



내 PC



IntelliJ IDEA  
2022.1.3



웹\_프로젝트.java

# OPEN BANKING SERVICE

오픈뱅킹 2조 | 김재영 | 임지은 | 최경민 | 최민영 | 최유림 | 김관중



휴지통





내 PC



IntelliJ IDEA  
2022.1.3



웹\_프로젝트

## @Index

1. 시스템 아키텍처
2. WBS
3. 트러블 슈팅
4. 시연
5. 향후 계획



휴지통



시스템\_아키텍처

open\_banking\_service

architecture

시스템\_아키텍처

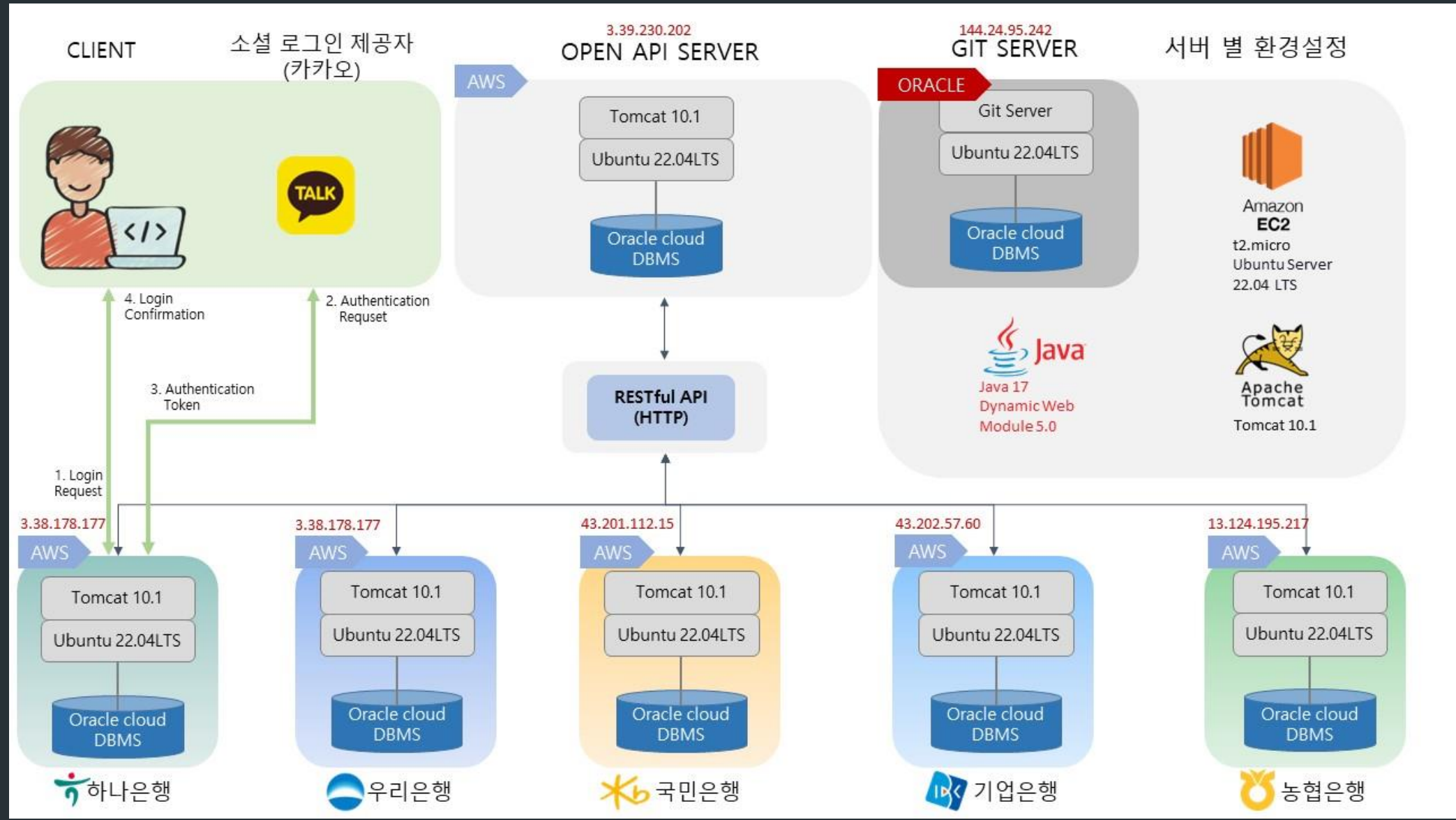
wbs

trouble\_shooting

demonstration

future\_plans

```
1 public class 시스템_아키텍처 {
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26 }
```



open\_banking\_service

architecture

시스템\_아키텍처

wbs

개발일정

trouble\_shooting

demonstration

future\_plans

개발일정

```
1 public class 개발일정 {
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26 }
```

구분	상세 업무	6/30	7/1	7/2	7/3	7/4	7/5	7/6	7/7	7/8	7/9	7/10	7/11	7/12	7/13	7/14	7/15	7/16	7/17	7/18	7/19		
주요 마일스톤	분석/설계(0.5주)																						
	개발/테스트/시연 (1.5주)																						
주요 회의 일정	착수회의																						
	수시회의																						
분석/설계	요구사항 정의 및 분석																						
	기능명세서																						
	설계서																						
	테이블정의서																						
	DB설계서																						
	화면설계서																						
개발	로그인																						
	로그인 및 회원가입 기능																						
	OAuth2.0 소셜 로그인 기능																						
	은행 API																						
	계좌 조회 기능																						
	계좌 개설 기능																						
