

소상공인과 함께하는 경영 · 세무 신고 지원 플랫폼

하나 WITH

하나금융티아이 교육생
광명융합기술원 데이터분석과 박종범

목차

- 01. 서비스 개요
- 02. 주요 기능 & 응용 기술
- 03. 개발환경 및 아키텍처
- 04. 시연 시나리오
- 05. 기대 효과
- 06. 보완점 및 향후 방향

하나 WITH

“소상공인을 위한 경영·세무 신고 지원 서비스”

서비스 목적

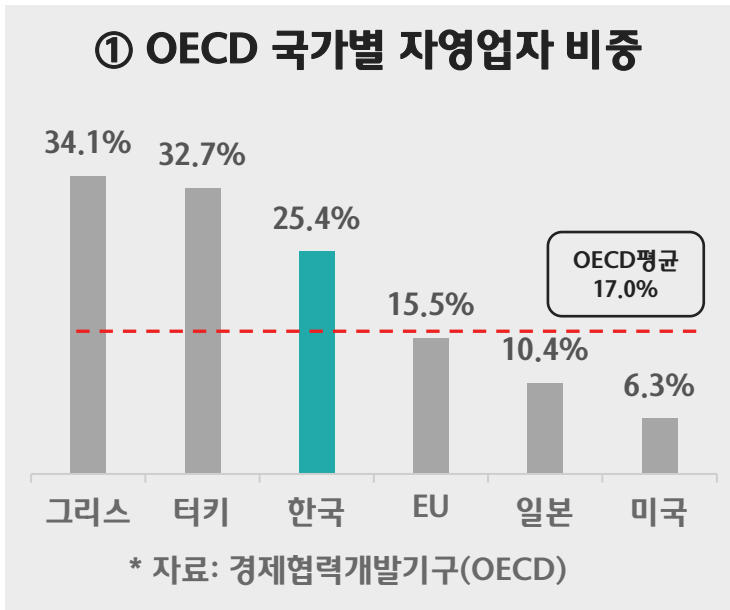
1. **소상공인**의 사업장 운영에 도움이 되는 서비스
2. **세무사** 업무 효율화
3. **은행**은 소상공인 고객 확보를 통한 비즈니스 기회 창출

서비스 장점

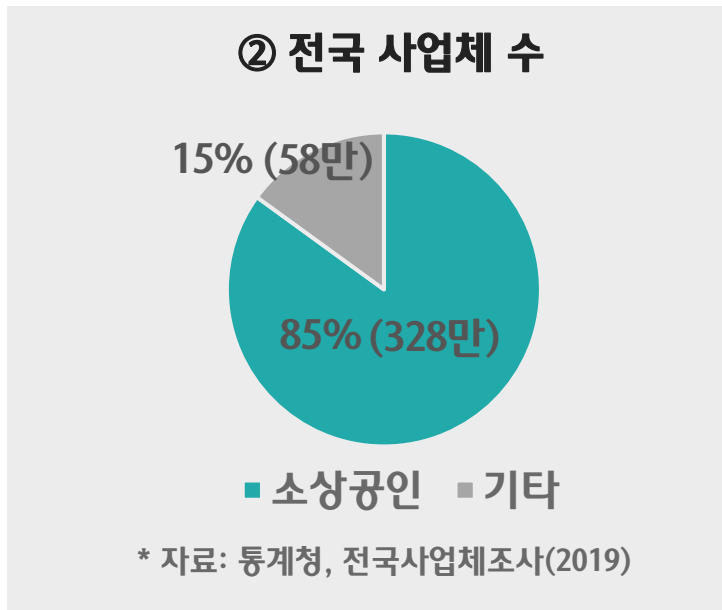
1. 세금신고에 필요한 증빙 자료 통합 관리
2. 한 눈에 파악할 수 있는 사업 현황
3. 편리한 세금 신고서 자동 생성

