



# CPT T1 STAGE\_2040

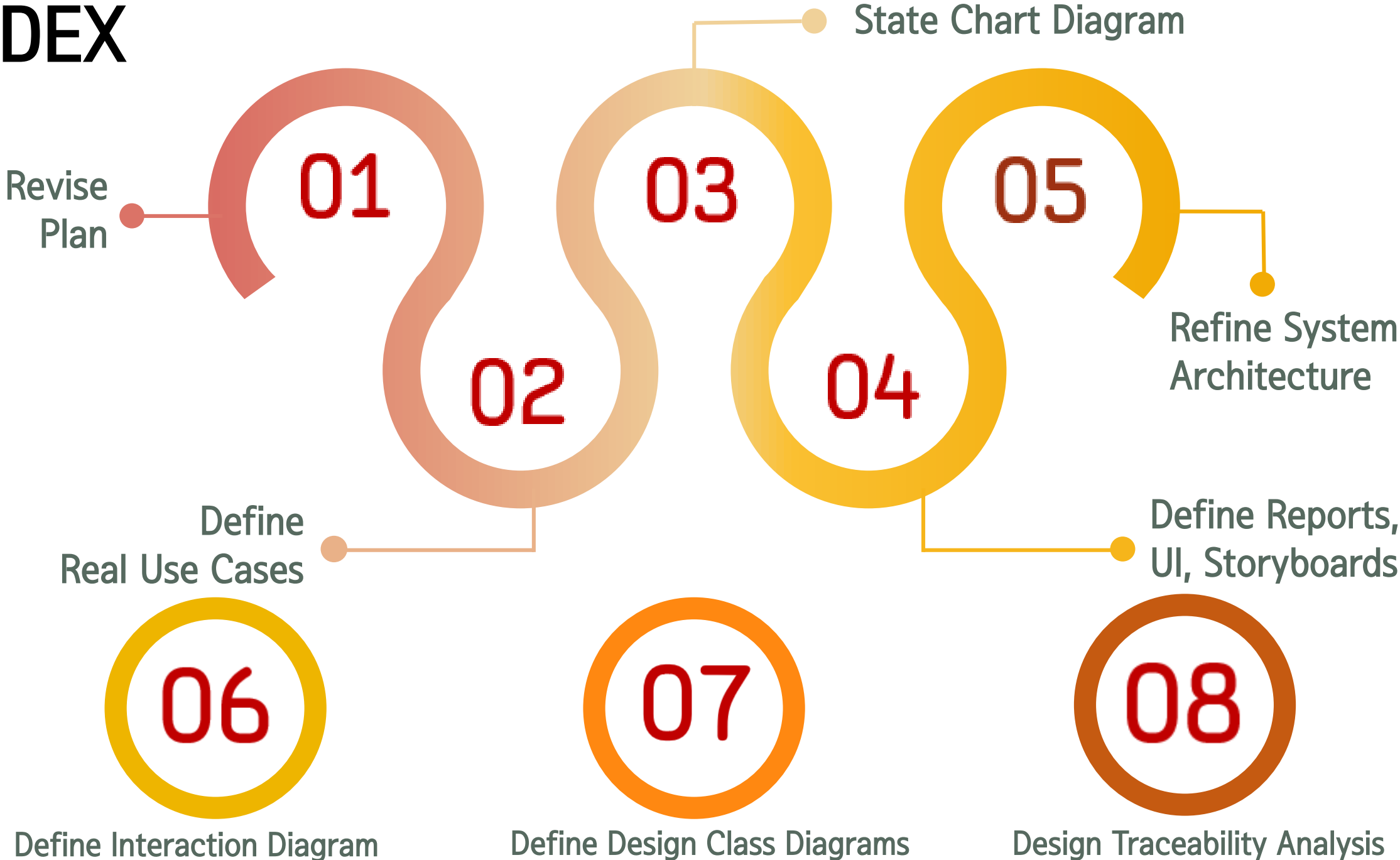
17.05.12 Third Presentation

201111334 김강욱 / 201211339 김진욱(Leader)

201312243 김동권 / 201510411 이소영



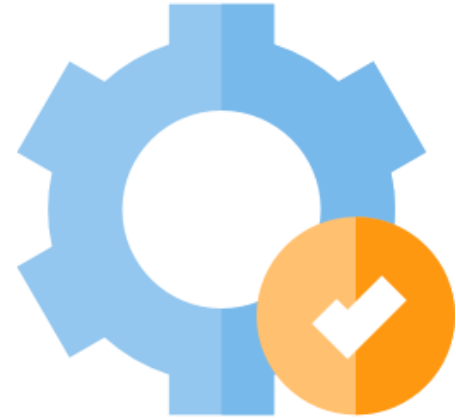
# INDEX





# Revise Plan

Project Scope



**How to guarantee?**



# Revise Plan

## Functional Requirements

### Version 2

| Function                  | Description                                    |
|---------------------------|--|
| Open File                 | 엑셀 파일을 연다.                                     |
| Parsing File              | 엑셀파일을 읽어 프로그램 내부에 저장.                          |
| Modify Weight             | 가중치 수정   |
| Input Essential Test Case | 중요 테스트 케이스 입력                                  |
| Compare with Test Case    | 프로그램에 의해 조합 된 테스트 케이스와 사용자가 입력한 중요 테스트 케이스를 비교 |
| Modify Constraints        | 제한 조건 수정                                       |
| Make Test Case            | 테스트 케이스 생성                                     |
| Arrange by Weight Average | 가중치 평균에 따른 정렬                                  |
| Make Final Test Case File | 최종 테스트 케이스 파일 생성                               |



### Version 3

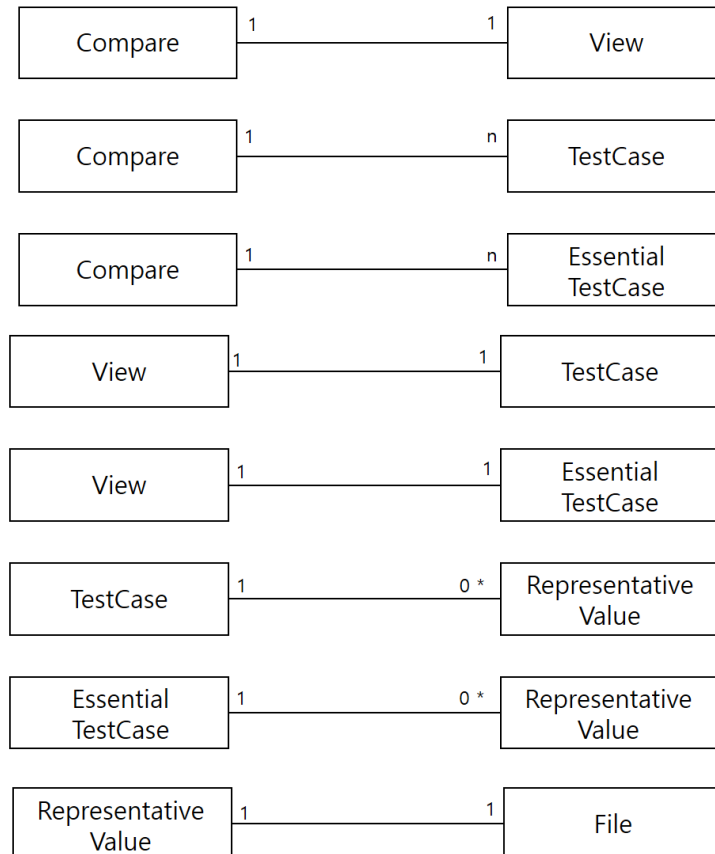
| Function                  | Description                                    |
|---------------------------|--|
| Open File                 | 엑셀 파일을 열고 조건에 맞게 프로그램 내부에 저장                   |
| Modify Weight             | 가중치 수정   |
| Input Essential Test Case | 중요 테스트 케이스 입력                                  |
| Compare with Test Case    | 프로그램에 의해 조합 된 테스트 케이스와 사용자가 입력한 중요 테스트 케이스를 비교 |
| Modify Constraints        | 제한 조건 수정                                       |
| Make Test Case            | 테스트 케이스 생성                                     |
| Arrange by Weight Average | 가중치 평균에 따른 정렬                                  |
| Make Final Test Case File | 최종 테스트 케이스 파일 생성                               |



