

Media Streaming with IBM Cloud Video Streaming

TEAM MEMBER

312621243007:S.BHARATH

Phase-5 Document submission

Project Title: Cloud Media Streaming with IBM Cloud Video Streaming

Problem Statement

Objective: Develop a seamless and reliable cloud-based media streaming platform using IBM Cloud Video Streaming.

Problem identified:

In today's digital age, the demand for high-quality, uninterrupted media streaming experiences is higher than ever. Whether it's for entertainment, education, or business purposes, users expect smooth and reliable playback across various devices and platforms. However, setting up a robust and scalable cloud media streaming solution can be a complex task, involving considerations for video encoding, content delivery, and user experience optimization.

Introduction:

In response to this growing need, our project, "Cloud Media Streaming with IBM Cloud Video Streaming," aims to create a cutting-edge platform for delivering high-quality video content over the internet. Leveraging the capabilities of IBM Cloud Video Streaming, we will design a solution that ensures seamless playback, regardless of the viewer's location or device.

The project starts by addressing the core issue: providing a reliable and scalable infrastructure for media streaming. This involves tasks such as video transcoding, adaptive bitrate streaming, content delivery network (CDN)

integration, and user interface design. We will utilize the powerful features of IBM Cloud Video Streaming to handle these tasks efficiently.

Our project focuses on a well-structured process that spans from content upload and encoding to user interface customization and integration with third-party services. We will also implement features like analytics and user authentication to enhance the overall streaming experience.

In the following sections, we will delve into the intricacies of our project, describing the methods, tools, and technologies we utilize to build a robust cloud media streaming platform. Our goal is to address the increasing demand for high-quality video content delivery, empowering content creators and businesses to reach their audience with a seamless streaming experience.

Data: The primary data for this project will consist of various video files in different formats and resolutions. Additionally, we will collect user engagement metrics to analyze viewer behavior and optimize the streaming experience. This data will be instrumental in fine-tuning our platform for optimal performance.

LITERATURE SURVEY

1." Cost Minimization of Cloud Services for On-Demand Video Streaming", Mahmoud Darwich [2022]

This research explores the utilization of cloud technology for video stream processing, leveraging the benefits of virtual machines and cost-effective storage servers. To cater to diverse user devices, multiple video formats are typically prepared, posing a challenge in terms of storage costs. This study presents an approach to optimize cloud storage by strategically determining which videos, in what formats, should be stored, thereby minimizing overall cloud service expenses. Promising results were obtained, demonstrating

effectiveness with a growing number of frequently accessed videos and increasing view counts. The proposed method resulted in a noteworthy reduction of up to 22% in cloud service costs.

2.” Cost-Efficient Storage for On-Demand Video

Streaming on Cloud ”, Ishihara [2021]

This paper addresses the challenge of video transcoding, where videos are converted into various formats to accommodate different user devices. This process demands significant storage and resources. With the rise of cloud technology, video streaming companies have shifted to processing videos in the cloud. Traditionally, multiple pre-transcoded video formats are stored and streamed, leading to high storage costs. This paper proposes a solution for cost-effective video storage in the hierarchical cloud storage. The method optimizes video pre-transcoding decisions based on suitable cloud storage to minimize costs. Experimental results demonstrate the effectiveness of the approach, particularly in scenarios with a high percentage of frequently accessed videos, leading to potential cost reduction of up to 40%.

3.” PERFORMANCE ANALYSIS OF VIDEO ON-DEMAND AND LIVE VIDEO STREAMING USING CLOUD BASED SERVICES - 2021”, UJASH PATEL [2022]

The paper explores Cyber-Physical Systems (CPS) and their integration with video streaming for applications like smart grids and health monitoring. It emphasizes the importance of Quality of Experience (QoE), cost, and bandwidth impact on cloud-based video analysis for Video-On-Demand Streaming (VoDS) and Live Video Streaming (LVS). Content Delivery Networks (CDNs) are discussed as crucial for achieving optimal user experience across various cloud providers

4.” Improving Hierarchy Storage for Video Streaming

in Cloud ”, Yasser Ismail [2021]

To address the cost implications of storing various video stream formats, this research focuses on utilizing cloud technology for efficient storage. Traditionally, storing multiple formats incurred high expenses. By adopting cloud services, video stream companies aim to mitigate costs. However, storing all streams in a single cloud type escalates costs, worsened by changing access patterns. To optimize storage costs, the paper introduces a method that leverages hierarchical cloud storage. The algorithm identifies frequently accessed video segments and stores them in the appropriate cloud storage type. Experiments demonstrate a promising 18.75% reduction in cloud storage costs through this approach.

5." Point Cloud Video Streaming: Challenges and Solutions ", Weishan Zhang [2021]

This article addresses the emerging field of volumetric video, essential for VR/AR/MR experiences and well-suited for advanced wireless communication like 5G. It emphasizes the need for efficient volumetric video streaming and focuses on point cloud video as a popular way to represent volumetric media. The article introduces point cloud video technology and its applications, outlines challenges and solutions in point cloud video streaming, and discusses encoding, tiling, viewing angle prediction, decoding, quality assessment, and transmission optimization. A preliminary MPEG DASH-based point cloud video streaming prototype is explained with simulation results, and future research directions are highlighted for high-quality point cloud video streaming.

DESIGN THINKING

Design Thinking Approach for Media Streaming with IBM Cloud Video Streaming

Empathize:

Gain insights into users' needs and expectations, focusing on viewers and content creators.

Actions:

- Conduct user surveys or interviews to understand preferences and pain points.
- Analyze market trends and competitors to identify key success factors.
- Seek input and feedback from experts in content delivery, user experience design, and cloud infrastructure.

Define:

Set clear objectives and success criteria.

Objectives:

- Develop a robust, scalable cloud-based platform using IBM Cloud Video Streaming.
- Ensure high-quality video playback and provide a user-friendly interface.
- Achieve high user satisfaction metrics, including increased user engagement, minimal buffering, and positive feedback.

Ideate:

Brainstorm innovative approaches leveraging IBM Cloud Video Streaming.

Actions:

- Explore IBM Cloud Video Streaming capabilities and integration with HTML, CSS, PHP, and JS for hosting and streaming content.
- Consider various streaming protocols and codecs to deliver high-quality video content through the platform.
- Brainstorm features like content recommendation algorithms, user-generated content integration, and social sharing functionalities within the framework of IBM Cloud Video Streaming.

Prototype:

Visualize the platform using HTML, CSS, PHP, JS with IBM Cloud Video Streaming.

Actions:

- Set up an IBM Cloud environment to test content upload, storage, and streaming capabilities.
- Develop a user interface prototype using HTML, CSS, PHP, and JS, integrating with IBM Cloud Video Streaming.
- Conduct usability testing with a small group of users to validate the initial design concepts and user experience.

Test:

Evaluate technical and user experience performance with IBM Cloud Video Streaming.

Actions:

- Test the IBM Cloud Video Streaming infrastructure for scalability, security, and performance under various loads.
- Gather user feedback on the prototype's usability, navigation, and overall satisfaction, specifically in conjunction with IBM Cloud Video Streaming.
- Use metrics such as video buffering rates, load times, and user engagement to assess the prototype's effectiveness with IBM Cloud Video Streaming.

Implement:

Build and deploy the platform using HTML, CSS, PHP, and JS in conjunction with IBM Cloud Video Streaming.

Actions:

- Integrate content upload, storage, and streaming components with IBM Cloud Video Streaming capabilities.
- Implement user authentication and authorization mechanisms for secure access, leveraging IBM Cloud Video Streaming features.
- Conduct comprehensive testing to ensure seamless content delivery and user interactions with IBM Cloud Video Streaming.

