**소프트웨어 요구사항 명세서  
Software Requirements Specification**

2019.10.27   
전형진

**목차**

1. **프로젝트 명**3
2. **서론**3

(1) 개요

(2) 목적

(3) 대상

(4) 전체 기술

(5) 세부 요구사항

(6) 프로젝트 제약 조건

1. **정보 기술**8
2. 자료사전
3. **초보자 사용 매뉴얼**8
4. **검증 기준**10

**1. 프로젝트명**

음원관리 프로그램

**2. 서론**

**(1) 개요**

본 문서는 음원관리 프로그램에 대한 소프트웨어 요구사항 명세서로서, 고객이 바라는 요구사항을 문서화하여 기술한다.

소프트웨어의 요구사항은 개발하고자 하는 소프트웨어가 갖추었으면 하는 기능 하나하나를 의미하며, 본 문서는 소프트웨어 개발 프로세스의 시작인 소프트웨어의 요구사항을 분석하고 정의하는 단계에서 작성되는 최종 산출물이다.

서론 단락에서는 기술적인 목표, 사업 목표, 시스템 목표. 요구사항 명세서의 목적을 서술한다. 프로젝트의 목적을 기술하고, 본 문서를 읽을 독자를 분류하여 문서를 읽는 방법에 대해 기술한다. 제품의 조망 및 기능, 사용자 특성, 가정 등을 전체 기술하고 외부 인터페이스나 기능 요구사항 등을 세부적으로 기술한 뒤 제약조건을 설명한다.

기능 기술 단락에서는 정보의 흐름을 체계적이며 구체적으로 쉽게 기술하기 위한 단락으로, 자료 흐름도를 보여주고 프로세스 명세서를 통해 자료흐름도의 최하위 프로세스에 대한 기본 기능을 기술한다.

정보기술 단락에서는 시스템의 객체, 객체의 속성, 객체 사이의 연관성을 규명한다. 데이터의 구조 및 그에 수반한 제약 조건들을 설계한개체-관계 다이어그램 ERD (Entity-Relationship- Diagram)를 보여주고, 자료흐름도를 더 자세히 설명한 자료 사전을 기술한다.

초보 사용자 매뉴얼에서는 사용자가 누구인지 규명하고, 사용자 입장의 인터페이스에 대해 기술한다. 입출력 데이터를 중심으로 메뉴 구조, 스크린, 보고서 형식 등이 포함된다.

검증 기준 단락에서는 소프트웨어 요구사항 명세에 대한 표준 IEEE Std. 1012 (IEEE Std. 1012-2012, IEEE Standard for Software Verification & Validation)이 제시하는 요구사항 평가 항목과 IEEE Std. 830 문서에서 제시하고 있는 요구사항 명세 특징 기술에 따라 검증 기준을 제시한다.

**(2) 목적**

이 프로젝트는 잠재적 고객이 음원관리를 하고자 할 때, PC에 설치된 본 프로그램을 통해 음원을 추가하고 관리할 수 있도록 그 기능을 제공하는 것을 목적으로 한다.

**(3) 대상**

본 문서를 읽을 것으로 예상되는 독자는 프로그램을 설계하는 개발자와 프로그램을 사용하는 고객 두 분류로 나뉠 수 있다. 그 중 개발자의 경우 본 문서의 모든 내용을 숙지하여야 한다.

사용자(고객)의 경우 서론, 초보사용자 매뉴얼, 검증 기준 까지만 숙지하여도 괜찮다.

**(4) 전체 기술**

(1) 제품의 조망

① 시스템 인터페이스

- GUI 프로그램: 사용자(고객)로부터 입력된 음원, 가수, 플레이리스트에 대한 정보를 추가하고, 관리(수정, 삭제)한다. 특정 정보에 대한 내용이 수정 혹은 삭제될 시, 각 정보 간의 동기화가 이루어져 사용자에게 편의를 제공한다.