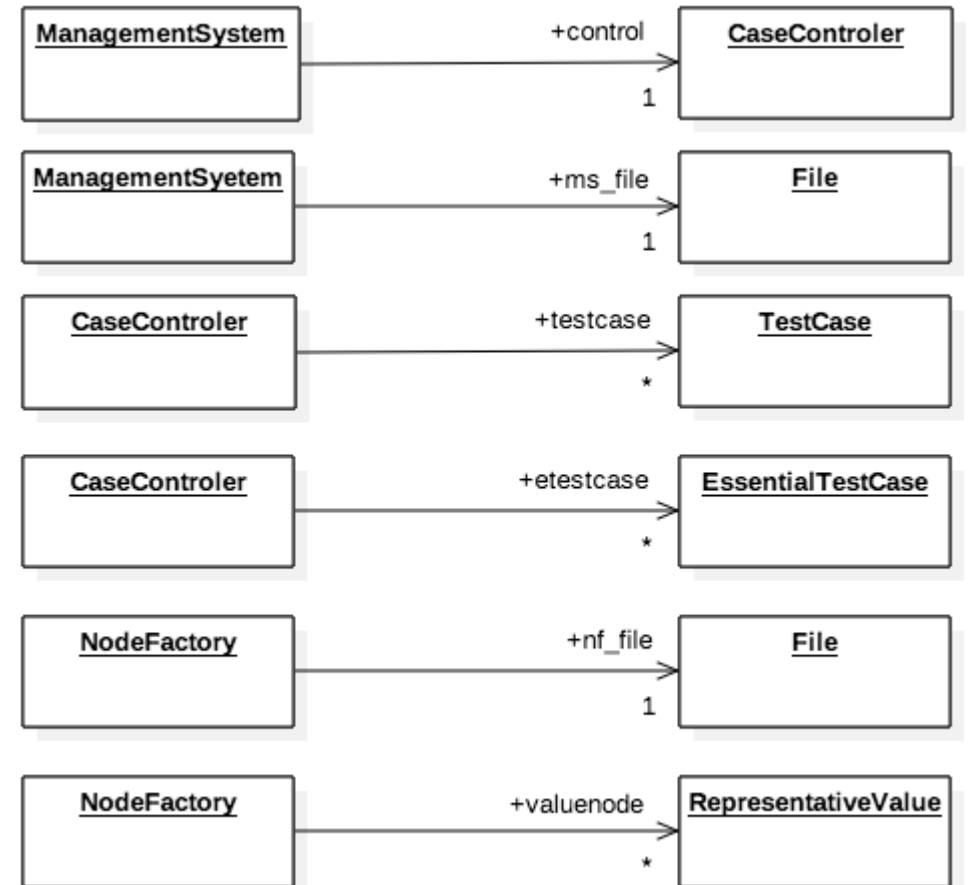
# Revise Plan

## Domain Model

### Version 1



### Version 2

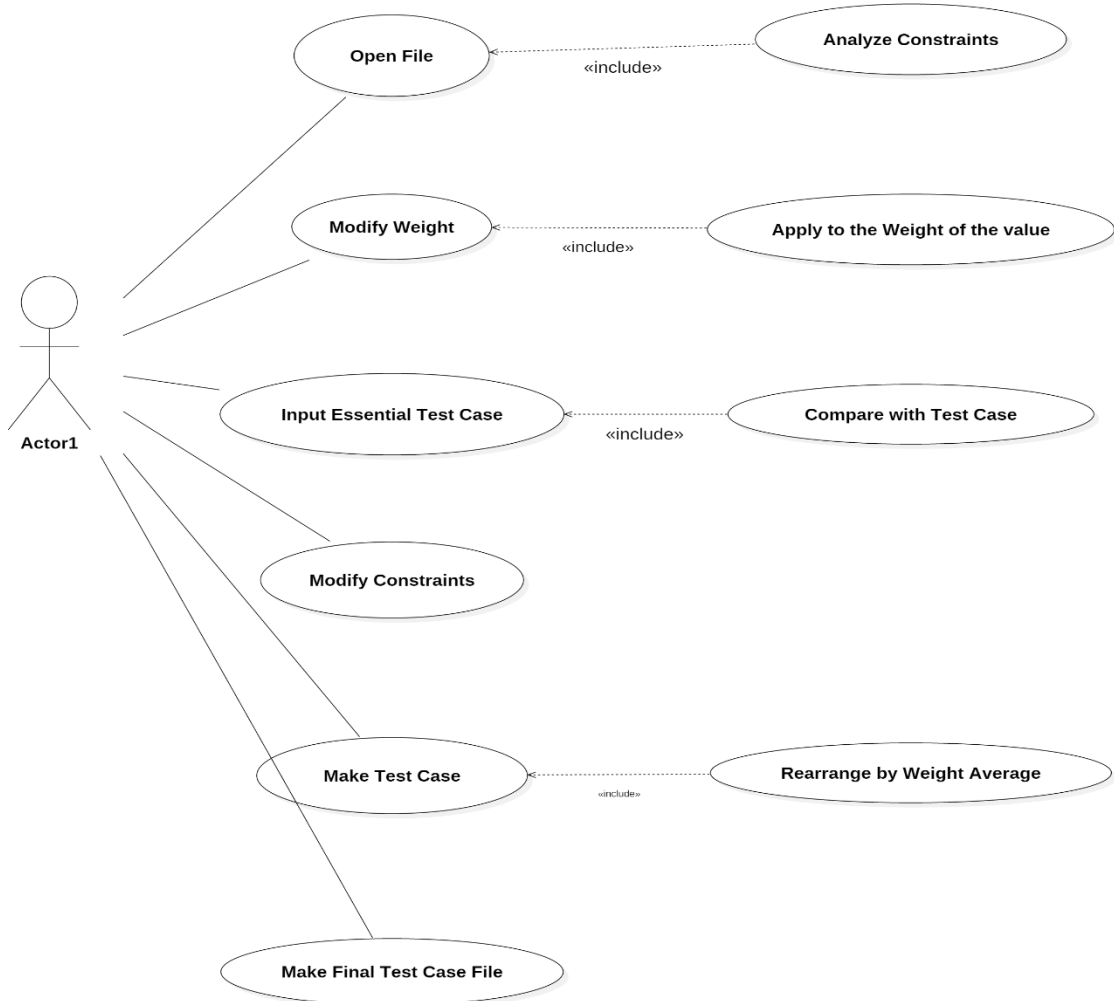




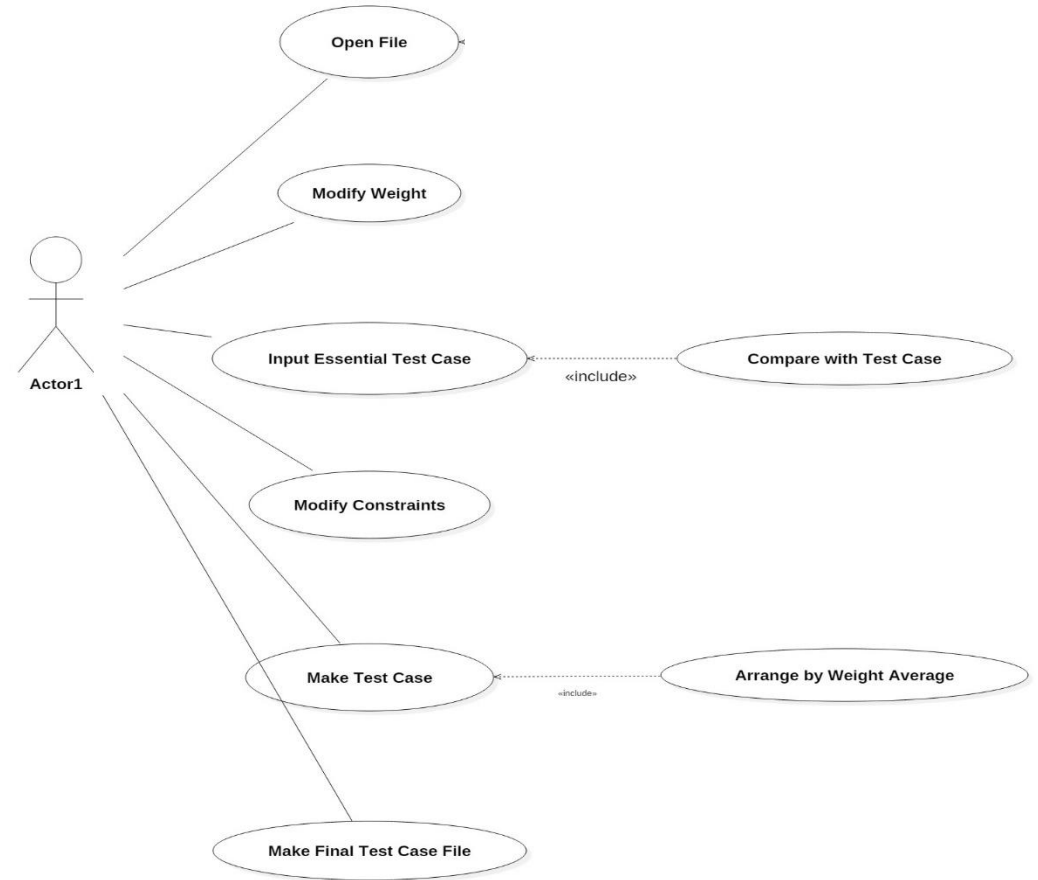
# Revise Plan

## Use Cases

### Version 2



### Version 3





# Define Real Use cases

## Open File



|                               |   |
|-------------------------------|---|
| Use Case                      | Open File   |
| Actor                         | User  |
| Purpose                       | Excel File을 열어서 프로그램에 넣어준다.   |
| Overview                      | Excel File을 열고, Excel File로부터 Category별 Representative Value의 정보를 읽고 각 정보를 Type에 맞는 변수에 넣어준다.   |
| Type                          | Primary and Essential   |
| Cross Reference               | System Function: R 1.2  |
| Pre-Requisites                | N/A   |
| UI Widgets                    | Sample UI 1.2<br>Sample UI 1.3  |
| Typical Courses of Events     | (A): Actor, (S): System<br>1. (A) User가 [파일열기버튼]을 누른다.<br>2. (S) [다이얼로그 박스]를 띄운다.<br>3. (A) [다이얼로그 박스]에서 파일을 선택한다.<br>4. (S) 선택한 파일을 불러온다.<br>5. (S) Excel file의 Representative Value와 해당 Value의 Index, Weight, Constraints의 정보를 읽는다.<br>6. (S) 확인한 정보를 각각 지정해둔 변수 배열에 넣어준다.<br>7. (S) [파일이름]에 파일명을 보여준다. |
| Alternative Courses of Events | 4. (S) 파일의 확장자명이.xlsx 혹은.xls 와 다르면 "파일 확장자명 오류입니다" 라는 오류메시지를 출력한다.<br>4. (S) 확장자명 이외의 오류로 파일을 가져올 수 없을 경우 "파일 열기 오류입니다" 라는 오류메시지를 출력한다.<br>5. (S) 읽어온 파일에 정보가 하나라도 없으면 오류 난 부분의 행 열 값을 띄워준다.<br>5. (S) 읽어온 파일의 양식이 다르면 "양식 오류입니다" 라는 오류메시지를 출력한다  |
| Exceptional Courses of Events | N/A   |



# Define Real Use cases

## Modify Weight



| Use Case                      | Modify Weight  |
|-------------------------------|--|
| Actor                         | User   |
| Purpose                       | Representative Value의 가중치를 변경한다.   |
| Overview                      | Representative Value에 할당되어 있는 가중치를 초기값과 다르게 변경한다.  |
| Type                          | Primary and Essential  |
| Cross Reference               | System Function: R 5.2<br>Use case: Arrange by Weight Average  |
| Pre-Requisites                | Test Case가 조합되어 있어야 한다.  |
| UI Widgets                    | Sample UI 1.2  |
| Typical Courses of Events     | (A): Actor, (S): System<br>1. (A) User가 수정할 Representative Value의 Weight 값을 [테스트케이스 테이블]에 입력한다<br>2. (S) User로부터 수정 요청 받은 Representative Value의 weight 변수를 사용자가 입력한 값으로 갱신한다.<br>3. (S) 수정 요청 받은 Representative Value가 속해 있는 Test Case의 Weight를 다시 계산한다. |
| Alternative Courses of Events | 1. (A) 양의 정수가 아닌 수정 값을 입력하면 System이 재입력을 요청한다.   |
| Exceptional Courses of Events | N/A  |





