|  |  |
| --- | --- |
| Team No | Team2 |
| 과목 | Software Modeling and Analysis |
| 담당교수 | JUNBEOM YOO Associate Professor / Ph.D |
| 팀 구성원 | 201014184 김도윤 |
| 201111367 여승훈 |
| 201111347 김태호 |
| 제출일자 | 2016-03-16 |

[SMA2016]T2 - OOPT stage 1000 ver 1.0

**Software Modeling & Analysis**

**Clone Checker with User**

OOPT stage 1000

Plan & Elaboration

1

[SMA2016]T2 - OOPT stage 1000 ver 1.0

**Index**

Activity1001.**Define Draft Plan**

Activity1002.**Create Preliminary Investigation Report**

Activity1003.**Define Requirements**

Activity1004.**Record Terms in Glossary**

Activity1005.**Implement Prototype**

Activity1006.**Define Draft System Architecture**

Activity1007.**Define Business Use Case**

Activity1008.**Define Business Concept Model**

Activity1009.**Define System Test Case**

Activity1010.**Refine Plan**

2

[SMA2016]T2 - OOPT stage 1000 ver 1.0

**Activity 1001. Define Draft Plan**

**1. Motivation**

소프트웨어의 개발이 활발해짐에 따라 소스코드의 복제, 도용이 증가하고 있다. 따라서 소

스코드 복제 여부에 대한 검사 또한 매우 중요해지고 있다. 소스코드가 복제되면 기업, 개

인 간의 분쟁이 일어날 수 있다. 따라서 사전에 소프트웨어 유사도 검사가 필요하다. 이것

은 학교에서 과제로 진행되는 프로젝트에도 해당되며, 배움에 있어 학생들의 소스코드 복

제 및 도용은 옳지 않다. 따라서 Clone Checker 프로그램을 개발하게 되었다.

**2. Project Objectives**

다수의 C 프로그램을 대장으로 상호 cheating 여부를 정량적으로 판단하고, 해당 내용을 User

가 검토하여 결과를 수정할 수 있는 Clone Checker

- 코드 유사성의 단순 수치화를 통해 코드의 유사성을 직관적으로 판단할 수 있다.

- 각 코드들간의 유사성을 등급으로 표시하여 알려준다.

- 복제 및 도용이 의심되는 코드를 User가 다시 한번 확인하여 판단할 수 있다.

**3. Functional requirements**

- Load Code

- Delete Code

- Display Code List

- Detect Duplication Code

- Clone Check

- Display Code Set List

- Show All

- Show Red

- Show Orange

- Show Yellow

- Show Green

- Select Code Set

3

[SMA2016]T2 - OOPT stage 1000 ver 1.0

- Display Main Code

- Display Sub Code

- Display Similarity

- User Accept

- User Reject

- Save

**4. Non-functional requirements**

- 검사시간을 3min/50file 이내로 유지한다.

**5. Resource Estimation**

5.1 Human efforts (M/M)

3M/3M

5.2 Human resources

3명 – 프로그래머 3명

5.3 Duration

3개월

5.4 Budget

T.F Wage : 30 \* 3man \* 3month = 270

**6. Other Information**

- Future Version

향상된 Performance, 한번이라도 Clone Checker 프로그램을 통해 객체화된 Code는 정

보가 살아있어 다음 Clone Check에 이용된다.

4

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk** | **Probability** | **Significance** | **Weight** |
| Un-Skilled OOPT | 3 | 5 | 15 |
| Lack of knowledge Clone Checker | 2 | 4 | 8 |
| Assignment Stack Overflow | 5 | 2 | 10 |
| Spring Love | 1 | 5 | 5 |
| Absence | 1 | 10 | 10 |

|  |  |
| --- | --- |
| **Risk** | **Reduction Plan** |
| Un-Skilled OOPT | 교수님 혹은 강의조교에게 자문을 구한다. |
| Lack of knowledge Clone Checker | Clone Checker 구동 방식에 대해 공부한다. |
| Assignment Stack Overflow | Time Sharing 기법을 활용하여 동시에 처리한다. |
| Spring Love | 합숙 프로젝트를 진행한다. (가능성 배제) |
| Absence | 설득한다 |

[SMA2016]T2 - OOPT stage 1000 ver 1.0

**Activity 1002. Create Preliminary Investigation Report**

**1. Alternative solution**

1.1 시중에 판매하는 Clone Check Program을 구매한다.