② 사용자 인터페이스

-사용자(고객)는 배포된 프로그램을 통해 원하는 음원의 가수에 대한 정보를 선입력 후, 음원의 구체적인 정보를 입력한다. 해당 정보는 사용자에게 화면으로 출력되며 해당 정보를 수정, 삭제할 수 있다.

-입력한 정보를 파일 출력을 통해 텍스트(.txt) 파일로 개별적으로 저장할 수 있으며, 파일 입력을 통해 외부의 텍스트 파일을 통해 전에 저장한 정보를 불러올 수 있다.

③소프트웨어 인터페이스

- Windows 운영체제와 GUI 프로그램 개발 크로스 플랫폼 프레임워크인 ‘Qt’를 사용해 개발한다. 배포 프로그램은 사용자의 마우스와 키보드를 통해 입력하여 작동되며, 필요한 정보를 화면에 제공할 수 있다.

④ 운영

-정상적인 경우: 프로그램은 음원에 대한 정보를 사용자에게 제공함으로써, 사용자는 쉽게 음원에 대한 정보에 접근할 수 있다.

-종료: 사용자가 프로그램을 종료할 때, 사후에 음원관리 정보를 로딩하기 위해 필요한 정보를 텍스트 파일 형태로 저장할 수 있다.

(2) 제품의 기능

- 고객이 이 제품을 사용하기 위해서는 배포된 프로그램을 다운받아야 한다.

-회원가입 및 로그인 기능을 통해 자신만의 음원 및 플레이리스트 기능을 제공한다.

- 고객은 마우스 클릭과 키보드 입력을 통해 음원 관리 정보를 쉽게 입력하고 디스플레이 모니터를 통해 해당 정보를 직관적으로 얻을 수 있다.

- 가수에 대한 정보를 선 입력한 후, 음원의 구체적인 정보를 입력한다.

-플레이리스트 기능이 제공되어 추가한 음원을 개인의 플레이리스트에 따로 저장할 수 있다.

-플레이리스트에서는 재생 순서를 변경하거나 해당 리스트에서 음원을 제외시킬 수 있다.

-메인 창에서 가수의 이름을 클릭할 시 가수에 대한 구체적인 정보를 출력한다.

(3) 사용자 특성

- 배포 프로그램을 사용하는데 있어서 편리한 Graphic Interface를 제공하기 위해 직관적인 사용이 가능하도록 설계한다.

**(5) 세부 요구사항**

(1) 외부 인터페이스

사용자로부터의 입력: 음원 정보, 가수 정보, 플레이리스트 정보

1. 기능 요구사항

* 메인 윈도우: 사용자는 음원에 대한 정보를 메인 윈도우의 음원 테이블을 통해 제공받을 수 있다. 곡 추가, 가수 추가, 플레이리스트, 전체 삭제, 선택 삭제, 선택 수정, 프로그램 종료 버튼을 통해 추가적인 기능을 제공한다. 윈도우 상단에 위치한 메뉴바는 파일 입출력을 위한 기능을 제공한다. 음원 검색을 통해 사용자가 원하는 정보를 출력한다.
* 곡 추가 버튼과 가수 추가 버튼을 통해 별도의 위젯을 통해 음원 정보와 가수 정보를 등록할 수 있다.
* 가수 정보 등록: 사용자는 가수 추가 위젯을 통해 가수의 이름, 생일, 성별, 데뷔날짜를 포함한 가수에 대한 정보를 입력한다. 추가적으로 가수의 사진을 업로드가 필요할 시 옵션으로 추가할 수 있다. 나이는 생일을 통해 자동으로 입력되며 만 나이로 저장 및 출력된다.
* 음원 정보 등록: 사용자는 음원 추가 위젯을 통해 음원의 고유번호와 노래 제목, 가수 이름, 작곡가, 종류(노래 혹은 연주곡), 장르를 입력한다. 추가적으로 음원의 앨범 커버와 가사 정보를 옵션으로 추가 입력할 수 있다.
* 플레이리스트 등록: 메인 윈도우에 출력된 음원 테이블에서 각 음원정보에 위치한 플레이리스트 추가 버튼을 클릭하여 플레이리스트에 추가할 수 있다. 플레이리스트 출력을 위해서는 메인 윈도우에 위치한 플레이리스트 버튼을 클릭하여 해당 정보를 확인할 수 있다.
* 가사 정보 출력: 가사 정보는 테이블에 위치한 가사정보 출력 버튼을 클릭해 별도의 가사 위젯을 통해 제공받는다. 팝업 가사창에는 해당 음원의 가사와 더불어 앨범커버와 가수 이름, 음원 제목을 함께 출력하여 사용자에게 제공한다.
* 가수 정보 출력: 메인 윈도우의 테이블에 출력된 음원의 가수를 더블 클릭하면 해당 가수에 대한 정보를 출력한다.

