**Simple Merge Specification**

**Version 2.0**

**June 9th, 2017**

**Project Team:**

김동섭, 김성민, 유진수, 이상진, 정채진, 최철곤

**Document Author(s):**

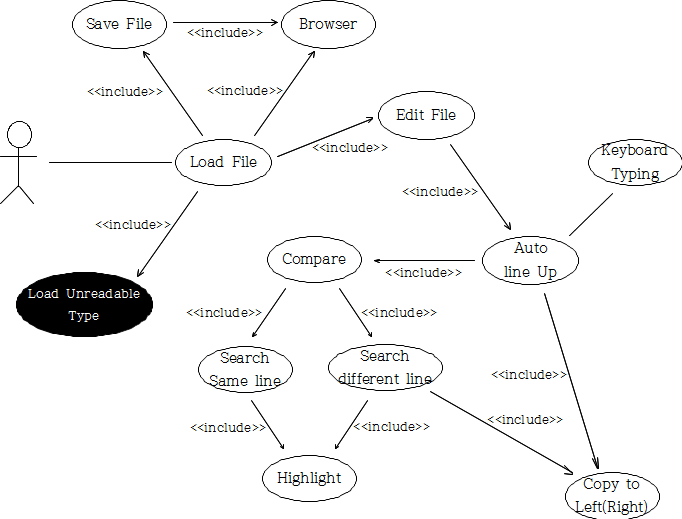
김동섭, 김성민, 유진수, 이상진, 정채진, 최철곤

**Customer Representative(s):**

이찬근 교수님

**I. Introduction**

이 프로젝트의 목표는 Simple Merge 프로그램을 만드는 것이다. 이 프로그램을 이용하여 서로의 코드를 비교하고, 그 비슷한 코드를 구분하여 두 코드 중 좋은 코드를 병합시킬 수 있으며, 저장, 불러오기가 가능한 프로그램이다.



**II. Use Cases**

UC1 Load File

UC2 Edit File

UC3 Save File

UC4 Auto line up

UC5 Keyboard Typing

UC6 Search Different line

UC7 Copy to left

UC8 Copy to right

UC9 Browser

UC10 Highlight

UC11 Load Unreadable File Type

**UC1 Flow of Events for the *Load File* Use Case**

**1.1 Preconditions:**

2개의 Text 파일이 있어야 함.

**1.2 Main Flow:**

프로그램 실행 시 상단 메뉴바에 File 메뉴를 클릭, 하위 메뉴에 ‘왼쪽 파일 불러오기’ 와 ‘오른쪽 파일 불러오기’를 클릭 시 browse창이 생성됨. 텍스트 파일을 선택하면 인터페이스 창에 텍스트파일 출력.

**1.3 Subflows:**

None

**1.4 Alternative Flows:**

[E1] 선택한 파일이 Text 파일이 아닌 경우 Load Unreadable File Type[UC13] 수행.

**UC2 Flow of Events for the *Edit File* Use Case**

**2.1 Preconditions:**

2개의 Text 파일이 Load된 상태

**2.2 Main Flow:**

Load된 파일을 Auto line up[UC4], Copy to left[UC9], Copy to right[UC10] 기능을 수행할 수 있는 상태로 바꿀 수 있는 버튼을 표시. 또한 편집사용 아이콘 혹은 버튼을 사용하여 불러온 텍스트파일의 편집이 가능하게 함. 또한 편집이 끝난 후, 편집사용을 종료하여 다시 편집을 못하도록 만듦.

**2.3 Subflows:**

None.

**2.4 Alternative Flows:**

None.

**UC3 Flow of Events for the *Save File* Use Case**

**3.1 Preconditions:**

Load한 Text File들에 변경 사항이 있어야 함.

**3.2 Main Flow:**

두 개의 파일의 변경 사항을 서로 다른 이름으로 저장함..

**3.3 Subflows:**

None

**3.4 Alternative Flows:**

None.

**UC4 Flow of Events for the *Auto line up* Use Case**

**4.1 Preconditions:**

줄바꿈, 들여쓰기, 중괄호의 위치 등의 형태가 일관성이 없는 코드가 존재함.

Search Different Line[UC6]이 실행되어야 함.

**4.2 Main Flow:**

편집 패널 상단에 “Edit”이라는 레이블이 붙은 단추를 누르고 Auto Line Up버튼을 누른다.

AutoLineUp을 누르게 되면 [UC6]을 실행한 상태에서 서로 같은 부분을 찾아 같은 열로 줄을 맞춰주게 된다.

**4.3 Subflows:**

None

**4.4 Alternative Flows:**

None.

**UC5 Flow of Events for the *keyboard typing* Use Case**

**5.1 Preconditions:**

Edit에서 편집사용이 실행되어야 한다.

**5.2 Main Flow:**

GUI를 이용하여 keyboard typing이 실행되도록 만든다. 키보드 입력으로 양 쪽 패널에서 텍스트를 입력할 수 있다.

**5.3 Subflows:**

None

**5.4 Alternative Flows:**

None.

**UC6 Flow of Events for the *Search Different line* Use Case**

**6.1 Preconditions:**

두 개의 코드를 display에 load한 후, AutoLineUp이 수행되어야 한다

**6.2 Main Flow:**

정렬된 두 개의 코드를 처음부터 끝까지 읽으면서 tokenizing을 한다. Tokenizing된 데이터의 각각의 line을 비교한다. line을 비교하면서 서로 다른 line을 찾아낸다.(LCS를 이용)

**6.3 Subflows**

None

**7.4 Alternative Flows:**

비교한 line이 동일하다면 건너뛴다.