1.2 인력을 동원한다.

1.3 개발업체에 제작을 의뢰한다.

**2. Project justification**

1) Cost : 2,700,000\

2) Duration : 3 Month

3) Risk : OOPT에 대한 이해력 부족. Clone Checker에 대한 지식 부족, 과제 Stack Overflow,

벚꽃사랑. 팀원의 휴학

4) Effect : 프로그램 유지 보수가 수월하다. 양산되는 과제물에 대한 Check가 용이하다.

비용 측면에서 저렴하다.

**3. Risk & Risk reduction plan**

5

3.1  Risk Management

3.2  Risk Reduction Plan

[SMA2016]T2 - OOPT stage 1000 ver 1.0

**4. Market Analysis**

- Alphago에 사회적 issue화에 의해 증가할 것으로 전망되는 Computer Engineering 시장.

**5. Managerial issue**

- 2016년 5월까지 proto-type 개발이 완료되어야 한다.

- 2016년 7월까지 Final version 개발이 완료되어야 한다.

6

[SMA2016]T2 - OOPT stage 1000 ver 1.0



**Activity 1003. Define Requirements**

**1. Functional Requirements**

**Function** **Description**

Load Code 검사할 Code를 List에 Load한다.

Delete Code List에 올라와있던 Code를 List에서 삭제한다.

Display Code List  Load된 Code의 List를 보여준다.

Detect Duplicate Code  Load된 모든 Code간의 Duplicate 여부를 Detect 시작한다..

Clone Check Load된 모든 Code간의 Duplicate 여부를 Detect 한다.

Display Code Set List Code Set List를 display한다.

Show All  Clone Check 후의 모든 Code Set의 유사성을 표시한다.

Show Red Clone Check 후 두 코드간 유사성이 100~90%인 Code Set을

표시한다.

Show Orange Clone Check 후 두 코드간 유사성이 90~70%인 Code Set을

표시한다

Show Yellow Clone Check 후 두 코드간 유사성이 70~50%인 Code Set을

표시한다

Show Green Clone Check 후 두 코드간 유사성이 50% 미만인 Code Set을

표시한다

Select Code Set  Check List에서 Code Set을 선택한다.

Display Main Code  List에서 선택한 비교항목의 Main Code를 display한다.

Display Sub Code  List에서 선택한 비교항목의 Sub Code를 display한다.

Display Similarity 선택한 두 Code의 Similarity를 Progress Bar에 Display한다.

User Accept User가 선택한 두 Code의 Duplicate가 의심되는 구간을 확인

하고 Clone으로 판단한다.

User Reject User가 선택한 두 Code의 Duplicate가 의심되는 구간을 확인

하고 Clone이 아니라고 판단한다.

Save  User가 검토한 정보를 저장하여 list에 적용한다.

7

|  |  |  |
| --- | --- | --- |
| **Reference No.** | **Function** | **Category** |
| R1.1 | Load Code | Evident |
| R1.2 | Delete Code | Evident |
| R1.3 | Display Code List | Hidden |
| R2.1 | Detect Duplicate Code | Evident |
| R2.2.1 | Clone Check | Hidden |
| R2.2.2 | Display Code Set List | Hidden |
| R3.1 | Show All | Evident |
| R3.2 | Show Red | Evident |
| R3.3 | Show Orange | Evident |
| R3.4 | Show Yellow | Evident |
| R3.5 | Show Green | Evident |
| R4.1 | Select Code Set | Evident |
| R4.2 | Display Main Code | Hidden |
| R4.3 | Display Sub Code | Hidden |
| R4.4 | Display Similarity | Hidden |
| R5.1 | User Accept | Evident |
| R5.2 | User Reject | Evident |
| R6 | Save | Evident |

|  |  |  |  |
| --- | --- | --- | --- |
| **Source** | **Case1** | **Case2** | **Case3** |
| Operating Software | Win7 – 64bit | Win8 – 64bit | Mac OS – 64bit |
| CPU | i3-4005U | i5-4200U | i5-4200U |
| RAM | 8GB | 4GB | 8GB |
| Library | 128SSD | 128GB SSD | 128GB SSD |

[SMA2016]T2 - OOPT stage 1000 ver 1.0

- 검사시간은 3min/50File 이내로 유지되어야 한다.

