

Requirement Engineering

20145736 장우진

20143774 박찬우

20146420 김소영

20142167 최현경

목차

Requirement for Simple Merge.....3

제시된 Program Requirement.....	3
시작화면	3
파일 비교.....	3
파일 Merge	3
Requirements 정리.....	4
Functional requirement.....	4
Non - Functional requirement.....	6

Requirement Engineering of Model, View, Controller.....6

Requirement of Model.....	7
Requirement of View	8
Requirement of Controller	13

Usecase15

User Scenario	15
Sequence Diagram.....	16
Case1) Compare	17
Case2 Merge.....	18
Case3) File Open	16

Requirement for Simple Merge

제시된 Program Requirement

시작화면

프로그램을 시작하면, 두 개의 edit 패널을 가진 메인 윈도우가 보여진다. 각각의 패널의 위쪽에는 "Load", "Edit", "Save" 버튼이 존재한다.

만약 사용자가 "Load" 버튼을 누른다면, 프로그램을 사용자가 컴퓨터에 저장된 파일을 선택하여 프로그램으로 불러올 수 있도록 하며, 불러온 파일의 내용은 edit 패널에 나타나게 된다.

만약 사용자가 "Edit" 버튼을 누른다면, 프로그램은 사용자가 edit 패널에서 문장을 수정할 수 있도록 한다.

만약 사용자가 "Save" 버튼을 누른다면, 프로그램은 edit 패널의 내용을 컴퓨터의 파일에 저장한다.

파일 비교

유저는 두 파일을 선택하여 그 내용을 비교할 수 있어야 한다. 프로그램은 두 파일의 차이점이 있는 라인을 폰트 나 다른 색의 배경을 통해 나타내야 한다. 또한 프로그램의 비교 결과는 Diff 프로그램의 결과와 같아야 하며, Diff 프로그램은 두 파일을 비교하는데 "LCS 알고리즘" 을 사용한다. 이 실행에서 merge 기능을 위해 블록은 identical 해야 한다.

파일 Merge

유저는 파일을 비교 한 후 나타난 차이점을 선택하여 그 차이점을 Merge 할 수 있어야 한다. 기본적인 Merge 버튼으로는 "**Copy to Right**" 와 "**Copy to left**" 가 있다. "**Copy to Right**" 버튼은 왼쪽 패널의 선택된 블록을 복사하여 오른쪽 패널의 차이점에 붙여 넣는 것이다. 비슷하게 "**Copy to Left**" 버튼은 오른쪽 패널의 선택된 블록을 왼쪽 패널의 차이점에 복사해 넣는 것이다.

Requirements 정리

Functional requirement

- 프로그램이 시작될 때, 하나의 탭이 존재하며, 어떠한 파일도 열려있지 않는다.
- 탭은 한번에 하나만 보여준다.
- 각 탭은 독립적이며, 두 개의 파일을 위한 text area 를 가진다.
- Each text areas included by tab showing now shows text data which had by the tab.
- "new tab" 이벤트가 일어나면, 현재 존재하는 탭들의 가장 마지막에 탭이 하나 추가되며, Window 메뉴에도 해당하는 메뉴 아이템이 생성된다. 컨트롤러는 이것을 감지하여 적절한 일을 수행하며, 모델에게도 이를 알려 새로운 탭에 대한 작업을 할 수 있도록 한다.
- "Close tab" 이벤트가 발생하면, 종료시킨 탭이 화면에서 사라지며 Window 메뉴에서도 해당하는 메뉴 아이템이 사라지며, 컨트롤러는 이것을 감지하여 해당 tab 에 대한 정보를 삭제하고, 모델에도 이를 알려 모델 역시 해당하는 데이터를 삭제시킬 수 있도록 한다.
- 다른 탭을 선택하는 이벤트가 일어나면, 컨트롤러는 이를 감지하여 메인 윈도우와 메뉴 아이템들의 활성화를 조절한다.
- "Load file" 이벤트가 발생하면, 컨트롤러는 이를 감지하여 파일 패스를 가져올 수 있는 새 창을 만들어주고 모델에 유저가 선택한 파일의 패스를 넘겨준다. 모델은 이를 받아 컨트롤러에게 파일의 데이터를 주고, 컨트롤러는 이것을 패널에 넣어 화면에 보여준다.
- "Save file" 이벤트가 발생하면, 컨트롤러는 이것을 감지하여 모델에게 알려주고 모델은 파일을 저장한다.
- 파일을 읽거나 저장할 때, 오직 그 때만 파일 객체는 존재하여야 하며, 일이 완료된 후 파일 객체는 사라져야 한다.
- "Compare" 이벤트가 일어나면, 컨트롤러는 이것을 감지하여 모델에게 파일의 비교를 요청하고, 그 결과값을 받아서 화면에 비교된 결과를 block 과 색깔을 통하여 보여주어야 한다
- 각 탭마다 오직 단지 하나의 비매칭 데이터 그룹이 선택될 수 있다. '비매칭 데이터 그룹'의 의미는 연속된 매칭되지 않는 라인들을 말한다.

- "Move up" / "Move down" 이벤트는 compare 완료된 후 발생할 수 있고, 이 이벤트가 발생하면 컨트롤러는 이것을 감지하여 선택된 차이점을 그 밑의 차이점이나 그 위의 차이점으로 옮겨 주고 그 위치에 맞게 화면의 스크롤 바가 움직여야 한다.
- 움직임에 관련된 기능은 5개로 **"First Difference"**, **"Last Difference"**, **"Previous Difference"**, **"Next Difference"** 와 **"Current Difference"** 가 버튼 및 메뉴 아이템으로 존재한다.
- **"First Difference"** 는 선택된 차이점을 첫 번째 차이점으로 옮겨준다.
- **"Last Difference"** 는 선택된 차이점을 마지막 차이점으로 옮겨준다.
- **"Next Difference"** 는 선택된 차이점을 그 다음 차이점으로 옮겨준다
- **"Previous Difference"** 는 선택된 차이점을 그 이전의 차이점으로 옮겨준다.
- **"Current Difference"** 는 현재 선택된 블록이 차이점이 아닐 경우 가까운 차이점을 선택된 차이점으로 한다.
- "Merge" 이벤트는 오직 compare 가 완료된 후 발생할 수 있고, 이 이벤트가 일어나면 컨트롤러는 이것을 감지하여 모델에 선택된 block 의 merge 를 요청하고 그 결과를 받아서 화면에 업데이트 시켜준다.
- "merge" 에 관련된 기능은 4개로 **"Copy to Right"**, **"Copy to Left"**, **"Copy to Right All"**, **"Copy to Left All"** 이 버튼과 메뉴 아이템으로 존재한다.
- **"Copy to Right"** 는 선택된 차이점의 왼쪽 패널의 내용을 오른쪽 패널에 복사해 넣는다.
- **"Copy to Left"** 는 선택된 차이점의 오른쪽 패널의 내용을 왼쪽 패널에 복사해 넣는다.
- **"Copy to Right All"** 은 전체 차이점을 왼쪽 패널의 내용과 동일하게 한다.
- **"Copy to Left All"** 은 전체 차이점을 오른쪽 패널의 내용과 동일하게 한다.