01 서비스 개요 – 기획 배경

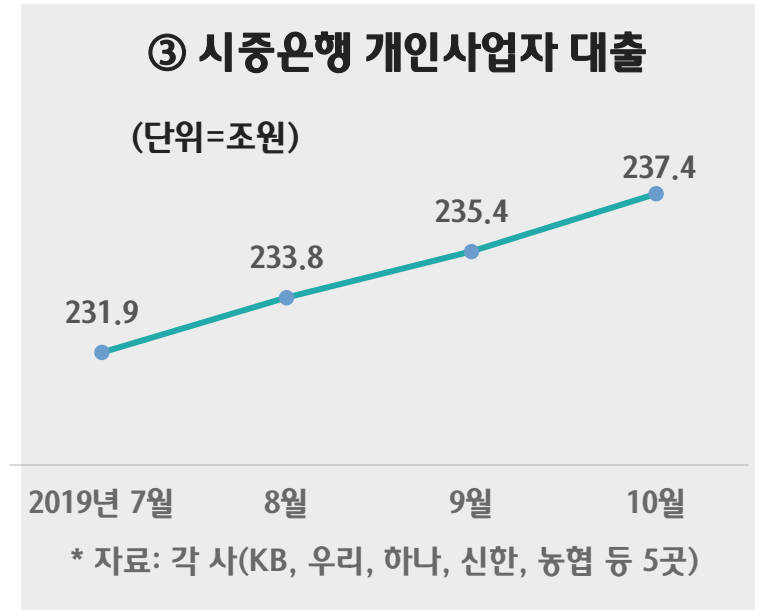
- 국내 경제활동인구의 자영업자 비중 **25% 상회**



- 전국 사업체 중 소상공인 비중 **85% (328만 개)**



- 개인사업자 시중은행 거래 규모 **증가 추세**



소상공인 전용 서비스 출시 필요

01 서비스 개요 – 경쟁사 분석

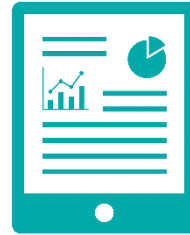
	<div> NH소상공인 Partner</div>	<div> KB Brigde</div>	<div> One 소상공인</div>	<div> 하나WITH</div>
1. 소상공인 전용 서비스	O	O	O	O
2. 매입/매출 통합 관리	X	X	X	O
3. 사업보고서 제공	O	X	X	O
4. 세금신고서 생성	X	X	X	O
5. 수기 영수증 자동 입력	X	X	X	O
6. 홈택스 연동	O	X	O	O

↓
보다 직접적인 소상공인 지원 서비스
“하나 WITH”



사업장 매입 / 매출 통합 관리

- ① 홈택스, 여신금융협회 연동을 통한 전자 영수증 관리
- ② OCR을 통한 수기 / 개인 영수증 전산화



사업보고서 제공

- ① 매입 / 매출 사업보고서 업데이트
- ② 일간, 주간, 월간 매입 / 매출 현황 파악



세금신고서 작성

- ① 담당 고객 증빙 자료 조회/수집
- ② 세금신고서 자동 생성 및 파일 변환 기능

02 주요 기능 & 응용 기술

#1 CLOVA OCR

등록된 수기 영수증 텍스트 자동추출
→ 영역 지정, 이미지 전체 텍스트 추출 → 전산화

#2 JSON Web Token

홈택스 API에 액세스하려는
클라이언트 유효성 검증

#3 Web Socket

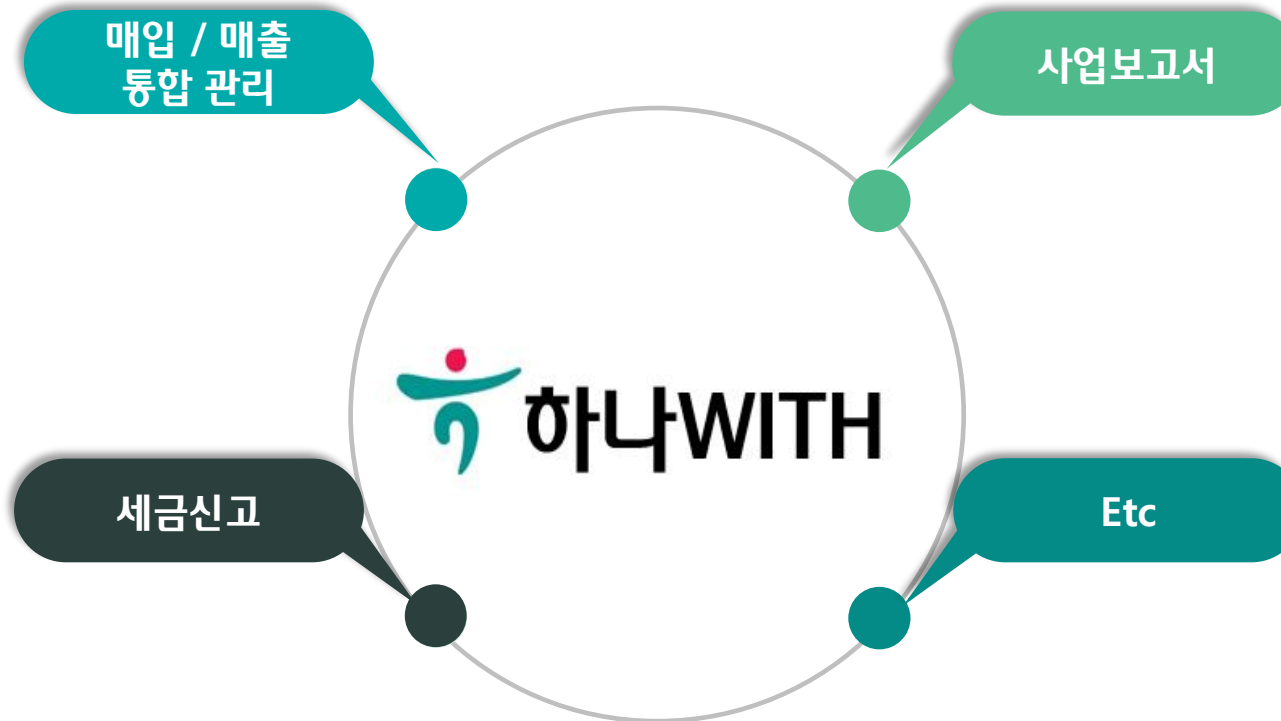
관리자 수기 영수증 승인 / 반려 시
실시간 알림

#4 Apache POI

소상공인, 세무사에게 필요한 내역 출력
(매입 / 매출 내역, 고객 증빙자료)

