



PORTFOLIO

(MZ 세대를 위한 수익 예측 기반 귀농 추천 서비스)



김상호





TURN TO FARMING

CONTENTS

1. 프로젝트 기획서

기획 의도

개발 목표

2. 개발 환경 및 사용 기술 경험

사용 기술 목록표

사용 기술 경험

3. 개발 스케줄표

Project I(Big Data & UI)

Project II(Servlet & JSP)

Project III(Spring Framework)

4. 빅데이터 분석 결과서

데이터 수집

데이터 전처리 및 검증

모델링 및 분석 결과

주요 시각화

5. 요구사항 정의서

6. 화면 설계서

7. UML

Use Case Diagram

Class Diagram

Sequence Diagram

8. ER Diagram

9. 주요 서비스 기능 및 소스 코드

주요 서비스 기능

소스 코드

10. Software Architecture

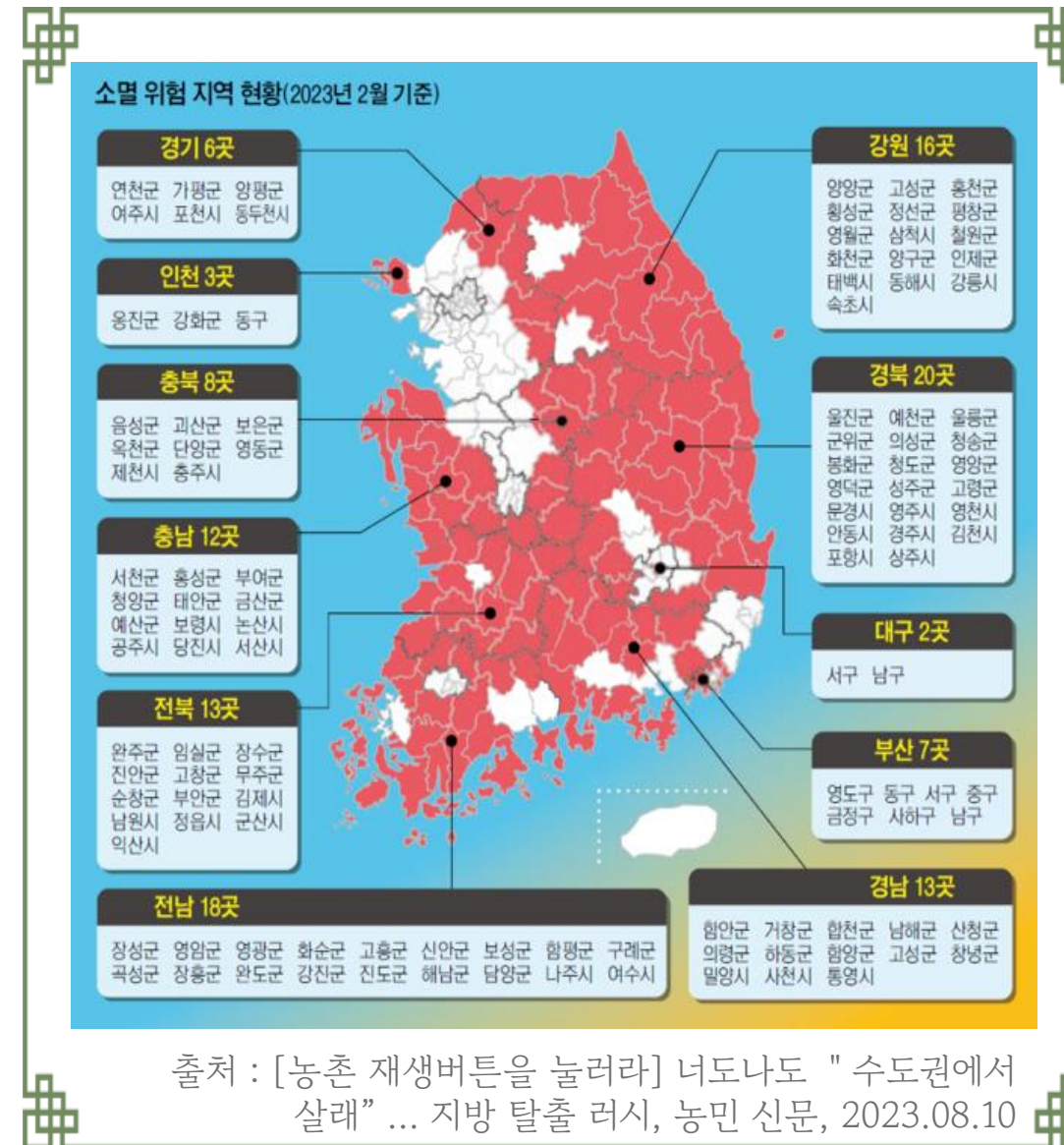
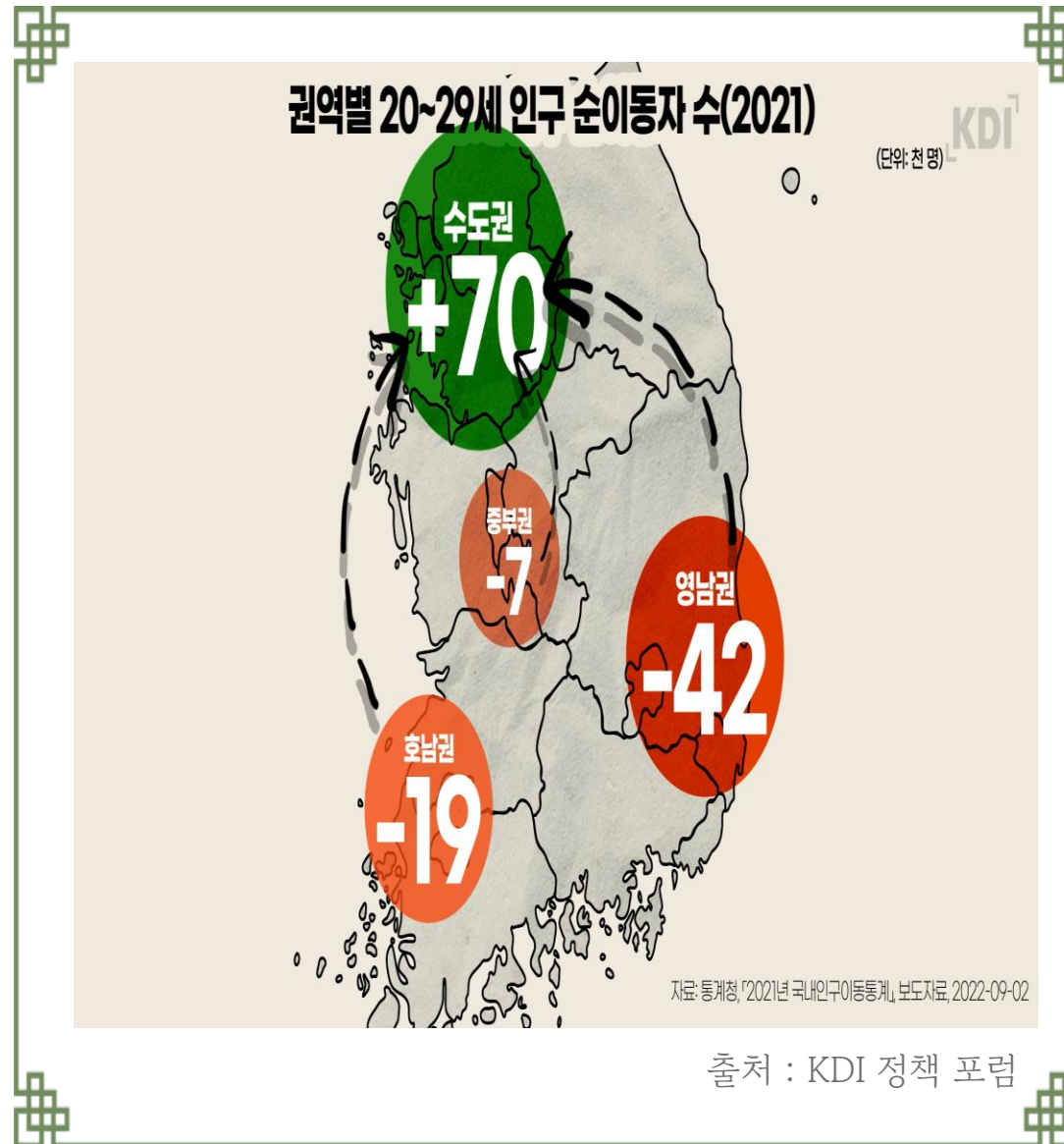
11. 시연