Non - Functional requirement

- 사용성 요구사항 : 이 프로그램은 직관적으로 사용될 수 있어야 한다. 이 문장에서 직관적이라는 것의 의미를 컴퓨터를 사용할 줄 아는 네 명 정도의 사람을 무작위로 선택하였을 때 프로그램의 기능을 어려움 없이 사용할 수 있어야 한다는 것으로 정의한다.
- 효율성 요구사항 : 이 프로그램은 100000줄 이하의 텍스트의 비교, 병합에 대하여 5초 안에 시행되어야 한다.
- 신뢰성 요구사항 : 프로그램을 허용된 조작 하에서 사용 시 어떤 방법으로도 기능이 제대로 수행되지 않는 경우는 없어야 한다.
- 이식성 요구사항 : 이 프로그램은 JAVA가 작동되는 일반적인 윈도우 운영체제 하에서 실행된다.
- 구현 요구사항 : 테스트 기반 디자인 모델을 채용하여야 하며, 테스트 케이스는 Junit과 EasyMock , Gui Testing Framework 를 사용해서 만들어야 한다.
- 구현 요구사항 : MVC 디자인 모델을 채용하여야 한다.
- 구현 요구사항 : OOP 컨셉에 할 수 있는 한 맞추어서 디자인하여야 한다.
- 인도 요구사항 : 보고서에는 테스트 기반으로 만들었다는 사실의 서술, mvc 컨셉이 어떻게 반영되었는지에 대한 서술, UI 테스트 프레임워크에 대한 서술, oop 원칙을 어떻게 적용하였는지에 대한 서술, 사용례와 스크린샷, 테스트 케이스와 그 결과가 포함되어야 하며 이 보고서는 제출 기한 전까지 인도되어야 한다.
- 표준 요구사항 : 프로그램 개발자는 JAVA, JAVAfx를 사용하여 프로그램을 만들어야 하며, Git을 통하여 프로그램 개발 상태를 공유하여야 한다.

Requirement Engineering of Model, View,

Controller

Requirement of Model

Merge	
Next Difference	현재 선택된 블록의 인덱스를 +2 해준다. 단, 다음 인덱스가 파일의 끝을 넘길 경우 더이상 이동하지 않고 비활성화된다.
Prev Difference	현재 선택된 블록의 인덱스를 -2 해준다. 단, 다음 인덱스가 파일의 끝을 넘길 경우 더이상 이동하지 않고 비활성화된다.
First Difference	블록의 인덱스를 각 케이스에 따라 0 혹은 1로 변경해준다. prev Difference를 비활성화한다.
Now Selected Difference	클릭된 인덱스에서 가장 가까운 오류 블록 인덱스로 인덱스를 이동시킨다. 보통 바로 다음 인덱스로 이동하지만 다음 인덱스가 없는 경우 이전으로 이동한다.
Last Difference	블록의 인덱스를 가장 끝 오류지점으로 변경해준다. next Difference를 비활성화.
Copy To Right	현재 선택된 차이점 블록에서, 왼쪽의 내용을 오른쪽으로 복사하고 화면에 출력한 후 다시 Compare 한다.
Copy To Left	현재 선택된 차이점 블록에서, 오른쪽의 내용을 왼쪽으로 복사하고 화면에 출력한 후 다시 Compare 한다.
Copy To Right All	오른쪽의 내용 자체를 왼쪽의 내용으로 옮기고 Compare, 화면에 출력한다.
Copy To Left All	왼쪽의 내용 자체를 왼쪽의 내용으로 옮기고 Compare, 화면에 출력한다.
Compare	현재 탭에서, 불러온 양 파일의 내용을 비교하고, 서로 맞은 부분과 다른 부분을 블록별로 나누어 ArrayList에 저장하도록 한다. 선택된 블록의 인덱스를 맨 처음으로 바꾸고, 케이스에 따라 다른 ListLayout을 출력하게 한다. 파일의 차이점이 없을 경우, ~Difference와 Copy To~ 버튼들을 비활성화, 차이점이 있을경우 모두 활성화한다.

Requirement of View

View 의 Requirement 는 다음과 같이 파워포인트를 이용하여 시각적으로 표현하여 직관적으로 이해할 수 있게 하였다.

