

Simple Merge Program (SRS)



과목	소프트웨어 공학
학과	컴퓨터공학부
팀	09
팀원	백인혁 20141266 백찬희 20142503 이동규 20140490 함지훈 20146512

목차

1. Introduction

: Usecase Diagram

2. Use Cases

: UC 1 ~ 9

3. Nonfunctional Requirements

: Usability, Reliability, Performance, Constraints

4. Requirements Dependency Traceability Table

5. Development and Target Platforms

6. Project Glossary

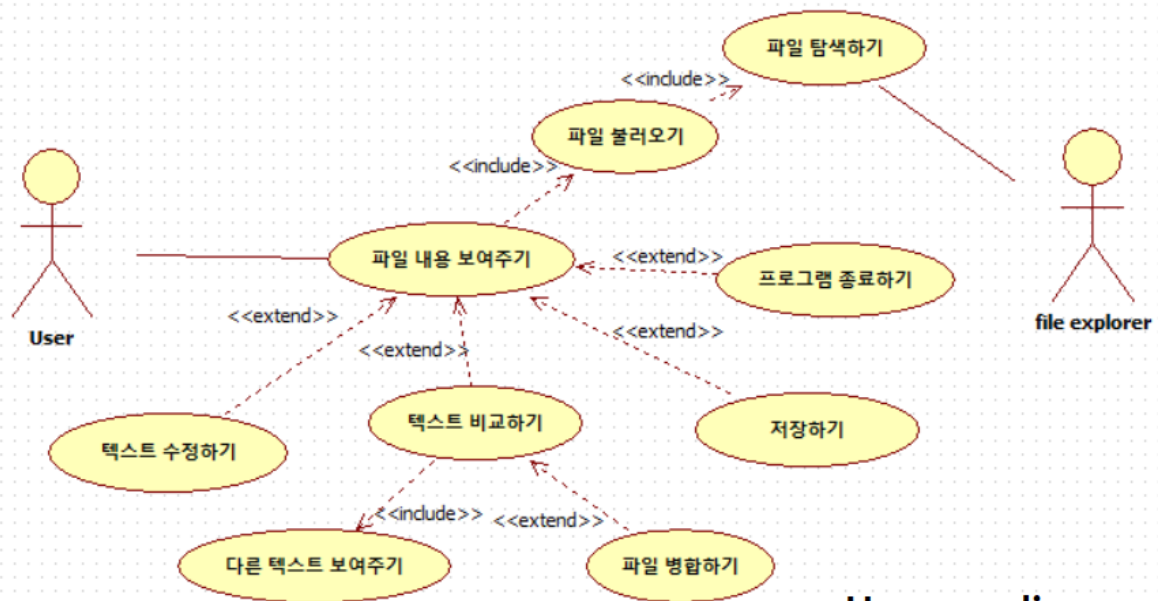
: View, Load, Compare, Merge, Save, Edit

7. Document Revision History

1. Introduction

Development of Simple Merge 프로젝트의 목표는 두 개의 다른 텍스트 파일을 비교하여 다른 부분을 찾아서 병합(수정) 할 수 있게 하는 것이다. 이를 위해서 프로젝트에서 텍스트 파일과 관련된 기능들을 제공할 것이다. 그 기능에는 '불러오기', '저장하기', '수정하기', '비교하기', '병합하기' 등이 있다. 사용자가 비교하고자 하는 파일을 불러오면 두 개의 텍스트를 비교하여 서로 다른 부분을 글자색, 배경색, 글꼴(강조) 등으로 표시해줄 것이다. 사용자는 판단하여 둘 중 하나의 텍스트 파일로 덮어쓸 수 있다. 이 프로젝트는 JAVA 프로그램 언어를 사용하여 개발될 것이며 자세한 기능 및 시나리오는 다음 목차에서 설명할 예정이다.

UC1	파일 내용 보여주기
UC2	파일 탐색하기
UC3	파일 불러오기
UC4	텍스트 수정하기
UC5	텍스트 비교하기
UC6	다른 텍스트 보여주기
UC7	파일 병합하기
UC8	저장하기
UC9	프로그램 종료하기



Usecase diagram

2. Use Cases

UC1 파일 내용 보여주기

가. Preconditions:

사용자가 LOAD 버튼을 눌러서 파일 불러오기를 완료한 상태이다.