12. 향후 계획 및 프로젝트 수행 소감



TURN TO FARMING

프로젝트 기획 의도



점차적인 도시 중심 생활
선호로 인해 농촌지역에서의
청년 인구가 감소

지역 경제 비활성화 및 지방
소멸 우려 증대



TURN TO FARMING

프로젝트 기획 의도



문제점별(1)	2022
사례수 (명)	3,000
관련 정보를 얻기 어려움 (%)	32.9
지원 자격이 너무 까다로움 (%)	24.2
지원절차, 서류 등이 너무 복잡함 (%)	12.9
지원금액이 적음 (%)	8.6
일회성 지원에 그침 (%)	5.0
특정시기에 지원시기가 집중되어 있음 (%)	2.7
실질적으로 필요한/도움이 되는 정책이 없음 (%)	13.7
기타 (%)	0.1

출처 : 통계청

어려움별(1)	2022
사례수 (명)	3,000
주거 (%)	3.8
농사 (%)	19.9
소득 (%)	38.3
지역 주민과의 관계 (%)	1.9
지역 인프라 부족 (%)	11.5
기타 (%)	0.6
어려움 없음 (%)	24.2

출처 : 통계청

지역 경제 비활성화 및 지방
소멸 해결방안
→ 청년들의 농촌 복귀

그러나, 귀농 관련 정보에 대한
어려움 및 귀농 시 얻을 소득에
대한 불확실성 존재



TURN TO FARMING

프로젝트 개발 목표



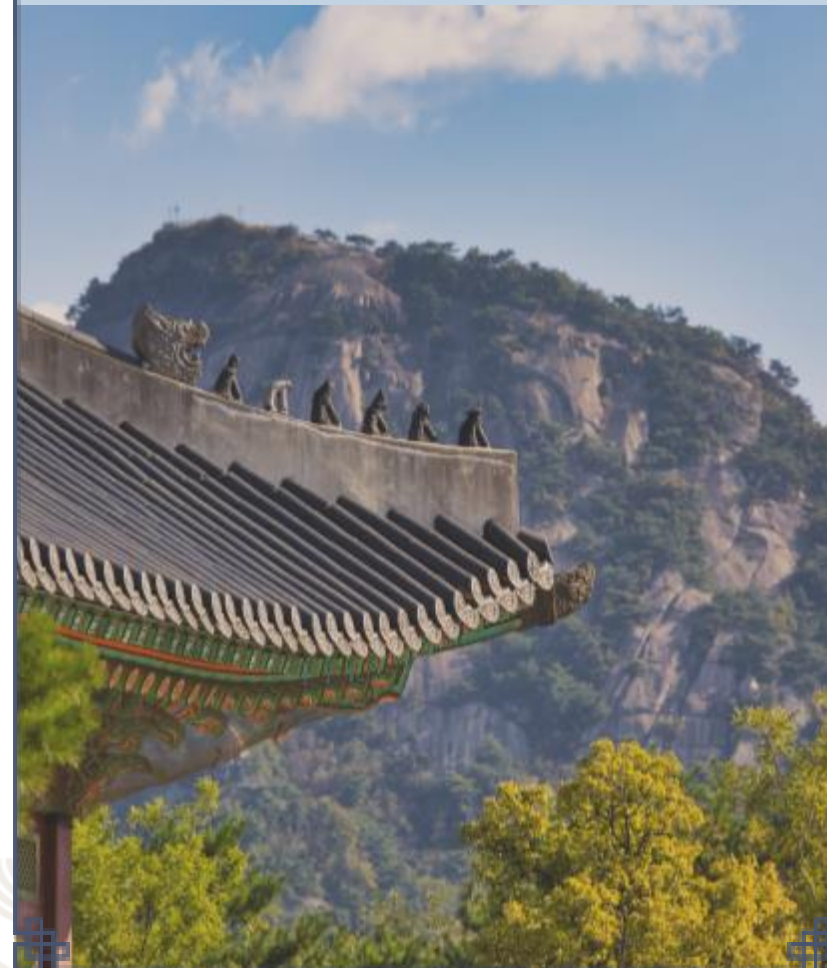
농작물 매출 예측

(다중 선형 회귀 분석을 이용한
농작물 매출 예측)



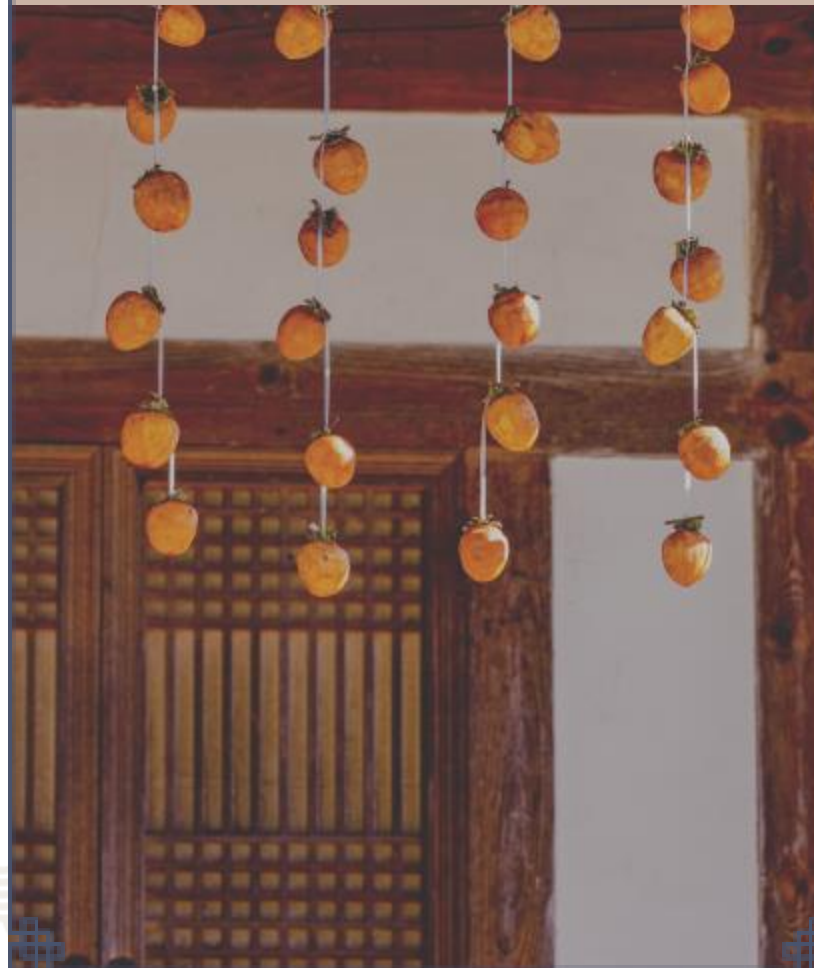
비용 조회

(농작물, 지역을 기준으로 초기비용,
유지비용, 정책 지원금 조회)



수익 확인

(예상 매출과 비용 및 정책 지원금을
통한 수익 예측)