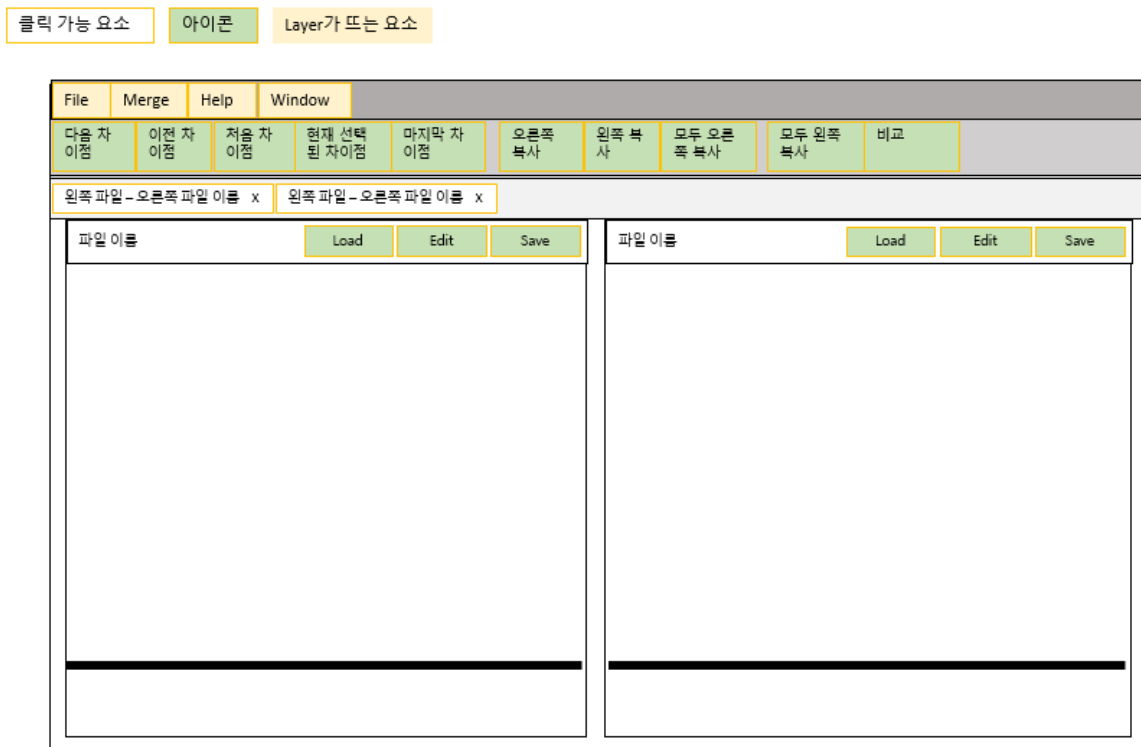


Figure 1 메인 윈도우의 화면

- 메인 윈도우 프로그램의 실행될 때 나타나게 되는 화면이며 다음과 같다. Menu 로는 "File", "Merge", "Help", "Window" 가 존재하며 각각의 설명은 아래쪽에 설명되어있다.
- 버튼은 총 10개로 Difference 관련이 5개, Merge 관련이 4개, Compare 가 하나이다
- 창은 Tab 으로 관리되며, 각 탭에는 각각 왼쪽과 오른쪽 파일에 대한 패널이 존재한다.
- 각 패널은 파일 이름을 알려주는 라벨과, "Load", "Edit", "Save" 버튼이 존재하고, 파일 내용을 담을 Text area 가 중앙에 존재하고 프로그램의 로그를 찍을 Text area 하단에 존재한다