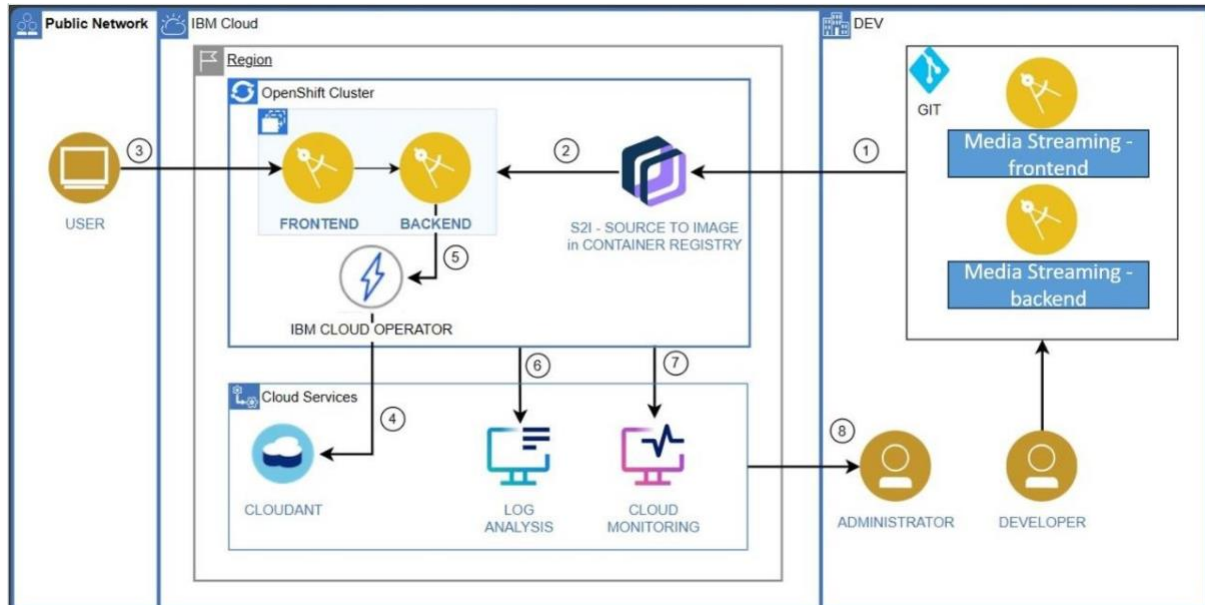
Iterate:

Continuously gather feedback and make improvements, in collaboration with IBM Cloud Video Streaming.

Actions:

- Monitor system performance, user engagement, and content popularity to identify areas for improvement, particularly leveraging IBM Cloud Video Streaming capabilities.
- Address user feedback and implement enhancements, such as refining recommendation algorithms or adding social sharing features, while integrating with IBM Cloud Video Streaming.
- Stay updated with emerging technologies and trends in media streaming to incorporate innovations into the platform, in collaboration with IBM Cloud Video Streaming.

TECHNOLOGY ARCHITECTURE



Design Thinking Approach for Media Streaming with IBM Cloud Video Streaming

1. Content Acquisition and Storage:

- Content Sources: Utilize various sources to acquire video content, including user uploads, licensed content, and streaming services.
- Cloud Storage: Store video files in IBM Cloud Object Storage for efficient retrieval and distribution.

2. Content Preprocessing:

- Quality Assurance: Ensure video content meets specified quality standards, addressing issues like resolution, aspect ratio, and compression artifacts.
- Transcoding: Convert videos into multiple formats and resolutions to accommodate various user devices and network conditions.

3. Platform Development and Integration:

- IBM Cloud Video Streaming: Leverage IBM Cloud Video Streaming for hosting and managing video content.
- API Integration: Integrate IBM Cloud Video Streaming APIs for seamless content upload, retrieval, and streaming.

4. User Interface (UI) Design:

- Create an intuitive UI for users to browse and interact with the media streaming platform.
- Implement features for content search, recommendations, and user-generated playlists.

5. Video Player Customization:

- Customize the video player using HTML, CSS, and JavaScript to provide a branded and user-friendly viewing experience.
- Implement features like adaptive bitrate streaming for optimal playback quality.

6. User Authentication and Authorization:

- Implement secure user authentication mechanisms to control access to premium content and user-specific features.
- Utilize IBM Cloud Identity and Access Management (IAM) for authentication.

7. Monetization (Optional):

- Implement monetization strategies such as subscription models, pay-per-view, or ad-based revenue streams using IBM Monetize.

8. Analytics and Insights:

- Integrate analytics tools to gather user engagement data, including views, watch time, and popular content.
- Utilize IBM Cloud Analytics for in-depth insights into user behavior and content performance.

9. Content Recommendation Engine (Optional):

- Develop a recommendation engine using machine learning algorithms to suggest personalized content to users.

10. Social Integration (Optional):

- Implement social sharing features to allow users to share content on social media platforms.

11. Content Moderation (Optional):

- Integrate content moderation tools to ensure compliance with community guidelines and prevent inappropriate or offensive content from being published.

12. Continuous Improvement and Innovation:

- Stay updated with emerging technologies in media streaming and leverage IBM Cloud services for ongoing enhancements.
- Gather user feedback and conduct usability testing to refine features and optimize the user experience.

MODULES DESCRIPTION

1. Content Acquisition and Storage Module:

- Objective: Acquire and store video content efficiently for the media streaming platform.
- Key Tasks:
 - Gather video content from various sources, including user uploads and licensed content.
 - Store video files in IBM Cloud Object Storage for easy retrieval.

2. Content Preprocessing Module:

- Objective: Prepare video content for optimal streaming quality and compatibility.
- Key Tasks:
 - Ensure video quality meets specified standards, addressing resolution, aspect ratio, and compression.
 - Transcode videos into multiple formats and resolutions for diverse user devices and network conditions.

3. Platform Development and Integration Module:

- Objective: Utilize IBM Cloud Video Streaming for hosting and managing video content.
- Key Tasks:
 - Integrate IBM Cloud Video Streaming APIs for seamless content upload, retrieval, and streaming.

4. User Interface (UI) Design Module:

- Objective: Design an intuitive user interface for seamless interaction with the media streaming platform.

- Key Tasks:

- Create a user-friendly UI with features for content search, recommendations, and user-generated playlists.

5. Video Player Customization Module:

- Objective: Customize the video player to provide an enhanced viewing experience.

- Key Tasks:

- Utilize HTML, CSS, and JavaScript to customize the video player for branding and optimal playback.

6. User Authentication and Authorization Module:

- Objective: Implement secure access control mechanisms for user authentication and authorization.

- Key Tasks:

- Integrate IBM Cloud Identity and Access Management (IAM) for authentication and authorization.

7. Monetization Strategies Module (Optional):

- Objective: Implement revenue-generating models for the media streaming platform.

- Key Tasks:

- Introduce monetization strategies like subscription models, pay-per-view, or ad-based revenue streams.

8. Analytics and Insights Module:

- Objective: Gather and analyze user engagement data to gain valuable insights.

- Key Tasks:

- Integrate analytics tools to track views, watch time, and popular content.
- Leverage IBM Cloud Analytics for comprehensive user behavior analysis.

9. Continuous Improvement and Innovation Module:

- Objective: Stay up-to-date with emerging technologies and continuously enhance the platform's features and performance.
- Key Tasks:
 - Gather user feedback and conduct usability testing to refine features and optimize the user experience.
 - Explore and incorporate new technologies and trends in media streaming.

ALGORITHM AND TECHNOLOGY USED

1. Content Acquisition and Storage:

- Technology: IBM Cloud Object Storage
- Description: Utilize IBM Cloud Object Storage for efficient storage and retrieval of video content. This technology allows seamless handling of large video files.

2. Content Preprocessing:

- Technology: HTML, CSS, JavaScript (for frontend processing)
- Description: Use HTML, CSS, and JavaScript to ensure video content meets specified quality standards, addressing resolution, aspect ratio, and compression. This step ensures optimal viewing experience.

3. Platform Development and Integration:

- Technology: IBM Cloud Video Streaming APIs, HTML, CSS, PHP, JS
- Description: Leverage IBM Cloud Video Streaming APIs for hosting and managing video content. Additionally, utilize HTML, CSS, PHP, and JS for seamless integration and user interaction within the platform.

4. User Interface (UI) Design:

- Technology: HTML, CSS, JavaScript
- Description: Create an intuitive user interface using HTML, CSS, and JavaScript. This UI allows users to browse, search, and interact with the media streaming platform effortlessly.

5. Video Player Customization:

- Technology: HTML, CSS, JavaScript
- Description: Customize the video player using HTML, CSS, and JavaScript to provide a branded and user-friendly viewing experience. This customization ensures optimal playback quality.

6. User Authentication and Authorization:

- Technology: HTML, CSS, PHP, JS
- Description: Implement secure user authentication and authorization mechanisms using HTML, CSS, PHP, and JS. This step ensures controlled access to premium content and user-specific features.

7. Monetization Strategies (Optional):

- Technology: HTML, CSS, PHP, JS
- Description: If required, implement revenue-generating models using HTML, CSS, PHP, and JS. This may include subscription models, pay-per-view, or ad-based revenue streams.

8. Analytics and Insights:

- Technology: IBM Cloud Analytics, HTML, CSS, PHP, JS
- Description: Integrate analytics tools, like IBM Cloud Analytics, to gather user engagement data. Use HTML, CSS, PHP, and JS for comprehensive user behavior analysis, tracking views, watch time, and popular content.