커뮤니티

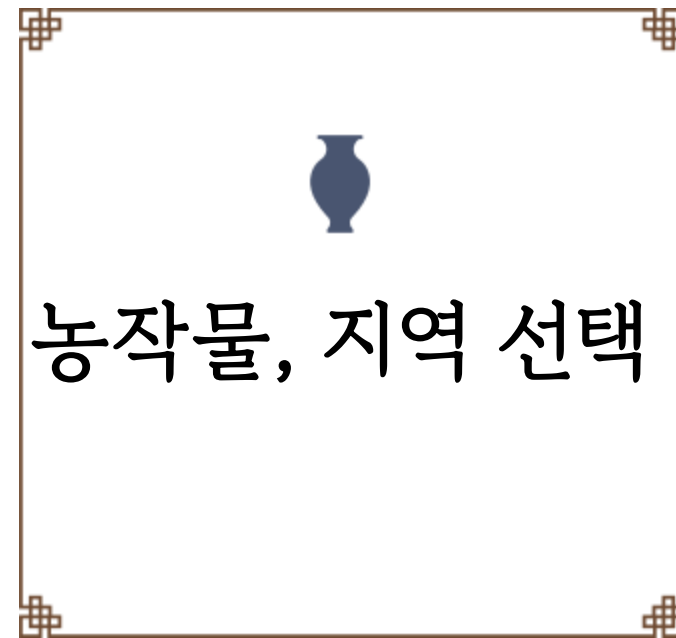
(지역 축제 및 문화행사 정보 제공,
귀농 정보 공유 게시판)



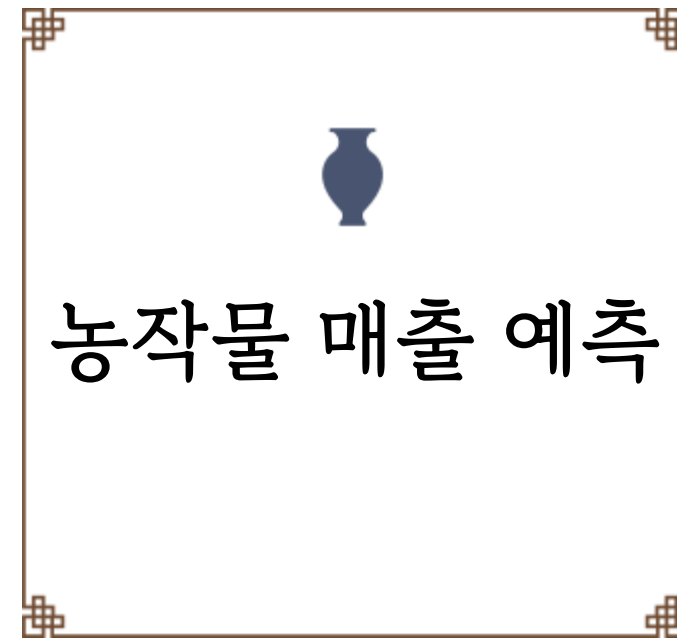


TURN TO FARMING

프로젝트 개발 목표



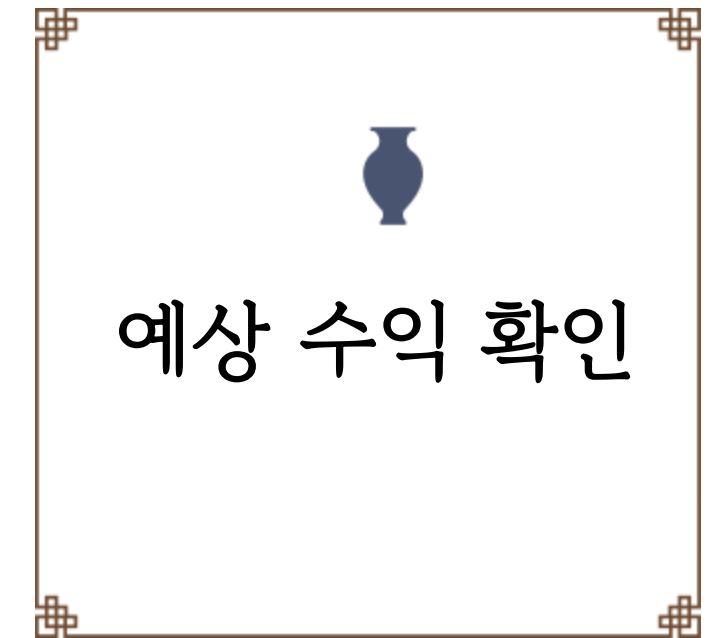
사용자 선호에 따른
농작물 우선, 지역 우선
2가지 매출 예측 서비스 제공



다중 선형 회귀 분석
2003년~2021년 데이터 분석
독립변수 : 기온, 강수량, 일사량,
일조시간, 생산비용, 유통 비율,
GDP, 경작면적, 생산량, 경매 단가



농작물과 지역에 따른
비용 정보 제공
(소요비용, 정책 지원금)



예상 매출, 비용, 정책
지원금을 종합한 예상 수익
그래프로 시각화



TURN TO FARMING

사용 기술 목록

개발 환경



[Spring Framework]
4.3.9



[Eclipse]
2019.09



[R-Studio]
4.1.1(2021-08-10)



[eGovFrame]
3.9.0



[Oracle]
11.2.0.2

사용 언어



[HTML5, CSS3, JavaScript]



[Bootstrap]
3.3.2



[JSP]
2.3

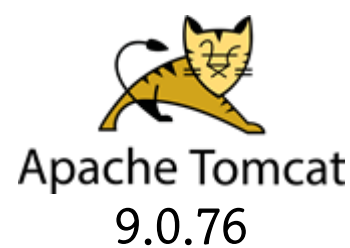


[Java]
1.8.0_361



[Python]
3.10.11

웹 서버



Apache Tomcat
9.0.76



TURN TO FARMING

사용 기술 경험



JAVA

- JAVA 1.8 기반 미니 프로젝트 경험
- 주소록 관리 시스템(입력, 출력, 검색, 삭제, 수정)
- Single Linked List구조(노드 삽입, 역순, 삭제)
- 상속, 인터페이스
- 컬렉션 프레임워크(List, Set, Map)



Spring Framework

- Bean설정 및 DI
- IOC
- AOP
- 프로젝트 - Spring WEB MVC 개발(Model, View, Controller)
- 프로젝트 - Spring JDBC(데이터 베이스와 연결)



Spring Boot

- Annotation을 사용한 DI
- 정적 리소스, JSP, Model, Form, Lombok 사용
- Validator, initBinder
- JDBCTemplate
- Spring Boot 배포



HTML5 / CSS3

- 프로젝트 - 전체적인 홈페이지 페이지
- 네비게이션 바 반응형 제작



JavaScript

- 프로젝트 - 농작물, 지역 정보, 매출, 비용, 수익 정보를 비동기적으로 업데이트 하면서 동적으로 표시(AJAX 사용)
- 프로젝트 - 농작물 매출, 예상 수익 정보 데이터를 시각적으로 표현(Chart.js 라이브러리 사용)



Bootstrap

- 프로젝트 - 전체적인 홈페이지 반응형으로 제작



JSP

- MVC Model2 패턴을 기반으로 한 수익 예측 기반 귀농 추천 서비스 웹사이트 개발
- 회원관리 및 게시판 개발



Database

- 프로젝트 - 농작물 매출, 비용, 수익, 게시판, 회원관리를 위한 SQL 처리 및 테이블 관리, 시퀀스 작성
- Oracle 데이터베이스를 이용한 다양한 쿼리문 실습
- 데이터 모델링을 이용한 설계



TURN TO FARMING

사용 기술 경험



R

- 프로젝트 - 다중 선형 회귀 분석을 통해 농작물 매출 예측
- 산점도, 상관 분석 및 상관계수
- 데이터 분석을 통한 시각화 방법
- 구글 API를 이용한 구글 맵 사용



Python

- 구글 코랩(colab) 사용하여 기본 문법 학습
- 데이터 시각화를 위한 Matplotlib 라이브러리 활용



AWS EC2

- AWS EC2 개발 환경 구축
- Ubuntu Server 20.04
- PuTTY와 FileZilla를 사용



Linux

- Ubuntu 기반 Linux 개발 환경 구축 실습
- 리눅스 커맨드 숙지
- Shell 프로그램 사용



MSA

- MSA Service Architecture 소개
- MSA Service Architecture 설계
- MSA 표준 구성 요소
- Spring Cloud