(3) 성능 요구사항

* 사용자는 Windows 운영체제가 설치된 PC를 통해 배포 프로그램을 실행하여 원하는 정보를 입력하고 출력 받을 수 있어야 하며, 별도의 네트워크 데이터 통신 없이도 정상적인 프로그램 실행이 원활해야 한다.
* 고객이 별도의 사용 설명서가 없이 직관적으로 프로그램을 사용할 수 있도록 직관적인 그래픽 디자인을 목표로 설계해야 한다.

**(6) 프로젝트 제약 조건**

* 사용자가 만족할 정도로 편리한 UI와 간편한 접근성으로 프로그램의 이용에 불편이 없어야 한다. 또한 각 자료구조 간의 데이터 동기화가 원활하게 이루어져야 한다.
* 사용자가 정보를 자료 구조에 입력하거나, 받아오는 과정에서 불편함이 없어야 한다. 입출력을 기다리는 대기시간이 적어야 하며, 사용자에게 필요한 정보만을 빠르게 필터링할 수 있어야 한다.
* 고객의 연령대를 고려하여 어떠한 연령대도 접근이 쉬운 간단하고 직관적인 GUI 프로그램을 목표로 해야 한다.

**3. 정보 기술**

**(1) 자료 사전**

1. 음원= ID + 제목 + 가수 이름+ 작곡가 + 타입 + 장르 + 가사 + 앨범커버

2. 가수 = 가수 이름 + 출생일자 + 나이 + 성별 + 데뷔날짜 + 가수사진 + 발매곡

2.1 발매곡 = ID + 제목

3. 플레이리스트 = ID + 재생횟수 + 추가순서

**4. 초보 사용자 매뉴얼**

1. 프로그램(고객용)

● 실행 및 회원 정보 입력

1. 배포 프로그램을 실행한다.

2. 회원일 경우 ID와 비밀번호를 입력한다.

3. 비회원일 경우 회원가입 버튼을 클릭하여 ID와 비밀번호를 설정한 후 로그인한다.

● 음원 리스트 관리

1. 어플리케이션 실행 후 최초 로그인 시, 메인 음원 리스트가 테이블 형태로 전시된다.

2. 음원 추가를 원할 경우, 우선적으로 좌측 하단의 가수 추가 버튼을 클릭하여 가수 정보를 새로 추가한다.

3.가수 추가를 완료한 후 가수 추가 버튼 옆에 위치한 곡 추가 버튼을 통해 새로운 음원을 추가한다.

4. 음원삭제는 전체삭제, 선택삭제가 있다. 전체삭제는 오른쪽 하단에 위치한 버튼을 클릭한다. 선택삭제의 경우, 테이블에 전시된 곡의 가장 왼쪽에 위치한 체크박스를 체크한 후 삭제한다. (복수선택 가능)

5. 음원수정의 경우 테이블에 전시된 곡의 가장 오른쪽에 위치한 체크박스를 체크한 후 선택 수정버튼을 수정한다. (복수선택 불가능)

6. 플레이리스트에 추가를 원할 시 테이블에 위치한 가장 오른쪽 Add 버튼을 클릭한다.