9. Continuous Improvement and Innovation:

- Technology: HTML, CSS, PHP, JS
- Description: Stay updated with emerging technologies and trends in media streaming. Continuously enhance the platform's features and performance using HTML, CSS, PHP, and JS. Gather user feedback and conduct usability testing for refinement.

Utilizing a combination of HTML, CSS, PHP, JS, and IBM Cloud Video Streaming, this project ensures an interactive and user-friendly media streaming platform with features for content customization, monetization (optional), and comprehensive analytics for user engagement insights. Continuous improvement and innovation are emphasized to adapt to evolving trends in media streaming.

PROJECT DEVELOPMENT STEPS AND SCREENSHOT

Front-End Development

```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5     <meta charset="UTF-8">
6     <meta http-equiv="X-UA-Compatible" content="IE=edge">
7     <meta name="viewport" content="width=device-width, initial-scale=1.0">
8     <link rel="stylesheet" href="homepage.css">
9     <title>Cloudflix - Watch your uploaded videos anywhere & anytime</title>
10 </head>
11 <body>
12
13     <div class="container">
14         <nav class="navbar">
15             <div class="left">
16                 
17             </div>
18             <div class="right">
19                 <select name="language" class="language">
20                     <option value="English">English</option>
21                     <option value="Hindi">Tamil</option>
22                 </select>
23                 <button><a href="login.html">Login</a></button>
24             </div>
25         </nav>
26
27         <div class="title">
28             <div class="content">
29                 <h1>Upload your videos and watch it anytime</h1>
30                 <h2>Watch anywhere. Cancel anytime.</h2>
31                 <form action="#">
32                     <h3>Ready to watch? Enter your email to create your account.</h3>
33                     <div class="email">
34                         <input type="email" name="email" placeholder="Email address">
35                         <button><a href="signup.html">Get Started </a></button>
36                     </div>
37                 </form>
38             </div>
39         </div>
40     </div>
41 </body>
42 </html>
```



```

1  *
2
3  padding: 0;
4  margin: 0;
5  background-color: #fff;
6
7  *
8  width: 100%;
9  height: 100%;
10 color: #333;
11 background-color: #fff;
12 font-family: sans-serif;
13
14 *
15 container: none;
16 content: "";
17 background: url(images/logo.png) no-repeat center center;
18 position: absolute;
19 height: 100%;
20 width: 100%;
21 opacity: 0.5;
22 z-index: 1;
23
24 *
25 container: none;
26 height: 100%;
27 width: 100%;
28
29 *
30 container: none;
31 height: 100%;
32 width: 100%;
33 position: absolute;
34
35 *
36 container: none;
37 height: 100%;
38 width: 100%;
39 position: absolute;
40
41 *
42 container: none;
43 height: 100%;
44 width: 100%;
45 position: absolute;
46
47 *
48 container: none;
49 height: 100%;
50 width: 100%;
51 position: absolute;
52
53 *
54 container: none;
55 height: 100%;
56 width: 100%;
57 position: absolute;
58
59 *
60 container: none;
61 height: 100%;
62 width: 100%;
63 position: absolute;
64
65 *
66 container: none;
67 height: 100%;
68 width: 100%;
69 position: absolute;
70
71 *
72 container: none;
73 height: 100%;
74 width: 100%;
75 position: absolute;
76
77 *
78 container: none;
79 height: 100%;
80 width: 100%;
81 position: absolute;
82
83 *
84 container: none;
85 height: 100%;
86 width: 100%;
87 position: absolute;
88
89 *
90 container: none;
91 height: 100%;
92 width: 100%;
93 position: absolute;
94
95 *
96 container: none;
97 height: 100%;
98 width: 100%;
99 position: absolute;
100
101 *
102 container: none;
103 height: 100%;
104 width: 100%;
105 position: absolute;
106
107 *
108 container: none;
109 height: 100%;
110 width: 100%;
111 position: absolute;
112
113 *
114 container: none;
115 height: 100%;
116 width: 100%;
117 position: absolute;
118
119 *
120 container: none;
121 height: 100%;
122 width: 100%;
123 position: absolute;
124
125 *
126 container: none;
127 height: 100%;
128 width: 100%;
129 position: absolute;
130
131 *
132 container: none;
133 height: 100%;
134 width: 100%;
135 position: absolute;
136
137 *
138 container: none;
139 height: 100%;
140 width: 100%;
141 position: absolute;
142
143 *
144 container: none;
145 height: 100%;
146 width: 100%;
147 position: absolute;
148
149 *
150 container: none;
151 height: 100%;
152 width: 100%;
153 position: absolute;
154
155 *
156 container: none;
157 height: 100%;
158 width: 100%;
159 position: absolute;
160
161 *
162 container: none;
163 height: 100%;
164 width: 100%;
165 position: absolute;
166
167 *
168 container: none;
169 height: 100%;
170 width: 100%;
171 position: absolute;
172
173 *
174 container: none;
175 height: 100%;
176 width: 100%;
177 position: absolute;
178
179 *
180 container: none;
181 height: 100%;
182 width: 100%;
183 position: absolute;
184
185 *
186 container: none;
187 height: 100%;
188 width: 100%;
189 position: absolute;
190
191 *
192 container: none;
193 height: 100%;
194 width: 100%;
195 position: absolute;
196
197 *
198 container: none;
199 height: 100%;
200 width: 100%;
201 position: absolute;
202
203 *
204 container: none;
205 height: 100%;
206 width: 100%;
207 position: absolute;
208
209 *
210 container: none;
211 height: 100%;
212 width: 100%;
213 position: absolute;
214
215 *
216 container: none;
217 height: 100%;
218 width: 100%;
219 position: absolute;
220
221 *
222 container: none;
223 height: 100%;
224 width: 100%;
225 position: absolute;
226
227 *
228 container: none;
229 height: 100%;
230 width: 100%;
231 position: absolute;
232
233 *
234 container: none;
235 height: 100%;
236 width: 100%;
237 position: absolute;
238
239 *
240 container: none;
241 height: 100%;
242 width: 100%;
243 position: absolute;
244
245 *
246 container: none;
247 height: 100%;
248 width: 100%;
249 position: absolute;
250
251 *
252 container: none;
253 height: 100%;
254 width: 100%;
255 position: absolute;
256
257 *
258 container: none;
259 height: 100%;
260 width: 100%;
261 position: absolute;
262
263 *
264 container: none;
265 height: 100%;
266 width: 100%;
267 position: absolute;
268
269 *
270 container: none;
271 height: 100%;
272 width: 100%;
273 position: absolute;
274
275 *
276 container: none;
277 height: 100%;
278 width: 100%;
279 position: absolute;
280
281 *
282 container: none;
283 height: 100%;
284 width: 100%;
285 position: absolute;
286
287 *
288 container: none;
289 height: 100%;
290 width: 100%;
291 position: absolute;
292
293 *
294 container: none;
295 height: 100%;
296 width: 100%;
297 position: absolute;
298
299 *
300 container: none;
301 height: 100%;
302 width: 100%;
303 position: absolute;
304
305 *
306 container: none;
307 height: 100%;
308 width: 100%;
309 position: absolute;
310
311 *
312 container: none;
313 height: 100%;
314 width: 100%;
315 position: absolute;
316
317 *
318 container: none;
319 height: 100%;
320 width: 100%;
321 position: absolute;
322
323 *
324 container: none;
325 height: 100%;
326 width: 100%;
327 position: absolute;
328
329 *
330 container: none;
331 height: 100%;
332 width: 100%;
333 position: absolute;
334
335 *
336 container: none;
337 height: 100%;
338 width: 100%;
339 position: absolute;
340
341 *
342 container: none;
343 height: 100%;
344 width: 100%;
345 position: absolute;
346
347 *
348 container: none;
349 height: 100%;
350 width: 100%;
351 position: absolute;
352
353 *
354 container: none;
355 height: 100%;
356 width: 100%;
357 position: absolute;
358
359 *
360 container: none;
361 height: 100%;
362 width: 100%;
363 position: absolute;
364
365 *
366 container: none;
367 height: 100%;
368 width: 100%;
369 position: absolute;
370
371 *
372 container: none;
373 height: 100%;
374 width: 100%;
375 position: absolute;
376
377 *
378 container: none;
379 height: 100%;
380 width: 100%;
381 position: absolute;
382
383 *
384 container: none;
385 height: 100%;
386 width: 100%;
387 position: absolute;
388
389 *
390 container: none;
391 height: 100%;
392 width: 100%;
393 position: absolute;
394
395 *
396 container: none;
397 height: 100%;
398 width: 100%;
399 position: absolute;
400
401 *
402 container: none;
403 height: 100%;
404 width: 100%;
405 position: absolute;
406
407 *
408 container: none;
409 height: 100%;
410 width: 100%;
411 position: absolute;
412
413 *
414 container: none;
415 height: 100%;
416 width: 100%;
417 position: absolute;
418
419 *
420 container: none;
421 height: 100%;
422 width: 100%;
423 position: absolute;
424
425 *