Git

- 프로젝트 관리를 위한 기본적인 git 명령어 사용 경험



TURN TO FARMING

개발 스케줄표

Project I (Big Data & UI)

업무 분장	이름	번호	항목	시작일	종료일	작업기간
기획	공통	1	아이디어 구상	2023-04-21	2023-04-24	4
		2	기획안 작성	2023-04-25	2023-04-27	3
		3	기획안 중간 발표	2023-04-27	2023-04-27	1
		4	개발 스케줄표 작성	2023-05-22	2023-05-22	1
		5	기획안 최종 발표	2023-05-26	2023-05-26	1
빅데이터	김상호	6	농작물 매출 회귀분석 독립변수 데이터 수집	2023-05-26	2023-05-31	6
		7	농작물 매출 회귀분석 모델링	2023-05-29	2023-05-31	3
		8	농작물 매출 회귀분석 보고서 작성	2023-06-08	2023-06-11	4
		9	농작물 매출, 수익 관련 API 프로그램 개발	2023-06-01	2023-06-02	2
		10	ERD 설계	2023-06-03	2023-06-04	2
		11	데이터 전처리작업	2023-06-05	2023-06-07	3
		12	데이터 분석 및 시각화	2023-06-08	2023-06-11	4
		13	데이터 결과보고서 작성	2023-06-12	2023-06-16	5
UI	공통	20	요구사항 정의서, 명세서 작성	2023-05-23	2023-05-25	3
		21	와이어프레임 작성	2023-05-27	2023-05-31	5
	김상호	22	메인 페이지 UI 구현	2023-06-17	2023-06-18	2
		23	농작물 매출 분석 및 예측 페이지 UI 구현	2023-06-19	2023-06-20	2
		26	성공사례 + 문화정보 게시판 페이지 UI 구현	2023-06-21	2023-06-21	1



TURN TO FARMING

개발 스케줄표

Project II (Servlet & JSP)

업무 분장	이름	번호	항목	시작일	종료일	작업기간
Web 개발	김상호	1	프로젝트 설계	2023-07-12	2023-07-14	3
		2	UML 작성 (Class Diagram, Sequence Diagram, Usecase Diagram)	2023-07-15	2023-07-18	4
		3	데이터 점검 및 수정, 데이터 임포트(Oracle)	2023-07-18	2023-07-18	1
		4	FrontController 구현	2023-07-19	2023-07-19	1
		5	비용 조회 서비스, 수익 확인 서비스 - DAO, DTO 구현	2023-07-19	2023-07-19	1
		6	비용 조회 서비스 - View 구현	2023-07-19	2023-07-19	1
		7	비용 조회 서비스 - 매출 조회 기능 구현	2023-07-19	2023-07-19	1
		8	비용 조회 서비스 - 비용 조회 기능 구현	2023-07-20	2023-07-20	1
		9	비용 조회 서비스 - 정책 지원금 조회기능 구현	2023-07-20	2023-07-20	1
		10	비용 조회 서비스 - 정책 리스트 정보 제공 기능 구현	2023-07-20	2023-07-20	1
		11	비용 조회 서비스 - 정책 지원금 변경 기능 구현	2023-07-21	2023-07-21	1
		12	비용 조회 서비스 - 총 소요 비용 정보 출력 기능 구현	2023-07-21	2023-07-21	1
		13	비용 조회 서비스 - 히스토리 저장 기능 구현	2023-07-21	2023-07-21	1
		14	수익 확인 서비스 - View 구현	2023-07-22	2023-07-22	1
		15	수익 확인 서비스 - 수익 정보 차트 시각화 기능 구현	2023-07-22	2023-07-23	2
		16	수익 확인 서비스 - 정보 출력 기능 구현	2023-07-24	2023-07-24	1
		17	단위 테스트 및 통합 테스트 수행	2023-07-25	2023-07-25	1
		18	버그 수정 및 코드 최적화	2023-07-25	2023-07-25	1

Project III (Spring Framework)

업무 분장	이름	번호	항목	시작일	종료일	작업기간
Spring Framework	김상호	1	Spring Framework 개발 환경 설정 세팅	2023-08-07	2023-08-07	1
		2	프로젝트 패키지 구성	2023-08-08	2023-08-08	1
		3	Bean 설정, DI 설정	2023-08-08	2023-08-08	1
		4	JDBC 연결	2023-08-08	2023-08-08	1
		5	농작물 매출 서비스 Spring Framework 개발	2023-08-09	2023-08-10	2
		6	비용 조회 서비스 Spring Framework 개발	2023-08-11	2023-08-11	1
		7	수익 확인 서비스 Spring Framework 개발	2023-08-12	2023-08-12	1
		8	성공 사례 게시판 Spring Framework 개발	2023-08-13	2023-08-13	1
		9	회원관리 Spring Framework 개발	2023-08-14	2023-08-15	2



TURN TO FARMING

데이터 수집

기후 데이터

온도, 강수량, 일조시간, 일사량,
기압, 풍속, 상대습도

출처 : 기상청 기상자료 개방포털

농작물 매출 관련 데이터

농작물 매출, 생산비용, 경작면적,
생산량

출처 : 통계청

상등품 도매가격, 유통 비율

출처 : 농수산물유통정보 KAMIS

GDP

출처 : e-나라지표

비용 관련 데이터

평균 투자비(초기비용),
연간 운영비(유지비용)

출처 : 귀농귀촌 종합센터

데이터 지역 : 전라남도
데이터 기간 : 2003년 ~ 2021년(19년)



TURN TO FARMING

데이터 전처리 및 검증

데이터 정합성 검증

- ✓ 농작물 매출 관련 결측치 데이터 NULL 값 처리
 - 쪽파 유통 비용 데이터 결측치
 - 참다래 생산 비용 데이터 결측치(2003년~2007년)
 - 복숭아 생산 면적, 생산량 데이터 결측치(2003년~2008년)
- ✓ NULL 값 변수들은 제외하고 분석 진행
- ✓ 기후 데이터는 해당 농작물 재배 기간의 평균값을 적용하여 사용
- ✓ 데이터 일관성을 유지하기 위해 소수점 두 자리로 통일

데이터 전처리

- ✓ 기후 데이터 중 기압, 상대습도, 풍속 데이터 삭제
(회귀분석을 사전에 진행한 결과 농작물 재배 상황에 큰 영향을 주지 않고, 실제 데이터로도 유의미한 차이를 나타내지 않음)
- ✓ 마지막 연도의 데이터 값만 존재하지 않는 경우, 3년 이동평균법을 사용하여 데이터 값을 대체
- ✓ 매출과 비용 데이터를 농지 면적(10a)를 기준으로 수치를 조절



TURN TO FARMING

모델링 및 분석 결과

양파 매출 분석 결과

설명 변수	B	t	p
Constant	-1.589e+07	-2.063	0.07298
온도	7.678e+05	2.408	0.04266
강수량	4.727e+03	1.335	0.21868
일조시간	-7.243e+03	-0.749	0.47539
일사량	1.972e+04	2.079	0.07126
생산비용	2.952e-01	0.228	0.82568
경작면적	-5.088e+02	-3.948	0.00425
생산량	-2.413e-01	-0.076	0.94093
상등품 도매가격	1.367e+02	1.373	0.20701
유통 비율	-8.279e+04	0.228	0.82568
GDP	-2.314e-02	-0.011	0.99178
Adjusted R ²	0.7878		
p - value	0.004056		

[표 1-1] 양파 매출 회귀 분석

Note : *p<0.05, **p<0.01, ***p<0.001
B : 가중치, t : 가중치/표준오차, p : 유의수준, p-value : 유의수준,
Adjusted R² : 조정된 결정계수

[표 1-1]의 회귀 분석 결과를 회귀 분석을 위한 변수 선택법으로
단계적 선택법(Stepwise Selection)을 사용하여 변수를 다시 선택
StepAIC(model, direction="both")