# Define Real Use cases

## Input Essential Test Case



| Use Case                      | Input Essential Test Case  |
|-------------------------------|--|
| Actor                         | User   |
| Purpose                       | 중요하다고 생각되는 Test case를 입력한다.  |
| Overview                      | Representative Value 의 Index를 이용해 Test case 조합을 프로그램에 입력하고 Essential Test case 목록에 추가한다.   |
| Type                          | Primary and Essential  |
| Cross Reference               | System Function: R 3.2<br>Use case: Compare with Test Case   |
| Pre-Requisites                | 파일이 파싱된 상태여야 한다.   |
| UI Widgets                    | Sample UI 1.1<br>Sample UI 1.2   |
| Typical Courses of Events     | (A): Actor, (S): System<br>1. (A) 100, 101, 200과 같이 Representative Value의 Index의 조합을 [중요 테스트케이스 입력창에 입력한다.<br>2. (S) index와 일치하는 Representative Value를 확인하고, 이를 제한조건을 고려하지 않고, 단순 조합하여 중요 Test Case를 생성한다.<br>3. (S) 조합된 중요 Test Case를 [중요테스트케이스 창]에 출력한다. |
| Alternative Courses of Events | 2. (S) User가 유효하지 않은 Index를 추가 했을 경우 "유효하지 않은 index입니다."라는 오류메시지를 출력한다.  |
| Exceptional Courses of Events | N/A  |



# Define Real Use cases

## Compare with Test Case



| Use Case                      | Compare with Test Case   |
|-------------------------------|--|
| Actor                         | N/A  |
| Purpose                       | Essential Test case 와 조합된 Test Case를 비교한다.   |
| Overview                      | 프로그램에 의해 조합된 Test Case와 사용자가 입력한 중요 Test Case를 비교하여 누락 여부와 원인을 확인한다.   |
| Type                          | Primary and Essential  |
| Cross Reference               | Functional Requirement: R 3.1<br>Use Case: Input Essential Test Case   |
| Pre-Requisites                | 1. 중요 Test Case가 입력되어야 한다.<br>2. Test Case가 조합되어 있어야 한다.   |
| UI Widgets                    |  |
| Typical Courses of Events     | (A): Actor (S): System<br>1. (S) 중요 Test Case와 프로그램에 의해 조합된 Test Case를 비교하여 중요 Test Case와 일치하는 조합된 Test Case를 자주색으로 표시한다<br>2. (S) 중요 Test Case와 프로그램에 의해 조합된 Test Case를 비교하여 누락되었다면 누락된 중요 Test Case를 노랑색으로 표시한다.<br>3. (S) 중요 Test Case가 누락된 원인이 되는 제한조건을 확인해서 제한조건을 빨간색으로 표시한다. |
| Alternative Courses of Events | N/A  |
| Exceptional Courses of Events | N/A  |



# Define Real Use cases

## Modify Constraints



| Use Case                      | Modify Constraints  |
|-------------------------------|---|
| Actor                         | User  |
| Purpose                       | 제한 조건을 수정한다.  |
| Overview                      | Representative Value의 제한조건을 수정한다.   |
| Type                          | Primary and Essential   |
| Cross Reference               | Functional Requirement: R 5.1<br>Use Case: Make Test Case   |
| Pre-Requisites                | Test Case가 조합되어 있어야 한다.   |
| UI Widgets                    | Sample UI 1.2   |
| Typical Courses of Events     | (A): Actor (S): System<br>1. (A) User가 수정할 Representative Value의 Constraints 값을 [테스트케이스 테이블]에 입력한다.<br>2. (S) User로부터 수정 요청 받은 Representative Value의 Constraints 변수를 사용자가 입력한 값으로 갱신한다. |
| Alternative Courses of Events | 1. 유효하지 않은 Constraints를 입력하면 "유효하지 않은 Constraints입니다."라는 오류메시지를 출력한다.   |
| Exceptional Courses of Events | N/A   |



# Define Real Use cases

## Make Test Case



| Use Case                      | Make Test Case  |
|-------------------------------|---|
| Actor                         | User  |
| Purpose                       | Test Case를 생성한다.  |
| Overview                      | 설정된 제한조건을 바탕으로 Test Case를 생성한다.   |
| Type                          | Primary and Essential   |
| Cross Reference               | Functional Requirement: R 5.2<br>Use Case: Arrange by Weight Average                |
| Pre-Requisites                | 파일이 파싱 되어있어야 한다.  |
| UI Widgets                    | Sample UI 1.2   |
| Typical Courses of Events     | (A): Actor (S): System<br>1. (A) [조합 버튼]을 누른다.<br>2. (S) 제한조건에 따라 Test Case를 조합한다.. |
| Alternative Courses of Events | N/A   |
| Exceptional Courses of Events | N/A   |



# Define Real Use cases

## Arrange by Weight Average



| Use Case                      | Arrange by Weight Average  |
|-------------------------------|--|
| Actor                         | N/A  |
| Purpose                       | Test Case의 가중치 평균에 따라 재정렬한다.   |
| Overview                      | User가 수정한 가중치를 반영하여 Test Case를 재정렬한다.  |
| Type                          | Primary and Essential  |
| Cross Reference               | Functional Requirement: R 2.1, R 5.1<br>Use Case: Modify Weight, Make Test Case  |
| Pre-Requisites                | Test Case가 조합되어 있어야 한다.  |
| UI Widgets                    | Sample UI 1.2  |
| Typical Courses of Events     | (A): Actor (S): System<br>1. (S) Test Case를 구성하는 Representative Value 들의 가중치의 평균을 구한다.<br>2. (S) Test Case의 가중치 평균을 기준으로 내림차순 정렬한다.<br>3. (S) Test Case를 정렬된 순서로 [테스트케이스 테이블]에 출력한다. |
| Alternative Courses of Events | 2. (S) 같은 가중치를 가진 Test Case 가 있으면 Representative Value의 Index 순으로 출력한다.  |
| Exceptional Courses of Events | N/A  |



# Define Real Use cases

## Make Final Test Case File

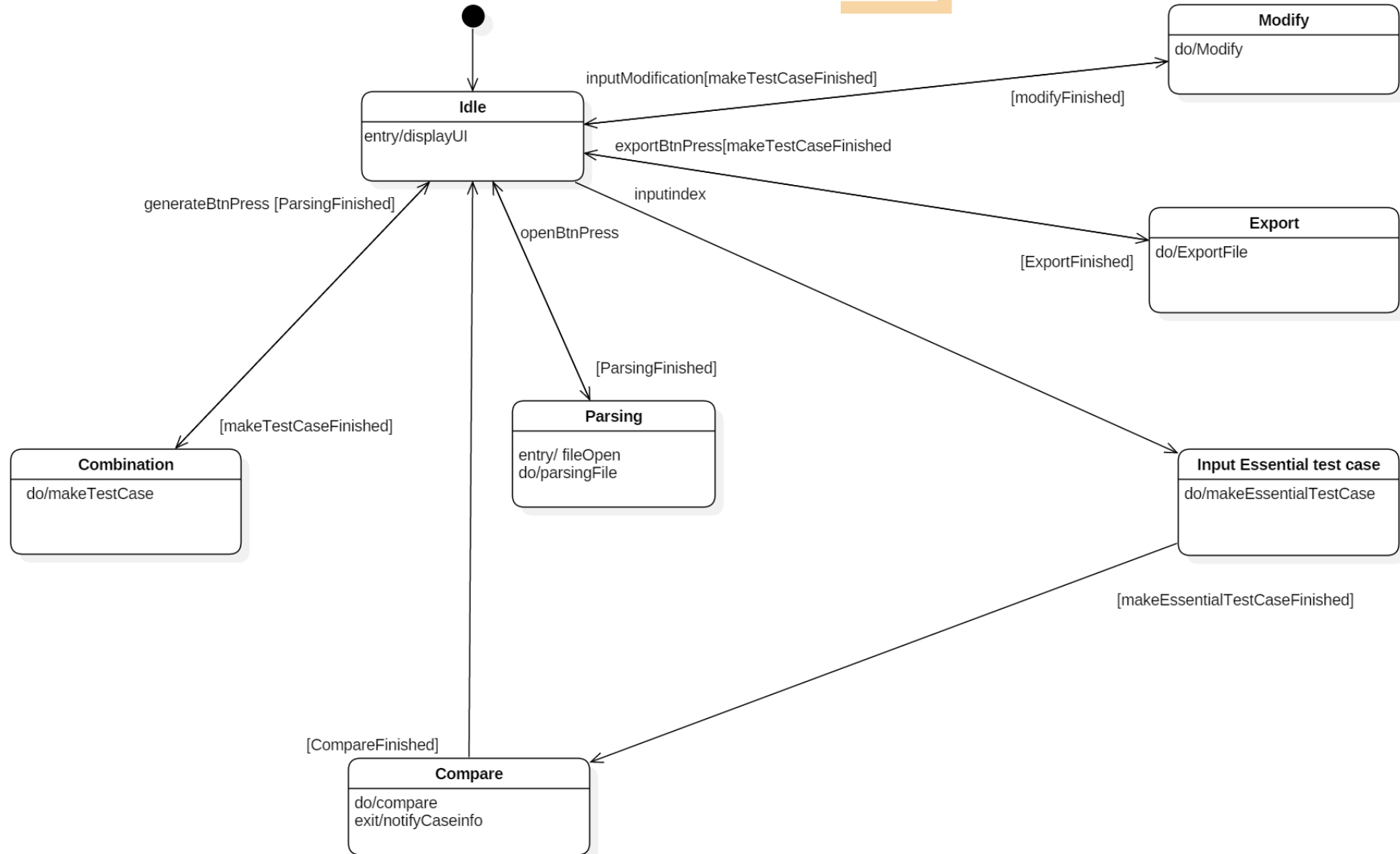


|                               |  |
|-------------------------------|--|
| Use Case                      | Make Final Test Case File  |
| Actor                         | User   |
| Purpose                       | 최종 Test Case 파일을 생성한다.   |
| Overview                      | User가 최종적으로 완성한 Test Case 파일을 생성한다.  |
| Type                          | Primary and Essential  |
| Cross Reference               | Functional Requirement: R 5.1<br>Use Case: Make Test Case  |
| Pre-Requisites                | Test Case가 조합되어 있어야 한다.  |
| UI Widgets                    | Sample UI 1.2<br>Sample UI 1.3   |
| Typical Courses of Events     | (A): Actor (S): System<br>1. (A) User가 [내보내기 버튼]을 누른다..<br>2. (S) [다이얼로그 박스]를 띄운다.<br>3. (A) [다이얼로그 박스]에서 최종 file 을 저장할 디렉토리를 선택하고 파일명을 입력한다.<br>4. (S) 해당 디렉토리에 Test Case List Fil을 .xlsx 형태로 출력한다. |
| Alternative Courses of Events | N/A  |
| Exceptional Courses of Events | N/A.   |