426 container: none;
427 height: 100%;
428 width: 100%;
429 position: absolute;
430
431 *
432 container: none;
433 height: 100%;
434 width: 100%;
435 position: absolute;
436
437 *
438 container: none;
439 height: 100%;
440 width: 100%;
441 position: absolute;
442
443 *
444 container: none;
445 height: 100%;
446 width: 100%;
447 position: absolute;
448
449 *
450 container: none;
451 height: 100%;
452 width: 100%;
453 position: absolute;
454
455 *
456 container: none;
457 height: 100%;
458 width: 100%;
459 position: absolute;
460
461 *
462 container: none;
463 height: 100%;
464 width: 100%;
465 position: absolute;
466
467 *
468 container: none;
469 height: 100%;
470 width: 100%;
471 position: absolute;
472
473 *
474 container: none;
475 height: 100%;
476 width: 100%;
477 position: absolute;
478
479 *
480 container: none;
481 height: 100%;
482 width: 100%;
483 position: absolute;
484
485 *
486 container: none;
487 height: 100%;
488 width: 100%;
489 position: absolute;
490
491 *
492 container: none;
493 height: 100%;
494 width: 100%;
495 position: absolute;
496
497 *
498 container: none;
499 height: 100%;
500 width: 100%;
501 position: absolute;
502
503 *
504 container: none;
505 height: 100%;
506 width: 100%;
507 position: absolute;
508
509 *
510 container: none;
511 height: 100%;
512 width: 100%;
513 position: absolute;
514
515 *
516 container: none;
517 height: 100%;
518 width: 100%;
519 position: absolute;
520
521 *
522 container: none;
523 height: 100%;
524 width: 100%;
525 position: absolute;
526
527 *
528 container: none;
529 height: 100%;
530 width: 100%;
531 position: absolute;
532
533 *
534 container: none;
535 height: 100%;
536 width: 100%;
537 position: absolute;
538
539 *
540 container: none;
541 height: 100%;
542 width: 100%;
543 position: absolute;
544
545 *
546 container: none;
547 height: 100%;
548 width: 100%;
549 position: absolute;
550
551 *
552 container: none;
553 height: 100%;
554 width: 100%;
555 position: absolute;
556
557 *
558 container: none;
559 height: 100%;
560 width: 100%;
561 position: absolute;
562
563 *
564 container: none;
565 height: 100%;
566 width: 100%;
567 position: absolute;
568
569 *
570 container: none;
571 height: 100%;
572 width: 100%;
573 position: absolute;
574
575 *
576 container: none;
577 height: 100%;
578 width: 100%;
579 position: absolute;
580
581 *
582 container: none;
583 height: 100%;
584 width: 100%;
585 position: absolute;
586
587 *
588 container: none;
589 height: 100%;
590 width: 100%;
591 position: absolute;
592
593 *
594 container: none;
595 height: 100%;
596 width: 100%;
597 position: absolute;
598
599 *
600 container: none;
601 height: 100%;
602 width: 100%;
603 position: absolute;
604
605 *
606 container: none;
607 height: 100%;
608 width: 100%;
609 position: absolute;
610
611 *
612 container: none;
613 height: 100%;
614 width: 100%;
615 position: absolute;
616
617 *
618 container: none;
619 height: 100%;
620 width: 100%;
621 position: absolute;
622
623 *
624 container: none;
625 height: 100%;
626 width: 100%;
627 position: absolute;
628
629 *
630 container: none;
631 height: 100%;
632 width: 100%;
633 position: absolute;
634
635 *
636 container: none;
637 height: 100%;
638 width: 100%;
639 position: absolute;
640
641 *
642 container: none;
643 height: 100%;
644 width: 100%;
645 position: absolute;
646
647 *
648 container: none;
649 height: 100%;
650 width: 100%;
651 position: absolute;
652
653 *
654 container: none;
655 height: 100%;
656 width: 100%;
657 position: absolute;
658
659 *
660 container: none;
661 height: 100%;
662 width: 100%;
663 position: absolute;
664
665 *
666 container: none;
667 height: 100%;
668 width: 100%;
669 position: absolute;
670
671 *
672 container: none;
673 height: 100%;
674 width: 100%;
675 position: absolute;
676
677 *
678 container: none;
679 height: 100%;
680 width: 100%;
681 position: absolute;
682
683 *
684 container: none;
685 height: 100%;
686 width: 100%;
687 position: absolute;
688
689 *
690 container: none;
691 height: 100%;
692 width: 100%;
693 position: absolute;
694
695 *
696 container: none;
697 height: 100%;
698 width: 100%;
699 position: absolute;
700
701 *
702 container: none;
703 height: 100%;
704 width: 100%;
705 position: absolute;
706
707 *
708 container: none;
709 height: 100%;
710 width: 100%;
711 position: absolute;
712
713 *
714 container: none;
715 height: 100%;
716 width: 100%;
717 position: absolute;
718
719 *
720 container: none;
721 height: 100%;
722 width: 100%;
723 position: absolute;
724
725 *
726 container: none;
727 height: 100%;
728 width: 100%;
729 position: absolute;
730
731 *
732 container: none;
733 height: 100%;
734 width: 100%;
735 position: absolute;
736
737 *
738 container: none;
739 height: 100%;
740 width: 100%;
741 position: absolute;
742
743 *
744 container: none;
745 height: 100%;
746 width: 100%;
747 position: absolute;
748
749 *
750 container: none;
751 height: 100%;
752 width: 100%;
753 position: absolute;
754
755 *
756 container: none;
757 height: 100%;
758 width: 100%;
759 position: absolute;
760
761 *
762 container: none;
763 height: 100%;
764 width: 100%;
765 position: absolute;
766
767 *
768 container: none;
769 height: 100%;
770 width: 100%;
771 position: absolute;
772
773 *
774 container: none;
775 height: 100%;
776 width: 100%;
777 position: absolute;
778
779 *
780 container: none;
781 height: 100%;
782 width: 100%;
783 position: absolute;
784
785 *
786 container: none;
787 height: 100%;
788 width: 100%;
789 position: absolute;
790
791 *
792 container: none;
793 height: 100%;
794 width: 100%;
795 position: absolute;
796
797 *
798 container: none;
799 height: 100%;
800 width: 100%;
801 position: absolute;
802
803 *
804 container: none;
805 height: 100%;
806 width: 100%;
807 position: absolute;
808
809 *
810 container: none;
811 height: 100%;
812 width: 100%;
813 position: absolute;
814
815 *
816 container: none;
817 height: 100%;
818 width: 100%;
819 position: absolute;
820
821 *
822 container: none;
823 height: 100%;
824 width: 100%;
825 position: absolute;
826
827 *
828 container: none;
829 height: 100%;
830 width: 100%;
831 position: absolute;
832
833 *
834 container: none;
835 height: 100%;
836 width: 100%;
837 position: absolute;
838
839 *
840 container: none;
841 height: 100%;
842 width: 100%;
843 position: absolute;
844
845 *
846 container: none;
847 height: 100%;
848 width: 100%;
849 position: absolute;
850
851 *
852 container: none;
853 height: 100%;
854 width: 100%;
855 position: absolute;
856
857 *
858 container: none;
859 height: 100%;
860 width: 100%;
861 position: absolute;
862
863 *
864 container: none;
865 height: 100%;
866 width: 100%;
867 position: absolute;
868
869 *
870 container: none;
871 height: 100%;
872 width: 100%;
873 position: absolute;
874
875 *
876 container: none;
877 height: 100%;
878 width: 100%;
879 position: absolute;
880
881 *
882 container: none;
883 height: 100%;
884 width: 100%;
885 position: absolute;
886
887 *
888 container: none;
889 height: 100%;
890 width: 100%;
891 position: absolute;
892
893 *
894 container: none;
895 height: 100%;
896 width: 100%;
897 position: absolute;
898
899 *
900 container: none;
901 height: 100%;
902 width: 100%;
903 position: absolute;
904
905 *
906 container: none;
907 height: 100%;
908 width: 100%;
909 position: absolute;
910
911 *
912 container: none;
913 height: 100%;
914 width: 100%;
915 position: absolute;
916
917 *
918 container: none;
919 height: 100%;
920 width: 100%;
921 position: absolute;
922
923 *
924 container: none;
925 height: 100%;
926 width: 100%;
927 position: absolute;
928
929 *
930 container: none;
931 height: 100%;
932 width: 100%;
933 position: absolute;
934
935 *
936 container: none;
937 height: 100%;
938 width: 100%;
939 position: absolute;
940
941 *
942 container: none;
943 height: 100%;
944 width: 100%;
945 position: absolute;
946
947 *
948 container: none;
949 height: 100%;
950 width: 100%;
951 position: absolute;
952
953 *
954 container: none;
955 height: 100%;
956 width: 100%;
957 position: absolute;
958
959 *
960 container: none;
961 height: 100%;
962 width: 100%;
963 position: absolute;
964
965 *
966 container: none;
967 height: 100%;
968 width: 100%;
969 position: absolute;
970
971 *
972 container: none;
973 height: 100%;
974 width: 100%;
975 position: absolute;
976
977 *
978 container: none;
979 height: 100%;
980 width: 100%;
981 position: absolute;
982
983 *
984 container: none;
985 height: 100%;
986 width: 100%;
987 position: absolute;
988
989 *
990 container: none;
991 height: 100%;
992 width: 100%;
993 position: absolute;
994
995 *
996 container: none;
997 height: 100%;
998 width: 100%;
999 position: absolute;
1000