#5 Html2Canvas & JsPDF

세금 신고서 작성 후 PDF 파일 변환



#6 Spring Scheduler

카드사 통합 매출 batch 처리 (익일 04시)
업데이트 알림 메시지 발송

#7 Chart.js

일간, 주간, 월간 매입 / 매출 보고서
시각화

#8 SMS API

매출 내역 전송, 사업보고서 업데이트
알림 메시지

#9 공공데이터 포털 API

사업자 회원가입 시 사업장 진위여부
확인 및 상태조회 서비스

#10 Ajax

비동기 통신을 활용한 화면 갱신

개발환경

-Language



-DB



-Server



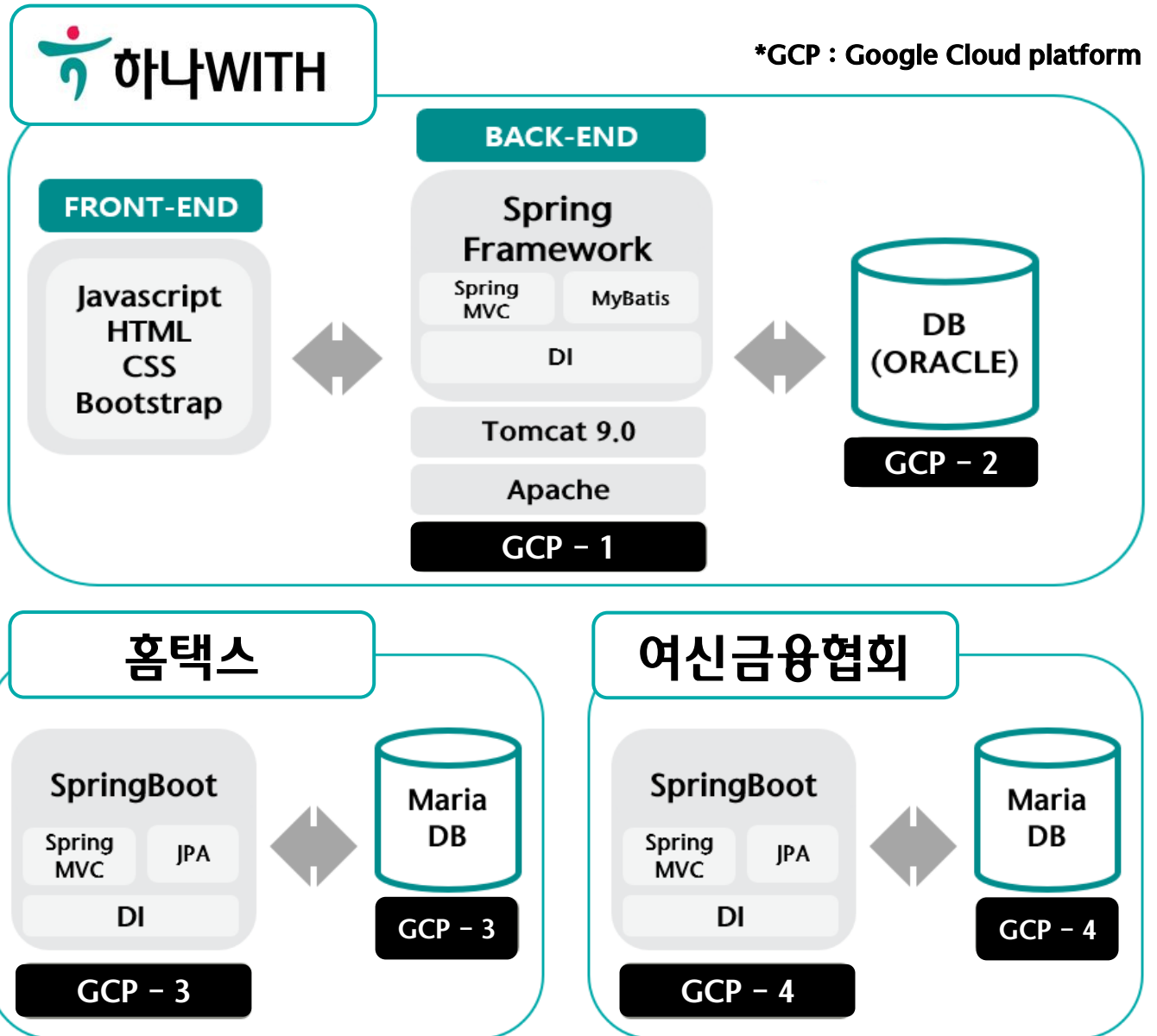
-Framework



-VCS



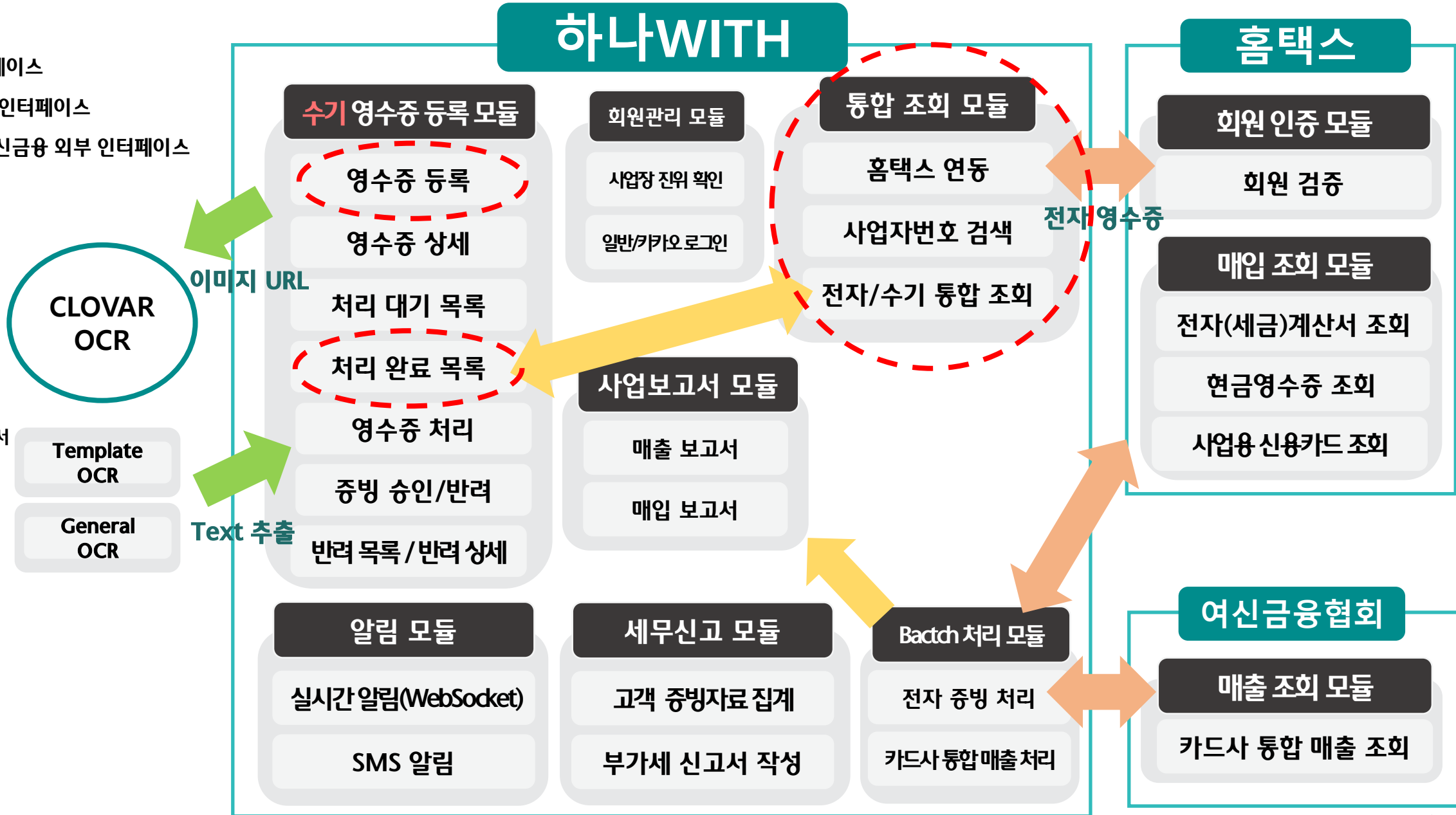
-Etc



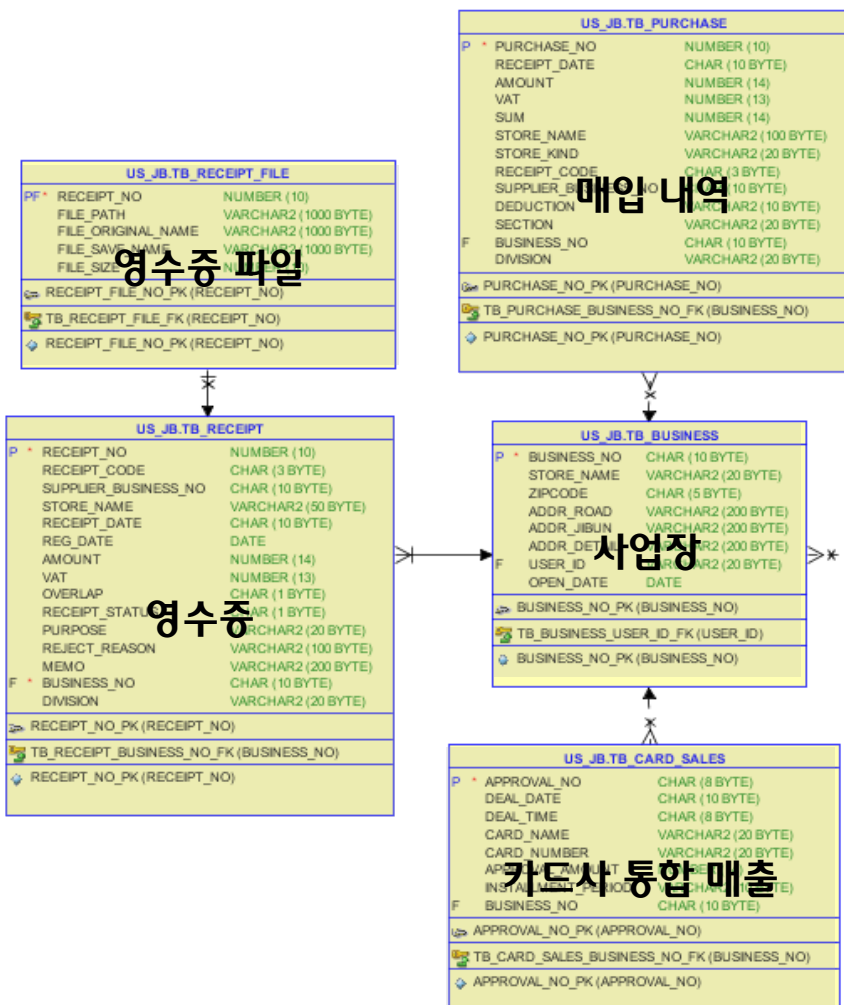
03 개발환경 및 아키텍처 – 서비스 아키텍처