나. Main Flow:

프로그램이 실행되고 사용자가 LOAD 버튼을 눌러 원하는 파일을 불러온다. 불러온 파일은 View 에 등록되어 내용이 보여진다 [UC2][UC3][S1].

다. Subflows:

[S1] LOAD 버튼이 왼쪽, 오른쪽 View 에 하나씩 있다. 각 View 에 해당하는 LOAD 버튼을 누르면 해당 View 에 텍스트 파일이 불러지며 내용이 보여진다.

라. Alternative Flows:

None.

UC2 파일 탐색하기

가. Preconditions:

사용자가 LOAD 버튼을 누른다.

나. Main Flow:

사용자가 LOAD 버튼을 눌러서 파일 탐색창을 띄운다. 이후 비교를 희망하는 파일을 선택한다.

다. Subflows:

None.

라. Alternative Flows:

None.

UC3 파일 불러오기

가. Preconditions:

- 1) 사용자가 LOAD 버튼을 누른다.
- 2) 탐색하기를 통해서 텍스트 파일 탐색을 완료한 상태이다.

나. Main Flow:

탐색하여 선택한 txt 파일을 View 에 불러온다 [S1][E1].

다. Subflows:

[S1] 사용자가 왼쪽, 오른쪽 View 중 LOAD 버튼을 누른 View 에 불러온다.

라. Alternative Flows:

[E1] 먼저 불러온 파일이 있다면 저장하지 않고 새로 불러온 파일로 덮어씌운다.

UC4 텍스트 수정하기

가. Preconditions:

- 1) 파일을 불러와서 View 에 내용이 있어야한다.
- 2) 사용자가 EDIT 버튼을 눌러야 한다.

나. Main Flow:

사용자는 EDIT 버튼을 통해서 패널에 불러온 내용을 수정한다 [E1].

다. Subflows:

None.

라. Alternative Flows:

[E1] 패널에 불러온 내용이 없을 경우에는 새로운 문서를 작성할 수 있다.

UC5 텍스트 비교하기

가. Preconditions:

- 1) 파일을 불러와서 View 에 등록되어 있어야한다.
- 2) 사용자가 COMPARE 버튼을 눌러야 한다.

나. Main Flow:

사용자가 비교를 희망하는 텍스트 파일을 등록 후 COMPARE 버튼을 누른다. 다른 부분을 찾아서 저장한다 [S1].

다. Subflows:

[S1] 라인별로 비교를 하여 다른 부분을 비교한다.

라. Alternative Flows:

None..

UC6 다른 텍스트 보여주기

가. Preconditions:

- 1) 사용자가 두 개의 텍스트 파일을 불러와서 등록 되어있어야 한다.
- 2) 사용자가 비교하기를 완료한 상태이다.

나. Main Flow:

사용자가 COMPARE 버튼을 눌러 비교하기를 요청하고 다른 부분이 있으면 표시해준다 [UC5][S1][S2].

다. Subflows:

[S1] 양쪽 View 에 다른 부분을 표시해준다.

[S2] 텍스트의 다른 부분을 배경색을 변경하고 밑줄을 그어서 보여준다.

라. Alternative Flows:

None.

UC7 파일 병합하기

가. Preconditions:

- 1) 사용자가 두 개의 텍스트 파일을 불러와서 등록 되어있어야 한다.
- 2) 사용자로부터 병합하려는 View 의 MERGE 버튼을 누른다.

나. Main Flow:

사용자가 병합하고자 하는 View 의 MERGE 버튼을 다른 View 의 내용을 기준으로 차이점을 병합한다.[S1].

다. Subflows:

[S1] 왼쪽 Vie 와 오른쪽 View 중 어디에 덮어쓸건지 버튼으로 입력받는다.

라. Alternative Flows:

None.

UC8 저장하기

가. Preconditions:

None.

나. Main Flow:

사용자가 SAVE 버튼을 누름으로써 파일 탐색기를 실행시켜 저장하고자 하는 경로를 선택하여 현 상태의 텍스트 파일을 저장한다[S1][S2].