```



```
1  <!DOCTYPE html>
2  <!-- Coding By CodingNepal - codingnepalweb.com -->
3  <html lang="en" dir="ltr">
4    <head>
5      <meta charset="UTF-8">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title> Cloudflif - Registration or Sign Up </title>
8      <link rel="stylesheet" href="signup.css">
9    </head>
10   <body>
11     <div class="wrapper">
12       <h2>Sign Up</h2>
13       <form action="main.html">
14         <div class="input-box">
15           <input type="text" placeholder="Enter your name" required>
16         </div>
17         <div class="input-box">
18           <input type="text" placeholder="Enter your email" required>
19         </div>
20         <div class="input-box">
21           <input type="password" placeholder="Create password" required>
22         </div>
23         <div class="input-box">
24           <input type="password" placeholder="Confirm password" required>
25         </div>
26         <div class="policy">
27           <input type="checkbox">
28           <h3>I accept all terms & condition</h3>
29         </div>
30         <div class="input-box button">
31           <input type="Submit" value="Sign Up">
32         </div>
33         <div class="text">
34           <h3>Already have an account? <a href="login.html">Login now</a></h3>
35         </div>
36       </form>
37     </div>
38   </body>
39 </html>
```

```

1 direct url( https://fonts.googleapis.com/css?family=Poppins:400,500,600,700&display=swap );
2 *
3 margin: 0;
4 padding: 0;
5 box-sizing: border-box;
6 font-family: 'Poppins', sans-serif;
7 }
8 body{
9 min-height: 100vh;
10 display: flex;
11 align-items: center;
12 justify-content: center;
13 background: #4477f4;
14 background: url(https://wallpapers.com/wp/wp1805b7s.jpg) no-repeat center center/cover;
15 }
16 body:before{
17 content: '';
18 position: fixed;
19 top: 0;
20 left: 0;
21 width: 100%;
22 height: 100%;
23 background: rgba(0, 0, 0, 0.5); // adjust the opacity as needed //
24 z-index: 1; // Position it behind other content //
25 }
26 .wrapper{
27 position: relative;
28 max-width: 480px;
29 width: 100%;
30 background: rgba(0, 0, 0, 0.973);
31 padding: 5px;
32 border-radius: 6px;
33 box-shadow: 0 5px 10px rgba(0,0,0,0.2);
34 border: 1px solid #fff;
35 }
36 .wrapper h2{
37 position: relative;
38 font-size: 32px;
39 font-weight: 600;
40 color: white;
41 text-align: center;
42 padding-bottom: 10px;
43 }
44 .wrapper h2:before{
45 content: '';
46 position: absolute;
47 left: 0;
48 bottom: 0;
49 height: 100%;
50 width: 350px;
51 border-radius: 12px;
52 background: #000000;
53 }
54 .wrapper form{
55 margin-top: 30px;
56 }
57 .wrapper form .input-box{
58 height: 50px;
59 margin: 10px 0;
60 }
61 form .input-box input{
62 height: 100%;
63 width: 100%;
64 outline: none;
65 padding: 0 15px;
66 font-size: 17px;
67 font-weight: 400;
68 color: #fff;
69 border: 1.5px solid #C7C7C7;
70 border-bottom: 2px;
71 border-bottom-width: 2.5px;
72 border-bottom-color: #fff;
73 transition: all 0.3s ease;
74 }
75 .input-box input:focus,
76 .input-box input:valid{
77 border-color: #4477f4;
78 }
79 form .policy{
80 display: flex;
81 align-items: center;
82 }
83 form label{
84 color: white;
85 font-size: 14px;
86 font-weight: 500;
87 margin-left: 10px;
88 }
89 .input-box button input{
90 color: #fff;
91 letter-spacing: 1px;
92 border: none;
93 background: #4477f4;
94 cursor: pointer;
95 }
96 .input-box button input:hover{
97 background: #4477f4;
98 }
99 form .text h3{
100 color: white;
101 width: 100%;
102 text-align: center;
103 }
104 form .text h4{
105 color: #4477f4;
106 text-decoration: none;
107 }
108 form .text h3 a{
109 text-decoration: underline;
110 }

```

```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Cloudflifx - Login page </title>
7   <link rel="stylesheet" href="login.css">
8   <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/css/all.min.css"/>
9 </head>
10 <body>
11   <div class="wrapper">
12     <header>Login</header>
13     <form action="">
14       <div class="field email">
15         <div class="input-area">
16           <input type="text" placeholder="Email Address">
17           <i class="icon fas fa-envelope"></i>
18           <i class="error error-icon fas fa-exclamation-circle"></i>
19         </div>
20         <div class="error error-txt">Email can't be blank</div>
21       </div>
22       <div class="field password">
23         <div class="input-area">
24           <input type="password" placeholder="Password">
25           <i class="icon fas fa-lock"></i>
26           <i class="error error-icon fas fa-exclamation-circle"></i>
27         </div>
28         <div class="error error-txt">Password can't be blank</div>
29       </div>
30       <div class="pass-txt"><a href="#">Forgot password?</a></div>
31       <input type="submit" value="Login">
32     </form>
33     <div class="sign-txt"><p>Need an account?</p> <a href="signup.html">Signup now</a></div>
34   </div>
35   <script src="login.js"></script>
36 </body>
37 </html>

```

[illegible]

```

1  const form = document.querySelector("form");
2  eField = form.querySelector(".email"),
3  eInput = eField.querySelector("input"),
4  pField = form.querySelector(".password"),
5  pInput = pField.querySelector("input");
6
7  form.onsubmit = (e) => {
8    e.preventDefault(); //preventing from form submitting
9    //if email and password is blank then add shake class in it else call specified function
10   (eInput.value == "") ? eField.classList.add("shake", "error") : checkEmail();
11   (pInput.value == "") ? pField.classList.add("shake", "error") : checkPass();
12
13   setTimeout(() => { //remove shake class after 500ms
14     eField.classList.remove("shake");
15     pField.classList.remove("shake");
16   }, 500);
17
18   eInput.onkeyup = () => {checkEmail();} //calling checkEmail function on email input keyup
19   pInput.onkeyup = () => {checkPass();} //calling checkPassword function on pass input keyup
20
21   function checkEmail(){ //checkEmail function
22     let pattern = /^[^ ]+@[^ ]+\.[a-z]{2,3}$/; //pattern for validate email
23     if(!eInput.value.match(pattern)){ //if pattern not matched then add error and remove valid class
24       eField.classList.add("error");
25       eField.classList.remove("valid");
26       let errorTxt = eField.querySelector(".error-txt");
27       //if email value is not empty then show please enter valid email else show Email can't be blank
28       (eInput.value != "") ? errorTxt.innerText = "Enter a valid email address" : errorTxt.innerText = "Email can't be blank";
29     }else{ //if pattern matched then remove error and add valid class
30       eField.classList.remove("error");
31       eField.classList.add("valid");
32     }
33   }
34
35   function checkPass(){ //checkPass function
36     if(pInput.value == ""){ //if pass is empty then add error and remove valid class
37       pField.classList.add("error");
38       pField.classList.remove("valid");
39     }else{ //if pass is empty then remove error and add valid class
40       pField.classList.remove("error");
41       pField.classList.add("valid");
42     }
43   }
44
45   //if eField and pField doesn't contains error class that mean user filled details properly
46   if(!eField.classList.contains("error") && !pField.classList.contains("error")){
47     window.location.href = form.getAttribute("action"); //redirecting user to the specified url which is inside action attribute of form tag
48   }
49 }