설명 변수	B	t	p
Constant	-1.950e+07	-4.419	0.000837
온도	8.248e+05	3.799	0.002532
강수량	4.908e+03	1.789	0.098863
일사량	1.946e+04	3.760	0.002719
경작면적	-5.248e+02	-7.047	1.34e-05
상등품 도매가격	1.881e+02	3.203	0.007595
유통 비율	-7.163e+04	-3.807	0.002499
Adjusted R ²	0.8425		
p - value	3.106e-05		

[표 1-2] 단계적 선택법을 사용한 양파 매출 회귀 분석

온도, 강수량, 일사량, 경작면적, 상등품 도매가격, 유통 비율이 독립변수로 선택

양파 예상 매출 = (-1.950e+07) + (8.248e+05 * 온도) + (4.908e+03 * 강수량) +
(1.946e+04 * 일사량) + (-5.248e+02 * 경작면적) + (1.881e+02 * 상등품 도매가격)
+ (-7.163e+04 * 유통 비율)

p-value : 3.106e-05
Adjusted R² : 0.8425 -> 84.25%

결론적으로 p-value와 Adjusted R² 값에 의해 본 회귀분석 모델은
유의하게 도출됨



TURN TO FARMING

모델링 및 분석 결과

고구마 매출 분석 결과

설명 변수	B	t	p
Constant	-3.271e+06	-1.110	0.2992
온도	8.817e+04	0.694	0.5076
강수량	2.474e+03	2.441	0.0405
일조시간	1.947e+03	1.210	0.2607
일사량	4.104e+02	0.290	0.7791
생산비용	1.236e+00	2.278	0.0523
경작면적	-5.013e+01	-0.326	0.7530
생산량	-7.559e+00	-1.703	0.1271
상등품 도매가격	1.140e+01	0.686	0.5119
유통 비율	1.181e+04	0.845	0.4224
GDP	7.747e-01	1.375	0.2064
Adjusted R ²	0.9811		
p - value	3.402e-07		

[표 2-1] 고구마 매출 회귀 분석

Note : *p<0.05, **p<0.01, ***p<0.001
B : 가중치, t : 가중치/표준오차, p : 유의수준, p-value : 유의수준,
Adjusted R² : 조정된 결정계수

[표 2-1]의 회귀 분석 결과를 회귀 분석을 위한 변수 선택법으로
단계적 선택법(Stepwise Selection)을 사용하여 변수를 다시 선택
StepAIC(model, direction="both")

설명 변수	B	t	p
Constant	-3.271e+06	-1.110	0.2992
온도	8.817e+04	0.694	0.5076
강수량	2.474e+03	2.441	0.0405
일조시간	1.947e+03	1.210	0.2607
생산비용	1.236e+00	2.278	0.0523
생산량	-7.559e+00	-1.703	0.1271
유통 비율	1.181e+04	0.845	0.4224
GDP	7.747e-01	1.375	0.2064
Adjusted R ²	0.9811		
p - value	3.402e-07		

[표 2-2] 단계적 선택법을 사용한 고구마 매출 회귀 분석

온도, 강수량, 일조시간, 생산비용, 생산량, 유통 비율, GDP가 독립변수로 선택

고구마 예상 매출 = (-3.271e+06) + (8.817e+04 * 온도) + (2.474e+03 * 강수량)
+ (1.947e+03 * 일조시간) + (1.236e+00 * 생산비용) + (-7.559e+00 * 생산량) +
(1.181e+04 * 유통 비율) + (7.747e-01 * GDP)