# State Chart Diagram

## Open File



# 4

# Define Reports, UI, Storyboards

## UI

**1** **2** **3**

| Index   | Weight | Value1              | Value1 Weight | Value1 Constraints | Value2          |
|---|--------|---------------------|---------------|--------------------|-----------------|
| 102   | 1      | 소스파일 매우 많이 추가       | 1             | Single             |                 |
| 104   | 1      | 소스 파일이 아닌 파일 추가     | 1             | Error              |                 |
| 110   | 1      | 아무것도 선택하지 않고 파일 삭제  | 1             | Single             |                 |
| 100.200.300.310.320.330.340.350.400.410.420     | 0.7    | 소스 파일 1개 추가         | 0.5           | property AD1       | 클론제킹 버튼 클릭      |
| 101.111.200.300.310.320.330.340.350.400.410.420 | 0.6    | 서로 다른 소스파일 2개 이상 추가 | 0.7           |                    | 파일을 선택한 후 1개 삭제 |
| 101.112.200.300.310.320.330.340.350.400.410.420 | 0.6    | 서로 다른 소스파일 2개 이상 추가 | 0.7           |                    |                 |
| 101.112.200.300.310.320.330.340.350.400.410.420 | 0.6    | 서로 다른 소스파일 2개 이상 추가 | 0.7           |                    |                 |
| 101.112.200.300.310.320.330.340.350.400.410.420 | 0.6    | 서로 다른 소스파일 2개 이상 추가 | 0.7           |                    |                 |
| 101.112.200.300.310.320.330.340.350.400.410.420 | 0.6    | 서로 다른 소스파일 2개 이상 추가 | 0.7           |                    |                 |

| Index   | Weight | Value1        | Value1 Weight | Value1 Constraints | Value2          |
|---|--------|---------------|---------------|--------------------|-----------------|
| 100.200.300.310.320.330.340.350.400.410.420     | 0.7    | 소스 파일 1개 추가   | 0.5           | property AD1       |                 |
| 102.111.200.300.310.320.330.340.350.400.410.420 | 0.6    | 소스파일 매우 많이 추가 | 0.7           | Single             | 파일을 선택한 후 1개 삭제 |
| 100.112.200.300.310.320.330.340.350.400.410.420 | 0.6    | 소스 파일 1개 추가   | 0.7           | property AD1       | 파일을 삭제하지 않는다.   |

Input Essential Test Case

| 번호 | 이름         | 타입    | 설명                |
|----|------------|-------|-------------------|
| 1  | 배경 이미지     | Image | 전체 배경 이미지         |
| 2  | 테스트케이스 창   | Image | 테스트케이스 창 배경 이미지   |
| 3  | 중요테스트케이스 창 | Image | 중요테스트케이스 창 배경 이미지 |



# 4

## Define Reports, UI, Storyboards

### UI

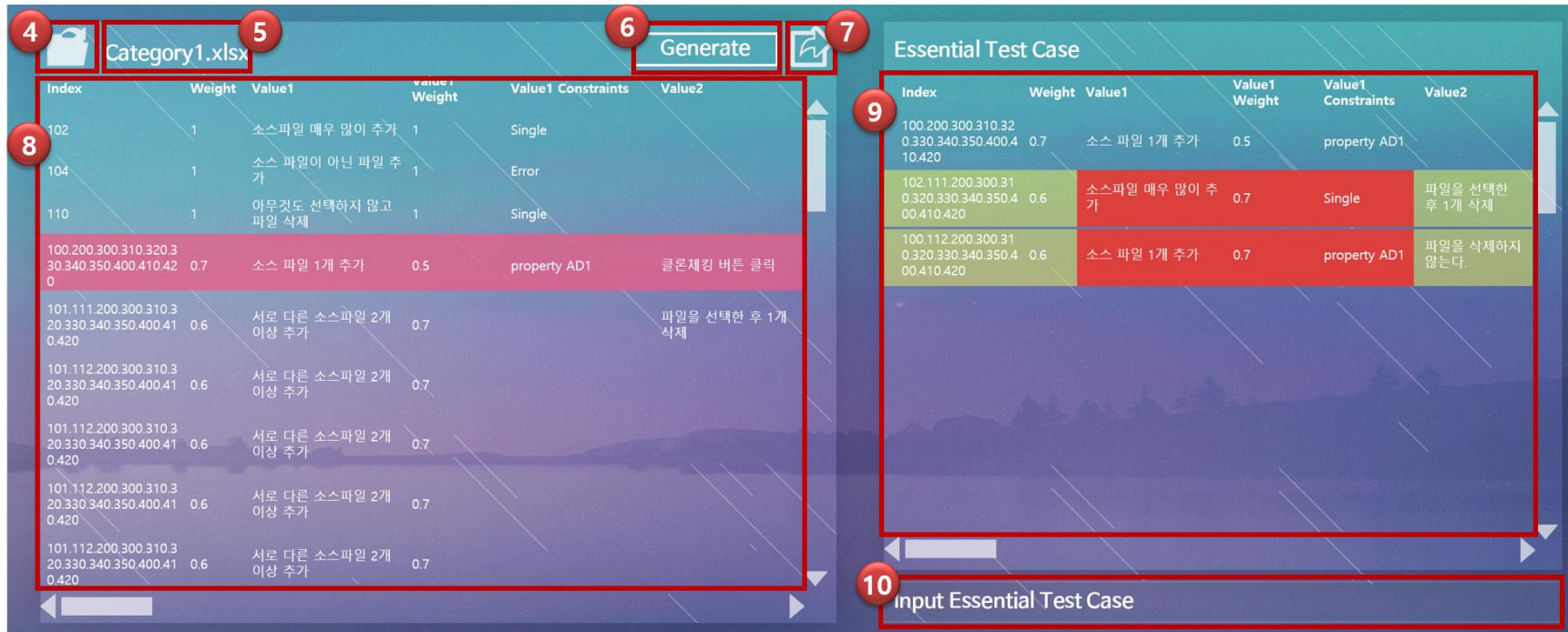
The screenshot displays two data tables side-by-side. The left table, titled 'Category1.xlsx', has columns: Index, Weight, Value1, Value1 Weight, Value1 Constraints, and Value2. The right table, titled 'Essential Test Case', has columns: Index, Weight, Value1, Value1 Weight, Value1 Constraints, and Value2. A 'Generate' button is located between the tables. A red box labeled '4' highlights a file icon, '5' highlights the file name 'Category1.xlsx', '6' highlights the 'Generate' button, '7' highlights a share icon, '8' highlights the left table content, '9' highlights the right table content, and '10' highlights a button labeled 'Input Essential Test Case' at the bottom.

| 번호 | 이름       | 타입     | 설명   |
|----|----------|--------|--|
| 4  | 파일 열기 버튼 | Button | 파일을 열 때 사용하는 버튼<br>버튼을 누르면 파일을 열 수 있는 Dialogue Box가 나타난다 |

# 4

## Define Reports, UI, Storyboards

### UI

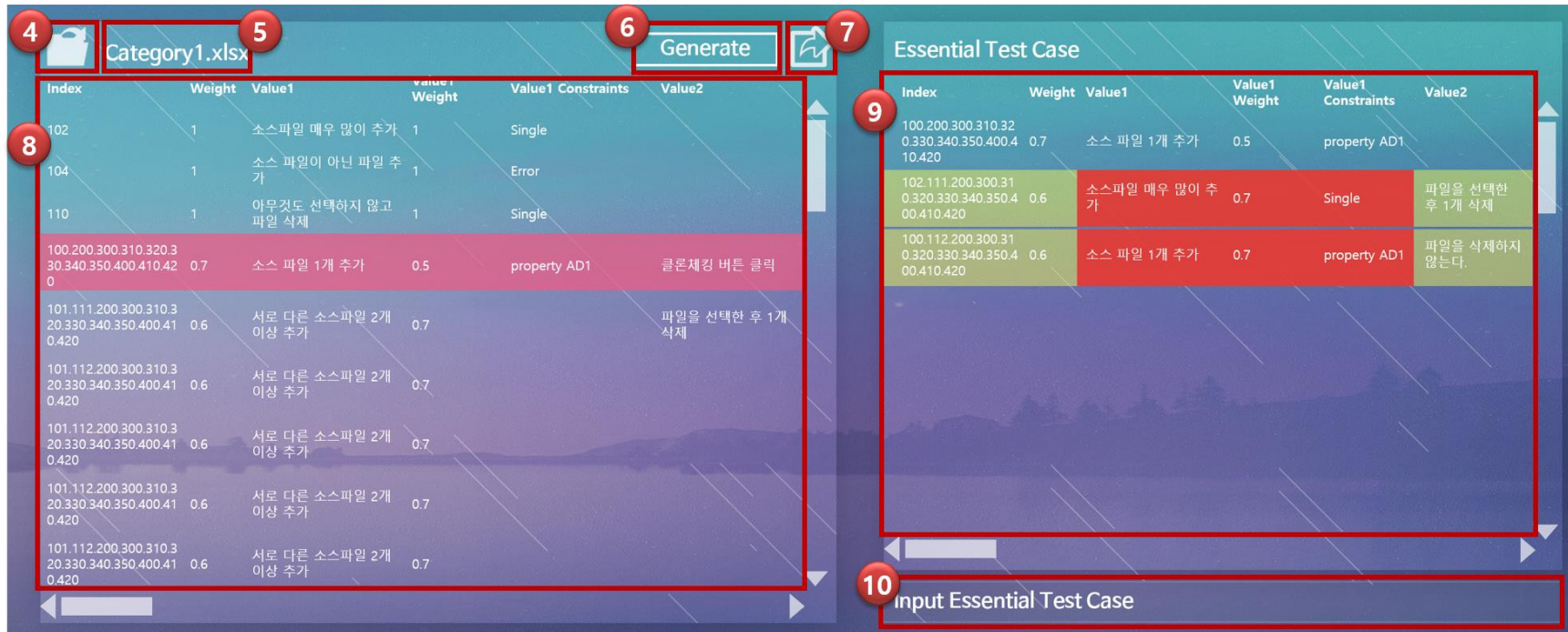


| 번호 | 이름    | 타입     | 설명         |                           |
|----|-------|--------|------------|---------------------------|
| 5  | 파일 이름 | String | 불러온 파일의 이름 |                           |
|    |       |        | 글자수        | 50글자(50글자 초과한 경우 글자크기 축소) |
|    |       |        | Default    | 공백으로 표시                   |