**2.  Functional Requirements (Categorized Table)**

**3.  Performance Requirements**

Develop Environment: Eclipse LUNA/MARS/KEPLER

**4.  Operating Environment**

8

[SMA2016]T2 - OOPT stage 1000 ver 1.0

**5. Interface Requirements**

- 유사도 percent에 따른 등급을 보기 좋게 표현해야 한다.

- 유사도 등급 기준에 따라 Code Set이 정확하게 분리되어야 한다.

- Duplicate가 의심되는 부분을 출력하여 User가 판단할 수 있도록 한다.

- User가 유사도 그래프를 확인하여 직관적인 관찰이 가능하도록 한다.

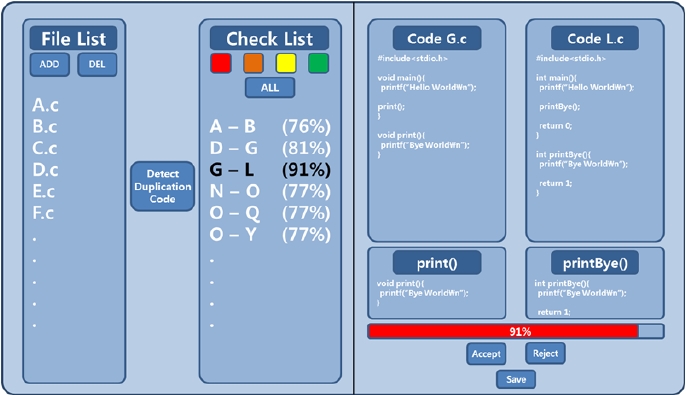
**6. Other Requirements**

- N/A

9

|  |  |  |
| --- | --- | --- |
| **Term** | **Description** | **Remarks** |
| User | Clone Checker 프로그램을 사용하여 Clone 여부를  판단하는 객체 |  |
| Value | Code 에서 선언된 Value 의 객체 |  |
| Operation | Code 에서 선언된 Operation 의 객체 |  |
| Code | User 가 Load 한 Code 객체 |  |
| Progress Bar | 유사도를 보여주는 Progress Bar |  |
| Main Controller | 프로그램을 제어하는 객체 |  |

