

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Team Leader 김정민 |  | Team Member  김상헌, 양종만, 이동현, 임기찬, 황선준 |  | Writer  김상헌, 이동현 |

WinMerge – Team7

Software Requirements Specification

COTENTS

1. [Introduction to the System 2](#_Toc349734314)

[Definition](#_Toc349734316)

[Target Platforms](#_Toc349734317)

[Development Platforms](#_Toc349734318)

1. [Usecase Event Flow 2](#_Toc349734319)

[Usecase](#_Toc349734320) Diagram

[Usecase Descriptions](#_Toc349734324)

1. System Sequence Diagrams [2](#_Toc349734331)

[시장 분석](#_Toc349734332)

[시장 세분화](#_Toc349734333)

[경쟁업체](#_Toc349734334)

[가격 책정](#_Toc349734335)

1. [Non-Functional requirements 2](#_Toc349734336)

[Quality Requirements](#_Toc349734337)

[Constraints](#_Toc349734338)

1. [Appendix 2](#_Toc349734336)

[Project Glossary](#_Toc349734337)

[Requirement Dependecy](#_Toc349734338) Traceability

[Document Revision History](#_Toc349734338)

# Introduction to the System

## Definition

User는 프로그램을 통해 기존에 있는 텍스트 파일을 불러오거나, 수정하고, 파일을 저장할 수 있다. 만약 2개의 파일에 내용이 다른 줄이 존재하면, 차이를 나타내고 한쪽으로 병합할 수 있는 프로그램이다.

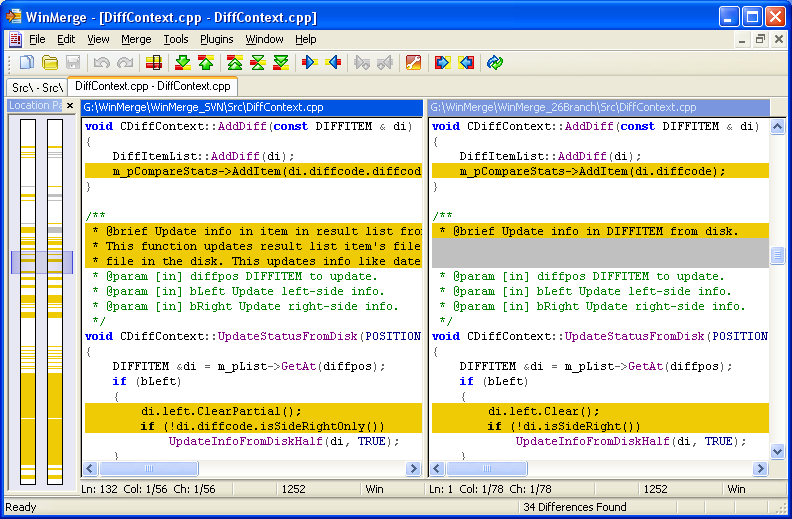


Figure 비슷한 프로그램의 사용 예시(WinMerge)

## Target Platforms

Windows 10 기반 운영체제

## Development Platforms

|  |  |
| --- | --- |
| JDK | 8.0.1 |
| Graddle | 4.7 |
| Eclipse | Oxygen 4.7.3a |

# UseCase Event Flow

## UseCase Diagram

추후 추가 예정(5.14 이전)

## UseCase Description

1. UC1 : LOAD
   1. Preconditions : 없음
   2. Main Flow ~~

# System Sequence Diagrams

추후 추가 예정(5.14이후)

# Non-Functional Requirements

## Quality Requirements

1. 유닛 테스트를 위해 Junit, EasyMock 와 같은 Testing Framework를 이용해야 한다.
2. MVC architecture pattern을 사용하여 프로그램을 설계해야 한다.
3. 유닛 테스트 시 정상적인 상황에서 올바른 행동과 비정상적 상황에서의 반응을 모두 포함해야 한다.

## Constraints

1. Compare버튼을 이용하여 파일을 비교할 때 LCS 알고리즘을 이용하여 비교하며 비교된 결과를 전면혹은 배경색을 통해 나타내야 하며 파일은 줄 단위로 비교가 이루어 진다.
2. 두 파일을 병합할 경우 Copy to right혹은 Copy to left버튼을 사용하며 Compare버튼을 사용하여 나타난 차이점을 블록으로 지정하여 블록을 선택한 방향으로 복사한다.
3. 파일을 수정하기 위해서는 Edit 버튼을 누르고 난 후 수정을 가능케 한다.

# Appendix

## Project Glossary

|  |  |
| --- | --- |
| 파일 | 일반적인 문서파일들을 의미한다. |
| Testing Framework | 유닛 테스트를 위해 쓰이는 틀을 지칭한다. |
| 유닛 테스트 | 전체프로그램이 아닌 프로그램의 작은 단위들의 정상 동작 여부를 확인하는 과정을 의미한다. |
| MVC | Model/View/Controller를 의미하며 모델은 프로그램의 정보를, 뷰는 화면과 같은 사용자 인터페이스를, 컨트롤러는 뷰와 모델 간의 상호동작을 관리한다. 영역을 구분함으로써 다른 영역의 영향없이 한 영역을 쉽게 고칠 수 있게 해주는 개발 방법이다. |
| LCS 알고리즘 | Longest Common Subsequence의 약어로 두 개의 문서파일을 비교할 때 자주 쓰이는 방식이다. |

## Requirements Dependency Traceability

## Document Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| 작성날짜 | 상세 내용 | 작성자 | 버전 |
| 2018/05/06 | 최초 작성 | 김상헌 | V0.1 |
| 2018/05/07 | 문서 형식 수정 | 김상헌 | V0.15 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |