자료구조 프로젝트

<음원 관리 프로그램>

**음원관리 프로그램**

우리는 음악을 듣기 위해 MP3 player 또는 PC에 설치된 음악 플레이어를 이용했고 스마트폰 등 여러 전자기기들이 등장하면서는 어플을 이용하는 추세이다. 여러 음악플레이어가 나왔지만 각각 장단점이 존재한다. 명세서에 설명될 플레이어는 원래의 기본기능에 충실하되 불편한 기능을 개선해서 만들어 질 것이다.

**기능\_사용자**

1. 전체재생, 반복재생, 한 곡 재생, 셔플 재생 가능
2. 원하는 곡을 다운받아 추가하거나 삭제할 수 있다.
3. 원하는 곡을 플레이 리스트 내에서 검색 가능하다.
4. 자주 들었던 곡들을 추천해 준다
5. 나만의 플레이 리스트를 만들 수 있고 그 안에서 곡의 추가 및 삭제가 가능하다.
6. 재생목록과 플레이리스트는 별개이다.
7. 플레이리스트에서 재생목록으로 음원을 가져와서 추가 가능하다
8. 예약시간에 맞춰 자동 종료된다.
9. 곡에 따른 가사를 표시해준다.
10. 최근 추가된 항목들을 보여준다.
11. 들은지 오래 된 곡 들을 보여준다.

**기능\_관리자**

1. 사용자가 할 수 있는 모든 기능들은 관리자 또한 똑같이 가능하다.
2. 음원에 대한 삭제가 가능하다.
3. 멤버 삭제가 가능하다

**시나리오**

실습 때 했던 틀을 기본으로 구현할 것이다.

로그인 또는 관리자로 로그인

ID, PW 입력

주요 기능은 다음과 같다.(사용자)

1. 곡 추가

* 곡 명과 가수를 작성해 기본 List에 추가 할 수 있다.

1. 모든 곡 표시

* 기본 List에 있는 곡에 대한 정보들을 표시해 준다.

1. 곡 검색

* 기본 List에 있는 곡을 곡명 또는 가수 명으로 검색 할 수 있다.

1. 곡 삭제

* 기본 List에 있는 곡을 삭제할 수 있다. 이때는 다른 플레이 리스트에 있는 동일한 곡도 삭제되어야 한다.

1. 플레이 리스트 생성

* 사용자는 다른 플레이 리스트를 생성할 수 있고 기본 List에 있는 곡들 중 마음에 드는 곡들을 모아 따로 플레이 리스트에 추가할 수 있다. 이 때는 포인터를 이용한다.

1. 사용자 현황

* 사용자가 자주 들었던 곡들, 오랫동안 듣지 않은 곡들을 정리해서 보여준다.

1. 예약종료

* 타이머 설정가능. 시간이 다 되면 자동으로 로그아웃 된다.

1. 로그아웃

- 로그아웃.

관리자기능

1. 사용자 정보보기

* 각 사용자들의 재생목록들을 볼 수 있고 리스트에 있는 곡들에 대해 삭제가 가능하다.
* 사용자 삭제가 가능하다.

1. 로그아웃

* 로그아웃

**ADT**

**Class Application**

<Member functions>

* Void Run();//프로그램 실행함수
* int GetMainCommand();//첫화면
* int GetMemberComman();//사용자기준 로그인시 첫 화면
* int GetManagerCommand();//관리자 기준 로그인시 첫 화면
* void SignUp();//가입하기
* void SearchMember();//관리자 기준 멤버 검색
* void Login();//로그인

=============Manager==========================

* Void PrintUserInfo();//사용자 정보 출력
* Void DelUse();//원하는 사용자 삭제

<Member variables>

* SortedLinkedList<User> m\_MemberList//이용자 리스트;
* Int m\_menu\_num;

**Class User**

<Member functions >

* void AddSong();// 곡을 추가한다.
* void DelSong();//원하는 곡을 삭제한다 또는 전체곡
* void PrintSongs();//가지고 있는 곡들을 보여준다
* void SearchSong();//List내의 곡을 검색한다.
* void AddPlayList();//Playlist를 추가한다
* void UpdataData();//음원에대해 수정할 사항이 있을시 업데이트한다.
* void UsersFavor();//사용자의 곡 선호도에 따라 음원 리스트를 보여준다
* void Timer();//타이머설정 함수
* void LogOut();//로그아웃
* void MemberLeave();//탈퇴하기

<Member variables >

* Int m\_id;
* String m\_name;
* SortedLinkedList<Music> m\_musicList;

**Class Music**

<Member functions >

* void Add(Itemtype data);//데이터 추가
* int Delete(Itemtype data);//데이터 삭제
* void PrintAll();
* int Search(Itemtype data);//데이터 검색
* void GetData(Itemtype data);// 데이터 가져오기
* void SetData(ItemType Data);//데이터 업데이트
* void MakePlayList();
* void Favor();
* void ResetList();//리스트 초기화 함수

<Member variables >

* LinkedList <Itemtype> item;
* Int m\_length;

**Class Itemtype**

<Member functions >

* Void SetRecord();//레코드 설정
* Void SetName();//이름 설정하는 함수
* Void SetSinger();//음악가 설정하는 함수
* String GetName();//이름 반환하는 함수
* String GetSinger();//음악가 반환하는 함수
* Int GetCnt();//Cnt가져오는 함수

<Member variables>

* String name;
* String singer;
* Int cnt;
* Member functions과 Member variables은 완성된 것이 아니며 주요기능들 위주로 설계했다. 디테일한 함수들은 추가가 될 것이며 문서에는 List밖에 언급을 하지 않았지만 List, Stack, Queue 등 여러 자료구조를 적합한 형태로 적용 시킬 것이다.