국민대학교
전자정보통신대학
컴퓨터공학부

1 캡스톤 디자인 I

종합설계 프로젝트

프로젝트 명	*사용자 맞춤형 contents와 App 실행을 위한 보안 framework*
팀 명	*Rest*
문서 제목	2 차 중간보고서

Version	1.3
Date	2012-MAY-03

팀원	김 하랑 (조장)
	김 용태
	김 현주
	박 정훈
	양 희숙
	왕 효함
지도교수	한 재일 교수님

문서 정보 / 수정 내역

Filename	2차 중간보고서 - Rest.doc
원안작성자	김하랑, 김용태, 김현주, 박정훈
수정작업자	김현주, 양희숙

수정날짜	대표 수정자	Revision	추가/수정 항목	내 용
2012-04-27	김용태	0.8	최초 작성	1차 작성 완료
2012-04-29	양희숙	0.9	내용 수정	프로젝트 목표 수정
2012-04-30	김현주	1.0	내용 수정	2차 작성 완료
2012-05-01	박정훈	1.1	내용 수정	수행 내용 수정
2012-05-01	김하랑	1.2	내용 수정	수정된 연구 내용 수정
2012-05-03	김하랑	1.3	최종 작성	최종 작성 완료

목 차

2 프로젝트 목표

웹 브라우저에서 필요한 정보를 찾았을 때 사용자가 그 웹 페이지의 정보를 다시 찾아 볼 수 있는 '즐겨찾기' 혹은 '북 마크'라는 서비스가 제공되고 있으며, 이 서비스를 통해서 주소 창에 URL를 입력하지 않고 클릭하는 것만으로 사용자가 원하는 웹사이트에 접속할 수 있다. 이 서비스는 웹 페이지의 URL을 기억, 제공하여 페이지의 일부가 아닌 전체를 대상하고 있지만, 사용자는 보통 웹 페이지의 전체가 아닌 부분적인 컨텐츠에 관심을 두고 있다.

이러한 사용자 맞춤형 컨텐츠를 제공하기 위해서는 여러 웹 페이지의 일부분을 하나의 프레임 상에 컨텐츠화 하며, 앱 서비스까지 확장하여 사용자 맞춤형 서비스를 제공할 수 있는 서비스 모델이 필요하고 이를 제공할 때 서버에 과부하 문제가 발생할 수 있다.

이 프로젝트에서는 이러한 서버 과부하 문제를 해결하기 위해 REST 아키텍처 기반의 웹 서비스를 제안하며, 사용자 맞춤형 컨텐츠와 앱 서비스를 위한 보안모델의 설계 및 구현을 목표로 한다.

3 수행 내용 및 중간결과

3.1 계획서 상의 연구내용

		4 월			
		4/6 ~ 4/12	4/13 ~ 4/19	4/20 ~ 4/26	4/27 ~ 5/3
설계	App 저장/삭제				
	App 갱신				
	App 검색				
설계 보완	App 등록/삭제				
	App 저장/삭제				
	Oauth 인증				
	App 갱신				
	App 검색				
	UI				
구현	App 실행/종료				
	App 등록/삭제				
	App 저장/삭제				
	Oauth 인증				
	App 갱신				
	App 검색				
	UI				

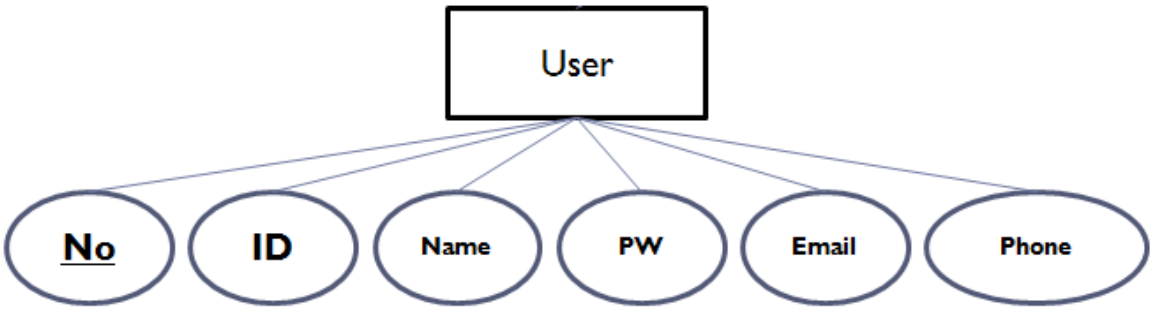
- App 저장, 삭제, 갱신, 검색 설계.
- App 등록, 저장, 삭제 설계 보완.
- Oauth 인증방식 설계 보완.
- UI 설계 보완.
- App 실행/ 종료 모듈 구현.
- App 저장/ 삭제 모듈 구현.
- OAuth 인증 모듈 구현.
- App 갱신, 검색 구현.
- UI 구현.