# 4

## Define Reports, UI, Storyboards

### UI

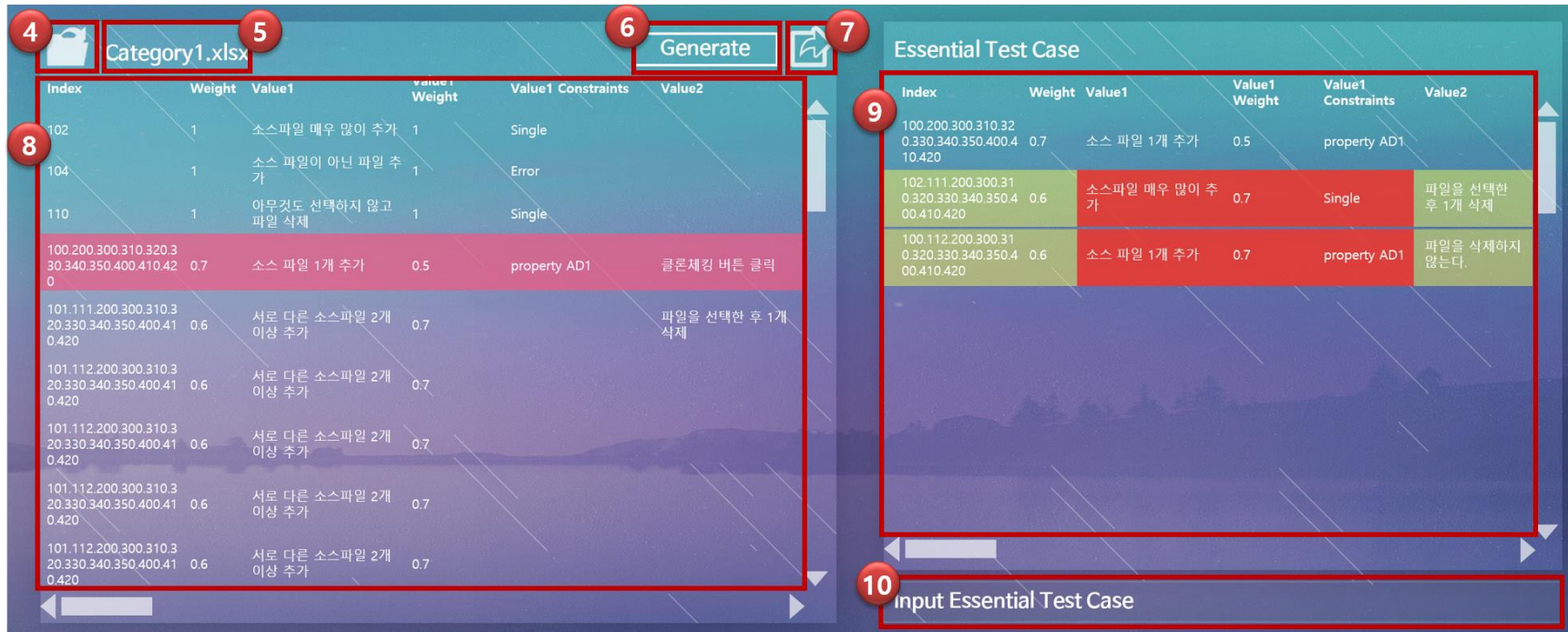


| 번호 | 이름      | 타입     | 설명   |
|----|---------|--------|--|
| 6  | 조합 버튼   | Button | 테스트 케이스를 조합해주는 버튼,<br>버튼을 누르면 테스트 케이스를 조합해서 출력해 준다.            |
| 7  | 내보내기 버튼 | Button | 파일을 내보낼 때 사용하는 버튼<br>버튼을 누르면 파일을 저장 할 수 있는 Dialogue Box가 나타난다. |

# 4

## Define Reports, UI, Storyboards

### UI

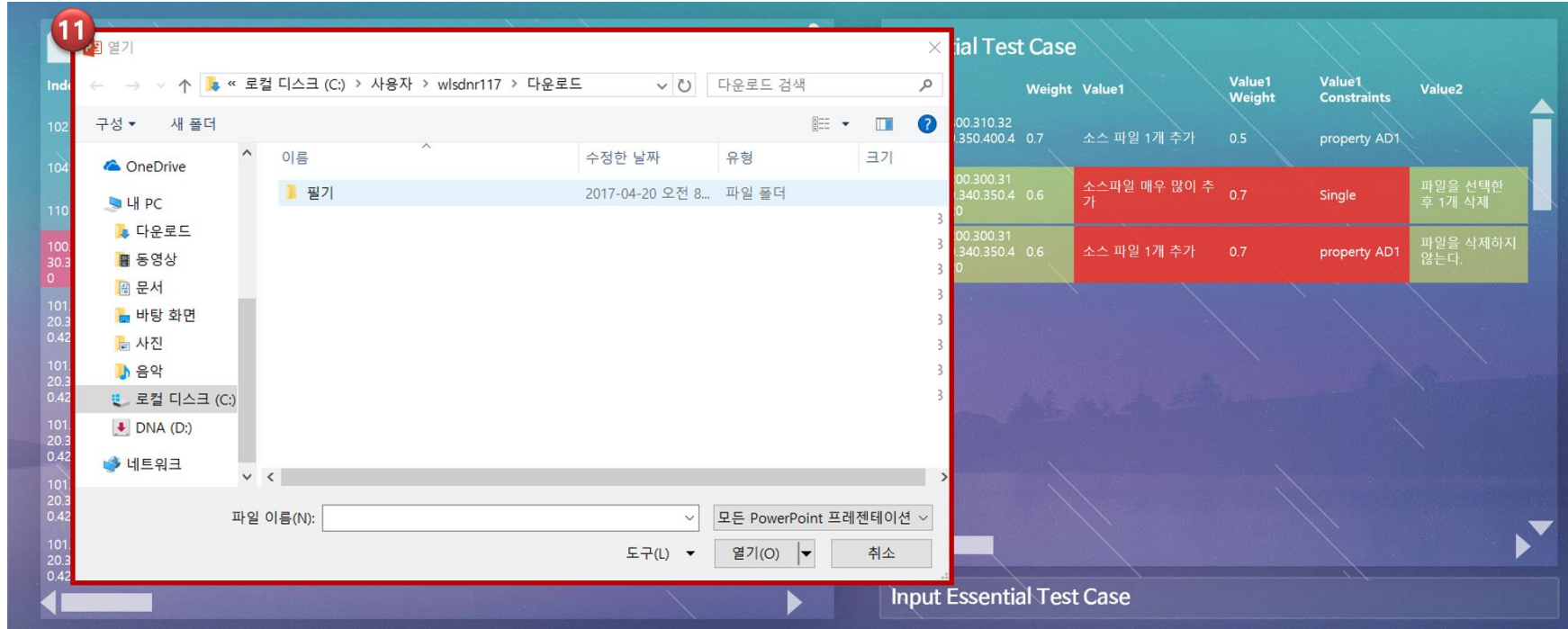


| 번호 | 이름            | 타입                | 설명  |
|----|---------------|-------------------|---|
| 8  | 테스트 케이스 테이블   | Table, ScrollView | 조합된 테스트케이스를 보여주는 테이블                                |
| 9  | 중요 테스트케이스 테이블 | Table, ScrollView | 중요 테스트케이스를 보여주는 테이블                                 |
| 10 | 중요테스트케이스 입력 창 | EditText          | 중요 테스트케이스를 인덱스 형태로 입력하는 창<br>인덱스를 입력하고 엔터키를 누르면 된다. |