p-value : 3.402e-07
Adjusted R² : 0.9811 -> 98.11%

결론적으로 p-value와 Adjusted R² 값에 의해 본 회귀분석 모델은
유의하게 도출됨

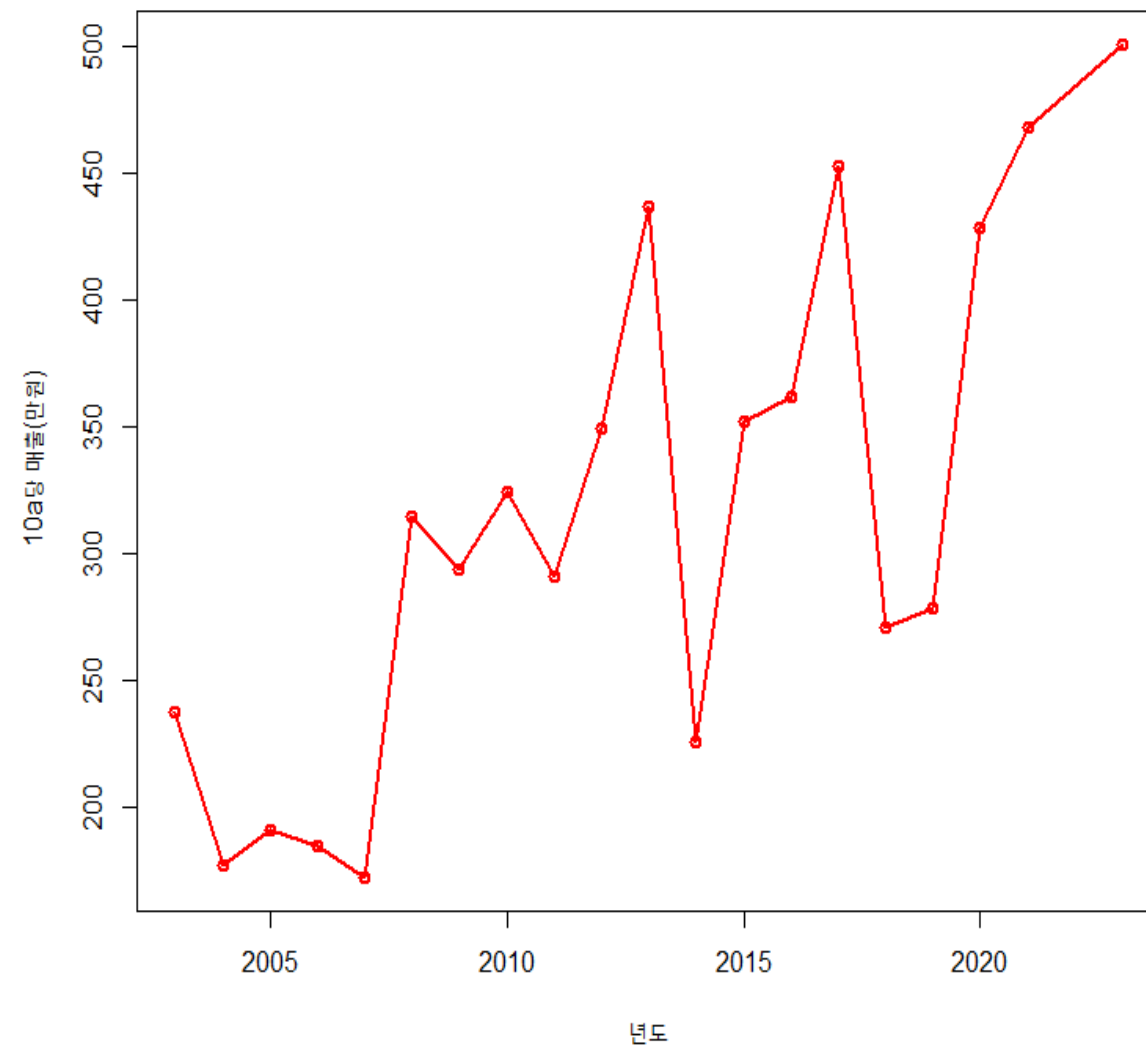


TURN TO FARMING

분석 결과 시각화

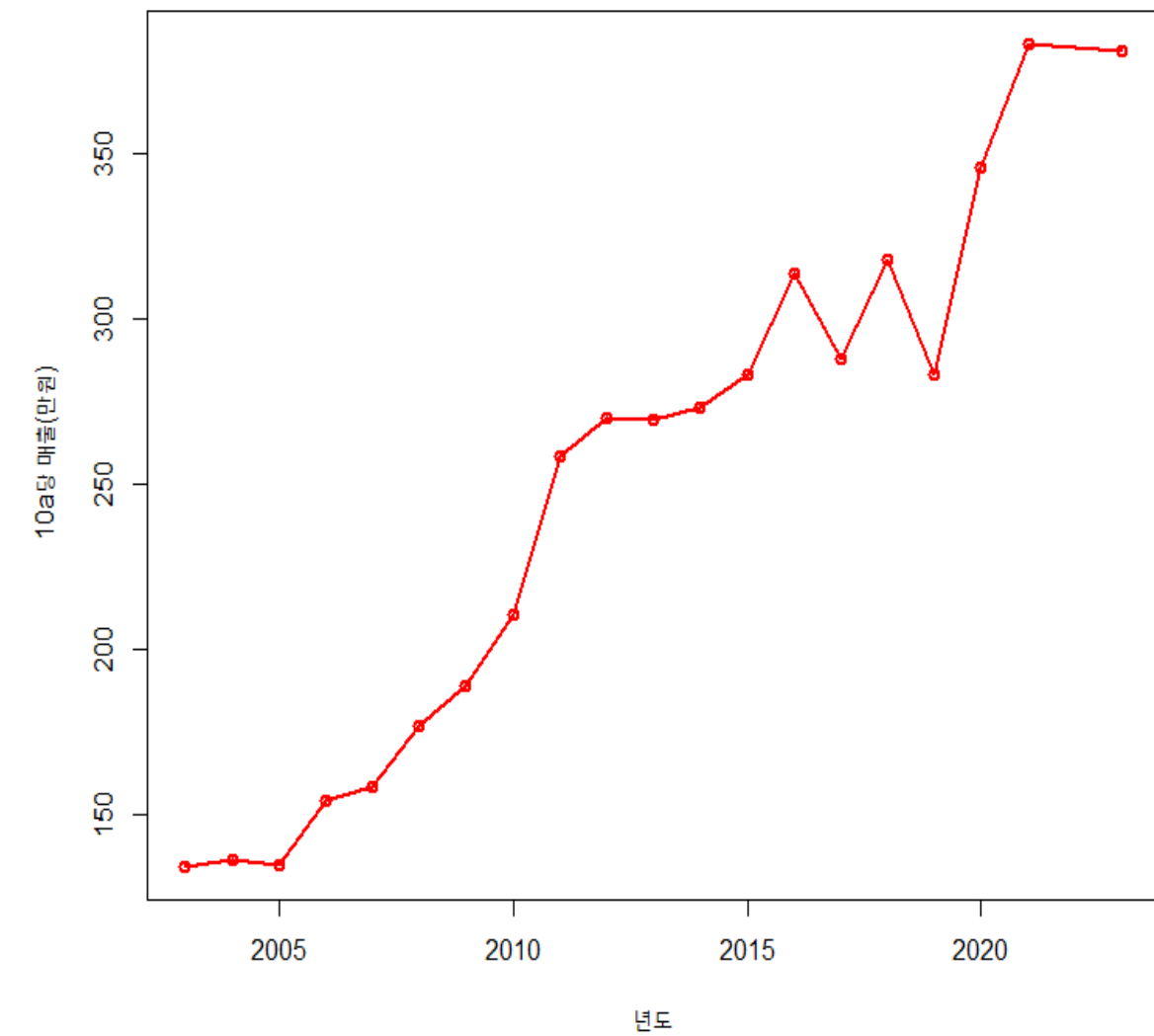
양파 매출 분석

양파 매출 변화



고구마 매출 분석

고구마 매출 변화



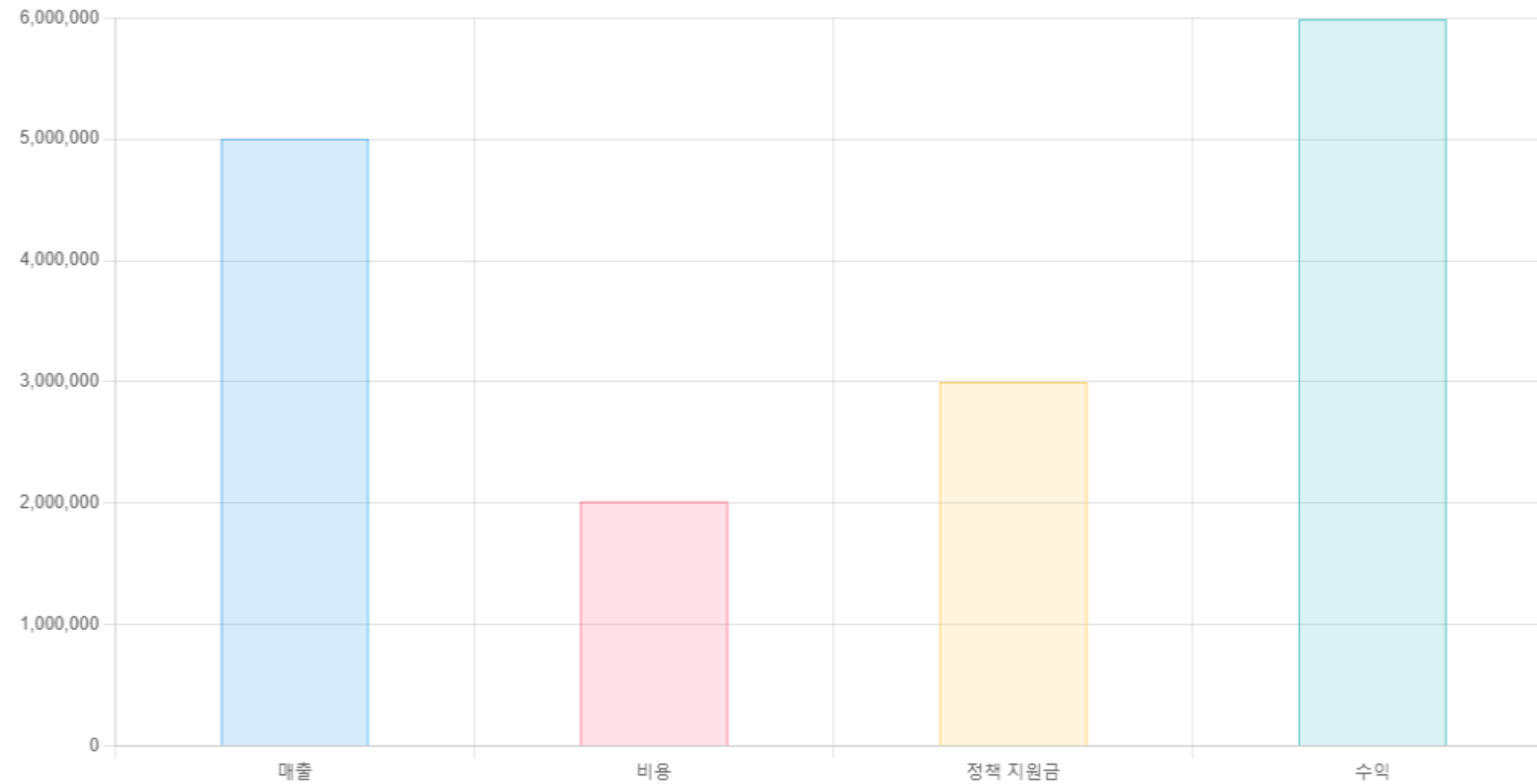
위 선그래프에서 마지막 점은 다중 선형 회귀 분석을 통한 예측값을 나타냅니다.



TURN TO FARMING

분석 결과 시각화

보성군 양파 수익 예측



보성군 양파 수익 분석				
예상 매출	평균 투자비	연간 운영비	정책 지원금	수익
5,005,700원	1,281,201원	735,620원	3,000,000원	5,988,879원

위 금액은 경작면적(10a), 노동인구(1인) 기준으로 분석하여 도출해낸 금액입니다.