3.2 수행내용

3.2.1 데이터베이스 구조

3.2.1.1. User Table

기본적인 인증에 필요한 사용자의 정보를 저장하는 User테이블을 구성한다. User 테이블에는 사용자의 No, id, Name, Password, Email, Phone Number 필드를 갖는다.

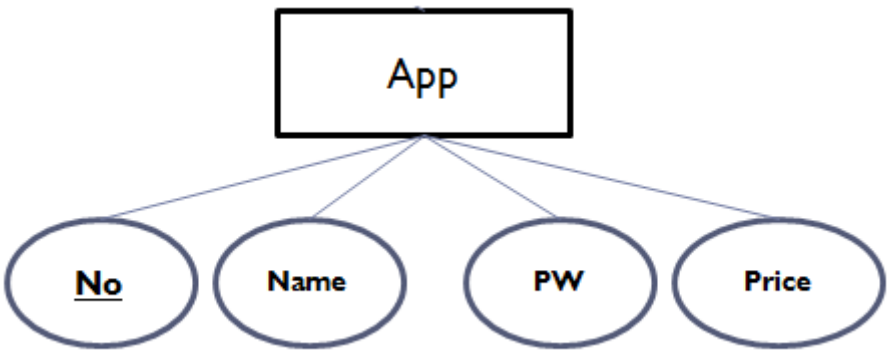


[그림1] User 테이블 구조

User 테이블에는 이것보다 많은 필드들이 정의되어야 한다. 하지만 본 프로젝트에는 인증 서버에서 이용할 기본적인 정보 필드들만 구성한다.

3.2.1.2. App Table

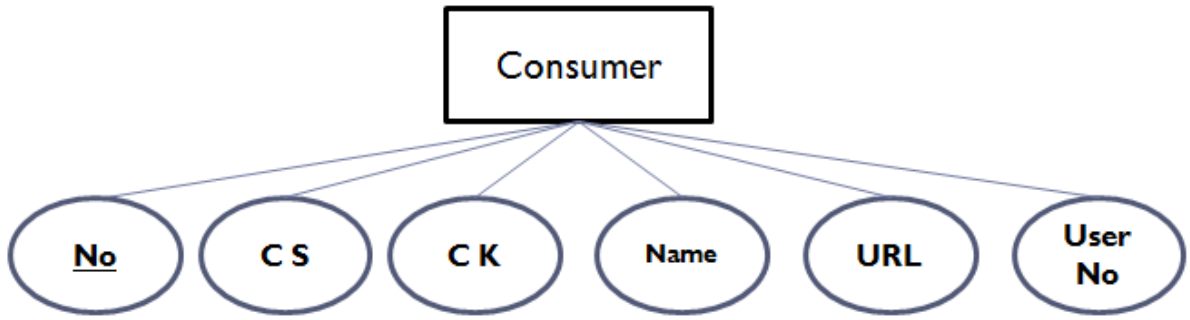
App에 대한 정보를 저장하는 App 테이블을 구성한다. App 테이블에는 App의 No, Name, Password, 그리고 Price 필드를 갖는다.