# 4

## Define Reports, UI, Storyboards

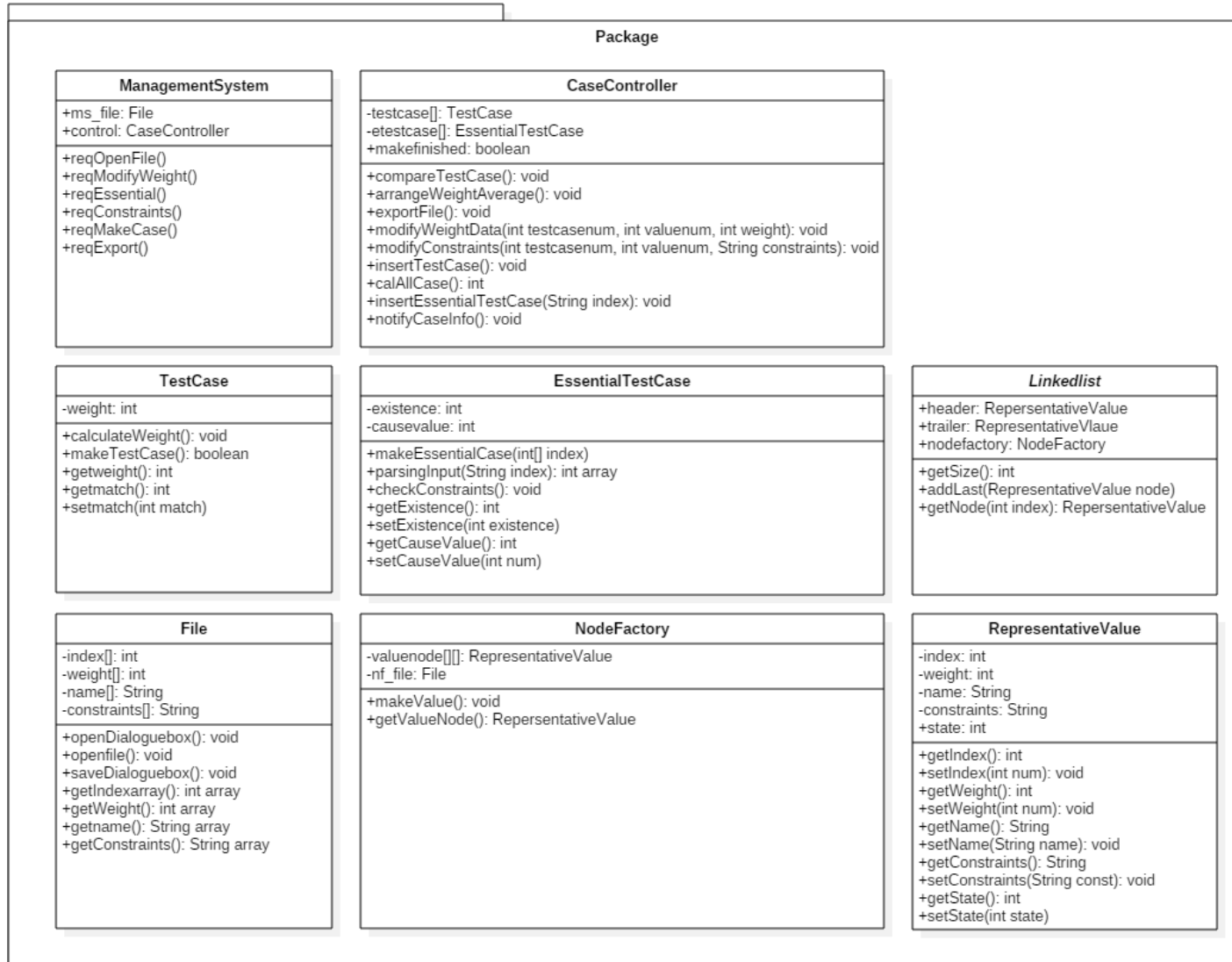
### UI



| 번호 | 이름       | 타입     | 설명                      |
|----|----------|--------|-------------------------|
| 11 | 다이얼로그 박스 | window | 파일 열기와 저장이 가능한 다이얼로그 박스 |