클릭 가능 요소

아이콘

Layer가 뜨는 요소

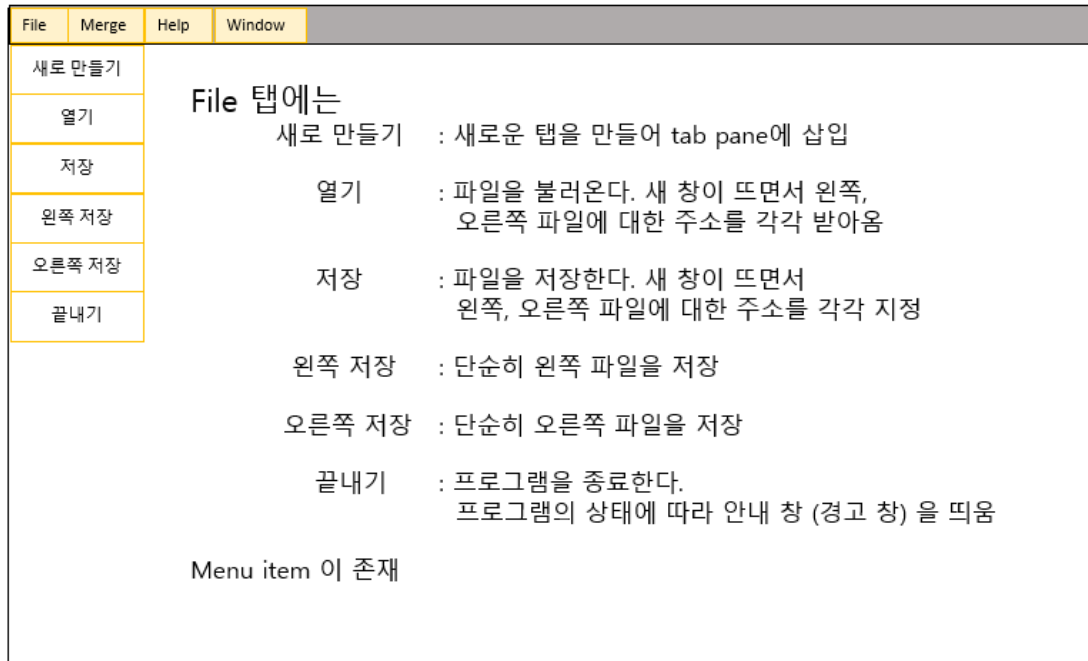


Figure 2 "File" 탭 구성요소

클릭 가능 요소

아이콘

Layer가 뜨는 요소

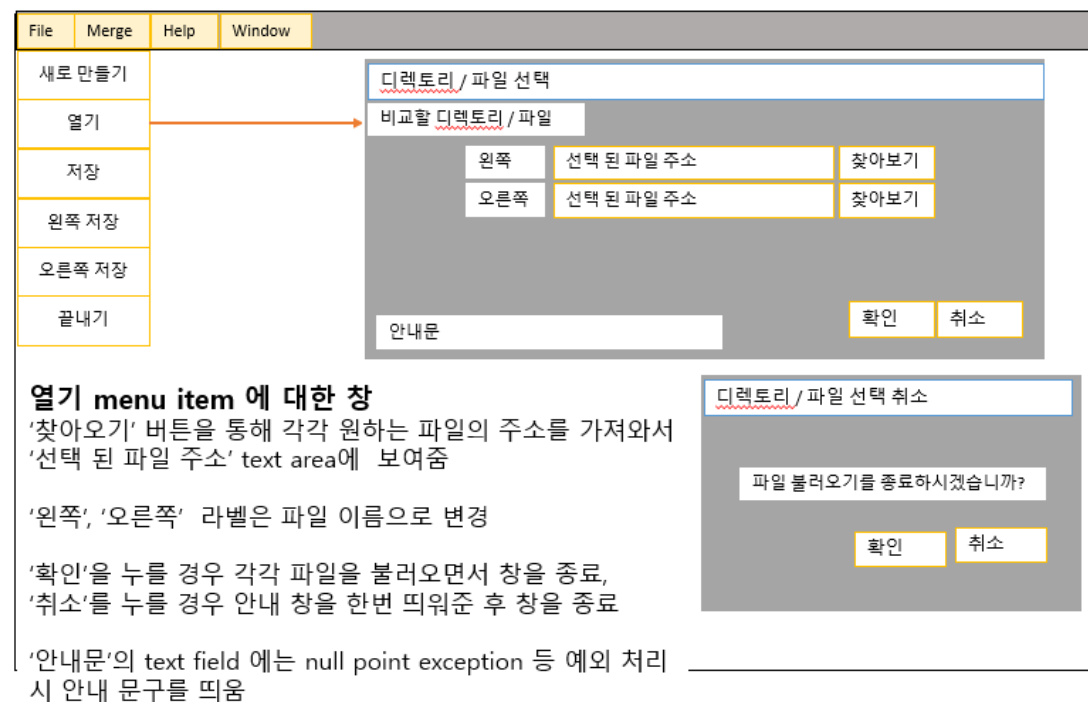


Figure 3 Open File Window

클릭 가능 요소 아이콘 Layer가 뜨는 요소

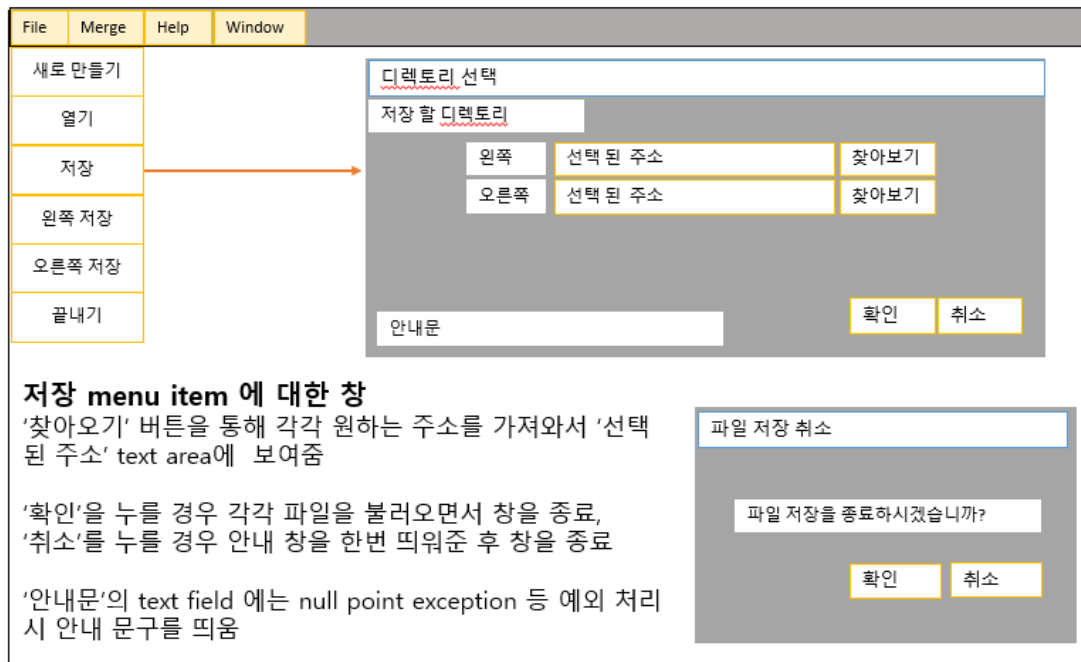


Figure 4 Save File Window

클릭 가능 요소 아이콘 Layer가 뜨는 요소

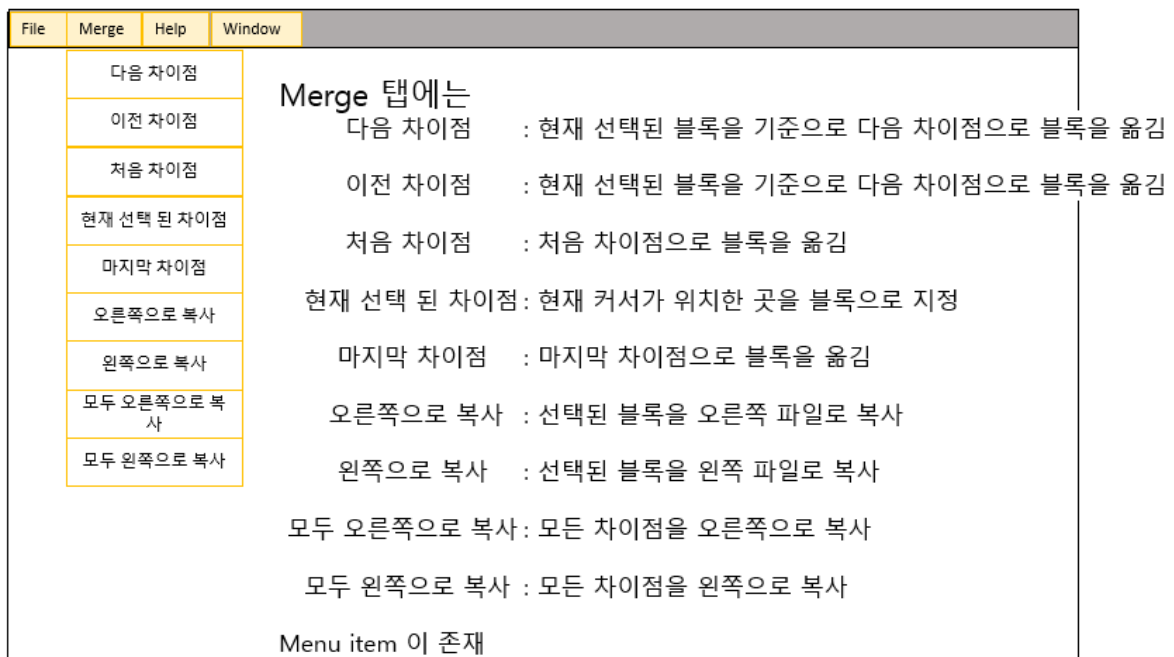


Figure 5 "Merge" 탭 구성요소

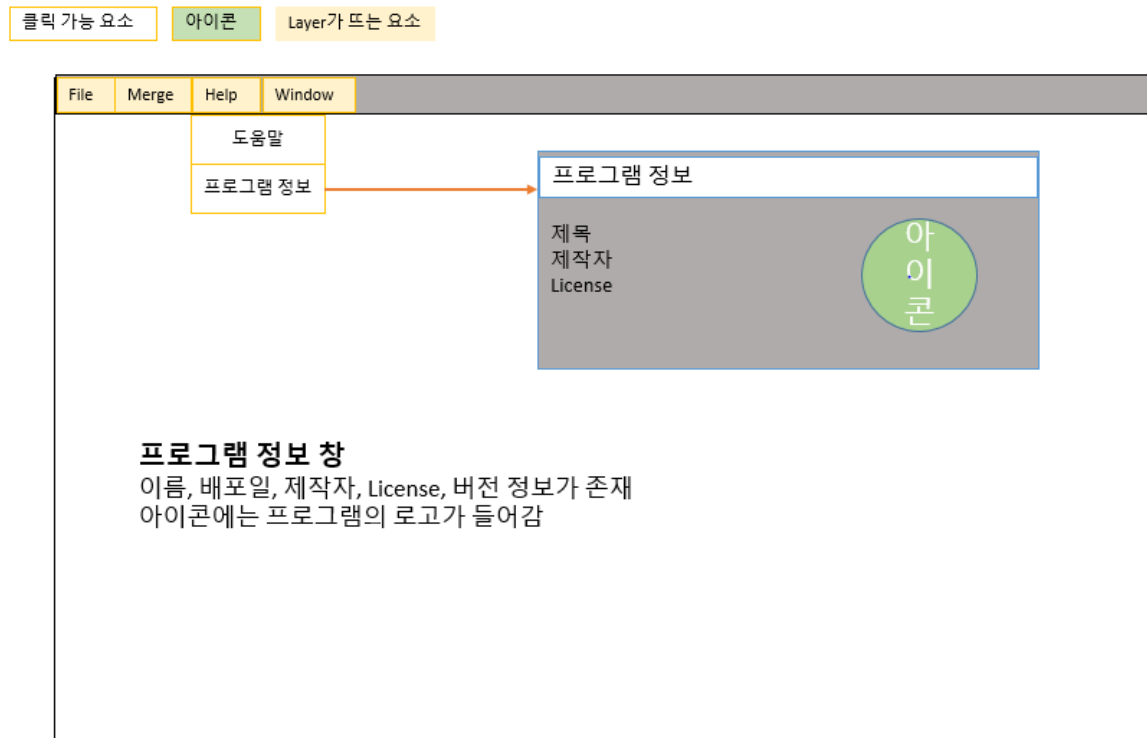


Figure 6 "Help" 탭 구성요소

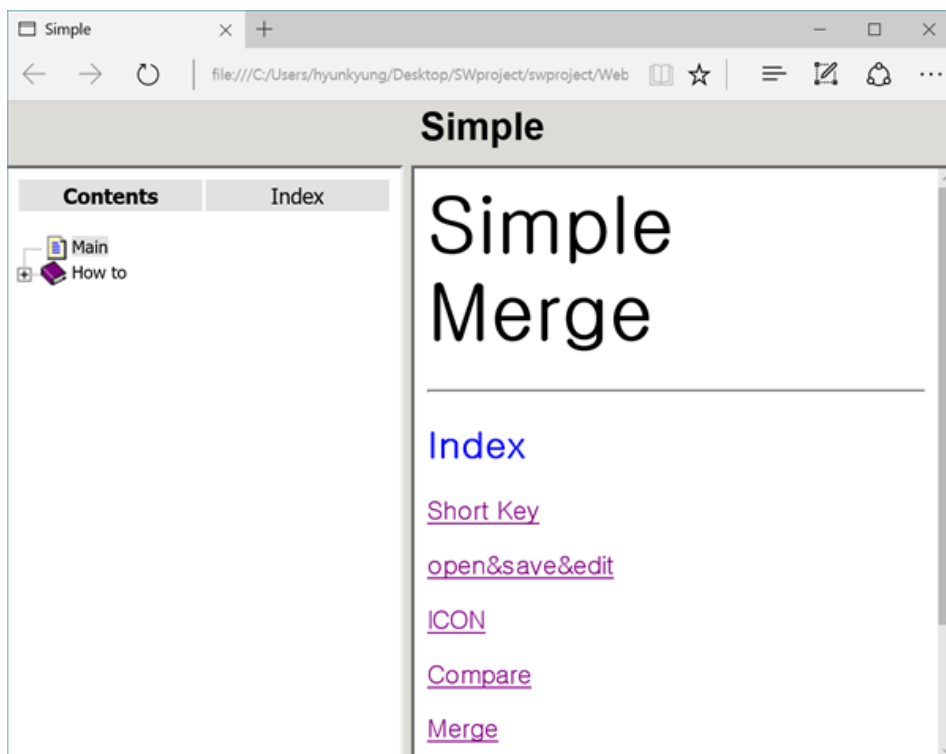


Figure 7 "Help" 메뉴 "Help" 메뉴 아이템 - 실제 도움말 창

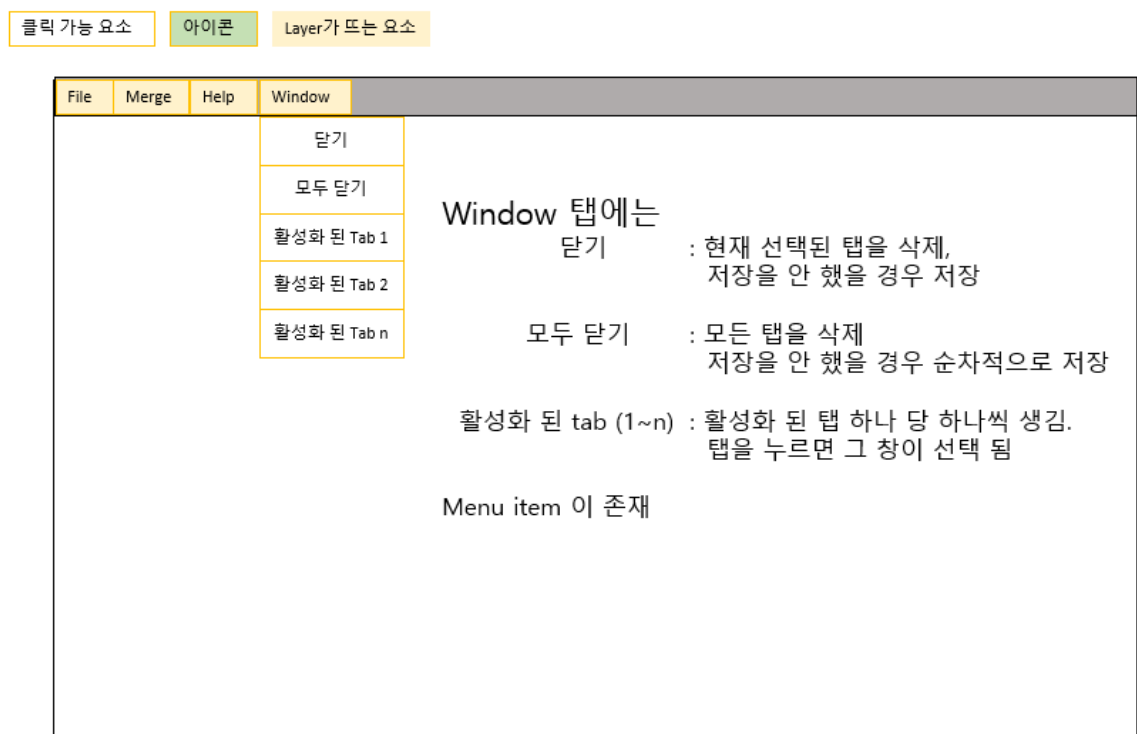


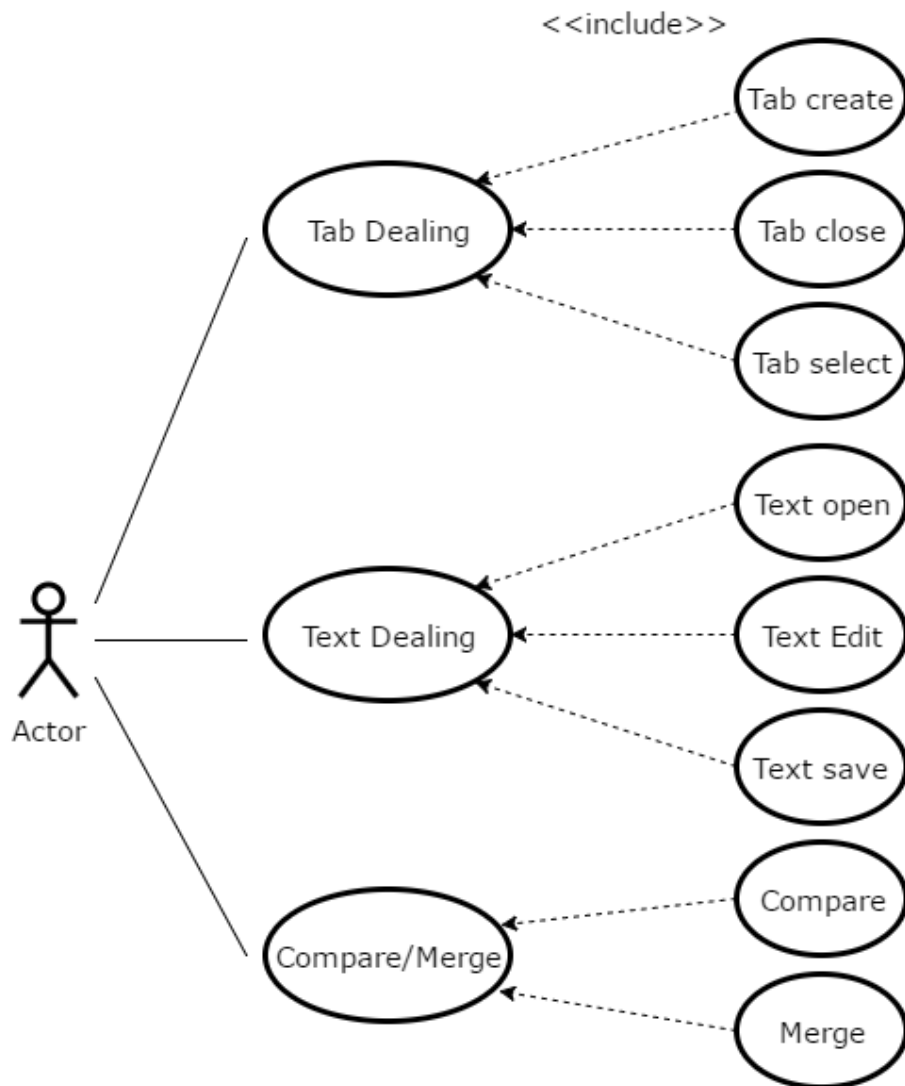
Figure 8 "Window" 탭 구성요소

Requirement of Controller

* 블록 인덱스 :일치 혹은 불일치된 줄의 묶음 단위를 순서대로 정렬했을 때, 처음부터 나뉘어진 각 묶음의 순서	