```

Back-end Development

```

1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>video upload php and mysql</title>
7      <style>
8          body {
9              display: flex;
10             justify-content: center;
11             align-items: center;
12             flex-direction: column;
13             min-height: 100vh;
14         }
15         input {
16             font-size: 2rem;
17         }
18         a {
19             text-decoration: none;
20             color: #006CFF;
21             font-size: 1.5rem;
22         }
23     </style>
24 </head>
25 <body>
26     <a href="view.php">Videos</a>
27     <?php if (isset($_GET['error'])) { ?>
28         <p><?=$_GET['error']?></p>
29     <?php } ?>
30     <form action="upload.php"
31         method="post"
32         enctype="multipart/form-data">
33
34         <input type="file"
35             name="my_video">
36
37         <input type="submit"
38             name="submit"
39             value="Upload">
40     </form>
41 </body>
42 </html>

```

```

1  <?php
2
3  $sname = "localhost";
4  $uname = "root";
5  $password = "";
6
7  $db_name = "test_db";
8
9  $conn = mysqli_connect($sname, $uname, $password, $db_name);
10
11  if (!$conn) {
12      echo "Connection failed!";
13      exit();
14  }

```

```

1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>View</title>
7      <style>
8          body {
9              display: flex;
10             justify-content: center;
11             align-items: center;
12             flex-wrap: wrap;
13             min-height: 100vh;
14         }
15         video {
16             width: 640px;
17             height: 360px;
18         }
19         a {
20             text-decoration: none;
21             color: #006CFF;
22             font-size: 1.5rem;
23         }
24     </style>
25 </head>
26 <body>
27     <a href="index.php" >UPLOAD</a>
28
29     <div class="alb">
30         <?php
31             include "db_conn.php";
32             $sql = "SELECT * FROM videos ORDER BY id DESC";
33             $res = mysqli_query($conn, $sql);
34
35             if (mysqli_num_rows($res) > 0) {
36                 while ($video = mysqli_fetch_assoc($res)) {
37
38                     <video src="uploads/<?=$video['video_url']?>"
39                        controls>
40
41                     </video>
42
43                 <?php
44                     }
45                 }else {
46                     echo "<h1>Empty</h1>";
47                 }
48             }
49         </div>
50 </body>
51 </html>

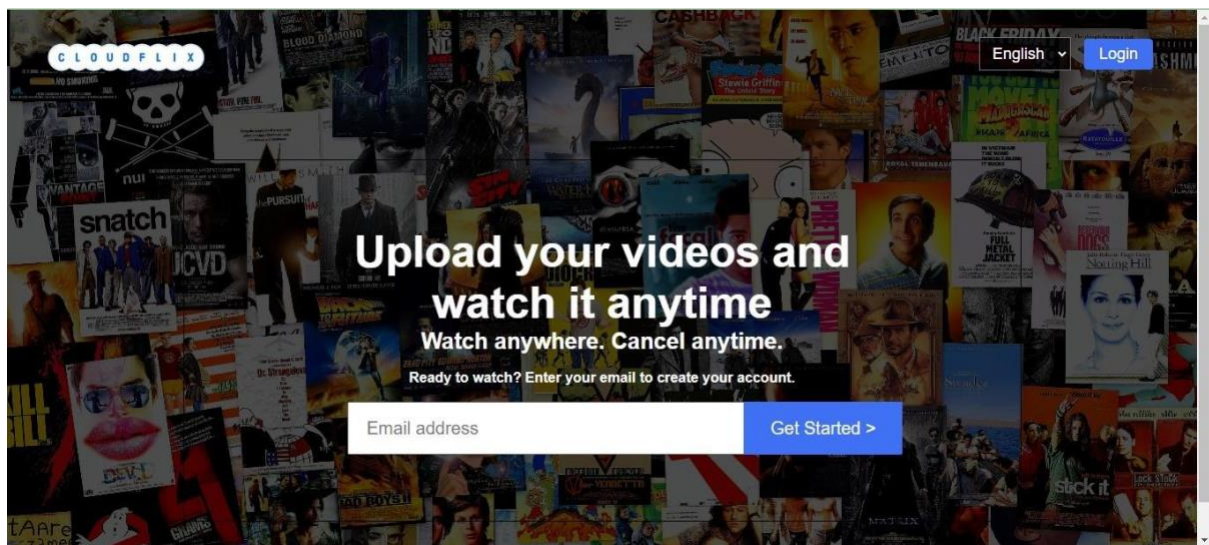
```

```

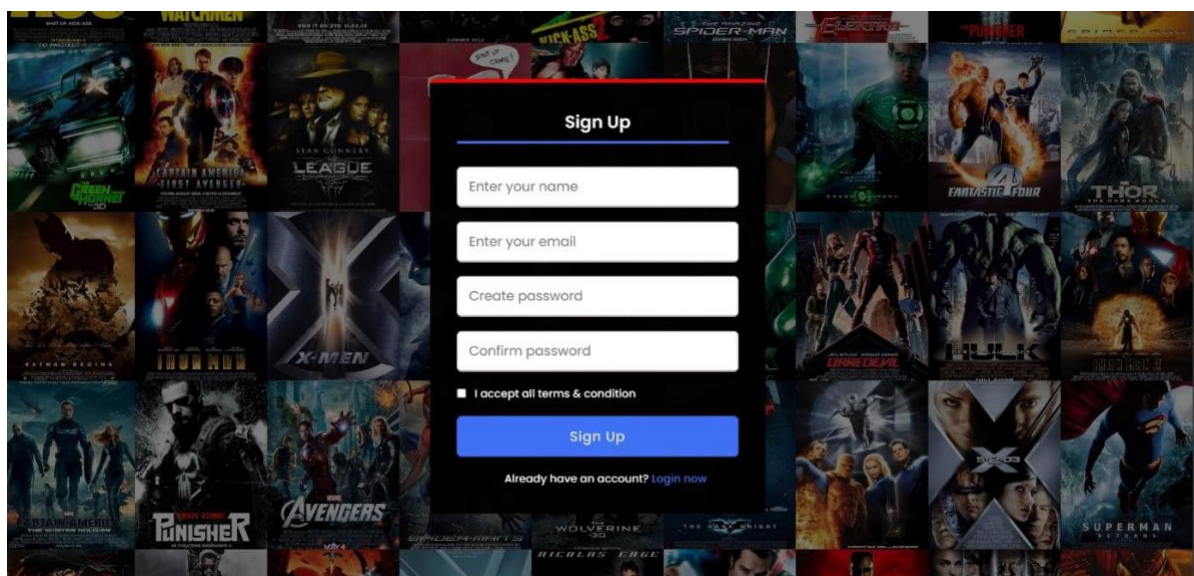
1  <?php
2      $file_name = $_FILES['file']['name'];
3      $tmp_name = $_FILES['file']['tmp_name'];
4      $file_up_name = time() . $file_name;
5      move_uploaded_file($tmp_name, "files/" . $file_up_name); // Specify the path where you want to save the video files
6  ?>

```

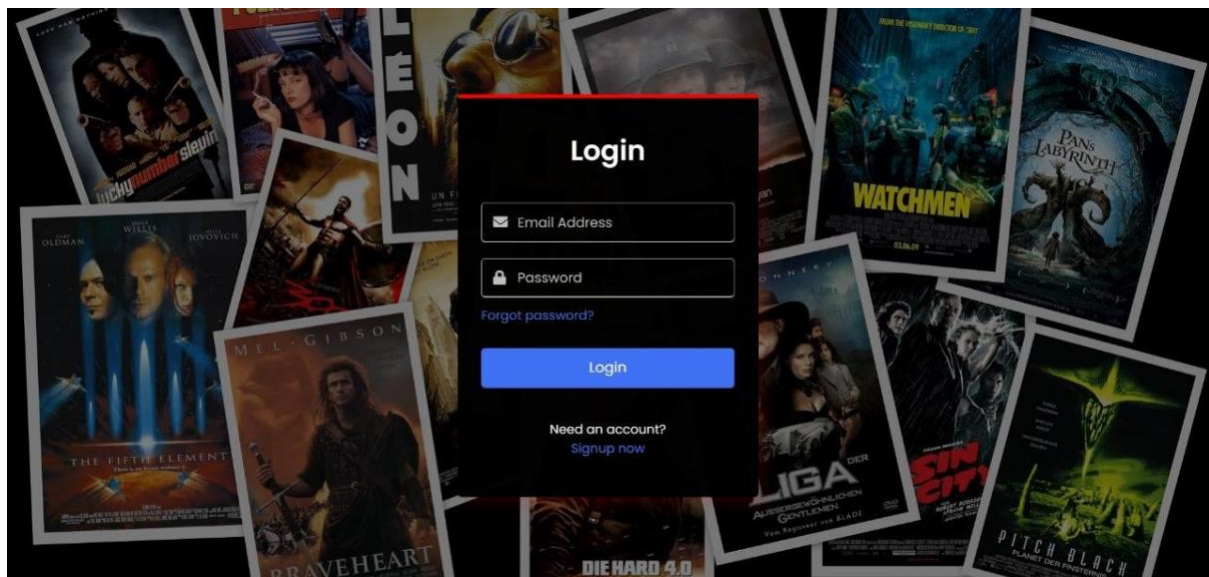

Step 1: Enter your e-mail id to create your account in cloudflx.