7. 가사출력은 테이블에 위치한 Show Lyrics 버튼을 클릭한다.

● 플레이리스트 관리

1. 플레이리스트 출력 및 관리를 원할 경우, 우측 상단에 위치한 PlayList 버튼을 클릭한다.

2. 대부분의 기능은 메인 음원 리스트와 그 기능이 동일하거나 비슷하다.

3. 플레이리스트 삭제에는 음원삭제는 전체삭제, 선택삭제가 있다. 전체삭제는 좌측 상단에 위치한 버튼을 클릭한다. 선택삭제의 경우, 테이블에 전시된 곡의 가장 왼쪽에 위치한 체크박스를 체크한 후 삭제한다. (복수선택 가능)

4. 플레이리스트의 순서를 수정하고 싶을 시, 두 개의 곡을 선택한 후 오른쪽 하단의 순서 바꿈 버튼을 클릭한다.

● 정보 저장

1. 설정한 모든 정보를 저장하고 싶을 때는 왼쪽 상단의 메뉴 중 file을 클릭한다.

2. 하단에 전시되는 save를 클릭하여 현재까지 설정한 모든 정보를 저장한다.

**5. 검증 기준**

검증 기준은 소프트웨어 요구사항 명세에 대한 표준 IEEE Std. 1012 (IEEE Std. 1012-2012, IEEE Standard for Software Verification & Validation)이 제시하고 있는 요구사항 평가 항목과 IEEE Std. 830 문서에서 제시하고 있는 특징 기술에 따른다.

* **IEEE Std 830의 좋은 요구사항 명세서가 지녀야 할 속성**

|  |  |
| --- | --- |
| 정확성 (Correct) | 명세서 상의 모든 요구사항이 소프트웨어에서 만족되고, 제품이 원래 요구사항을 만족시킬 수 있다면, 명세서는 정확성을 갖는다고 할 수 있다. |
| 명백성 (Unambiguous) | 명세서의 기술은 명확해야 하며, 애매모호한 부분이 있어서는 안 된다. |
| 완전성 (Complete) | 기능, 성능, 제약사항, 속성, 인터페이스 요구사항 등 필요한 모든 정보가 누락되지 않고 모두 기술되어 있다면, 명세서는 완전성을 갖는다고 할 수 있다. |
| 일관성 (Consistent) | 같은 요구사항 정의 내에서 요구사항들 간에 모순이 없고, 다른 소프트웨어 요구사항이나 상위 요구사항(시스템 요구사항)과 충돌되는 부분 없이 정의되어 있다면, 명세서는 일관성을 갖는다고 할 수 있다. |
| 중요도 안정 우선순위 (Ranked for importanceand /or stability) | 모든 요구사항의 중요도는 동등하지 않으며, 소프트웨어의 동작에 치명적인 사항이 존재하기도 하므로 중요도 혹은 우선순위에 따라 작성되어야 한다. |
| 검증 가능성 (Verifiable) | 기술된 요구사항이 성공이나 실패에 대한 평가 기준이 될 수 있어야 한다. |
| 수정 가능성 (Modifiable) | 기술된 요구사항들의 구조와 형태를 유지하며 다른 요구사항에 영향을 최소화하며 변경할 수 있도록 기술된다면, 명세서는 수정 가능성을 갖는다고 할 수 있다. |
| 추적성 (Traceable) | 기술된 요구사항이 관련된 문서, 근거 혹은 산출물이 명확하게 참조 가능하다면, 명세서는 추적 가능성을 갖는다고 할 수 있다. |