[SMA2016]T2 - OOPT stage 1000 ver 1.0

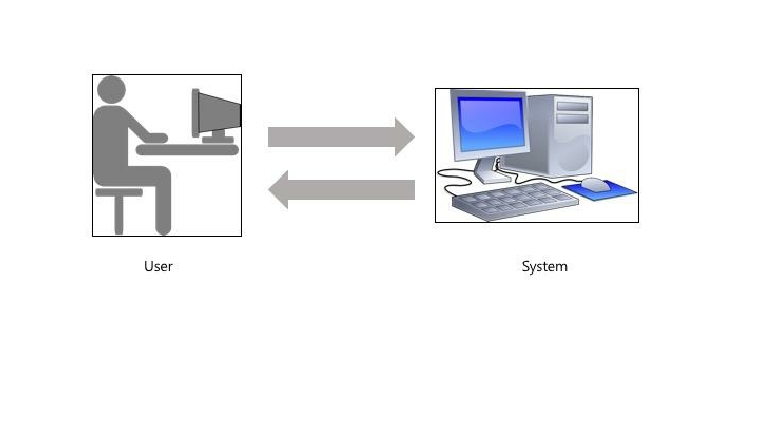


**Activity 1005. Implement Prototype**

**Activity 1004. Record Terms in Glossary**

10

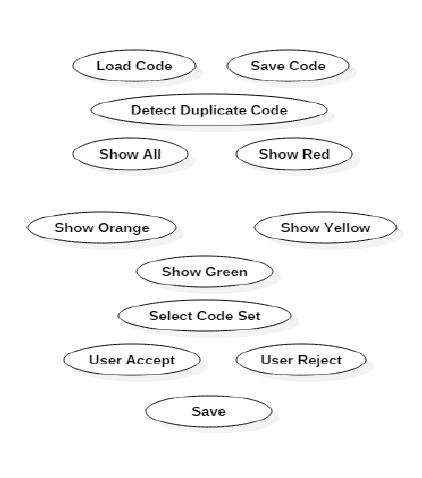
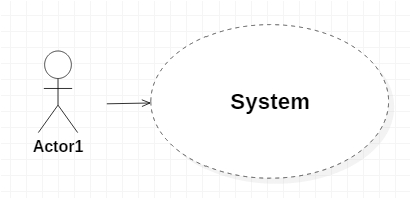
[SMA2016]T2 - OOPT stage 1000 ver 1.0



**Activity 1006. Define Draft System Architecture**

11

[SMA2016]T2 - OOPT stage 1000 ver 1.0



**Activity 1007. Define Business Use Case**

**1. Define system boundary**

**2. Identify and describe actors**

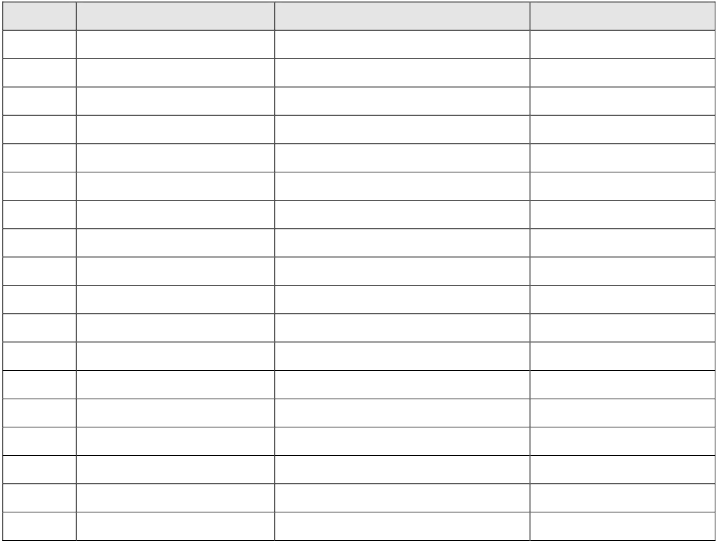
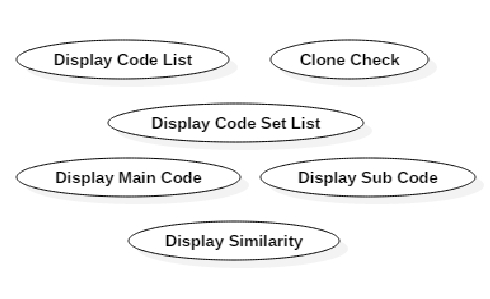
- User : Duplicate Code를 Detect하는 대상.