File	
New Tab	새 탭 생성, 모델에 새 탭을 생성하게 하고 탭의 최대 숫자를 증가시킨다. 해당 탭의 단축키 활성화 상태를 초기화한다.
Open	왼쪽/오른쪽 파일을 열 수 있는 창을 열고, 파일이 성공적으로 열렸으면 새 탭을 생성하고 모델에 파일을 받아온다.
Save As	각 파일을 다른 이름으로 저장하는 창을 열고 각각 입력받은 이름과 형식으로 파일을 출력한다.
Save Left File	프로그램에서 변경한 왼쪽 파일의 내용을 원래 파일로 출력한다.
Save Right File	프로그램에서 변경한 오른쪽 파일의 내용을 원래 파일로 출력한다.
Close	모델을 모두 닫고 프로그램을 종료시킨다.
Window	
Close Tab	열려있는 탭의 숫자를 줄이고 파일을 반환한 후 현재의 탭을 닫는다.
Close All Tab	모든 탭에서 열린 파일을 종료하고 모든 탭을 닫는다.
SplitPane Function	
Edit	사용자가 불러온 파일의 텍스트 영역에서 직접 수정한 파일의 내용을 모델에 적용시킨다.

Merge	
Next Difference	현재 선택된 블록의 인덱스를 +2 해준다. 단, 다음 인덱스가 파일의 끝을 넘길 경우 더이상 이동하지 않고 비활성화된다.
Prev Difference	현재 선택된 블록의 인덱스를 -2 해준다. 단, 다음 인덱스가 파일의 끝을 넘길 경우 더이상 이동하지 않고 비활성화된다.