TURN TO FARMING

요구사항 분석서

3. 수익분석 기능

3.1. 매출 예측 기능

3.1.1. 농작물 우선 선택 기능

- ▶ 먼저, 재배하고자 하는 농작물을 선택할 수 있어야 한다.
- ▶ 농작물을 선택할 시, 재배가 가능한 지역을 알려주고, 지도에 표시해서 보여줄 수 있어야 한다.
- ▶ 매출 확인 버튼을 클릭 시, 매출 예측 화면으로 이동할 수 있어야 한다.
- ▶ 매출 예측 화면은 모달창을 통해 보여줄 수 있어야 한다.
- ▶ 매출에 대한 정보는 그래프를 통해 보여주고, 과거부터 지금까지의 매출 추이를 나타내고 마지막 연도는 예측하려고 하는 연도의 값을 넣어서 보여줄 수 있어야 한다.
- ▶ 버튼을 통해 매출 예측 화면을 닫을 수 있어야 한다. 이 때, 화면을 닫아도 조건 선택화면이 리셋 되지 않아야 한다.
- ▶ 제출 버튼을 통해 매출 예측치를 개인 히스토리에 저장할 수 있어야 한다.

3.1.2. 지역 우선 선택 기능

- ▶ 먼저, 귀농하고자 하는 지역을 선택할 수 있어야 한다.
- ▶ 각 지역을 선택 시, 해당 지역에서 재배 가능한 농작물을 추천해서 보여줄 수 있어야 한다.
- ▶ 재배 가능한 농작물을 선택 시, 각 농작물에 대한 세부 정보(이름, 품종, 수확시기, 특징)를 보여줄 수 있어야 한다.
- ▶ 재배 가능한 농작물을 선택 시, 각 농작물에 대한 경매단가와 생산량에 대한 5년 추이를 그래프를 통해 보여줄 수 있어야 한다.
- ▶ 각 과정을 진행하기 위한 선택 과정 시, 자동으로 다음과과정으로 화면이 전환될 수 있어야 한다.
- ▶ 다시 선택버튼 클릭 시, 입력했던 조건이 모두 사라질 수 있어야 한다.
- ▶ 매출확인 버튼 클릭 시, 매출 예측 화면으로 이동할 수 있어야 한다.
- ▶ 매출 예측 화면은 모달창을 통해 보여줄 수 있어야 한다.
- ▶ 매출에 대한 정보는 그래프를 통해 보여주고, 과거부터 지금까지의 매출 추이를 나타내고 마지막 연도는 예측하려고 하는 연도의 값을 넣어서 보여줄 수 있어야 한다.
- ▶ 버튼을 통해 매출 예측 화면을 닫을 수 있어야 한다. 이 때, 화면을 닫아도 조건 선택화면이 리셋 되지 않아야 한다.

3.2. 비용 조회 기능

- ▶ 각 조건(농작물, 지역, 나이, 귀농 경력)을 선택할 수 있어야 한다.
 - ▷ 각 조건은 리스트 형식으로 제공할 수 있어야 한다.
 - ▷ 농작물 조건을 선택한 후, 해당 농작물에 재배 가능 지역을 선택할 수 있어야 한다.
 - ▷ 선택된 조건은 다른 색상으로 표현할 수 있어야 한다.
- ▶ 각 조건을 선택할 때, 먼저 선택한 조건들이 사라지지 않고 남아 있어야 한다.
 - ▷ 선택된 조건은 조건 리스트 아래에 저장될 수 있어야 한다.
- ▶ 조건(농작물, 지역)을 선택할 시, 검색을 통해 찾을 수 있어야 한다.
 - ▷ 검색창을 사용할 때, 단어가 완성될 때마다, 해당 단어가 포함된 결과를 보여줄 수 있어야 한다.
- ▶ 초기화 버튼을 클릭 시, 초기 상태로 돌아갈 수 있어야 한다.
- ▶ 검색 버튼을 클릭 시, 현재까지 입력했던 조건들에 맞는 정부 지원 정책을 게시판 리스트 형태로 보여줄 수 있어야 한다.
 - ▷ 검색을 누르면, 조건에 맞는 정책 리스트가 보여 지면서, 기존의 조건들은 초기화될 수 있어야 한다.
 - ▷ 지역, 정책 이름, 지원금의 형태로 정책 리스트가 구성될 수 있어야 한다.
- ▶ 정책 지원금 항목을 선택 할 수 있어야 한다.
 - ▷ 자신이 원하는 정책 지원금을 선택 한 후 정책 지원금 적용 버튼을 누르면 사용자가 선택한 정책 지원금으로 정보가 업데이트 되어야 한다.
- ▶ 조건에 해당하는 연간 운영비용과 선택한 정책 지원금을 계산하여 연간 총 비용을 보여줄 수 있어야 한다.
 - ▷ 계산된 비용이 화면에 보여 질 수 있어야 한다.
- ▶ 수익 확인 버튼을 클릭 시, 연간 총 운영비용의 정보를 수익 페이지로 전달할 수 있어야 한다.

3.3. 수익 확인 기능

- ▶ 농작물, 지역, 경작면적, 노동인구를 선택할 수 있어야 한다.
 - ▷ 조건 선택 항목은 드롭다운 형태로 제공해야 한다.
- ▶ 검색 버튼을 클릭하기 위해서는 농작물, 지역, 경작면적, 노동인구 조건을 모두 선택해야 한다.
- ▶ 초기화 버튼을 클릭 시, 선택한 조건들을 초기화 할 수 있어야 한다.
- ▶ 검색 버튼을 클릭 시, 선택한 조건을 기반으로 예상 매출, 비용, 정책 지원금, 수익을 그래프와 텍스트 형태로 보여줄 수 있어야 한다.
 - ▷ 매출, 비용, 정책 지원금, 예상 수익 정보를 막대그래프의 형태로 표현할 수 있어야 한다.
 - ▷ 매출, 비용, 정책 지원금, 예상 수익의 정보를 원 단위로 텍스트 형태로 표현할 수 있어야 한다.