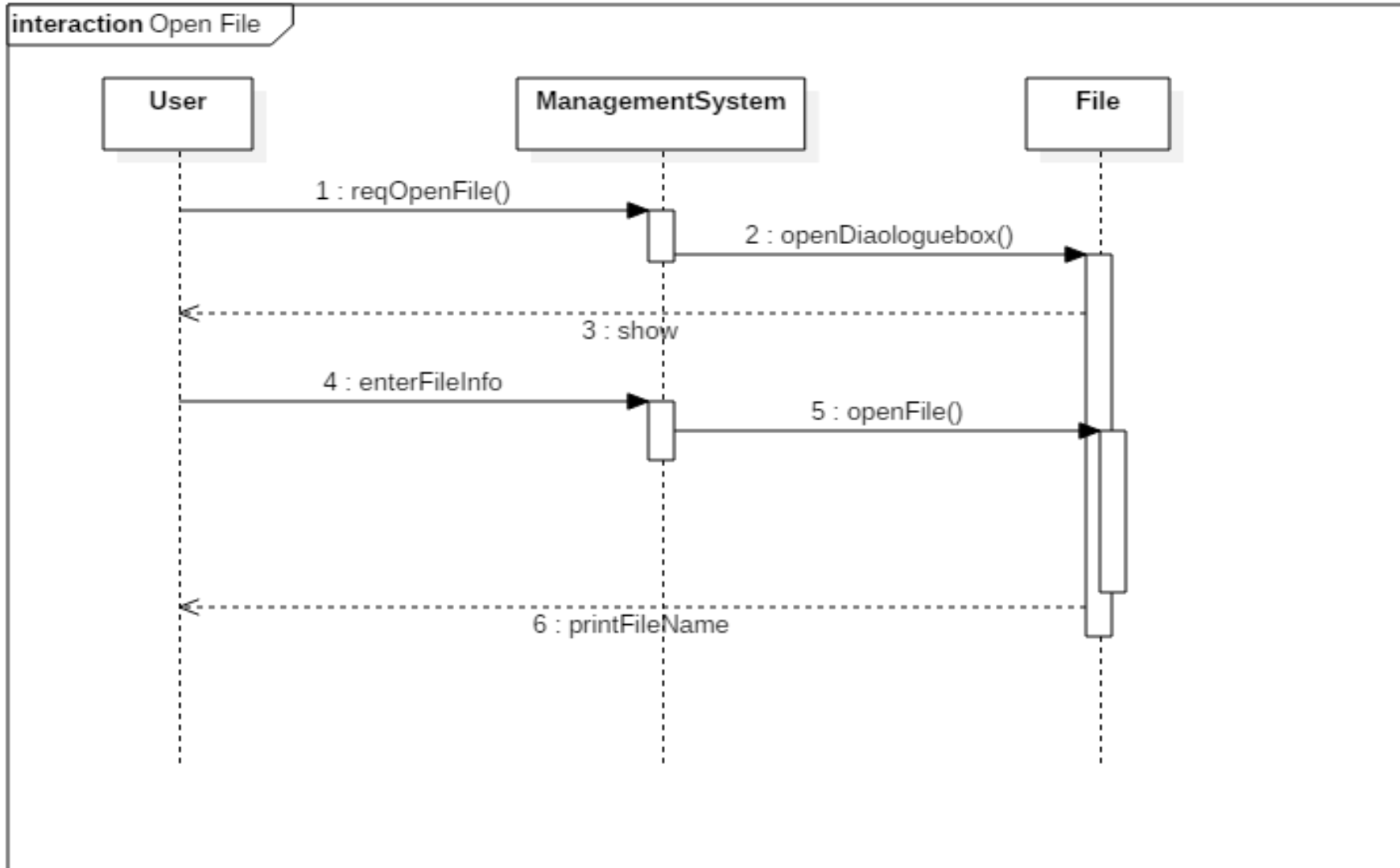
# Refine System Architecture



# 6

## Define System Sequence Diagram

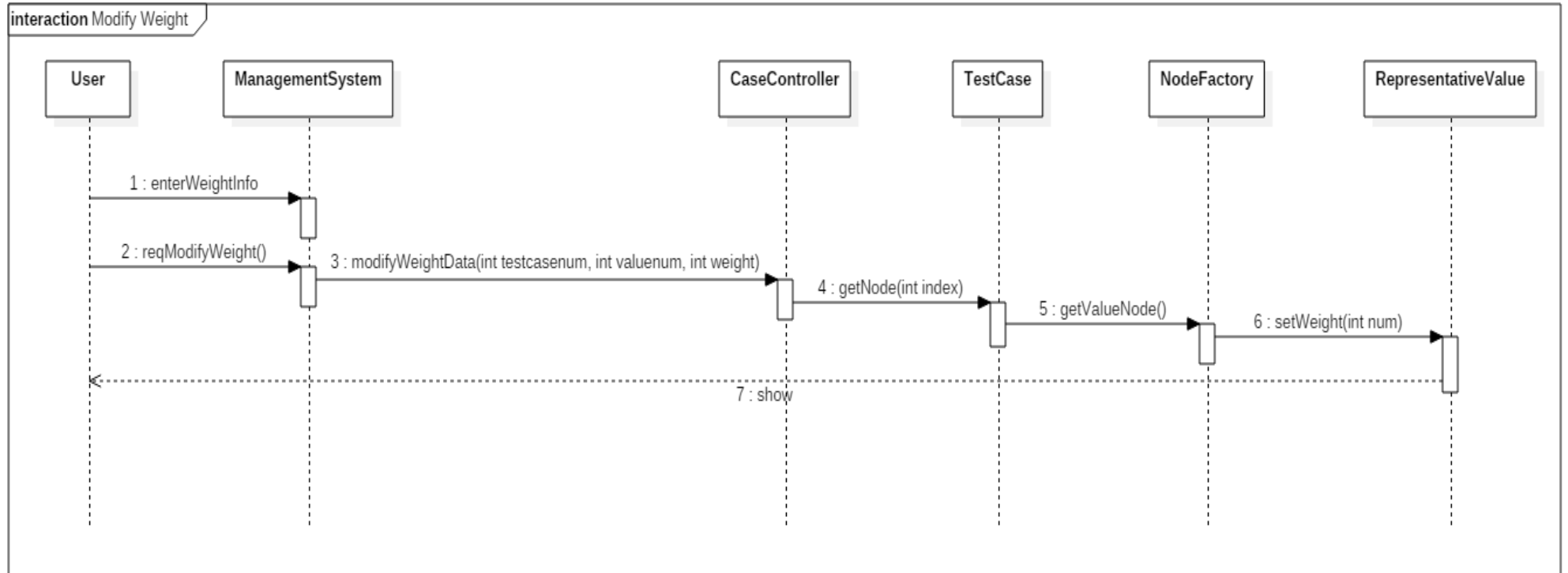
### Open File



# 6

## Define System Sequence Diagram

### Modify Weight

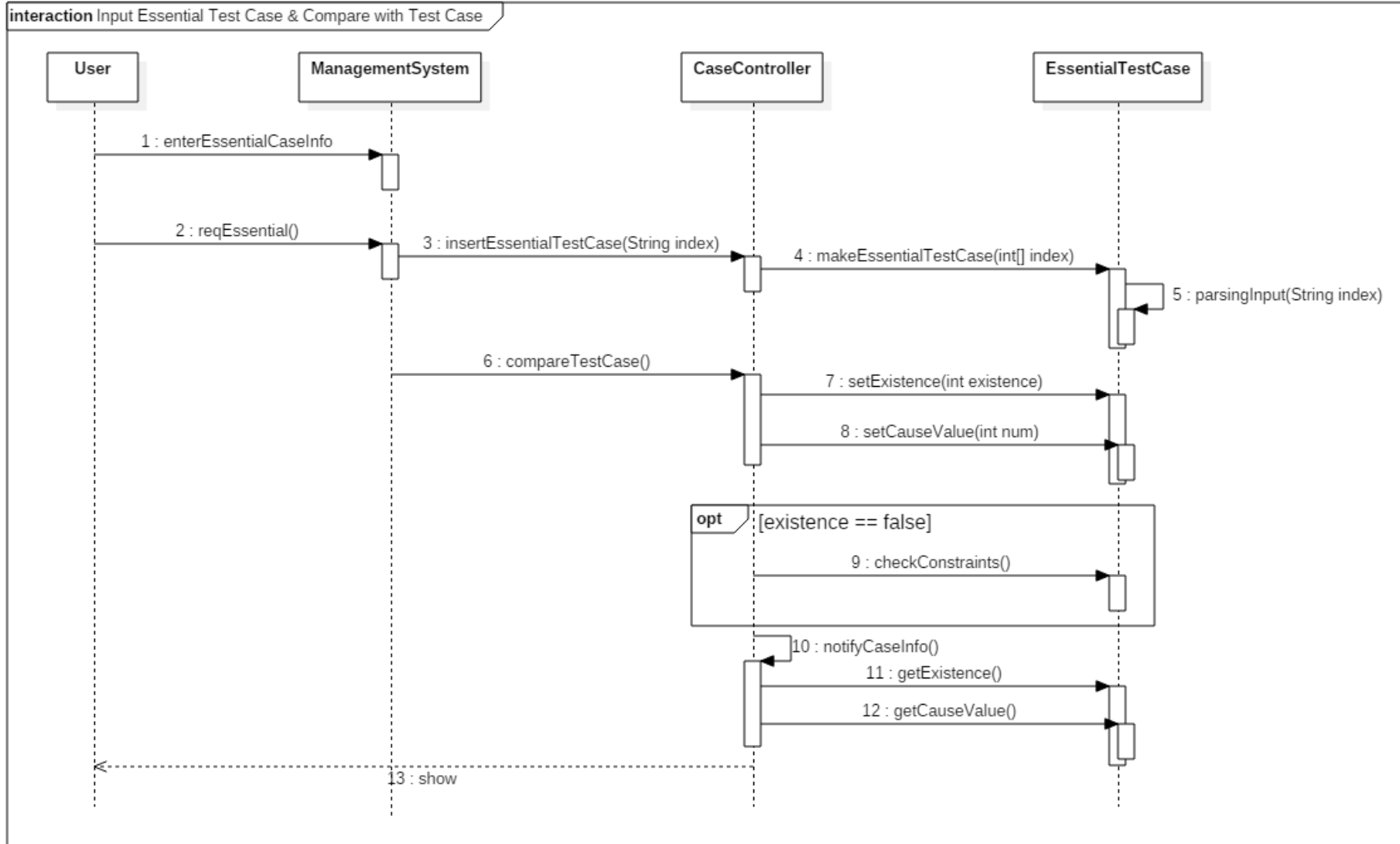




# 6

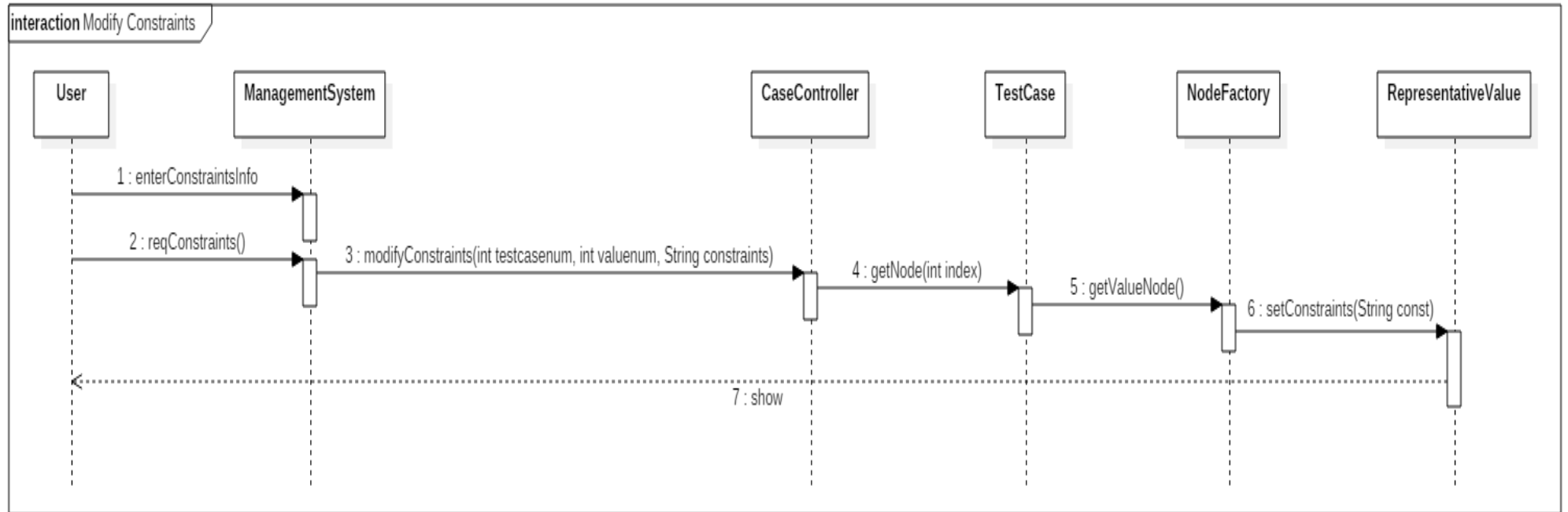
## Define System Sequence Diagram

### Input Essential Test Case & Compare with Test Case



# 6

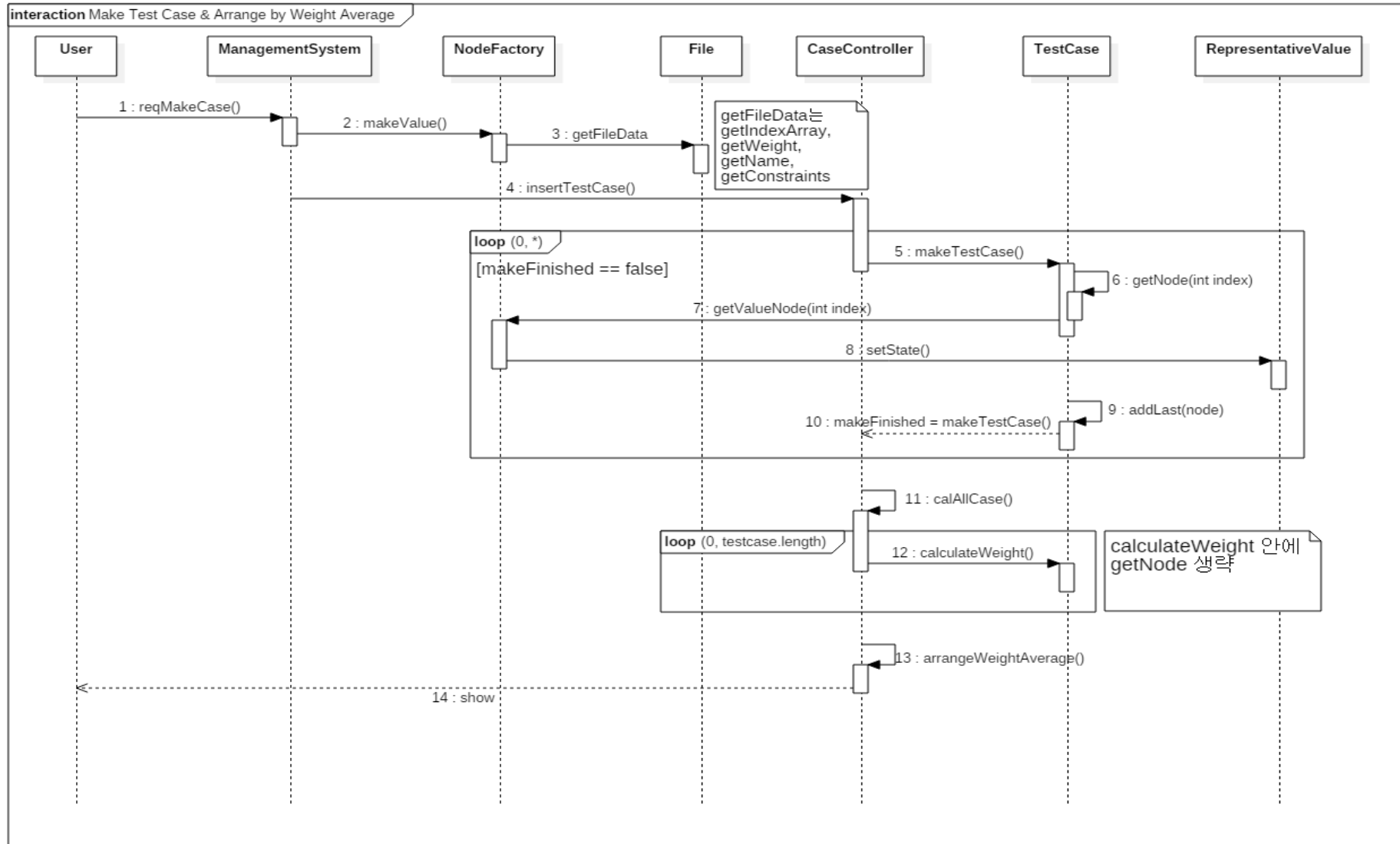
## Define System Sequence Diagram Modify Constraints