	계좌 이체 기능																						
	오픈뱅킹 계좌 등록																						
	연결																						
	VIEW 연결																						
	인프라																						
	개별 은행 및 OPEN API 서버 용 클라우드 환경 구축																						
	형상 관리를 위한 GIT 서버 구축																						
	관리자 기능 및 고객센터																						
	게시글 CRUD																						
	게시글 답변																						
	관리자 회원 승인																						
테스트	단위 테스트																						
	통합 테스트																						
시연	PT 준비																						

}

📌 (기준은행)API 명세서 ...				
Aa 기능명	☰ 메소드 구분	🔗 URL	☰ 요청 유형	☰ Value
내가 가진 모든 계좌 조회	GET	benchmark-bank/accounts	Param	?PersonalIdNumber={value}
내가 연동한 모든 계좌 조회	GET	benchmark-bank/my-accounts	Param	?PersonalIdNumber={value}
오픈뱅킹에 등록된 본인의 타행 계좌 조회	GET	benchmark-bank/outer-my-accounts	Param	?bankName={value}&personalIdNumber={value}
내 모든 계좌 조회 (자행계좌 + 타행 연동계좌)	GET	benchmark-bank/my-bank-accounts	Param	?bankCode={value}&personalIdNumber={value}
내 타행 계좌 잔액 조회	GET	benchmark-bank/outer-balance	Param	?accountNumber={value}&bankCode={value}
계좌주 실명조회	GET	benchmark-bank/username	Param	?bankCode={value}&accountNumber={value}
오픈뱅킹 등록	PUT	benchmark-bank/registrate	Body	[           {             "accountNumber": "12345678",             "bankCode": "하나"           },           {             "accountNumber": "87654321",             "bankCode": "우리"           }         ]