**3. Identify use cases**

3.1 Actor-Based

12

[SMA2016]T2 - OOPT stage 1000 ver 1.0



3.2 Event-Based

**4. Allocate system functions into related use cases**

**Ref. #.** **Function** **Use Case Number & Name** **Category**

R1.1 Load Code 1. Load Code

R1.2 Delete Code 2. Delete Code

R1.3 Display Code List 3. Display Code List

R2.1 Detect Duplicate Code 4. Detect Duplicate Code

R2.2.1 Clone Check 5. Clone Check

R.2.2.2 Display Code Set List 6. Display Code Set List

R3.1 Show All 7. Show All

R3.2 Show Red 8. Show Red

R3.3 Show Orange 9. Show Orange

R3.4 Show Yellow 10. Show Yellow

R3.5 Show Green 11. Show Green

R4.1 Select Code Set 12. Select Code Set

R4.2 Display Main Code 13. Display Main Code

R4.3 Display Sub Code 14. Display Sub Code

R4.4 Display Similarity 15. Display Similarity

R5.1 User Accept 16. User Accept

R5.2 User Reject 17. User Reject

R6 Save 18. Save

13

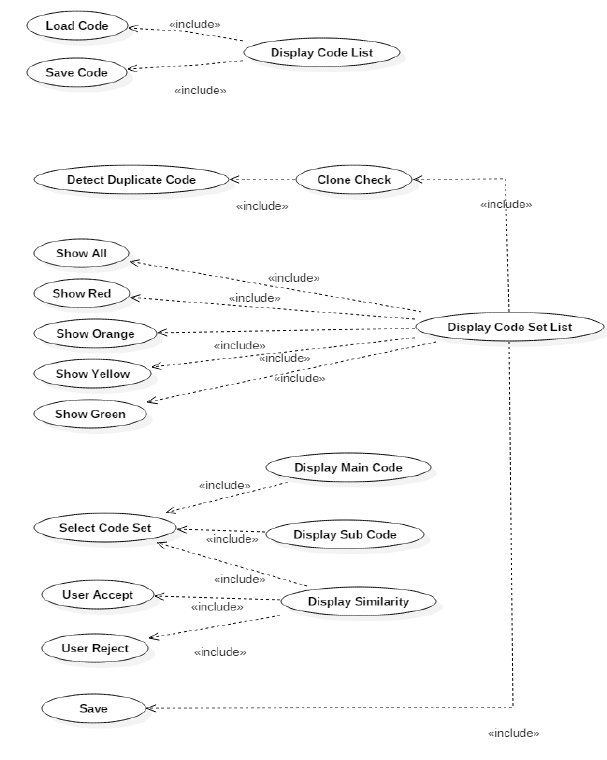
|  |  |  |  |
| --- | --- | --- | --- |
| **Ref. #.** | **Function** | **Use Case Number & Name** | **Category** |
| R1.1 | Load Code | 1. Load Code | Primary |
| R1.2 | Delete Code | 2. Delete Code | Primary |
| R1.3 | Display Code List | 3. Display Code List | Primary |
| R2.1 | Detect Duplicate Code | 4. Detect Duplicate Code | Primary |
| R2.2.1 | Clone Check | 5. Clone Check | Primary |
| R2.2.2 | Display Code Set List | 6. Display Code Set List | Primary |
| R3.1 | Show All | 7. Show All | Primary |
| R3.2 | Show Red | 8. Show Red | Primary |
| R3.3 | Show Orange | 9. Show Orange | Primary |
| R3.4 | Show Yellow | 10. Show Yellow | Primary |
| R3.5 | Show Green | 11. Show Green | Primary |
| R4.1 | Select Code Set | 12. Select Code Set | Primary |
| R4.2 | Display Main Code | 13. Display Main Code | Primary |
| R4.3 | Display Sub Code | 14. Display Sub Code | Primary |
| R4.4 | Display Similarity | 15. Display Similarity | Primary |
| R5.1 | User Accept | 16. User Accept | Primary |
| R5.2 | User Reject | 17. User Reject | Primary |
| R6 | Save | 18. Save | Primary |