[그림2] App 테이블 구조

3.2.1.3. Consumer Table

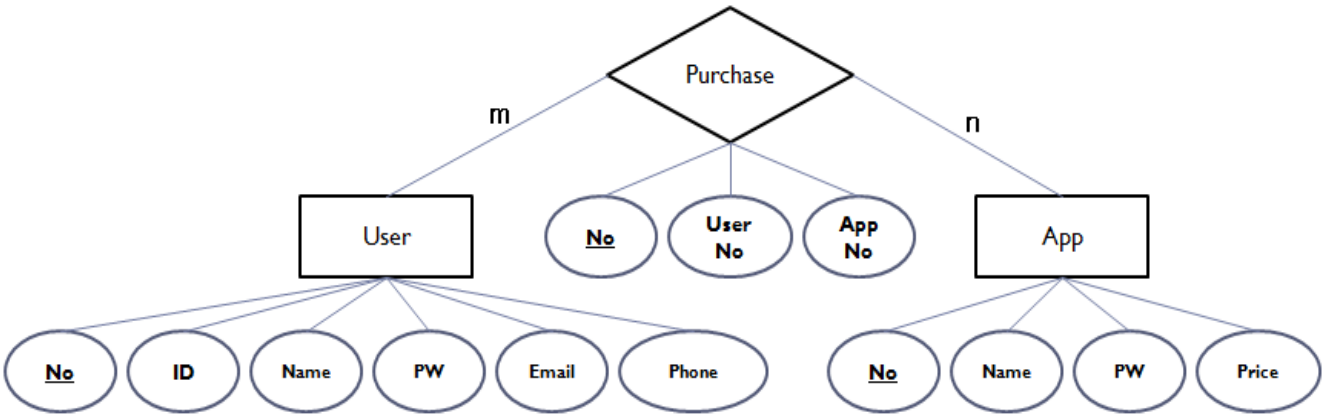
OAuth 프로토콜을 위한 어플리케이션 정보 테이블을 구성한다. Consumer 테이블에는 어플리케이션의 No, Customer Secret Key, Customer Key, Name, 그리고 App Service Server의 URL 필드를 갖는다.



[그림 4] Customer 구조

Customer 테이블의 UserNo 필드는 User 테이블의 No를 참조하는 외래키이다.

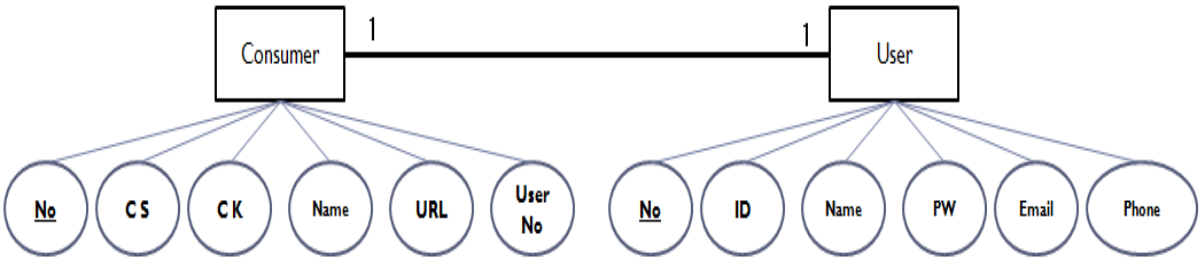
3.2.1.4. Relation between User Table and App Table



[그림3] User 테이블과 App 테이블 관계

User테이블과 App테이블의 관계는 위의 [그림3]과 같다. 하나의 유저는 여러 개의 App을 사용가능하며, 하나의 App은 여러 개의 User로부터 접근될 수 있다. 즉 Purchase M:N의 관계를 가진다. Purchase의 User No은 User table의 no를 참조하는 외래키이며, App no는 App의 no를 참조하는 외래키이다.

3.2.1.5. Relation between User Table and Consumer Table



[그림4] User 테이블과 Consumer 테이블 관계

Customer의 등록은 로그인 된 사용자에 한해서 등록이 가능하다. 또한 하나의 사용자는 하나의 Consumer를 등록할 수 있다. 본 시스템에서는 하나의 사용자가 하나의 Consumer를 등록 가능하도록 설계 하였으나, 실제 상용화될 때에는 정책에 따라 수정 가능하다.

3.2.2 OAuth 인증

이번 2차 수행에서는 OAuth 인증의 Consumer와 Provider 서버를 설계, 구현하였다. App Market Server는 OAuth의 Provider서버로 http://oauth.googlecode.com/의 소스 코드를 수정하여 구현하였으며, App Service Server는 OAuth의 Consumer로 Google API를 사용하여 구현하였다.

아래의 Token 요청 및 응답 과정이 이루어지기 위해서는 사전에 Consumer가 Provider에 등록하여 Consumer Key 및 Secret를 가지고 있어야 한다.

3.2.2.1. Request Token 요청

Consumer는 Provider 서버에게 Request Token을 요청할 때 가지고 있는 Consumer Key 및 Secret 을 전송하게 된다. 아래의 메시지는 실제 구현한 Consumer가 Provider 서버에게 전송하는 메시지 내용이다.

request tocken servlet
OAuthMessage(GET,
http://localhost/AppMarketServer/request_token?oauth_consumer_key=5d0ba4781c0d9e624f8df30806a4c4dd&oauth_signature_method=HMAC-SHA1&oauth_timestamp=1336085189&oauth_nonce=3806997533029&oauth_version=1.0&oauth_signature=Zj%2FN7L2ZRy272iz0fUkUSAIv%2FI8%3D,
[oauth_token=61284cc5313d0d26f66ed69bfa43262e, oauth_token_secret=
0fe2d91ac381e4dad3fa36f32d9a35c8])

메세지 전송 시 Consumer Key 및 Secret은 md5로 암호화 후 전송하게 된다.