☰

open\_banking\_service

▼ architecture

C 시스템\_아키텍처

▼ wbs

C 개발일정

▼ trouble\_shooting

C 소통문제

C 세션문제

> demonstration

> future\_plans

C 세션문제

1 public class 세션문제 {

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26 }

	USERNA...	OSUSER	PROCESS	PROGRAM	STATUS
1	DA2304	DA2304	1234	JDBC Thin Client	INACTIVE
2	DA2304	DA2304	1234	JDBC Thin Client	INACTIVE
3	DA2304	DA2304	1234	JDBC Thin Client	INACTIVE
4	DA2304	DA2304	1234	JDBC Thin Client	INACTIVE
5	DA2304	DA2304	1234	JDBC Thin Client	INACTIVE
6	DA2304	DA2304	16824	SQL Developer	INACTIVE
7	DA2304	DA2304	1234	JDBC Thin Client	INACTIVE
8	DA2304	DA2304	1234	JDBC Thin Client	INACTIVE
9	DA2319	DA2319	20404	SQL Developer	INACTIVE
10	DA2321	DA2321	18220	SQL Developer	ACTIVE
11	DA2321	DA2321	1234	JDBC Thin Client	INACTIVE
12	DA2321	DA2321	1234	JDBC Thin Client	INACTIVE
13	DA2321	DA2321	1234	JDBC Thin Client	INACTIVE
14	DA2321	DA2321	1234	JDBC Thin Client	INACTIVE
15	DA2321	DA2321	1234	JDBC Thin Client	INACTIVE
16	DA2321	DA2321	1234	JDBC Thin Client	INACTIVE
17	DA2321	DA2321	1234	JDBC Thin Client	INACTIVE
18	DA2321	DA2321	1234	JDBC Thin Client	INACTIVE
19	DA2321	DA2321	1234	JDBC Thin Client	INACTIVE
20	DA2321	DA2321	1234	JDBC Thin Client	INACTIVE
21	DA2321	DA2321	1234	JDBC Thin Client	INACTIVE
22	DA2321	DA2321	1234	JDBC Thin Client	INACTIVE
23	DA2321	DA2321	1234	JDBC Thin Client	INACTIVE
24	DA2321	DA2321	1234	JDBC Thin Client	INACTIVE
25	DA2321	DA2321	7432	SQL Developer	INACTIVE
26	DA2321	DA2321	1234	JDBC Thin Client	INACTIVE
27	TEAM4	TEAM4	11529	SQL Developer	INACTIVE
28	TEAM4	TEAM4	2680	SQL Developer	INACTIVE

세션문제 해결 전

```
@Override
public boolean withdraw(String accountNumber1, int amount) {
    PreparedStatement pstmt = null;
    Connection conn = null;
    boolean result = true;
    String query =
        "UPDATE account_info_woori SET balance = balance - ? WHERE account_number = ?";

    try {
        conn = DriverManager.getConnection(DB_URL, USER, PASS);
        pstmt = conn.prepareStatement(query);
        pstmt.setInt(1, amount);
        pstmt.setString(2, accountNumber1);
        pstmt.executeUpdate();
    } catch (SQLException e) {
        e.printStackTrace();
    }
    result = true;
    return result;
}
```

자원을 할당받았지만 반납하는 코드의 부재

open\_banking\_service

architecture

시스템\_아키텍처

wbs

개발일정

trouble\_shooting

소통문제

세션문제

demonstration

future\_plans

세션문제

```
1 public class 세션문제 {
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26 }
```

세션문제 해결 후

	USERNAME	OSUSER	PROCESS	PROGRAM	STATUS
1	DA2304	DA2304	16400	SQL Developer	INACTIVE
2	DA2321	DA2321	18220	SQL Developer	ACTIVE
3	DA2319	DA2319	20404	SQL Developer	INACTIVE
4	TEAM4	TEAM4	11529	SQL Developer	INACTIVE
5	DA2321	DA2321	7432	SQL Developer	INACTIVE

```
@Override
public boolean withdraw(String accountNumber1, int amount) {
    boolean result = true;
    String query = "UPDATE account_info_woori SET balance = balance - ? WHERE account_number = ?";
    try (Connection conn = DriverManager.getConnection(DB_URL, USER, PASS);
        PreparedStatement pstmt = conn.prepareStatement(query)) {
        pstmt.setInt(1, amount);
        pstmt.setString(2, accountNumber1);
        pstmt.executeUpdate();
        result = true;
    } catch (SQLException e) {
        e.printStackTrace();
    }
    return result;
}
```