[SMA2016]T2 - OOPT stage 1000 ver 1.0

**5. Categorize use cases**

14

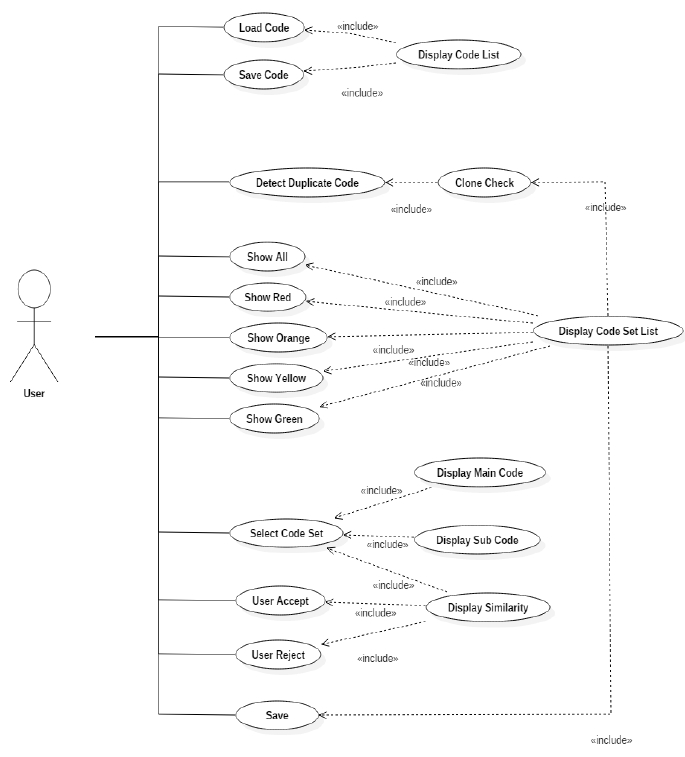
[SMA2016]T2 - OOPT stage 1000 ver 1.0



**6. Identify relationships between use cases**

15

[SMA2016]T2 - OOPT stage 1000 ver 1.0



**7. Draw a use case diagram**

16

|  |  |
| --- | --- |
| **Use Case** | 3. Display Code List |
| **Actors** | None |
| **Description** | Load된 Code의 List를 보여준다. |

|  |  |
| --- | --- |
| **Use Case** | 4. Detect Duplicate Code |
| **Actors** | User |
| **Description** | Load된 모든 Code간의 Duplicate 여부를 Detect 시작한다. |

|  |  |
| --- | --- |
| **Use Case** | 5. Clone Check |
| **Actors** | None |
| **Description** | Load된 모든 Code간의 Duplicate 여부를 Detect 한다. |

|  |  |
| --- | --- |
| **Use Case** | 1. Load Code |
| **Actors** | User |
| **Description** | 검사할 Code를 List에 Load한다. |

|  |  |
| --- | --- |
| **Use Case** | 2. Delete Code |
| **Actors** | User |
| **Description** | List에 올라와있던 Code를 List에서 삭제한다. |

|  |  |
| --- | --- |
| **Use Case** | 6. Display Code Set List |
| **Actors** | None |
| **Description** | Code Set List를 Display한다. |

|  |  |
| --- | --- |
| **Use Case** | 7. Show All |
| **Actors** | User |
| **Description** | Clone Check 후의 모든 Code Set의 유사성을 표시한다. |

[SMA2016]T2 - OOPT stage 1000 ver 1.0

17

**8.  Describe use cases**

|  |  |
| --- | --- |
| **Use Case** | 10. Show Yellow |
| **Actors** | User |
| **Description** | Clone Check 후 두 코드간 유사성이 70~50%인 Code Set을 표시한다. |

|  |  |
| --- | --- |
| **Use Case** | 8. Show Red |
| **Actors** | User |
| **Description** | Clone Check 후 두 코드간 유사성이 100~90%인 Code Set을 표시한다. |

|  |  |
| --- | --- |
| **Use Case** | 12. Select Code Set |
| **Actors** | User |
| **Description** | Check List에서 Code Set을 선택한다. |

|  |  |
| --- | --- |
| **Use Case** | 11. Show Green |
| **Actors** | User |
| **Description** | Clone Check 후 두 코드간 유사성이 50% 미만인 Code Set을 표시한다. |

|  |  |
| --- | --- |
| **Use Case** | 9. Show Orange |
| **Actors** | User |
| **Description** | Clone Check 후 두 코드간 유사성이 90~70%인 Code Set을 표시한다. |

|  |  |
| --- | --- |
| **Use Case** | 14. Display Sub Code |
| **Actors** | None |
| **Description** | List에서 선택한 비교항목의 Sub Code를 display한다. |

|  |  |
| --- | --- |
| **Use Case** | 13. Display Main Code |
| **Actors** | None |
| **Description** | List에서 선택한 비교항목의 Main Code를 display한다. |