3.2.2.2. Request Token 응답

Provider 서버는 Consumer에게 받은 요청을 받고 보내온 Consumer Key 및 Secret 을 확인하고 일치 할 경우 Request Token 을 전송하게 된다. 아래의 메시지는 실제 구현한 Provider 서버가 Consumer에게 Request Token을 응답하는 메시지 내용이다.

authorization servlet - get
OAuthMessage(GET, http://localhost/AppMarketServer/authorize, [oauth_callback=
http%3A%2F%2Flocalhost%2FAppServiceServer%2FReceive, oauth_token=
61284cc5313d0d26f66ed69bfa43262e])
authorization servlet - post

㉠http://localhost/AppServiceServer/Receive?oauth_token=61284cc5313d0d26f66ed69bfa43262e

㉡http://localhost/AppServiceServer/Receive

oauth_token=61284cc5313d0d26f66ed69bfa43262e

㉠은 Provider 서버가 Consumer에게 전송하는 Request Token 메세지이고 ㉡은 Consumer가 은 Provider 서버에게 받은 Request Token 메시지이며, 이 응답 메시지는 암호화 되지 않는다.

3.2.2.3. Access Token 요청 및 응답

Consumer가 Provider 서버에게 Request Token을 받은 뒤 Access Token을 요청하게 된다. 이때 Provider 서버는 Consumer에게 Request Token을 수신, 확인 한 뒤 Access Token을 전송한다.
아래의 메시지는 실제 구현한 Provider 서버가 Consumer에게 받은 Request Token과 보낸 Access Token을 보여준다.

access tocken servlet

Access

==========

KEY={5d0ba4781c0d9e624f8df30806a4c4dd} RT={61284cc5313d0d26f66ed69bfa43262e} ID={test}

==========

==========

KEY={5d0ba4781c0d9e624f8df30806a4c4dd} AT={76ae63646106f1ae915225b68ce5279d} ID={test}

==========

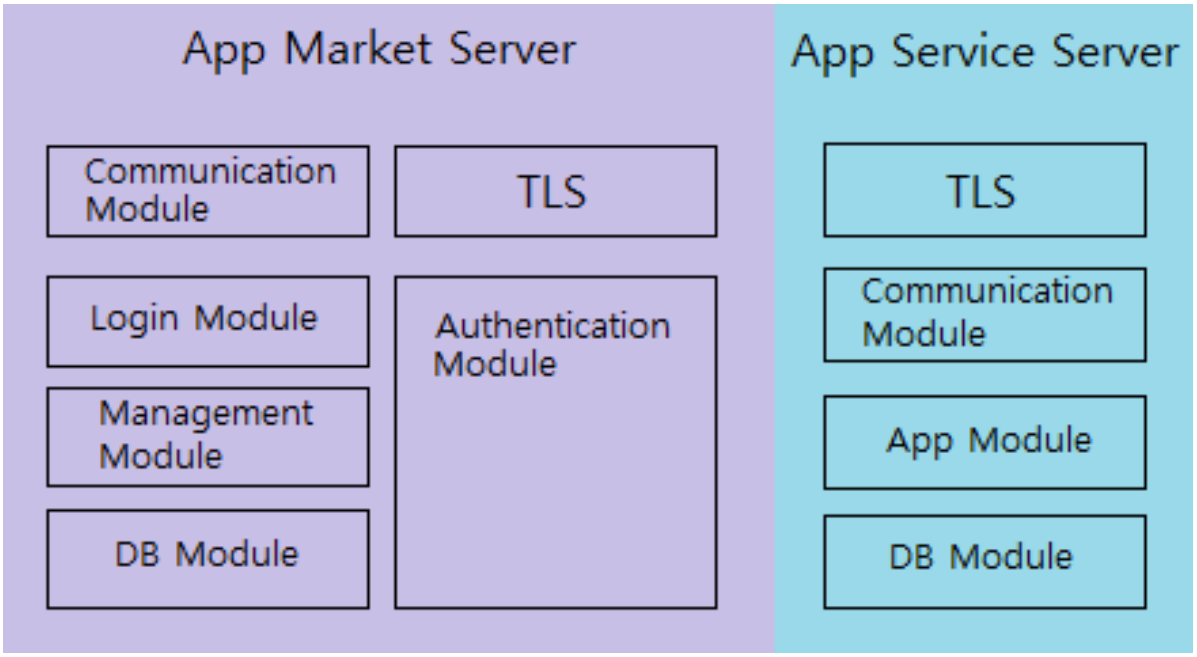
실제 Access Token은 md5로 암호화 후 전송된다.

