBrackets: true,
44             theme: 'panda-syntax'
45         };
46         if (object) {
47             const keys = Object.keys(object);
48             for (const key of keys) {
49                 defaultOptions[key] = object[key];
50             }
51         }
52         return defaultOptions;
53     }
54
55     const codeEditors = {
56         html: CodeMirror(htmlEditor, getDefaultOptions({
57             mode: 'text/html',
58             value: ''
59         })),
60         css: CodeMirror(cssEditor, getDefaultOptions({
61             mode: 'css',
62             value: '',
63             extraKeys: { 'Ctrl-Space': 'autocomplete' },
64             hintOptions: {
65                 completeSingle: false,
66                 closeOnUnfocus: false
67             }
68         })),
69         js: CodeMirror(jsEditor, getDefaultOptions({
70             mode: 'javascript',
71             value: ''
72         })),
73     };
74     return codeEditors;
75 }
76
77 // Function to set up the Live preview studio with CodeMirror editors and event listeners
78 function setupLivePreviewStudio() {
79     const codeEditors = initializeCodeEditors();
80
81     // Event Listener for changes in HTML editor
82     CodeMirror.on(codeEditors.html, 'change', () => {
83         updateLiveHTMLPreview(codeEditors);
84     });
85
86     // Event Listener for changes in CSS editor
87     CodeMirror.on(codeEditors.css, 'change', () => {
88         updateLiveCSSPreview(codeEditors);
89     });
90
91     // Event Listener for changes in HTML editor
92     CodeMirror.on(codeEditors.js, 'change', () => {
93         updateLiveJSPreview(codeEditors);
94     });
95 }
96
97 // Event Listener to set up the Live preview studio after the dom is fully loaded
98 document.addEventListener('DOMContentLoaded', () => {
99     initializeLivePreview();
100    setupLivePreviewStudio();
101 });

```