[SMA2016]T2 - OOPT stage 1000 ver 1.0

18

|  |  |
| --- | --- |
| **Use Case** | 16. User Accept |
| **Actors** | User |
| **Description** | User가 선택한 두 Code의 Duplicate가 의심되는 구간을 확인하고  Clone으로 판단한다. |

|  |  |
| --- | --- |
| **Use Case** | 17. User Reject |
| **Actors** | User |
| **Description** | User가 선택한 두 Code의 Duplicate가 의심되는 구간을 확인하고  Clone이 아니라고 판단한다. |

|  |  |
| --- | --- |
| **Use Case** | 15. Display Similarity |
| **Actors** | None |
| **Description** | 선택한 두 Code의 Similarity를 Progress Bar에 Display한다. |

|  |  |
| --- | --- |
| **Use Case** | 18. Save |
| **Actors** | User |
| **Description** | User가 검토한 정보를 저장하여 list에 적용한다. |

[SMA2016]T2 - OOPT stage 1000 ver 1.0

19

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Ref. #.** | **Function** | **Use Case Number & Name** | **Category** | **Rank** |
| R1.1 | Load Code | 1. Load Code | Primary | High |
| R1.2 | Delete Code | 2. Delete Code | Primary | High |
| R1.3 | Display Code List | 3. Display Code List | Primary | High |
| R2.1 | Detect Duplicate Code | 4. Detect Duplicate Code | Primary | High |
| R2.2.1 | Clone Check | 5. Clone Check | Primary | High |
| R2.2.2 | Display Code Set List | 6. Display Code Set List | Primary | High |
| R3.1 | Show All | 7. Show All | Primary | High |
| R3.2 | Show Red | 8. Show Red | Primary | High |
| R3.3 | Show Orange | 9. Show Orange | Primary | High |
| R3.4 | Show Yellow | 10. Show Yellow | Primary | High |
| R3.5 | Show Green | 11. Show Green | Primary | High |
| R4.1 | Select Code Set | 12. Select Code Set | Primary | High |
| R4.2 | Display Main Code | 13. Display Main Code | Primary | High |
| R4.3 | Display Sub Code | 14. Display Sub Code | Primary | High |
| R4.4 | Display Similarity | 15. Display Similarity | Primary | High |
| R5.1 | User Accept | 16. User Accept | Primary | High |
| R5.2 | User Reject | 17. User Reject | Primary | High |
| R6 | Save | 18. Save | Primary | High |

[SMA2016]T2 - OOPT stage 1000 ver 1.0

**9. Rank use cases**

20

[SMA2016]T2 - OOPT stage 1000 ver 1.0



**Activity 1008. Define Business Concept Model**

21

|  |  |  |
| --- | --- | --- |
| **Use-Case No.** | **Use-Case** | **Test Plan** |
| 1 | Load Code | User가 Load한 Code File이 List에 Load되는지 확인  한다. |
| 2 | Delete Code | User가 Delete한 Code File이 List에서 Delete되는지  확인한다. |
| 3 | Display Code List | User가 Load한 List가 Display 되는지 확인한다. |
| 4 | Detect Duplicate Code | User가 요청한 Detect Duplicate Code 명령이 수행되  는지 확인한다. |
| 5 | Clone Check | Load된 모든 Code간의 Clone Check가 수행되는지  확인한다. |
| 6 | Display Code Set List | Code Set List들이 Display 되는지 확인한다. |
| 7 | Show All | 모든 Code Set List가 Display 되는지 확인한다. |
| 8 | Show Red | 코드간 유사성이 100~90%인 Code Set이 Display 되  는지 확인한다. |
| 9 | Show Orange | 코드간 유사성이 90~70%인 Code Set이 Display 되  는지 확인한다 |
| 10 | Show Yellow | 코드간 유사성이 70~50%인 Code Set이 Display 되  는지 확인한다 |
| 11 | Show Green | 코드간 유사성이 50% 미만인 Code Set이 Display 되  는지 확인한다 |
| 12 | Select Code Set | Check List에서 Code Set이. 선택되는지 확인한다. |
| 13 | Display Main Code | List에서 선택한 비교항목의 Main Code가 Display 되  는지 확인한다. |
| 14 | Display Sub Code | List에서 선택한 비교항목의 Sub Code가 Display 되  는지 확인한다. |
| 15 | Display Similarity | 선택한 두 Code의 Similarity가 Progress Bar에  Display 되는지 확인한다. |
| 16 | User Accept | User Accept 명령이 정확하게 전달되는지 확인한다. |
| 17 | User Reject | User의 Reject 명령이 정확하게 전달되는지 확인한다. |
| 18 | Save | User의 Save 명령이 정확하게 전달되는지 확인한다. |