4 수정된 연구내용 및 추진 방향

4.1 수정사항

프로젝트의 소프트웨어 구조를 수정하고 App 실행 시 필요한 인증 방식을 적용 및 구현하여 이에 따른 Sequence Diagram이 변경 되었다.

4.1.1 소프트웨어 구조

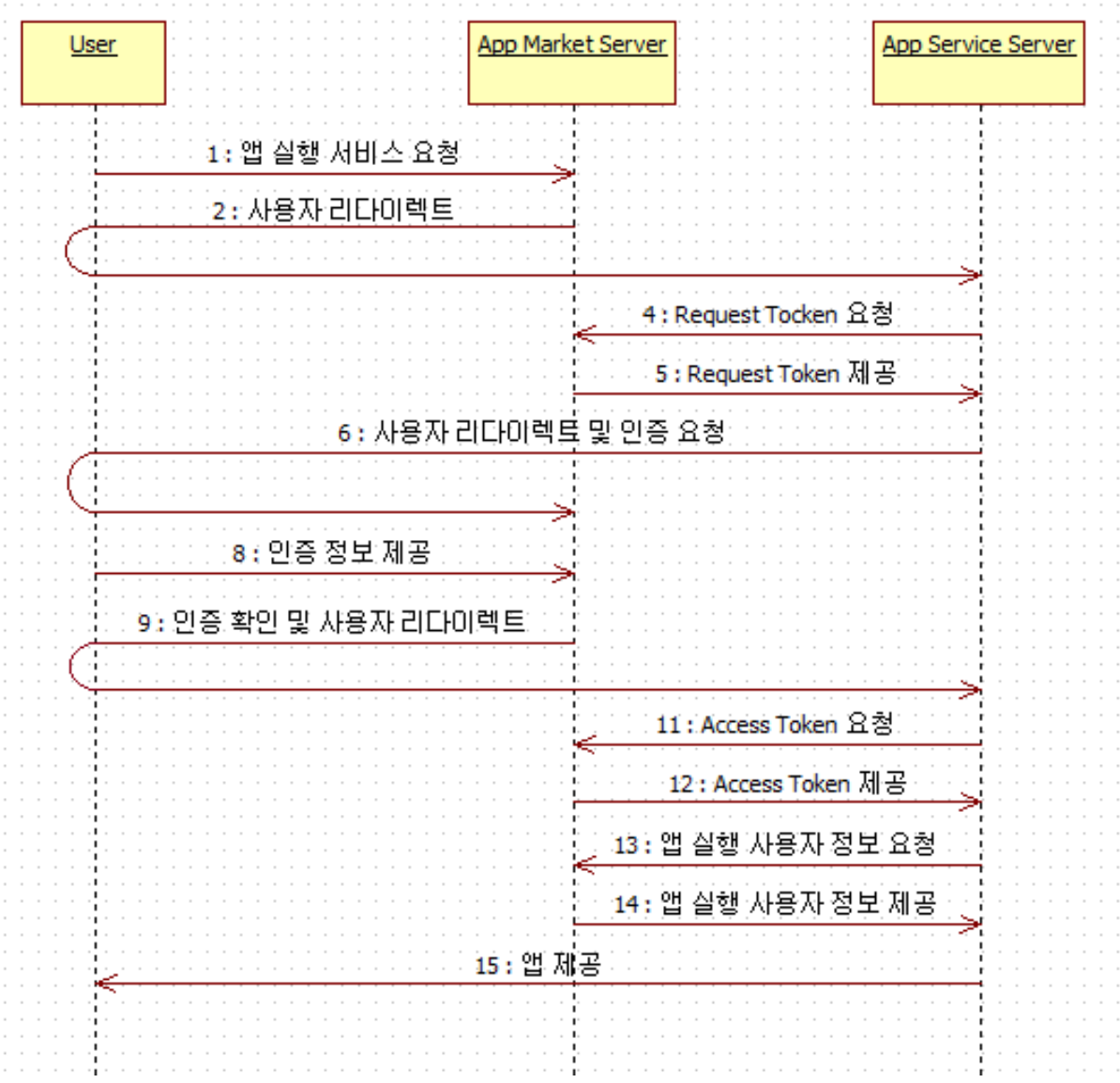


[그림 5]소프트웨어 구조

전체 시스템의 구조는 위의 그림과 같다. 사용자의 App Market 로그인을 위한 Login Module을 비롯해 DB Module, 각 서버와 사용자의 요청을 받아 정보를 제공하는 Communication Module, 서버와 사용자의 통신 보안을 위한 TLS, App과 User의 CRUD를 담당하는 Management Module. 그리고 App의 실행을 위한 App Module이 있다. App Marker Server는 OAuth의 Provider로서 OAuth 인증 시스템인 Authentication module이 존재하며, App Service Server는 OAuth 인증 시스템의 Consumer로서 작동하게 된다.

4.1.2 시퀀스 다이어그램

User 가 앱 실행 서비스를 이용하기 위해서는 App Service Server 가 사전에 App Market Server 에 자신을 등록하여 Consumer Key 와 Secret 을 발급 받아야 한다. Consumer Key 와 Secret 은 OAuth 프로토콜 상에 이용된다. OAuth 프로토콜에 의한 앱 실행 프로세스는 다음과 같다.



[그림 6]OAuth 인증과정이 적용된 App 실행 시퀀스 다이어그램

- User 는 App Market Server 로 앱 실행 서비스를 요청한다.
- App Market Server 는 User
- App Service Server 는 사전에 발급받은 Consumer Key 와 Secret 을 이용하여 App Market Server 에 Request Token 을 요청한다.
- App Market Server 는 App Service Server 가 제시한 Consumer Key 와 Secret 을 데이터베이스에서 검색하여 검증한 뒤 정상적이라면 Request Token 을 발급한다.
- App Service Server 는 발급받은 Request Token 을 Secret 과 함께 User 를 App Market Server 로 리다이렉트 시킨다.
