UML, 코드 linking 정상 시스템

**시작 조건**

사용자가 팀 스페이스 멤버이고 로그인 상태에서 팀 스페이스 내의 UML페이지 또는 코드 스크립트 페이지로 진입해야함.

**사건의 흐름**

1. 사용자가 UML페이지의 사이드바에 존재하는 Test1 클래스 다이어그램을 선택한다.

2. 사용자가 UML페이지의 사이드바에 존재하는 “UML Linking code”를 선택한다.

3. 시스템은 코드 스크립트 데이터에서 클래스 명이 Test1인 클래스를 탐색한다.

4. 시스템은 Test1클래스를 UML데이터와 코드 스크립트 데이터를 링킹한다.

5. 링킹 데이터에 Test1클래스가 추가된다.

6. 사용자가 UML 페이지에서 Test1 클래스명을 Test2로 변경한다.

7. 시스템은 링킹 데이터의 데이터를 수정해서 Test1을 Test2로 변경한다.

8. 시스템은 수정된 내용을 코드 스크립트에 반영하여 코드 스크립트 내의 Test1클래스를 Test2로 변경한다.

9. 사용자가 사이드바에 존재하는 IDE페이지를 선택한다.

10. 사용자가 IDE페이지에서 Test2클래스의 파일로 들어간다.

11. 사용자가 Test2 클래스에 멤버 함수로 test3함수를 생성한다.

12. 시스템은 생성된 test3를 링킹 데이터에 반영하여 test3 함수를 추가한다.

13. 시스템은 UML 수정된 내용을 UML 데이터에 반여하여 UML 페이지 내의 Test2클래스에 test3 함수를 추가한다.

**종료 조건**

사용자가 다른 페이지로 이동하거나 프로그램을 종료한다.

UML, 코드 linking 비정상 시스템

**시작 조건**

사용자가 팀 스페이스 멤버이고 로그인 상태에서 팀 스페이스 내의 UML페이지 또는 코드 스크립트 페이지로 진입해야함.

**사건의 흐름**

1. 사용자 1가 UML페이지의 사이드바에 존재하는 Test1 클래스 다이어그램을 선택한다.

2. 사용자 1가 UML페이지의 사이드바에 존재하는 “UML Linking code”를 선택한다.

3. 시스템은 코드 스크립트 데이터에서 클래스 명이 Test1인 클래스를 탐색한다.

4. 시스템은 Test1클래스를 UML데이터와 코드 스크립트 데이터를 링킹한다.

5. 링킹 데이터에 Test1클래스가 추가된다.

6. 사용자 1가 UML 페이지에서 Test1 클래스명을 Test2로 변경한다.

7. 시스템은 링킹 데이터의 데이터를 수정해서 Test1을 Test2로 변경한다.

8. 사용자 2가 IDE 라이브 코드 스크립트 환경에서 Test1클래스 파일의 이름을 Test0로 변경한다.

9. 시스템은 UML데이터와 코드 데이터의 변경사항을 고려하지 못해 오류를 발생시킨다.

**종료 조건**

사용자가 다른 페이지로 이동하거나 프로그램을 종료한다.