* **IEEE Std. 1012의 요구사항 명세서 체크 리스트**

|  |  |  |
| --- | --- | --- |
| 요구사항 평가 (Requirement Evaluation) | 정확성 (Correctness) | - 소프트웨어 요구사항이 시스템의 가정, 제약사항, 동작 환경 내에서 소프트웨어  에 할당된 시스템 요구사항을 만족하는지 확인한다.  - 소프트웨어 요구사항이 표준, 참조, 규격, 정책, 물리적 법칙, 비즈니스 등을 따르는지 확인한다.  - 도메인 지식, 프로토타입 결과, 공학 원리 등과 연결된 로직과 데이터 흐름상태의 흐름 및 변화를 입증한다.  - 데이터의 흐름과 제어가 기능과 성능 요구사항을 만족하는지 확인한다.  - 데이터 사용과 형식의 유효성을 확인한다. |
| 일관성 (Consistency) | - 모든 용어와 컨셉들이 일관성 있게 작성되어 있는지 확인한다.  - 기능 상호작용과 가정들이 일관성 있고, 시스템 요구사항을 만족하는지 확인한다.  - 소프트웨어 요구사항과 시스템 요구사항을 포함한 외부적 이 내부적으로 일관성을 갖는지 확인한다. |
| 완전성 (Completeness) | - 시스템의 가정과 제약사항 내에서 프로세스 정의 및 스케쥴링, 하드웨어/소프트웨어 사용자 인터페이스의 명세, 성능 규격, 치명적인 구성 데이터, 시스템 장치 /소프트웨어 제어 등이 SRS에 명시되어 있는지 확인한다.  - SRS가 명시된 구성 관리 절차를 만족하는지 확인한다. |
| 정확도 (Accuracy) | - 시스템 환경에서 로직 및 인터페이스의 정밀도가 요구사항을 만족하는지 확인한다.  - 만들어진 물리적 현상들이 시스템 요구사항과 물리적인 법칙을 따르는지 확인한다. |
| 판독성 (Readability) | - 문서가 알기 쉽고, 이해하기 쉬우며, 모호하지 않고 명백한 것인지 입증한다.  - 문서가 두문자어, 약어 등에 대한 정의가 명시되어 있는지 확인한다. |
| 시험성 (Testability) | - SRS의 요구사항을 입증하기 위한 객관적인 시험 기준이 되는지 확인한다. |
| 인터페이스 분석 (Interface Analysis) | 정확성 (Correctness) | - 외부 시스템, 내부 시스템, 소프트웨어 인터페이스 요구사항의 유효성을 확인한다. |
| 일관성 (Consistency) | - 인터페이스가 SRS와 IRS 사이의 일관성을 갖는지 확인한다. |
| 완전성 (Completeness) | - 각각의 인터페이스가 데이터 형식과 timing. bandwidth, accuracy, safety, security 등의 성능 기준을 포함하는지 확인한다. |
| 정확도 (Accuracy) | - 각각의 인터페이스가 정확도를 가진 정보를 제공하는지 확인한다. |
| 시험성 (Testability) | - 인터페이스 요구사항을 입증할 수 있는 객관적인 시험 기준이 되는지 확인한다. |
| 추적성 분석 (Traceability Analysis) | 정확성 (Correctness) | - 각각의 소프트웨어 요구사항과 시스템 요구사항 사이의 관계가 정확성을 갖는 것을 확인한다. |
| 일관성 (Consistency) | - 소프트웨어와 요구사항과 시스템 요구사항 사이의 관계가 일관된 수준으로 명시되어 있는 것을 확인한다. |
| 완전성 (Completeness) | - 모든 소프트웨어 요구사항이 시스템 요구사항에서 추적 가능한 것을 확인한다.  - 소프트웨어와 관련된 모든 시스템 요구사항이 소프트웨어 요구사항에서 추적 가능한 것을 확인한다. |
| 정확도 (Accuracy) | - 시스템 성능과 동작하는 특성들이 추적된 소프트웨어 요구사항에 의해 명시되었음을 입증한다. |