- 리다이렉트된 User 는 App Market Server 와의 인증절차를 거친다. 이때, 사용자가 가져온 Request Token 역시 인증된다.
- App Market Server 는 인증된 Request Token 과 User 를 다시 App Service Server 로 리다이렉트 시킨다.
- 이후 App Service Server 는 인증된 Request Token 과 Secret 을 이용하여 Access Token 을 요청한다.
- App Market Server 는 제공된 정보를 확인하여 정상적이라면 Access Token 을 발급한다.
- App Service Server 는 발급받은 Access Token 을 활용하여 User 의 정보를 요청하여 인증을 마무리한다.
- 사용자 인증이 완료된 App Service Server 는 User 에게 앱 실행 서비스를 제공한다.

₩

5 향후 추진계획

5.1 향후 계획의 세부 내용

		5 월			
		5/4 ~ 5/10	5/11 ~ 5/17	5/18 ~ 5/24	5/25 ~ 5/31
설계 보완	App 등록/삭제				
	App 저장/삭제				
	App 갱신				
	App 검색				
	UI				
구현	App 등록/삭제				
	App 저장/삭제				
	App 갱신				
	App 검색				
	UI				
테스트	App 실행/종료				
	App 등록/삭제				
	App 저장/삭제				
	Oauth 인증				
	App 갱신				
	App 검색				
	통합 테스트				
기타	최종 보고				

- App 실행/종료 : 구현이 완료 되었고 테스트를 실시하여 둘째 주까지 완성도를 높일 것이다.
- OAuth 인증 : 구현이 완료 되었고 테스트를 실시하여 둘째 주까지 완성도를 높일 것이다.
- App 등록/삭제 : 5 월 첫째 주까지 구현을 하며 설계 보완을 마치고 둘째 주까지 구현을 완료한다.
- App 저장/삭제 : 5 월 첫째 주까지 구현을 하며 설계 보완을 마치고 둘째 주까지 구현을 완료한다.
- App 갱신과 검색 : 5 월 첫째 주까지 구현을 하며 설계 보완을 마치고 둘째 주까지 구현을 완료한다.
- UI : 5 월 첫째 주까지 구현을 하며 설계 보완을 마치고 둘째 주까지 구현을 완료한다
- 통합 테스트 : 5 월 둘째 주부터 통합 테스트를 실시하여 구현을 완료한다.

6 애로 및 건의사항

없음.