**UC7 Flow of Events for the Copy to leftUse Case**

**7.1 Preconditions:**

두 개의 코드를 비교한 후 차이점이 있는 라인을 하이라이트처리 한다.

**7.2 Main Flow:**

두 개의 코드의 차이점이 있는 라인이 블록처리 되었을 시 Copy to left버튼을 누른다. 오른쪽 코드에 블록처리 되어있는 라인이 왼쪽 코드의 블록처리 되어있는 라인으로 복사된다. Copy의 연속실행은 해당 블록을 동일하게 만든다. 만일 옮기기 전, 왼쪽과 오른쪽 패널의 줄 수가 맞지 않는다면 적은 쪽의 줄 수를 늘려서 실행한다.

**7.3 Subflows:**

None

**7.4 Alternative Flows:**

하이라이트 처리가 없다면 아무일도 일어나지 않는다.

**UC8 Flow of Events for the Copy to rightUse Case**

**8.1 Preconditions:**

두 개의 코드를 비교한 후 차이점이 있는 라인을 하이라이트 처리 한다.

AutoLineUp이 실행되어야 한다.

**8.2 Main Flow:**

두 개의 코드의 차이점이 있는 라인이 블록처리 되었을 시 Copy to right버튼을 누른다. 왼쪽 코드에 블록처리 되어있는 라인이 오른쪽 코드의 블록처리 되어있는 라인으로 복사된다. Copy의 연속실행은 해당 블록을 동일하게 만든다. 만일 옮기기 전, 왼쪽과 오른쪽 패널의 줄 수가 맞지 않는다면 적은 쪽의 줄 수를 늘려서 실행한다.

**8.3 Subflows:**

None

**8.4 Alternative Flows:**

하이라이트 처리가 없다면 아무일도 일어나지 않는다.

**UC9 Flow of Events for the *Brower* Use Case**

**9.1 Preconditions:**

왼쪽 패널과 오른쪽 패널을 따로따로 받으면서 Load File 및 Save File 실행 요청이 들어옴.

**9.2 Main Flow:**

파일의 디렉토리 경로를 찾아 파일을 선택함.

**9.3 Subflows:**

None.

**9.4 Alternative Flows:**

None.

**UC10 Flow of Events for the *Highlight* Use Case**

**10.1 Preconditions:**

두 파일의 텍스트 간 다른 부분이 있음.

AutoLineUp이 실행되어야 함

**10.2 Main Flow:**

텍스트의 다른 부분을 음영 처리 혹은 Coloring으로 강조한다.

**10.3 Subflows:**

두 텍스트가 다른 내용일 경우 빨간색 하이라이트로 표현. 다른 공백이 존재할 경우 에도 빨간색으로 표현한다.

**10.4 Alternative Flows:**

None.

**III. Misuse Cases**

**UC11 Flow of Events for the *Load unreadable File Type* Use Case**

**13.1 Preconditions:**

User가 읽을 수 없는 파일을 Load File[UC1]브라우저에서 선택하였을 경우.

**11.2 Main Flow:**

[E1] 선택한 파일이 Text 파일이 아닌 경우, Load Fail Error Massage를 출력함.

**11.3 Sub-flows:**

[E2] Load 기능을 수행하지 않음.

**11.4 Alternative Flows:**

None.

**IV. Nonfunctional Requirements**

**NR1.Product Requirement**

The Merge System shall be available to all user during all the time.

**NR1.Time constaints**

2017년 6월 10일까지 모든 구현을 완료한다.

**NR2. User Interface**

메뉴 바에 File, Edit, Help를 만듦. Help에는 License로 우리 조의 학번과 이름을 기입하고, File에는 Save, Load 밑 종료 버튼이 있어야한다. 또한 Edit에는 편집사용 편집 불가 및 AutoLineUp 버튼을 만들며, 이 모든 버튼은 메뉴 바 하단 아이콘으로 만들어 간편하게 한다.

**V. Constraints**

All code development shall be done with the Java programming language.

All testing shall be done using JUnit and FIT.

**VI. Requirements Dependency Traceability Table**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | UC1 | UC2 | UC3 | UC4 | UC5 | UC6 | UC7 | UC8 | UC9 | UC10 | UC11 |
| UC1 |  | O |  |  |  |  |  |  |  |  |  |
| UC2 | O |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
| UC3 | O |  |  |  |  |  |  |  |  |  |  |
| UC4 |  | O |  |  | O |  |  |  |  |  |  |
| UC5 |  | O |  |  |  |  |  |  |  |  |  |
| UC6 |  |  |  | O |  |  |  |  |  |  |  |
| UC7 |  |  |  |  |  |  |  |  |  |  |  |
| UC8 |  |  |  |  |  |  |  |  |  |  |  |
| UC9 |  |  |  | O |  | O |  |  |  |  |  |
| UC10 |  |  |  | O |  | O |  |  |  |  |  |
| UC11 | O |  | O |  |  |  |  |  |  |  |  |
| UC12 |  |  |  |  |  | O |  |  |  |  |  |
| UC13 | O |  |  |  |  |  |  |  |  |  |  |

**VII. Development and Target Platforms**

1. Windows XP Operating System

2. Intel Pentium IV processors

3. Eclipse IDE

**VIII. Project Glossary**

**Auto line up:** 코드 파일의 각 line을 정렬하는 단계. 정렬되어야 Compare 단계를 수행할 수 있다.

**IX. Document Revision History**

Version 2.0

Name(s) Kim Dong Seop’s

Date June, 9th, 2017

Change Description Original creation of the SRS.