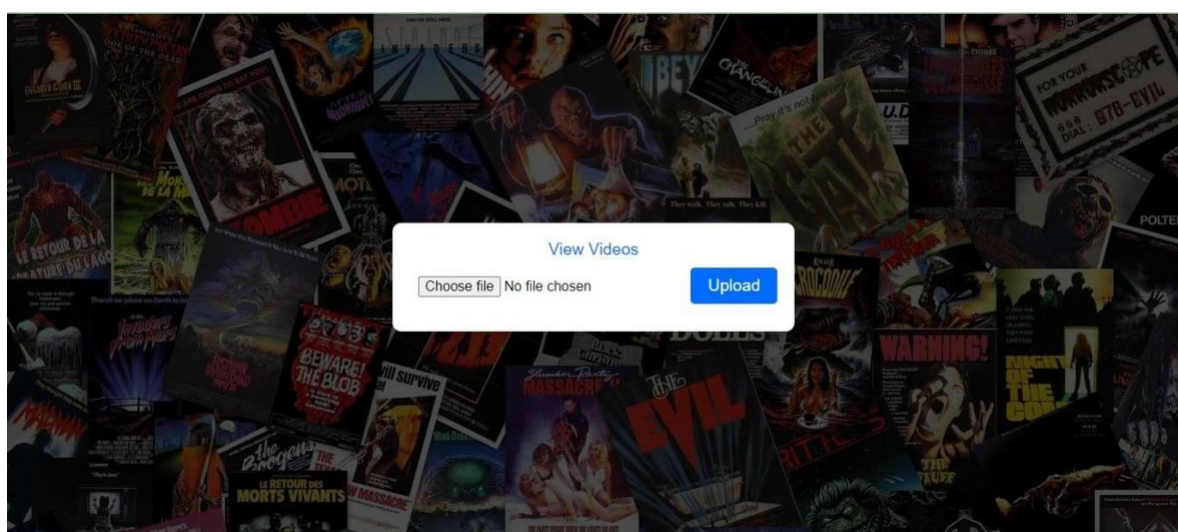
Step 2: Enter your name and create a strong password to complete the sign-up process.



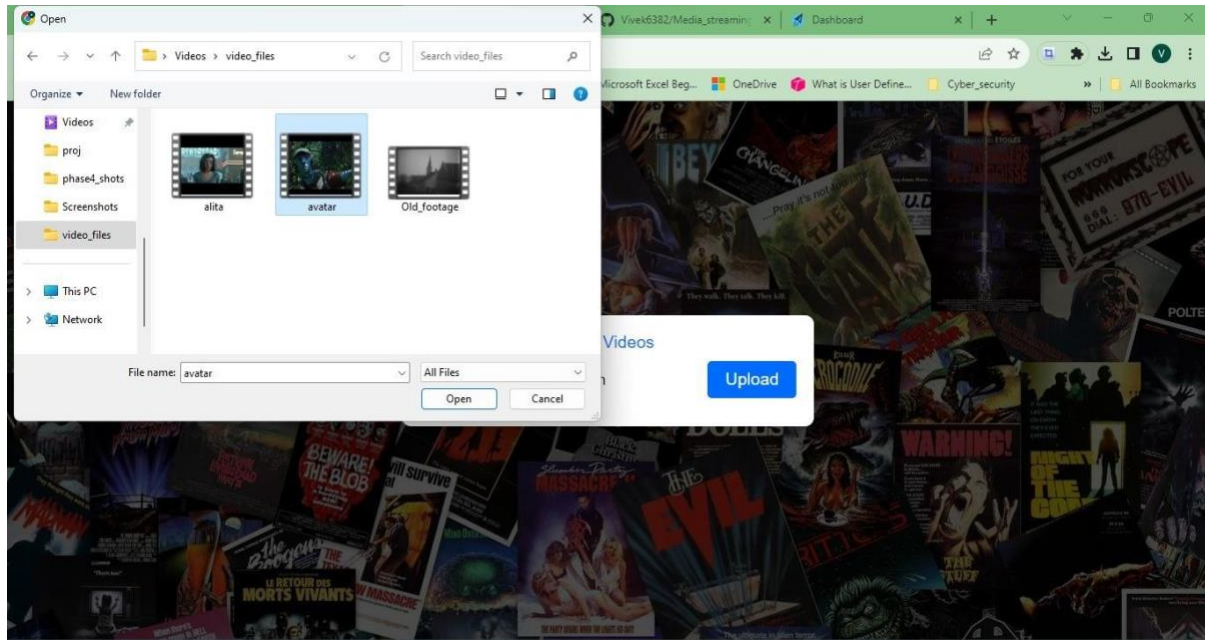
Step 3: Enter the registered mail id and password to log in.



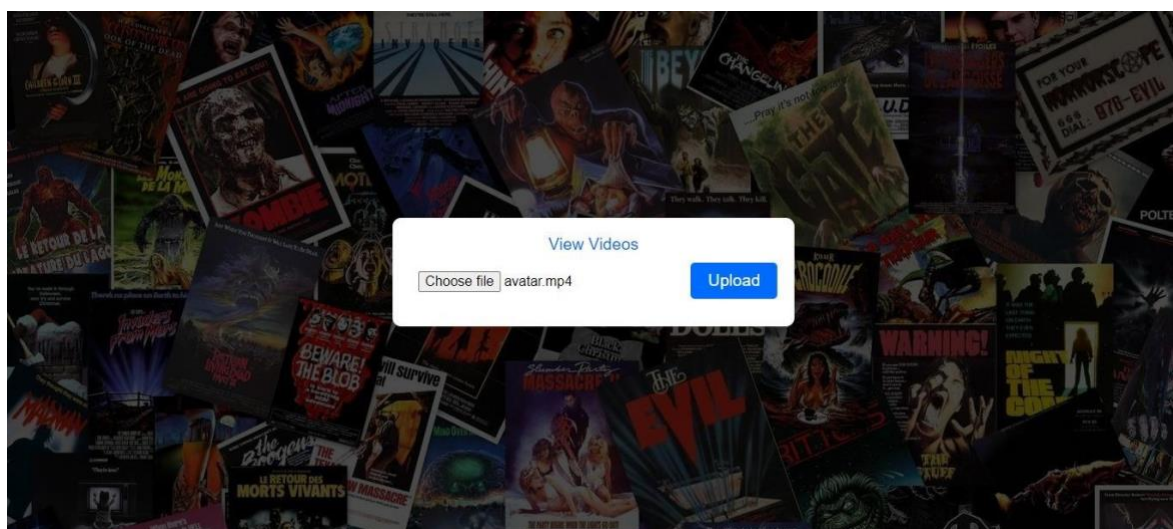
Step 4: Now click the choose file button to select the video to be uploaded.



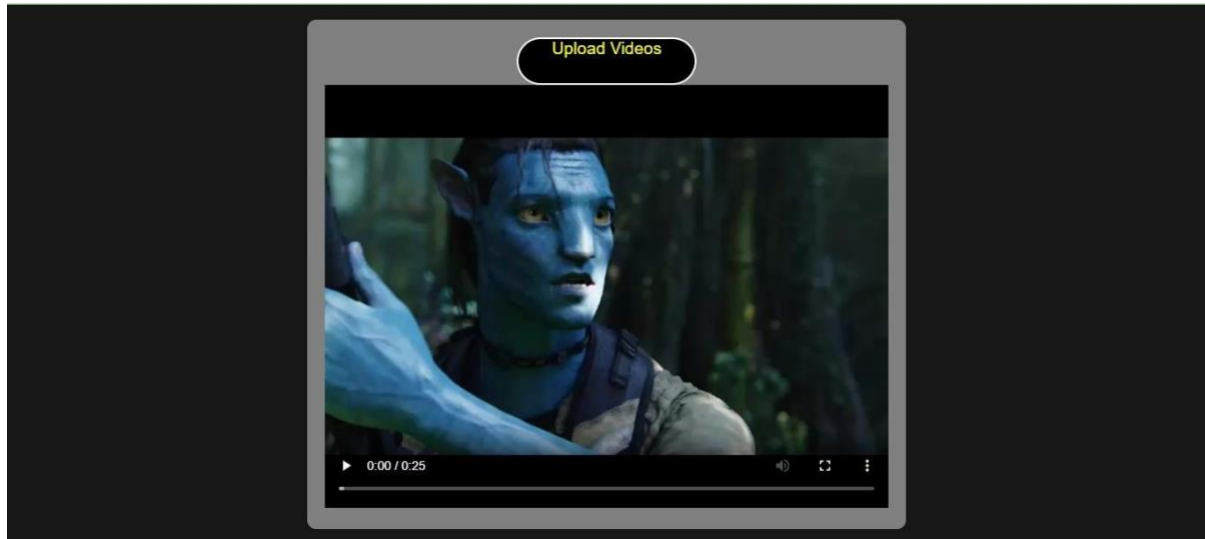
Step 5: Choose the video file which you want to upload and click open.



