|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **과제 보고서**  **(2024.07.09)** | | 결  재 | 작 성 자 | 선 임 | | 부 장 | | 위 원 |
|  |  | |  | |  |
| 작 성 자 | 서원형 | | | | 작 성 일 | | 24.07.09 | |

|  |  |  |  |
| --- | --- | --- | --- |
| 과제명 | 새로고침, reload시 frame 출력 개선 | 진행도(%) | 100 |
| 결론 | - 나라장터 새로고침, reload시 frame 출력을 개선함.  [남은 문제점]  URL이 이동하지 않는 frame에서는 오버레이가 이상출력되는 새로운 문제가 발생함. | | |
| 진행 과정 | **[문제점]**  2024년 7월 전임자 문서에 특정 페이지 접속시 오버레이가 이상출력되는 문제가 존재하였고, 페이지가 로드된 상태에서 셀렉터를 실행하는 것이 아닌 셀렉터를 실행하고 페이지를 로드하게 되면 오버레이가 이상 출력되는 문제가 발생했다.  이 보고서 이전 시점에 남은 오버레이 이상 출력은 다음과 같다.  1. iframe과 frame의 구분없이 동일한 코드로 적용시켜 어디서 에러가 나는지 특정이 안됨.  2. X, Y좌표가 초기화되지 않고 계속 오버레이 출력에 영향을 주는 것.  **[해결 방법]**  우선 문제 상황을 재현하고, 다양한 시도를 통해 문제를 규명하였다.  생각한 방안으로는 1. 페이지 로드시 에러, 2. 프레임 에러, 3. 세션, 쿠키 에러, 4. 코드 설계에러로 구분하여 다양한 상황에서 문제를 구현해보고 로그를 통해 코드를 역추적하였다.    (크롬 설정에서 볼 수 있는 배율 값)  <오버레이 이상 출력 상황 구현>  하지만, 이것으로는 에러를 규명할 수 없었고, frame 요소의 특성에 대해 조금 더 깊게 찾아보았다. 이때, 최신 구조인 iframe보다 느리며 페이지 로드에 시간이 오래 걸릴 수 있다는 점을 착안하여 페이지가 로드되는 네트워크 부분을 확인하였다. 또한, 간헐적으로 원인 불명한 에러가 발생했기 때문에 네트워크 로드 속도에 따라 에러가 발생할 수도 있겠다고 생각하였다.    C:\Users\USER\Desktop\서원형\1000ms 이상로드.png  <정상적으로 로드된 상황> <간헐적으로 네트워크가 느려져 frame이 늦게 로드되는 상황>  위 두 사진에서 볼 수 있듯, 간헐적으로 네트워크가 느려져 frame이 늦게 도착하여 에러가 발생하는 것으로 정확한 원인을 규명할 수 있었고 이에 대한 대응책으로 frame을 불러오는 코드는 비동기 처리를 통해 모든 frame이 로드된 이후 셀렉터에 정보를 보내는 함수를 실행하는 방향으로 코드를 설계하였다.    (익스텐션 – content.js – delayedContent() 內 조건문 notifyFrames()를 비동기 함수로 변환하여 await을 추가하였다.)  비동기처리를 통하여 frame이 로드되지 않을 경우에는 beBOT으로 데이터를 보내지 않고 정상적으로 frame이 로드되고 그 때부터 셀렉터의 오버레이가 정상작동되게 구현하였다.  **[결론]**  frame이 존재하는 페이지를 reload하거나 새로고침하여도 정상적으로 frame요소를 다시 불러와 셀렉터에 잘 출력된다.      <셀렉터 이상출력 문제 – 해결 전> <셀렉터 이상출력 문제 – 해결 후>  위 그림과 같이 새로고침을 하거나 reload를 하여도 정상적으로 출력된다. 네트워크 속도에 따라 차이가 있지만, 로드되지 네트워크로 인해 정상적으로 받아올 수 없었던 frame 정보가 있다면, 다시 frame 정보를 reload하는 비동기 로직을 구현하여 모두 정상적으로 구현된다.  단, **새롭게 발견한 에러 사항**으로는 frame 內 document가 변화할 때 에러가 존재한다. 크롬 익스텐션의 코드는 모두 URL이 새롭게 로드될 경우에만 새로 프레임을 불러오는 방향으로 설계하였는데, 해당 사이트는 URL이 변하지 않고 frame set > frame > frameset > frame안의 document만 변화하는 방식으로 이에 대한 해결 방안이 추가적으로 요구된다.  아래의 표는 예상되는 해결 방법이다.   |  |  |  | | --- | --- | --- | | 예상 해결방법 | 장점 | 단점 | | 나라장터 사이트 예외처리(1안) | 자원 효율성 증대, 특성화 가능 | 유연성 저하, 모듈화 불가 | | frame 구조 재귀적 탐색(2안) | 유연성 증대, 모듈화 가능 | 자원 효율성 저하, 특성화 불가 | | 1안, 2안 동시 사용 | 유연성 증대, 특성화 가능, 모듈화 가능 | 자원 효율성 저하 | | | |