- : 내부 인터페이스
- : OCR 외부 인터페이스
- : 홈택스, 여신금융 외부 인터페이스

- ① 전자 (세금)계산서
 - ② 현금영수증
 - ③ 간이영수증
-
- ④ 카드 영수증

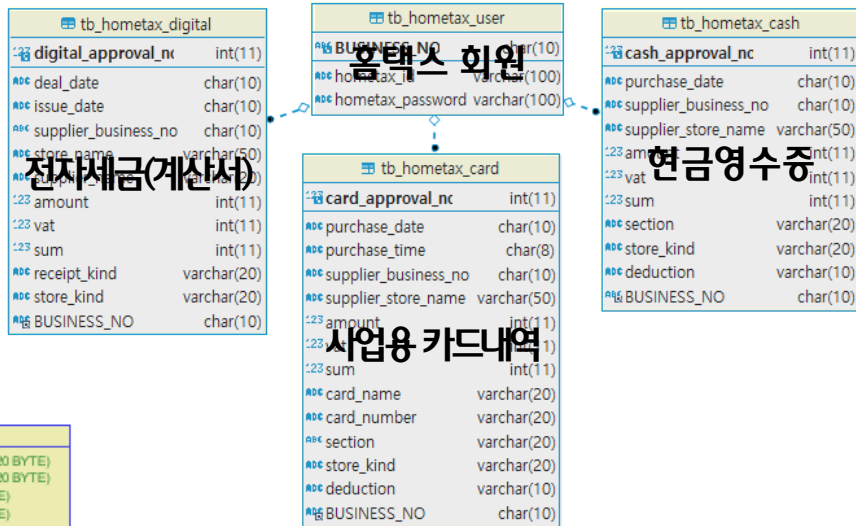


[하나WITH]



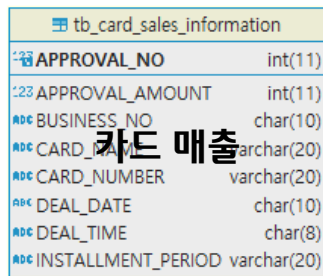
[홈택스]

(전자 매입 증빙)



[여신금융협회]

(카드사 통합 매출)