# 6

## Define System Sequence Diagram

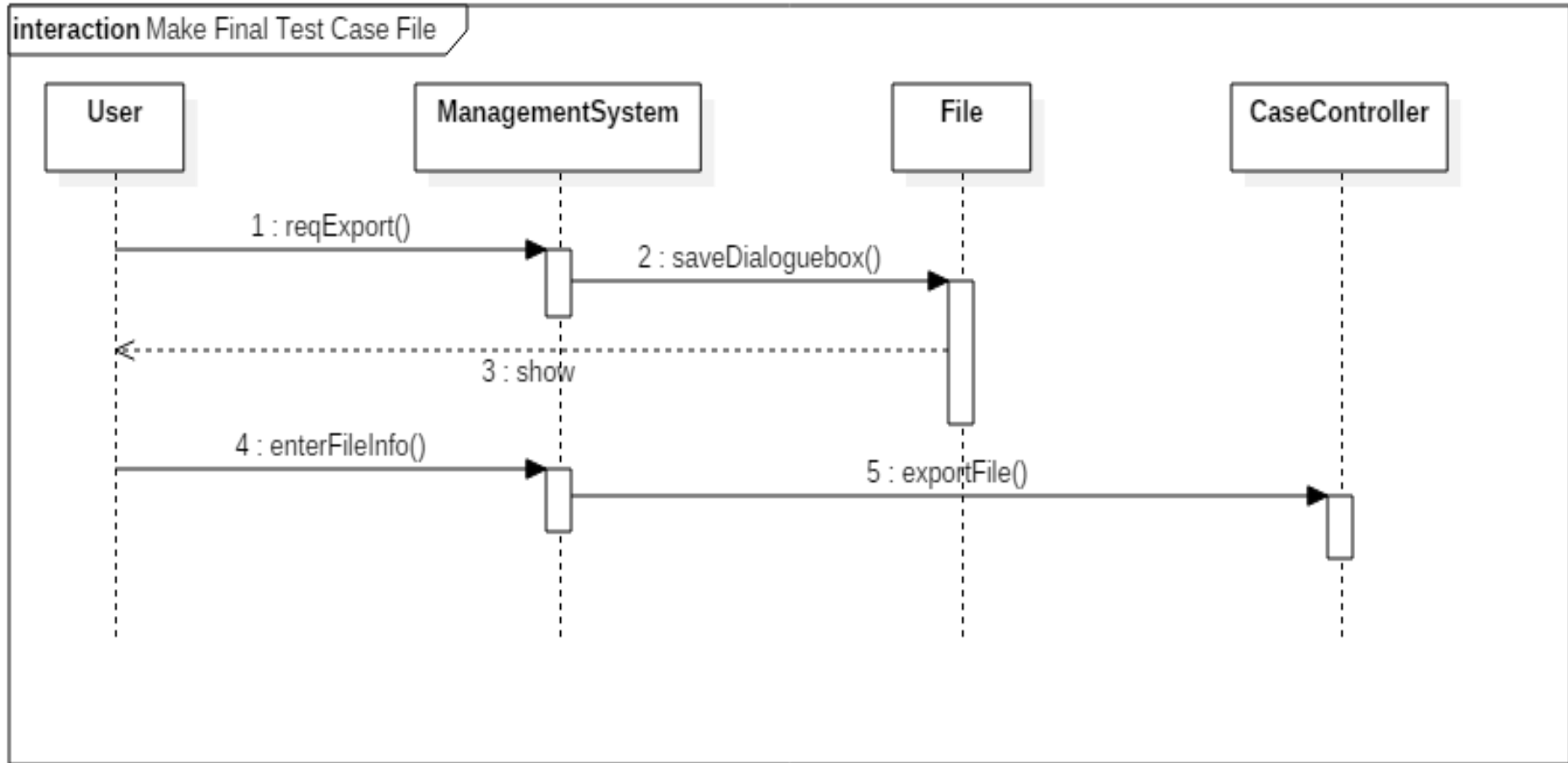
### Make Test Case & Arrange by Weight Average



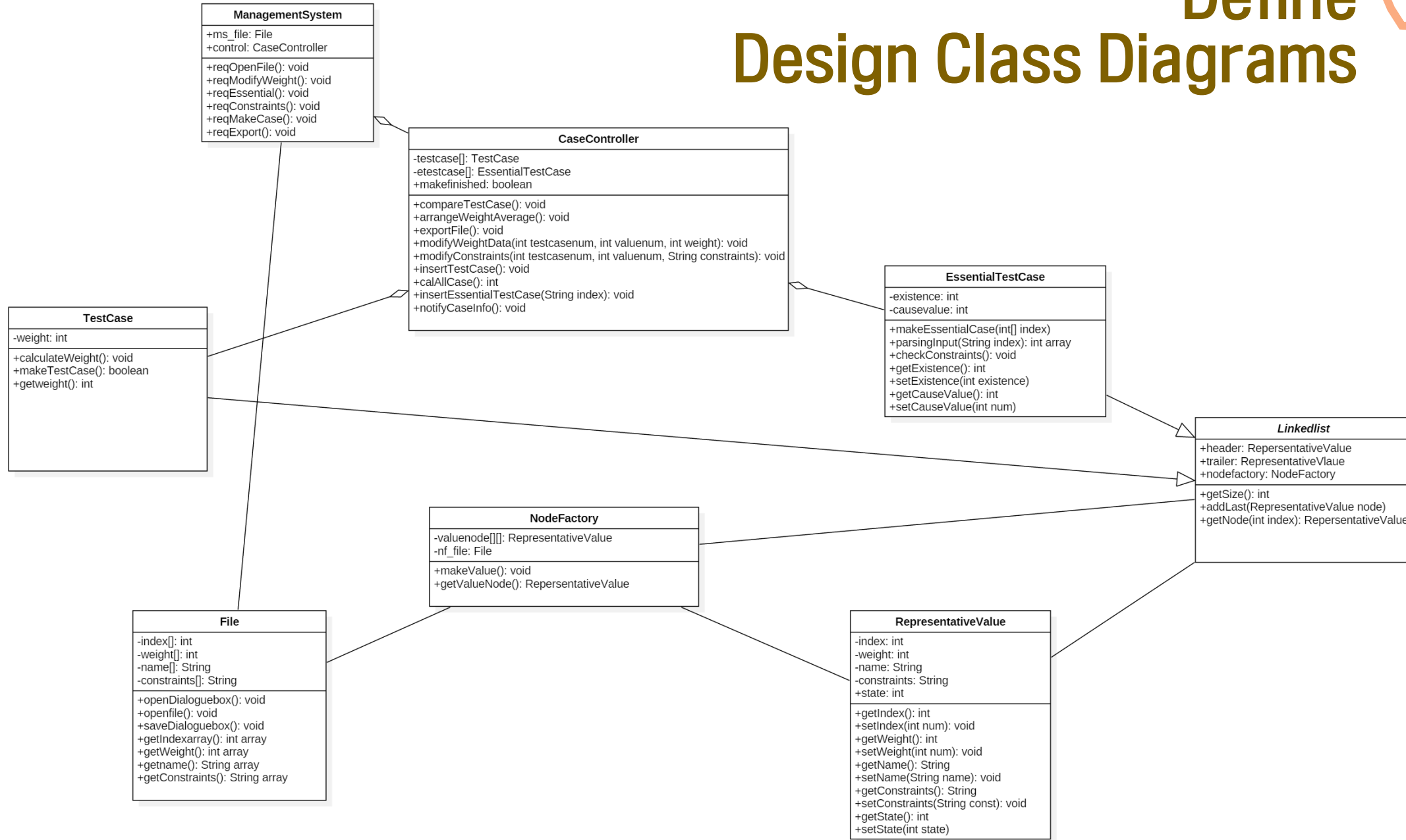
# 6

## Define System Sequence Diagram

Make Final Test Case File



# Define Design Class Diagrams





# Design Traceability Analysis

| Operation in sequencediagram | Operation in interction diagram                                      | Method  | Class               |     |
|------------------------------|--|---|---------------------|-----|
| 1: enterFileInfo()           | reqOpenFile()  | reqOpenFile(): void   | ManagementSystem    |     |
| 2: reqOpenFile()             | OpenDialoguebox()  | reqModifyWeight(): void   |                     |     |
| 3: enterWeightInfo()         | show   | reqEssential(): void  |                     |     |
| 4: reqModifyWeight()         | enterFileInfo  | reqConstraints(): void  | CaseController      |     |
| 5: enterEssentialInfo()      | openFile()   | reqMakeCase(): void   |                     |     |
| 6: reqEssential()            | printlnFileName  | reqExport(): void   |                     |     |
| 7: enterConstraintsInfo()    | enterWeightInfo  | compareTestCase(): void   |                     |     |
| 8: reqConstraints()          | reqModifyWeight()  | arrangeWeightAverage(): void  |                     |     |
| 9: reqMakeCase()             | modifyWeightData(int testcasenum, int value num, int weight)         | modifyWeightData(in int testcasenum, in int valuenum, in int weight): void          |                     |     |
| 10: enterExportInfo()        | getNode(int index)   | exportFile(): void  |                     |     |
| 11: reqExport()              | getValueNode()   | modifyConstraints(in int testcasenum, in int valuenum, in String constraints): void |                     |     |
|                              | setValueNode(int num)  | insertTestCase(): void  |                     |     |
|                              | show   | callAllCase(): int  |                     |     |
|                              | enterEssentialCaseInfo   | insertEssentialTestCase(in String index): void                                      |                     |     |
|                              | reqEssential()   | notifyCaseInfo(): void  | TestCase            |     |
|                              | insertEssentialTestCase(String index)                                | calculateWeight(): void   |                     |     |
|                              | makeEssentialTestCase(int[] index)                                   | makeTestCase(): boolean   |                     |     |
|                              | parseInput(String index)   | getWeight(): int  |                     |     |
|                              | compareTestCase()  | getMatch(): int   |                     |     |
|                              | setExistence(int existence)  | setMatch(in int match)  |                     |     |
|                              | setCauseValue(int num)   | makeEssentialCase(in int[] index)   |                     |     |
|                              | checkConstraints()   | parseInput(in String index): int array  |                     |     |
|                              | notifyCaseInfo()   | checkConstraints(): void  |                     |     |
|                              | getExistence()   | getExistence(): int   |                     |     |
|                              | getCauseValue()  | setExistence(in int existence)  | EssentialTestCase   |     |
|                              | enterConstrainsInfo  | getCauseValue(): int  |                     |     |
|                              | reqConstraints()   | setCauseValue(in int num)   |                     |     |
|                              | modifyConstraints(int testcasenum, int valuenum, String constraints) | getSize(): int  |                     |     |
|                              | getNode(int index)   | addLast(in RepresentativeValue node)  |                     |     |
|                              | getValueNode()   | getNode(in int index): RepresentativeValue  |                     |     |
|                              | setConstraints(String const)   | openDialoguebox(): void   |                     |     |
|                              | reqMakeCase()  | openfile(): void  |                     |     |
|                              | makeValue()  | saveDialoguebox(): void   |                     |     |
|                              | getFileData  | getIndexarray(): int array  |                     |     |
|                              | insertTestCase()   | getWeight(): int array  | File                |     |
|                              | makeTestCase()   | getName(): String array   |                     |     |
|                              | getNode(int index)   | getConstraints(): String array  |                     |     |
|                              | getValueNode(int index)  | makeValue(): void   |                     |     |
|                              | setState()   | getValueNode(): RepresentativeValue   |                     |     |
|                              | addLast(node)  | getIndex(): int   |                     |     |
|                              | makeFinished = makeTestCase()  | setIndex(in int num): void  |                     |     |
|                              | callAllCase()  | getWeight(): int  |                     |     |
|                              | calculateWeight()  | setWeight(in int num): void   |                     |     |
|                              | arrangeWeightAverage()   | getName(): String   |                     |     |
|                              | show   | setName(in String name): void   | RepresentativeValue |     |
|                              | reqExport()  | getConstraints(): String  |                     |     |
|                              | saveDialoguebox()  | setConstraints(in String const): void   |                     |     |
|                              | show   | getState(): int   |                     |     |
|                              | enterExportInfo  | setState(in int state)  |                     |     |
|                              | exportFile()   | input   |                     |     |
|                              |  | ouput   |                     | GUI |