다. Subflows:

[S1] 경로에 같은 이름의 파일이 존재하면 기존 파일에 덮어쓸 것인지 확인한다.

[S2] 새로운 이름으로 새 파일을 생성할 수 있다.

라. Alternative Flows:

None.

UC9 프로그램 종료하기

가. Preconditions:

None.

나. Main Flow:

사용자가 EXIT 버튼을 누름으로써 프로그램을 종료한다.

다. Subflows:

None.

라. Alternative Flows:

None.

3. Nonfunctional Requirements

Usability

프로그램을 사용자가 이해하기 쉽게 하고 텍스트의 가독성을 높인다.

NR1.1. 유저 인터페이스

따로 설명서를 참고하지 않아도 쉽게 프로그램의 기능을 이해할 수 있도록 쉬운 인터페이스를 만든다.

NR1.2. 유저 에러

유저가 원하지 않는 기능을 잘못 눌렀을 경우에 대비해 기능을 수행하기 전 해당 기능을 수행할지 확인하는 메시지를 띄운다.

Reliability

사용자가 주어진 기능이 아닌 잘못된 기능을 실행해서 시스템이 실행을 실패한 경우 경고 문고를 띄워서 실패했다고 알려줌으로써 시스템에 신뢰도를 높여야 한다.

NR2.1 텍스트 파일이 아닌 잘못된 파일을 불러오는 경우

텍스트 파일이 아닌 경우 비교 또는 수정, 병합이 불가능하기 때문에 텍스트 파일만 탐색기로 선택가능하게 구현한다..

Performance

또한 동시에 수정과 저장을 하는 등 다중 작업을 시행해서는 안되기 때문에 한 번에 하나의 업무만 처리해야한다. 각 작업을 수행하는 데 있어서 긴 시간을 보내게 된다면 다중 작업을 지원하지 않기 때문에 사용자가 오래 기다려야한다. 따라서 가능한 모든 작업을 1 초 내외로 처리하도록 한다.

NR3.1 File I/O

File I/O 에는 텍스트 파일의 크기에 따라 시간의 차이가 많이 생긴다. Time Complexity 가 $O(n^2)$ 을 넘지 않도록 하여 Runtime 이 급격하기 증가하지 않도록 제한한다.

NR3.2 텍스트 비교

텍스트를 비교할 때도 마찬가지로 텍스트 파일만큼의 시간이 걸린다. 불러왔다면 전체를 비교하는 데 1 초를 넘기지 않도록 구현한다.

Constraints

코드는 JAVA Programming Language 로, 구현환경은 Eclipse 에서 구현한다. 코드 테스트는 JUNIT 환경에서 실시한다.

4. Requirements Dependency Traceability Table

	UC1	UC2	UC3	UC4	UC5	UC6	UC7	UC8	UC9	NR1.1	NR1.2	NR2.1	NR3.1	NR3.2
UC1		X	X											
UC2														
UC3		X												
UC4	X													
UC5	X													
UC6	X					X								
UC7	X				X									
UC8	X	X	X											
UC9														
NR1.1														
NR1.2														
NR2.1	X		X											
NR3.1			X											
NR3.2						X								

5. Development and Target Platforms

가. Windows(7, 8, 8.1, 10) Operating System

나. Intel & AMD processors

다. JAVA Eclipse IDE

라. GitHub

6. Project Glossary

가. View : 텍스트를 보여주는 패널/창을 의미한다.

나. LOAD : 텍스트 파일을 불러온다.

다. COMPARE : 불러온 2 개의 텍스트 파일을 서로 비교한다.

라. MERGE : 비교한 결과를 가지고 서로 다른 부분들을 어느 한쪽을 기준으로 서로 파일의 내용을 동일하게 만드는 작업을 의미한다.

* 결과적으로 Merge 를 실행하면 왼쪽과 오른쪽 파일내용이 같아져야 한다.

마. SAVE : 작업한 파일의 내용을 저장한다.

바. EDIT : 텍스트 파일의 내용을 수정할 수 있게 된다.

7. Document Revision History

버전	3.0
이름	소프트웨어 공학 9 팀
날짜	2018 년 6 월 7 일
Change Description	Second modify of the SRS