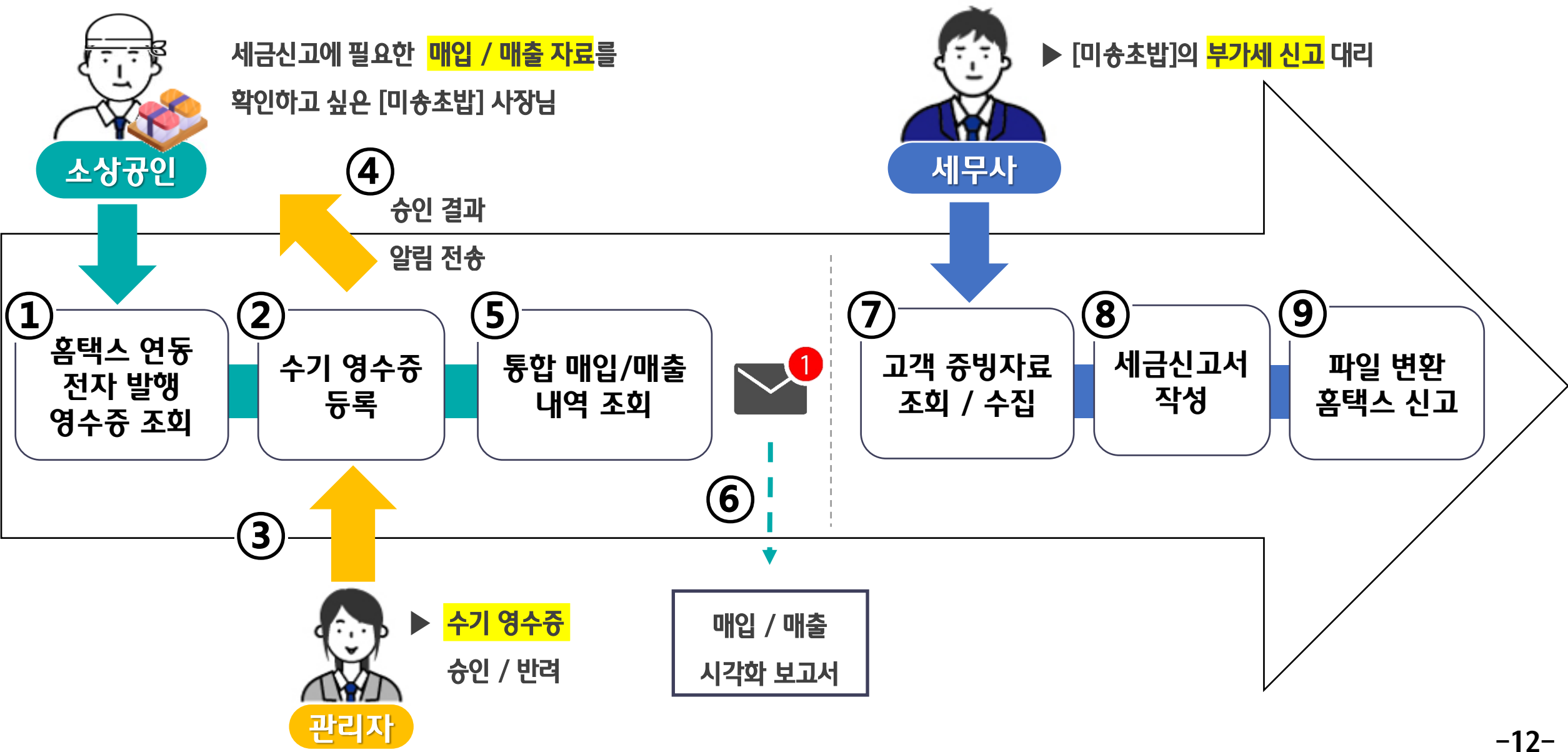


하나WITH

2021/09/06 – 2021/10/5

	9/6	9/9	9/12	9/16	9/23	9/26	9/30	10/3	10/5
서비스 기획									
DB 설계									
소상공인 기능 개발									
영수증 등록 기능 개발									
API 서버 개발									
사업보고서 기능 개발									
관리자 기능 개발									
세무사 기능 개발									
테스트									