First Difference	블록의 인덱스를 각 케이스에 따라 0 혹은 1 로 변경해준다. prev Difference 를 비활성화한다.
Now Selected Difference	클릭된 인덱스에서 가장 가까운 오류 블록 인덱스로 인덱스를 이동시킨다. 보통 바로 다음 인덱스로 이동하지만 다음 인덱스가 없는 경우 이전으로 이동한다.
Last Difference	블록의 인덱스를 가장 끝 오류지점으로 변경해준다. next Difference 를 비활성화.
Copy To Right	현재 선택된 차이점 블록에서, 왼쪽의 내용을 오른쪽으로 복사하고 화면에 출력한 후 다시 Compare 한다.
Copy To Left	현재 선택된 차이점 블록에서, 오른쪽의 내용을 왼쪽으로 복사하고 화면에 출력한 후 다시 Compare 한다.
Copy To Right All	오른쪽의 내용 자체를 왼쪽의 내용으로 옮기고 Compare, 화면에 출력한다.
Copy To Left All	왼쪽의 내용 자체를 왼쪽의 내용으로 옮기고 Compare, 화면에 출력한다.
Compare	<p>현재 탭에서, 불러온 양 파일의 내용을 비교하고, 서로 맞은 부분과 다른 부분을 블록별로 나누어 ArrayList 에 저장하도록 한다.</p> <p>선택된 블록의 인덱스를 맨 처음으로 바꾸고, 케이스에 따라 다른 ListLayout 을 출력하게 한다.</p> <p>파일의 차이점이 없을 경우, ~Difference와 Copy To~ 버튼들을 비활성화, 차이점이 있을경우 모두 활성화한다.</p>