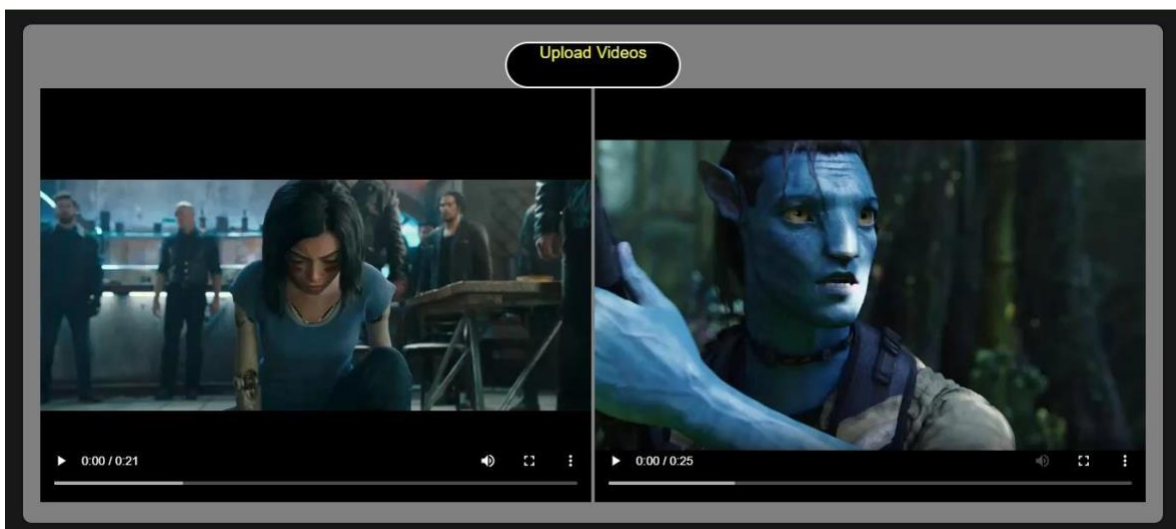
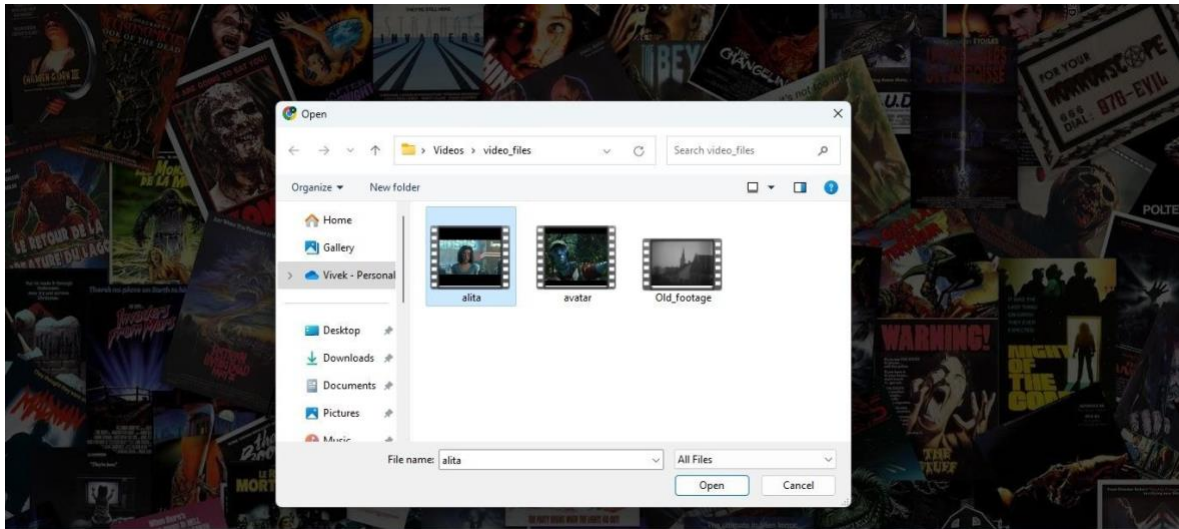
Step 6: Now the name of the chosen file will be displayed.

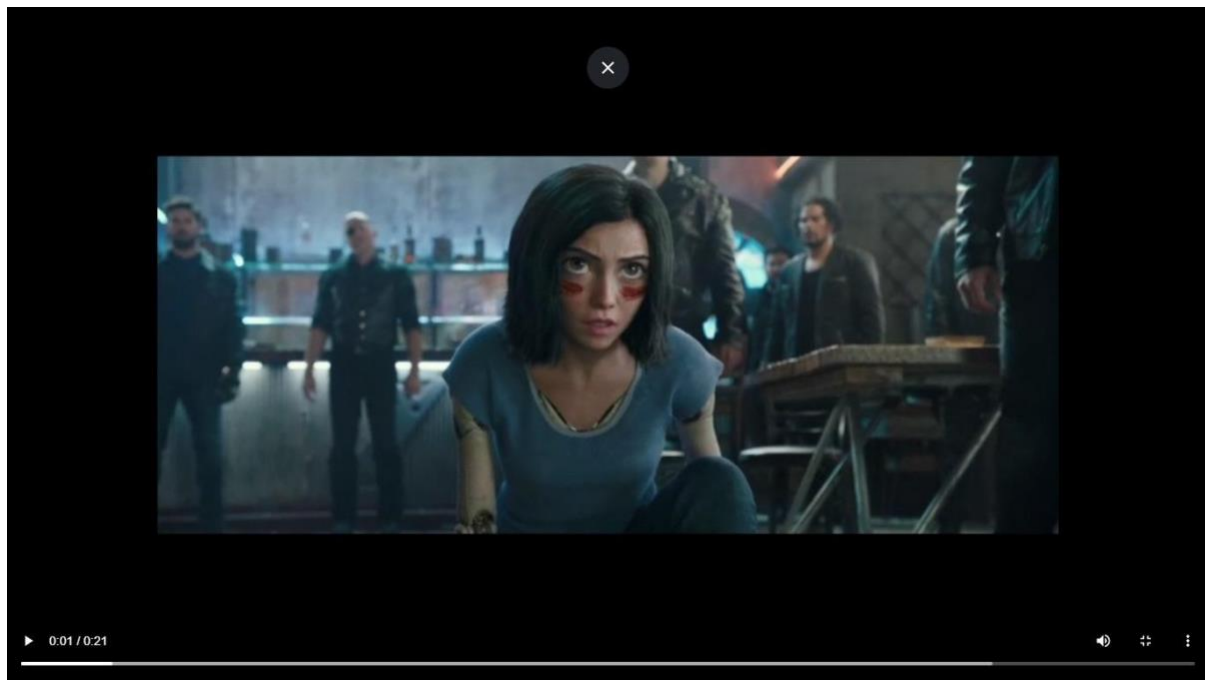


Step 7: Now click upload to upload the selected video file and now the video will be displayed .



Step 8: Likewise you can upload another video and it will be displayed with the previously uploaded video.





CONCLUSION

In conclusion, the implementation of cloud media streaming exemplifies the boundless opportunities technology offers in the realm of content delivery and consumption. With the ability to seamlessly stream media from remote servers to various devices, it not only revolutionizes the way we access and enjoy content but also underscores the immense potential of cloud-based solutions. The scalability, convenience, and cost-efficiency of cloud media streaming make it a game-changer for both consumers and providers, shaping the future of entertainment and information dissemination. As we continue to witness rapid advancements in this field, it is clear that cloud media streaming will play a pivotal role in redefining the digital landscape and how we interact with media.