22

[SMA2016]T2 - OOPT stage 1000 ver 1.0

**Activity 1009. Define System Test Case**

[SMA2016]T2 - OOPT stage 1000 ver 1.0

**Activity 1010. Refine Plan**

**1. Project Scope**

소스코드의 복제 및 도용으로 인한 문제를 방지하기 위해 소스코드간의 Cloning 여부를

판단한다.

**2. Project Objectives**

다수의 C 프로그램을 대장으로 상호 cheating 여부를 정량적으로 판단하고, 해당 내용을

User가 검토하여 결과를 수정할 수 있는 Clone Checker를 개발한다.

**3. Functional Requirements**

- Load Code

- Delete Code

- Display Code List

- Detect Duplicate Code

- Clone Check

- Display Code Set List

- Show All

- Show Red

- Show Orange

- Show Yellow

- Show Green

- Select Code Set

- Display Main Code

- Display Sub Code

- Display Similarity

- User Accept

- User Reject

- Save

23

|  |  |  |  |
| --- | --- | --- | --- |
| **Source** | **Case1** | **Case2** | **Case3** |
| Operating Software | Win7 – 64bit | Win8 – 64bit | Mac OS – 64bit |
| CPU | i3-4005U | i5-4200U | i5-4200U |
| RAM | 8GB | 4GB | 8GB |
| Library | 128SSD | 128GB SSD | 128GB SSD |

[SMA2016]T2 - OOPT stage 1000 ver 1.0

**4. Performance Requirements**

- 검사시간을 3min/50File 이내로 유지한다.

- 유사도 percent에 따른 등급을 보기 좋게 표현해야 한다.

**5.  Operating Environment**

**6.  User Interface Requirements**

- 유사도 등급 기준에 따라 Code Set이 정확하게 분리되어야 한다.

- Duplicate가 의심되는 부분을 출력하여 User가 판단할 수 있도록 한다.

- User가 유사도 그래프를 확인하여 직관적인 관찰이 가능하도록 한다.

**7. Other Requirements**

- N/A

**8. Resources**

8.1 Human efforts (M/M)

3M/3M

8.2 Human resources

3명 – 프로그래머 3명

8.3 Duration

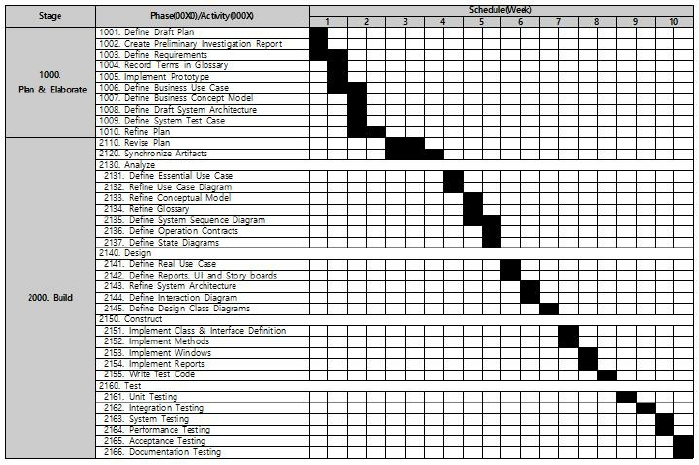
3개월

8.4 Budget

\2,700,000

24

[SMA2016]T2 - OOPT stage 1000 ver 1.0



**9. Scheduling**

25