Usecase

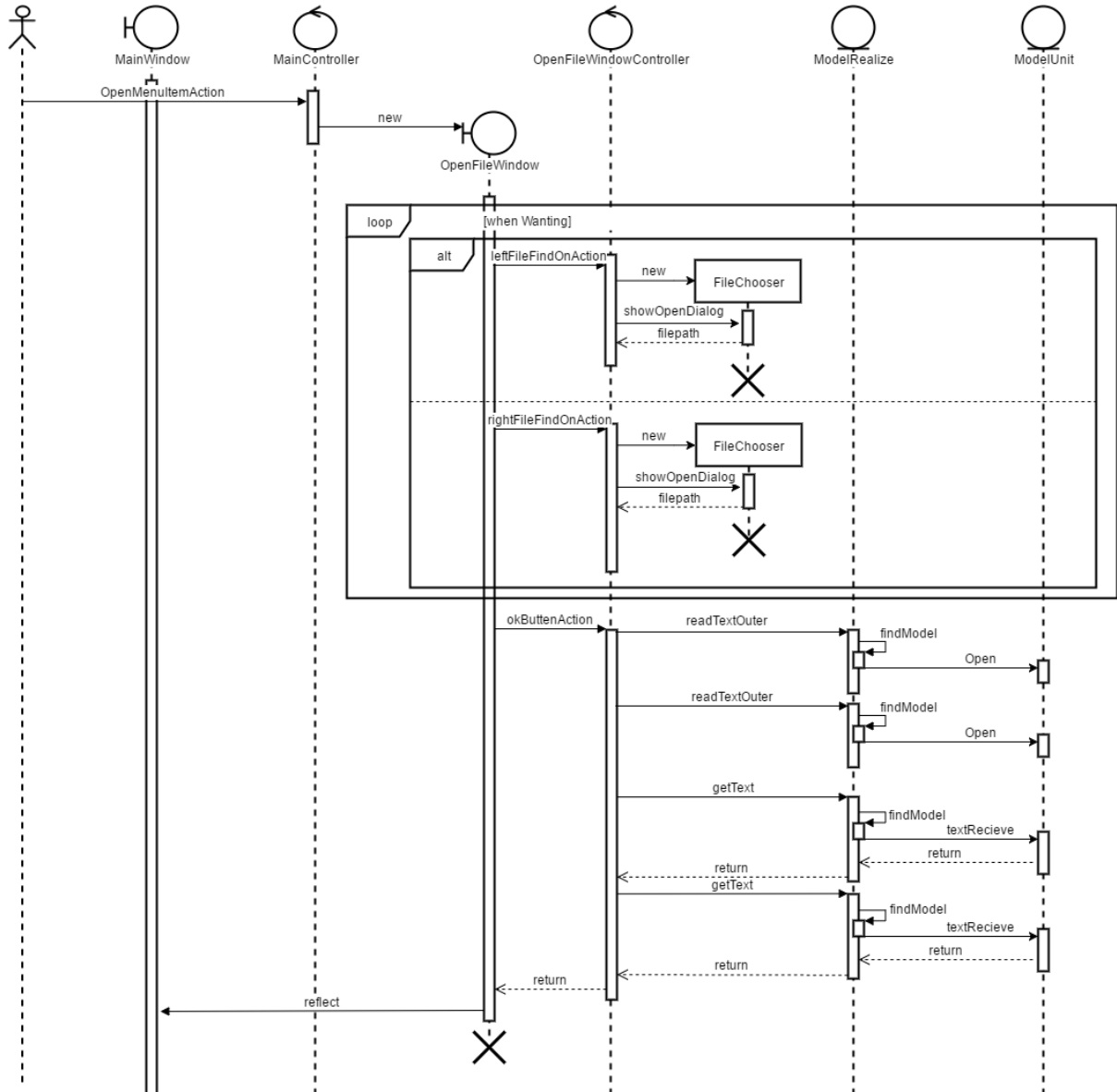


User Scenario

- Git이나 SVN 같은 기능들을 하는 Tool을 이용하지 않고 코드를 작성하였다. 그러나 지난 버전에 작성했던 부분하고 현재 수정한 부분의 차이점을 알고 싶었다. (Compare)
- 문서간의 차이점이 보다가 순간 처음 차이점으로 가고 싶었다 (First Diff)
- 이전 차이점 다음 차이점을 가고 싶었다. (Next Pre Diff)
- 마지막 차이점을 알고 싶었다. (Last Diff)
- 문서간의 차이점을 발견하고 특정 부분만 동기화 시키고 싶었다. (Copy ~)
- 문서간의 차이점을 발견하고 그 차이점이 너무 많아 그냥 정상 작동한 버전으로 덮어 씌우고 싶었다. (Copy to All ~)