try 조건 안에 자원 할당 코드를 넣어  
자동으로 자원을 반납하도록 구현



```
1 public class 시/연 {
```

2

3

○

8

10

11

12

10

14

15

16

17

17

20

21

22

03

25

26

```
public class 시/연 {
```





open\_banking\_service

architecture

시스템\_아키텍처

wbs

개발일정

trouble\_shooting

소통문제

세션문제

demonstration

시연

future\_plans

향후\_계획

향후\_계획

1public class 향후\_계획 {

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26}





## KOPO\_오픈뱅킹 2조

개발 일정 관리, 리소스 조정, 팀 내 의견 조율, 프로젝트 전략 수립, 리스크 관리 등 프로젝트 진행 전반의 모든 것

### 목차

#### 0. 프로젝트 진행 상황(~7/13 기준)

#### 1. 프로젝트 개요

- 1-1. 개발 일정
- 1-2. 기능 명세서
- 1-3. 역할분장
- 1-4. 화면설계
  - 1-4-1. 화면설계서(figma)
  - 1-4-3. 화면설계서(ppt)
- 1-5. DB 설계
  - 1-5-1. ERD 설계
  - 1-5-2. 테이블 정의서
  - 1-5-3. DDL
- 1-6. 서비스 아키텍처
- 1-7. API 명세서
- 1-8. 패키지 구조
- 1-9. Git svr

#### 2. 프로젝트 요구 사항

#### 3. 회의록

#### 4. .properties

#### 5.발표 리뷰

- KOPO\_오픈뱅킹 2조

> 은행 중계 공동서비스

> 팀원정보

> (기존은행)API 명세서

> (OPENAPI)API 명세서

> (대상은행)API 명세서

> (트리블슈팅)세션문제해결

> (개발로그)VIEW 추가 & 유효성검사

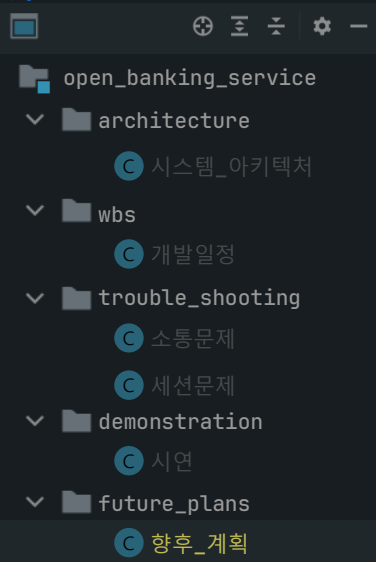
> (개발로그)UI/UX VIEW

> (개발로그)기존은행/대상은행

> (개발로그)OPENAPI


> (개발로그)관리자페이지

> 시연 시나리오(임시)




```
1 public class 향후_계획 {
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26 }
```



 **감사합니다.**  
오픈뱅킹 2조 | 김재영 | 임지은 | 최경민 | 최민영 | 최유림 | 김관중  
☐ 다시 묻지 않음


종료













취소



## KOPO\_오픈뱅킹 2조

개발 일정 관리, 리소스 조정, 팀 내 의견 조율, 프로젝트 전략 수립, 리스크 관리 등 프로젝트 진행 전반의 모든 것

 KOPO\_오픈뱅킹 2조

- >  은행 중계 공동서비스
- >  팀원정보
- >  (기준은행)API 명세서
- >  (OPENAPI)API 명세서
- >  (대상은행)API 명세서
- >  (트리블슈팅)세션문제해결
- >  (개발로그)VIEW 추가 & 유효성검사
- >  (개발로그)UI/UX VIEW
- >  (개발로그)기준은행/대상은행
- >  (개발로그)OPENAPI
- >  (개발로그)관리자페이지
- >  시연 시나리오(임시)

1-5. DB 설계

- 1-5-1. ERD 설계
- 1-5-2. 테이블 정의서
- 1-5-3. DDL

1-6. 서비스 아키텍처

1-7. API 명세서

1-8. 패키지 구조

1-9. Git svr

2. 프로젝트 요구 사항

3. 회의록

4. .properties

5. 발표 리뷰