04 시연 시나리오



모두의 기쁨 그 하나를 위하여

소상공인을 위한 경영·세무 신고 지원 플랫폼





- 편리한 매입 / 매출 증빙 자료 통합 관리
- 세금 신고 자료 훼손 / 분실 방지
- 사업보고서를 통한 사업 현황 파악

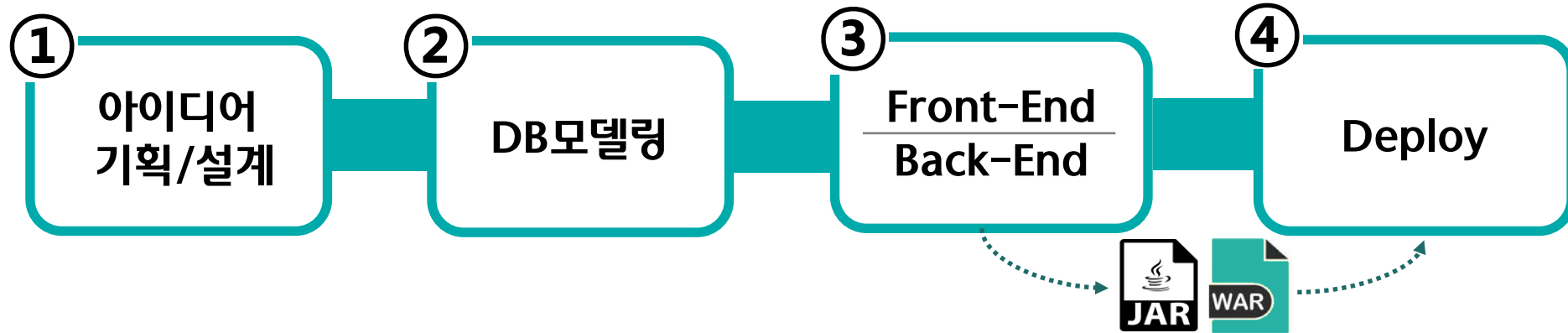


- 소상공인 고객 점유율 강화
- 금융 상품 연계를 통한 비즈니스 기회 창출



- 전산화된 수기 영수증을 통한 업무 효율 상승
- 편리한 세금 신고서 작성

- 01 **관리자 수기 영수증 처리 개선**
서버에 등록된 영수증 재 OCR후 입력 정보와 일치하지 않는
사진만 확인
- 02 **관리자 대시보드 구성**
소상공인 업종별 군집분석, 고객 데이터 분석
- 03 **소상공인 세금신고서 직접 작성**
세무사를 거치지 않은 셀프 신고



* 실제 서버 배포 시행착오

Linux환경 Oracle formatting, 배포 시 경로 설정,
서버 Shut Down 등..



Q&A

감사합니다.



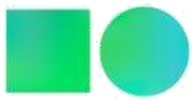
➤ 수기 영수증 등록 과정



수기 or 개인카드 영수증



Google Cloud
서버에 이미지 저장



네이버OCR로
이미지URL 전송



Template OCR

- 샘플을 통한 영역 지정
- 세금계산서, 간이영수증 등 일정한 양식에 사용

General OCR

- 일정하지 않은 카드영수증
- 이미지 전체 텍스트 추출
- 영수증에 커스터마이징

① 수기 영수증 등록 (일정한 양식 - 세금계산서, 계산서, 간이영수증)

- Template OCR 영수증 샘플 등록, 영역 지정

【별지 제11호 서식】 (공급받는자)

세금계산서 (공급받는자)

등록번호: 605-23-22491

상호: 진유동

사업장소: 부산시 진구 초읍동 263-2

업태: 도매

세액: 272,273

합계금액: 300,000

[세금계산서]

【별지 제28호 서식(1)】 (공급받는자)

계산서 (공급받는자)

등록번호: 605-40-37399

상호: 세계식품

사업장소: 부산시 진구 부전동 341-46

업태: 도,소매

세액: 500,000

합계금액: 500,000

[계산서] (부가세 0원)

0303-18 (95.6.26. 개정)

영수증 (공급받는자용)

등록번호: 605-27-35392

상호: 화성주차장

사업장소: 부산광역시 부산진구 부전동 401-25

업태: 서비스

작성년월일: 2021.3.3.

공급대가총액: 9,000

합계금액: 9,000

[간이영수증]



지정한 영역 Text 추출 후 Parsing

(1. supplier_business_no, 2. store_name, 3. receipt_date, 4. amount)

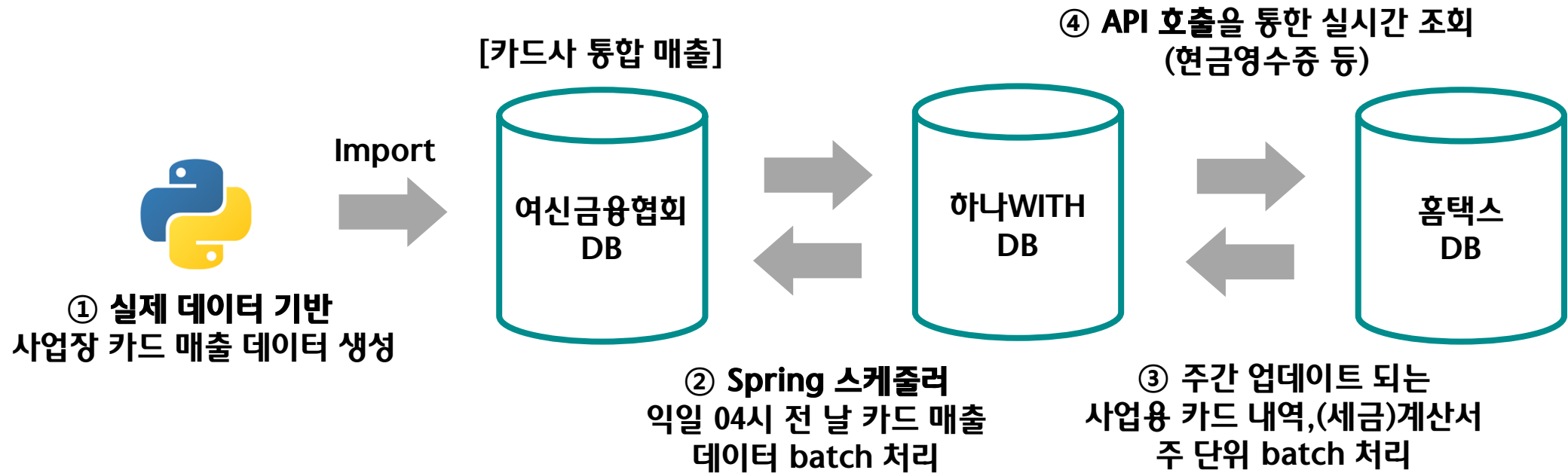
② 수기 영수증 등록 (일정하지 않은 양식 - 일반 카드 영수증)