Sequence Diagram

Case1) File Open

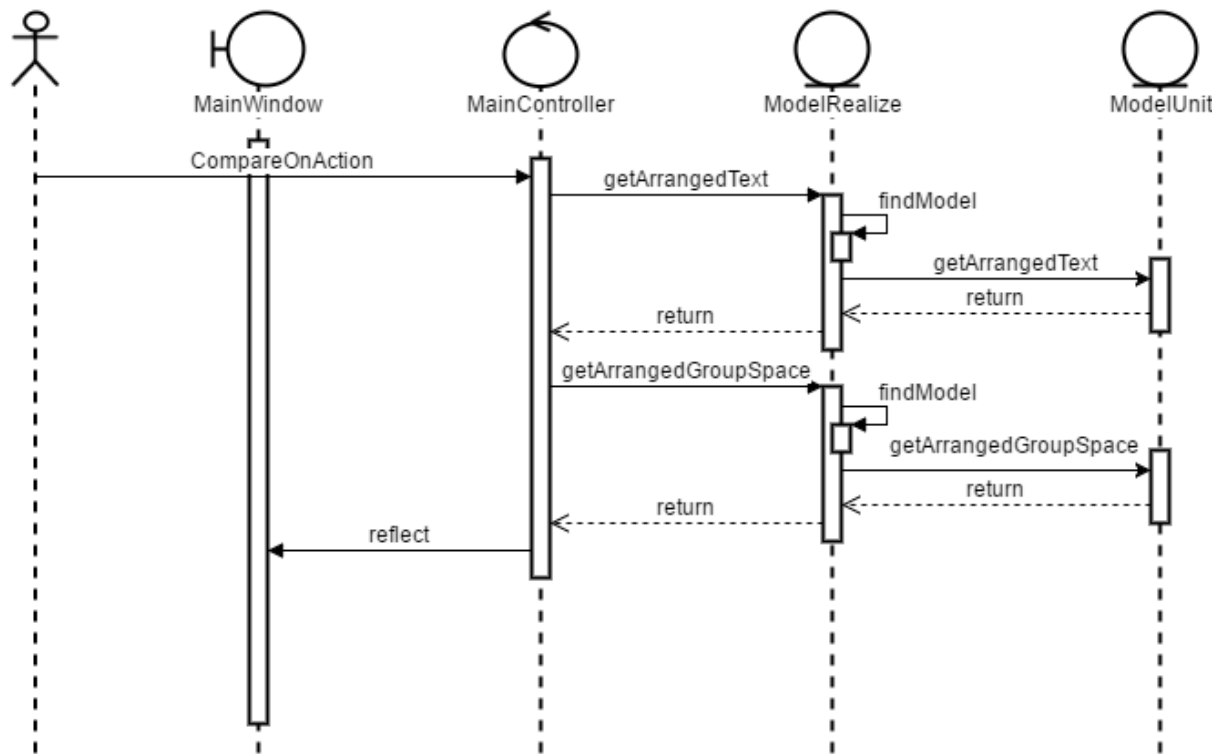


사용자가 두 파일을 불러오기를 원한다.

특정 버튼을 조작하면 컨트롤 오브젝트의 OpenMenuItemOnAction 메소드가 실행되어 OpenfileWindow를 띄운다.

OpenfileWindow는 FileChoser 클래스를 이용하여 파일 주소를 가져와 왼쪽 또는 오른쪽의 텍스트를 열도록 모델에게 요청하거나, 자기 자신을 닫으면서 모델에게 모델이 가진 텍스트를 요청하고 뷰의 내용을 갱신시킨다. 결과적으로 두 파일은 불러와져 사용자에게 보여지게 된다.

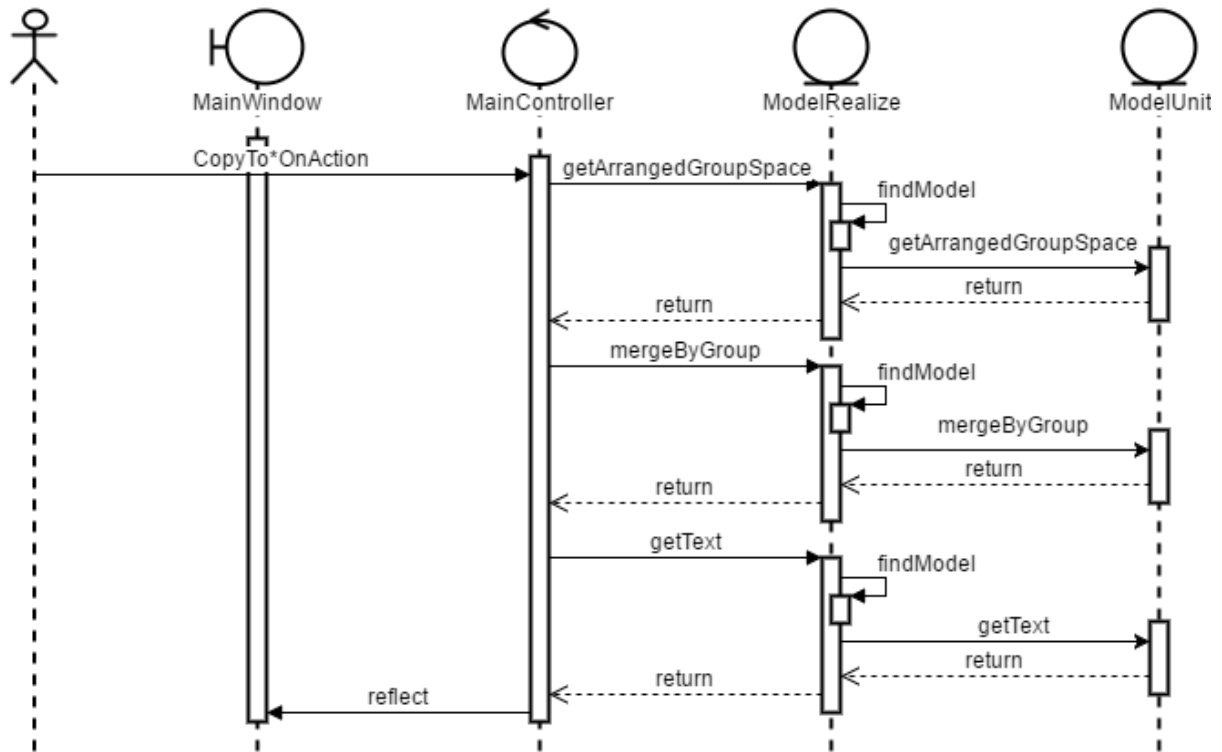
Case2) Compare



사용자가 텍스트 파일을 비교하기를 원한다.

특정 버튼을 조작하면 컨트롤 오브젝트의 CompareOnAction 메소드가 실행된다. 컨트롤러는 모델에 재배열된 텍스트와, 블록 단위의 각 줄 수를 요구하며, 모델은 그러한 값을 연산하여 넘겨준다. 그 자료로 뷰의 데이터는 갱신된다. 결과적으로 탭의 두 텍스트는 재배열되고 컬러링되어 보여지게 된다

Case3) Merge



사용자가 텍스트의 특정 불일치 블록 단위를 병합하기를 원한다. 특정 버튼을 조작하면 컨트롤러 오브젝트의 CopyToLeftOnAction 메소드, 또는 CopyToRightOnAction 메소드가 실행된다. 컨트롤러는 그룹 줄 수를 요구한 뒤, 특정 블록 단위를 병합해 달라고 모델에게 요구한다. 모델이 병합을 처리한 뒤, 컨트롤러는 다시 텍스트를 받아온다. 그 자료로 뷰의 데이터는 갱신된다. 결과적으로 탭의 두 재배열되고 컬러링된 텍스트는 선택된 부분이 병합되어 보여지게 된다.