**Software Engineering Q/A Sheet (#9)**

date:2020/11/14 number:2016310936 name: 우승민

Questions from Prof.

1. Describe the definition and benefits of the design pattern.

Design pattern은 해결해야 할 문제와 그에 대한 해결법에 대한 추상적인 지식을 재사용 하는 것이다.

Design Pattern 은 커뮤니케이션에서의 효율성을 향상시킬 수 있다. 또한 System을 objects로 분해하는 것에 효율적이며, object interface를 지정하는 것에도 효율적이다. 또한 자주 이용되는 Pattern을 이용하면 상당한 안정성이 기대된다.

1. Explain the benefits of software reuse.

개발 시간 및 비용이 감소한다. 이미 존재하는 system을 재사용하면 안정성이 기대된다. 개발 생산성과 품질이 향상된다.

1. Explain the need for configuration management.

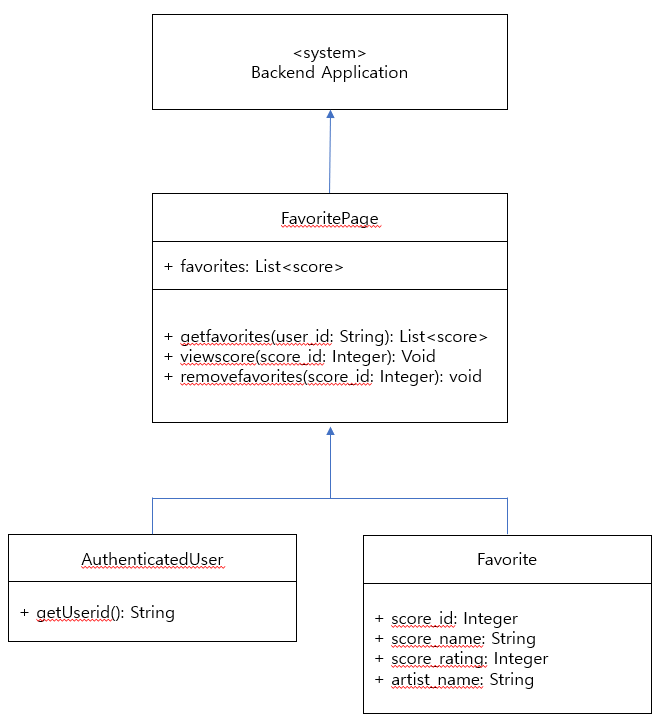
모든 개발자들이 제어된 방식으로 project의 코드와 문서들을 접근하고, 변경사항을 찾고, components를 compile, link하여 system을 만들 수 있도록 system integration process를 지원하기 위해 필요하다.

1. Explain the purpose of open source development from a OS(Open Source software) developer perspective and an OS user perspective.

OS developer : 개발자들이 자신들이 개발하는 system을 open source로 공개하면 외부에서 같은 문제를 고민하고 있는 다른 개발자들과 소통할 수 있고 더 나은 결과를 만들어 낼 수 있기 때문이다.

OS user : user 또한 open source를 이용하면서 자신들에게 더 필요한 내용을 발견하고 공유하면서 더 좋은 system을 이용할 수 있을 것이다.

1. Apply the design process learned in this chapter to your term project partially or as a whole. Practice is matter.

Favorite page – class diagram

1. FavoritePage 객체

A. attributes

+ favorites: favorite page내에 로드되어 있는 악보 목록

B. methods

+ getscores(criteria: Criteria): 사용자가 등록한 악보 목록을 backend server에서 가져온다.

+ viewsocre(score\_id: Integer): 특정 악보를 선택하여 본다. (Score 컴포넌트로 이동)

2. AuthenticatedUser – 인증 유저 객체

A. attributes

+ getUserid: 유저의 id를 입력 받는다.

2. Score – 악보 객체

A. attributes

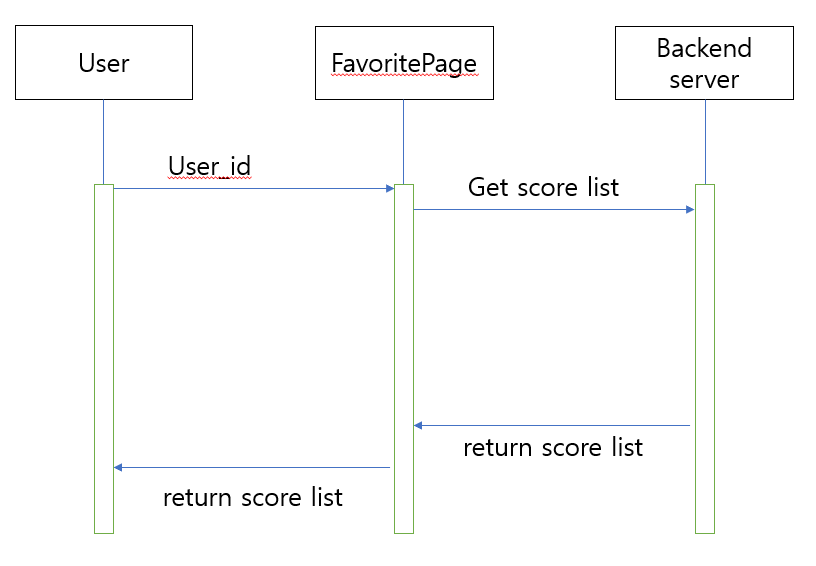
+ score\_id : 악보의 id

+ score\_name : 악보의 이름

+ score\_rating : 악보의 난이도

+ artist\_name : 작곡가(가수)의 이름

Favorite page - sequence Diagram



Questions from you

1. Software reuse에 어떤 level 들이 있고 무엇인지 설명하라.

Abstraction level : software를 직접 재사용하지 않고 성공적인 추상화된 지식을 이용한다.

Object level : 코드를 새로 작성하기 보다는 library의 objects를 직접 재사용한다.

Component level : application system의 objects와 object classes 등의 components를 재사용한다.

System level : application system 전체를 재사용한다.