|  |
| --- |
| **아이디어** **& 요구사항** |

* **외부인 방문 신청** 시 페이지(본격적인 방문 신청 페이지) **이동 전 필수 시청 동영상**(안전 교육 영상)을 시청해야 하는 경우가 존재
* 동영상 시청 중 다음 페이지로 이동하지 못하게 하기 (옵션)

|  |
| --- |
| **이슈** |

1. **브라우저 정책**(브라우저마다 조금씩 다름)으로 인해, 동영상 플레이가 **사용자의 상호작용 없이 자동 재생되지 않게** 되어있음
2. 방문 신청 페이지에서 방문 양식 작성 페이지로 넘어갈 때, **필요한 정보**(Request의 Query String 등...)를 넘겨줘야 함

|  |
| --- |
| **해결 방안** |

1. 페이지를 새로 만들기보다 **해당 페이지에서** 다음 단계로 진행하는 버튼(사용자의 상호작용) 입력 시 페이지가 넘어간 것처럼 보이게 하고 **영상을 재생**
   1. 방문 신청 페이지 로드 시 영상과 함께 로드
   2. 다음 버튼 클릭 시 숨겨진 요소(display: none)를 표현하고 기존 단계는 숨김
2. 필요한 정보는 기존 페이지에서 작성한 대로 사용 가능

|  |
| --- |
| **구현 기능** |

* **비동기 함수를 통해 비디오 플레이어 초기화** 
  + 비디오 요소의 id, 경로, 종료 시 콜백함수 등의 매개변수를 받으며, 표준화된 사용 기능을 여러 초기화 함수를 작성하지 않아도 초기화 하도록 설정
* **비디오 컨트롤 비활성화**
  + 기존 라이브러리에서 제공하는 마우스 우 클릭, 키보드 컨트롤 비활성화 기능을 묶어 둠.
  + 추가적으로 Progress Bar를 통해 사용자가 마음대로 탐색할 수 있던 부분(Seek 탐색)을 Event Listener 로 조작하지 못하도록 막음
* **비디오 소스를 JavaScript로 추가하도록 설정**
  + 하드 코딩 대신 문자열(string)로 비디오 소스를 추가하도록 함수 추가

|  |
| --- |
| **사용 방법** |

[Plyr 라이브러리](https://github.com/sampotts/plyr) 관련 문서는 [링크](https://github.com/sampotts/plyr)(https://github.com/sampotts/plyr) 참조

1. JavaScript 및 HTML 요소 삽입
   1. 폴더 복사 및 덮어쓰기 텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

      자동 생성된 설명
   2. JavaScript, CSS, HTML 요소 삽입

|  |
| --- |
| <!-- Video Player 소스 시작 -->  <div>  <video id="RequiredVideo" width="100%" playsinline></video>  </div>  <link href="/library/plyr/plyr.css" rel="stylesheet" type="text/css" />  <script src="/js/WatchVideo.js" type="text/javascript"></script>  <script>  videoManager.init(  'RequiredVideo',  '/Video/RequiredVideo.mp4',  onVideoEnd,// 콜백 함수 (이후 문서에서 나옴)  [  'play-large', // 큰 재생 버튼  'play', // 재생/일시정지 버튼  'progress', // 재생 시간 바  'current-time', // 현재 재생 시간  'mute', // 음소거 버튼  'volume', // 볼륨 조절  'fullscreen' // 전체 화면 버튼  ],  { selected: 1, options: [1] },  false  );  </script>  <!-- Video Player 소스 종료 --> |

1. <video> 태그의 id는 적당히 작성할 것
2. CSS의 경우 library 폴더에 있는 plyr 라이브러리 소스 사용
3. Script의 경우
   1. js 폴더에 있는 WatchVideo.js
   2. 라이브러리 그대로 사용하려는 경우(더 많은 기능을 필요로 하거나 막는 기능이 필요 없는 경우) ` /library/plyr/plyr.min.js` 파일을 사용  
      (현재는 WathchVideo.js에서 동적으로 로딩 중)
4. videoManager(WatchVideo에서 만든 객체)의 `init()` 함수를 통해 초기화
   1. 필요한 경우, 비디오가 끝나면 호출하는 콜백 함수를 정의해서 매개변수로 넣어줄 수 있음

|  |
| --- |
| // 비디오 시청 완료 여부  let completed = false;  // 콜백 함수  const onVideoEnd = () => {  completed = true;  // confirm 요소로 확인 시 fnSubmit을 호출  confirm('안전 교육 영상을 끝까지 시청하셨습니다. 다음 단계로 진행하시겠습니까?') && fnSubmit();  }; |

1. 현재 문서에서는 `onVideoEnd` 변수가 콜백 함수를 담고 있으며, completed 변수는 비디오 시청 완료 여부를 확인하는 변수로 사용
2. Confirm()이 true가 되면 fnSubmit() 호출

(ps1. 이후 `.cshtml` 변경 사항은 기존 문서의 주석으로 처리되어 있으나, JavaScript의 경우 `.min.js` 말고 `.js`의 주석을 참조할 것)

(ps2. `.min.js`는 배포용 문서)