- General OCR 이미지 전체 Text 추출, 영수증 정보 추출을 위한 커스터마이징 (영수금액, 부가세 추출)



1. 이미지 전체 Text 추출
2. '합', '계', '금액' 문자와 관련 있는 Index 추출
3. 추출한 Index + 1번 째부터 정규식을 이용하여 숫자 제외 공백 치환, Integer형 변환 시도
4. 형 변환 가능한 Index List 추출, 추출한 Index 중 Max value를 '합계금액'으로 저장

➤ 사업보고서 업데이트



➤ JPA를 활용한 API 서버 구축 (홈택스, 여신금융협회)

- Java 객체와 관계형 DB 테이블 사이의 Mapping 설정을 통해 **SQL 자동 생성**
- **중복된 쿼리 제거, 유지 보수성 향상**
- **복잡한 쿼리용으로는 적합하지 않다는걸 느낌...**

[홈택스-현금영수증 Entity 클래스]

[홈택스-현금영수증 조회 Controller]

```
@RequestMapping(path = "/hometax-cash", method = RequestMethod.GET)
ResponseBody<Map<String, Object>> selectHometaxCash(@RequestParam String purchaseDate, String businessNo){
    HometaxCashDto hometaxCashDto = new HometaxCashDto();
    hometaxCashDto.setBusinessNo(businessNo);
    hometaxCashDto.setPurchaseDate(purchaseDate);
    Map<String, Object> returnMap = hometaxCashService.selectHometaxCash(hometaxCashDto);

    return new ResponseEntity<>(returnMap, (HttpStatus) returnMap.get("status"));
}
```

[홈택스-현금영수증 조회 Repository]

```
@Nullable
List<HometaxCash> findByPurchaseDateAndBusinessNo(String purchaseDate, String businessNo);
SELECT * FROM HOMETAX_CASH WHERE PURCHASE_DATE
= purchaseDate AND BUSINESS_NO = businessNo;

@Nullable
List<HometaxCash> findByPurchaseDateStartsWithAndBusinessNo(String purchaseDate, String businessNo);
SELECT * FROM HOMETAX_CASH WHERE PURCHASE_DATE
LIKE 'purchaseDate%' AND BUSINESS_NO = businessNo;
```

```
@AllArgsConstructor
@Entity
@Table(name = "HOMETAX_CASH")
@Getter
public class HometaxCash {

    @Id
    @Column(length = 10, nullable = false)
    private String cashApprovalNo; //승인번호

    @Column(length = 8, nullable = false)
    private String purchaseDate; //매입일

    @Column(length = 10)
    private String supplierBusinessNo; //가맹점 사업자번호

    @Column(length = 30)
    private String supplierStoreName; //가맹점 상호

    @Column(length = 10)
    private String businessNo; //사업자번호

    @Column(length = 8)
    private String purchaseTime; //매입시

    @Column(length = 12)
    private long amount; //공급가액
```

➤ JWT (Json Web Token)

- 하나WITH - 홈택스 연동 시 회원 인증

[하나WITH]

```
// 토큰 생성
public static String createToken(HomeTaxInfoVO homeTaxInfo) {
```

```
// 비밀키
String key="jbhana";
```

```
// 헤더
Map<String, Object> headers = new HashMap<>();
headers.put("typ", "JWT");
headers.put("alg", "HS256");
```

```
Map<String, Object> payloads = new HashMap<>();
payloads.put("hometaxId", homeTaxInfo.getHometaxId());
payloads.put("hometaxPassword", homeTaxInfo.getHometaxPassword());
payloads.put("businessNo", homeTaxInfo.getBusinessNo());
```

```
String jwt = Jwts.builder()
    .setHeader(headers)
    .setClaims(payloads)
    .signWith(SignatureAlgorithm.HS256, key.getBytes())
    .compact();
```

```
return jwt;
```

```
}
```

헤더(Header)

내용(Payload)

서명(Signature)

- ① 헤더, 내용 각각 base64 Encoding
- ② 비밀키로 헤더에서 정의한 alg로 암호화
- ③ 다시 base64 Encoding

aaaaa.bbbbbb.ccccc

[홈택스]

```
public Map<String, Object> selectHometaxAuth(String token){
```

```
String secretKey = "jbhana";
```

```
Map<String, Object> returnMap = new HashMap<>();
```

```
Claims claims = Jwts.parser()
```

```
    .setSigningKey(secretKey.getBytes())
```

```
    .parseClaimsJws(token)
```

```
    .getBody();
```

```
String hometaxId = claims.get("hometaxId", String.class);
```

```
String hometaxPassword = claims.get("hometaxPassword", String.class);
```

```
String businessNo = claims.get("businessNo", String.class);
```

```
HometaxAuth hometaxAuth = hometaxAuthdRepository.findByBusinessNo(businessNo);
```