TURN TO FARMING

요구사항 명세서

RQ-ID	화면 명	요구사항 명	요구사항 상세
RQ-ID-000-01	전체 페이지	상단 메뉴바	Home, 수익분석▼(매출 분석(농작물 우선), 매출 분석(지역 우선), 비용 조회, 수익 확인), 정부지원 정책들, 커뮤니티▼(지역축제&문화행사, 귀농 성공사례), 로그인(회원관리)로 구분되어 있고 각 구분된 영역 클릭 시 해당 기능으로 이동
RQ-ID-000-02	전체 페이지	로고	웹 페이지 왼쪽 상단에 클릭 시, 메인 화면으로 이동할 수 있는 로고를 삽입
RQ-ID-000-03	전체 페이지	웹 페이지 최하단 링크	웹 페이지의 각 기능으로 이동할 수 있는 링크를, 최하단부에 삽입
RQ-ID-000-04	전체 페이지	웹 페이지 최상단 이동 버튼	최상단으로 이동할 수 있는 버튼을 웹페이지 오른쪽 하단에 고정 삽입
RQ-ID-001-01	메인화면	메인 화면 이미지	자동으로 움직이는(캐러셀) 이미지를 상단 부분에 설정, 수동으로 이미지를 조정할 수 있는 버튼을 양 옆에 삽입
RQ-ID-001-02	메인화면	몸통 부분	몸통 부분에 각 기능으로 이동할 수 있는 박스 및 이미지를 삽입
RQ-ID-002-01	매출 예측(농작물 우선)	농작물 선택 리스트	추천 농작물의 전체 리스트를 체크하여 선택할 수 있도록 설정
RQ-ID-002-02	매출 예측(농작물 우선)	재배 가능 지역 조회	선택된 농작물을 재배할 수 있는 지역들을 리스트로 제공하고 직관적으로 확인할 수 있도록 지도를 통해 지역 정보 제공
RQ-ID-002-03	매출 예측(농작물 우선)	매출 확인 버튼	클릭시 선택된 농작물에 대한 예상 매출을 확인할 수 있는 화면으로 넘어갈 수 있도록 설정
RQ-ID-003-01	매출 예측(지역 우선)	지역 선택 지도	귀농을 원하는 지역을 지도와 버튼 형태로 제공하여 클릭 시 해당 지역에서 재배 가능한 농작물을 선택할 수 있는 화면으로 넘어갈 수 있도록 설정
RQ-ID-003-02	매출 예측(지역 우선)	재배 가능 농작물 조회	선택된 지역에서 재배 가능한 농작물 리스트를 이미지와 텍스트가 합쳐진 형태로 제공하고 클릭 시, 해당 농작물의 상세정보를 확인할 수 있도록 설정
RQ-ID-003-03	매출 예측(지역 우선)	농작물의 상세 정보 조회	선택한 농작물의 세부 정보(이름, 품종, 수확 시기, 특징)를 보여주고 경매 단가와 수확량의 지난 5년간 추이를 그래프로 표시
RQ-ID-003-04	매출 예측(지역 우선)	매출 확인 버튼	클릭 시 선택한 작물의 예상 매출을 확인할 수 있는 화면으로 넘어갈 수 있도록 설정
RQ-ID-004-01	매출 확인(모달)	예상 매출 조회	과거의 매출 추이와 예측 연도의 예상 매출 정보를 그래프를 통해 표시
RQ-ID-004-02	매출 확인(모달)	닫기 버튼	클릭 시, 예상 매출 화면에서 벗어나도록 설정
RQ-ID-005-01	비용 조회	비용 조회를 위한 조건 선택	비용과 정책 지원금 산정을 위한 조건(농작물, 지역, 나이, 귀농 경력)을 입력할 수 있도록 박스 형태로 설정
RQ-ID-005-02	비용 조회	선택 조건 확인	선택한 조건을 실시간으로 확인할 수 있도록 아래 박스 형태로 선택 조건 정보 제공
RQ-ID-005-03	비용 조회	농작물, 지역 검색	원하는 농작물과 지역을 키워드를 통해 찾아낼 수 있도록 설정
RQ-ID-005-04	비용 조회	초기화버튼	클릭 시 기존에 입력했던 모든 조건들을 초기화 하도록 설정
RQ-ID-005-05	비용 조회	정책 지원금 조회	조건 입력 후 검색버튼을 입력하면, 해당 조건에 부합하는 정책 리스트를 체크할 수 있는 상태로 제공
RQ-ID-005-06	비용 조회	정책 지원금 적용 버튼	정책 지원금을 선택하고 해당 버튼을 클릭하면 정책 지원금이 해당 조건으로 변경
RQ-ID-006-01	수익 확인	수익 확인을 위한 조건 선택	농작물, 지역, 경작면적, 노동인구를 선택하여 조회
RQ-ID-006-02	수익 확인	초기화 버튼	클릭 시, 선택했던 농작물과 지역이 초기화 되도록 설정
RQ-ID-006-03	수익 확인	검색 버튼	클릭 시, 입력했던 히스토리 정보를 가지고 수익 데이터를 막대그래프를 이용한 시각화를 하여 제공
RQ-ID-006-04	수익 확인	예상 수익 데이터 조회	계산된 수익데이터를 텍스트와 그래프를 통해 표시



TURN TO FARMING

화면 설계서

매출 분석 화면(농작물 우선)

프로젝트명	MZ세대를 위한 수익 예측 기반 귀농 추천 서비스	설명	농작물 우선 선택	페이지	2
화면 경로	수익 분석->매출 분석(농작물 우선)				



매출 확인

선호하는 농작물의 재배가능 지역과 매출을 확인해보세요.

1 추천 농작물 리스트

- 0 고구마
- 고추
- 대파
- 딸기
- 마늘
- 복숭아
- 시금치

2 재배가능지역

- 무안군
- 영광군
- 영암군
- 해남군

3 고구마 상세 정보



- 이름: 고구마
- 종 분류: 매꽃과의 여러해살이 뿌리채소
- 재배 시기: 5월 상순~중순
- 특이사항: 고구마재배에 있어서 비닐로 멀칭(피복)을 하면 보온 효과/보습 효과 (토양수분유지)/토양유실방지/잡초발생 억제 등의 효과가 있다.

4 경매 단가 확인

5 생산량 확인

Description	
1	추천 농작물 리스트를 제공.
2	농작물을 선택하면 해당 농작물의 재배 가능지역을 리스트로 제공.
3	해당 농작물의 상세 정보를 제공.
4	경매 단가 확인 버튼 - 해당 농작물의 경매 단가 정보를 제공.
5	생산량 확인 버튼 - 해당 농작물의 생산량 정보를 제공.

프로젝트명	MZ세대를 위한 수익 예측 기반 귀농 추천 서비스	설명	농작물 우선 선택	페이지	2
화면 경로	수익 분석->매출 분석(농작물 우선)				

추천 농작물 리스트

- 0 고구마
- 고추
- 대파
- 딸기
- 마늘
- 복숭아
- 시금치
- 양파
- 오이
- 정곡
- 쪽파
- 참다래
- 포도

재배가능지역

무안군

고구마 상세 정보



과외 여러해살이 뿌리채소 상순~중순 고구마재배에 있어서 비닐로 하면 보온 효과/보습 효과 (토양수분유지)/토양유실방지/잡초발생 억제가 있다.

생산량 확인

2 매출 분석

고구마 매출 추이(10a 당 만원)



3 close

1 매출 확인

4 비용 분석

Description	
1	매출 확인 버튼 - 클릭 시 매출 확인창으로 이동
2	농작물 매출 변화 추이 모달 창
3	닫기 버튼 - 클릭 시 매출 분석 모달창이 사라짐.
4	비용 분석 버튼 - 클릭 시 비용 조회 페이지로 이동.



TURN TO FARMING

화면 설계서

비용 조회 화면

프로젝트명	MZ세대를 위한 수익 예측 기반 귀농 추천 서비스	설명	비용 조회 화면	페이지	4
화면 경로	수익분석 -> 비용 조회				

WYC

Home 수익 분석 ▾ 정부지원 정책 커뮤니티 ▾ 회원관리 ▾

비용 조회

비용 조회

농작물별 비용과 정책을 간단하게 확인 하세요.

농작물별 비용과 정책 계산하기

1 농작물 ▾ 2 정책 조건 ▾ 5 초기화 6 검색

3 지역 검색

지역	시군구	나이	귀농 경력
전라남도	무안군 영광군 영암군 해남군	20~30살 30~40살 40~50살 50~60살 60살 이상	1년차(예비 귀농인 포함) 2년차 3년차 4년차 이상

4 고구마 무안군 30~40살 1년차

Description	
1	농작물 버튼 - 전체 농작물 리스트를 보여짐
2	정책 조건 버튼 - 지역, 시군구, 나이, 귀농경력 항목이 보여짐
3	검색 박스
4	결과창 - 선택한 조건들이 표시
5	선택한 조건 초기화 버튼
6	선택한 조건으로 지원 정책 정보를 검색하는 버튼

프로젝트명	MZ세대를 위한 수익 예측 기반 귀농 추천 서비스	설명	비용 조회 화면	페이지	4
화면 경로	수익분석 -> 비용 조회				

7 지역

지역	정부 지원 정책	정책 지원금
고흥군	귀농 창업 지원 (용자)	3,000,000원 ○ 8
고흥군	귀농 주택구입 지원 (용자)	3,000,000원 ○
고흥군	청년 문화 예술 취미동아리 운영	4,000,000원 ○
고흥군	도전! 청년농부 농지임차료등 지원	4,000,000원 ○
고흥군	귀농 귀촌인 살리기만 지원	5,000,000원 ○
고흥군	기업승계 청년농업인 지원	5,000,000원 ○
고흥군	귀향청년 빈집주택 수리비 지원	4,000,000원 ○
고흥군	귀향청년 U-turn(플러스) 정착장려금 지원(가구당)	4,000,000원 ○

정책 지원금 적용 9

10 소요 비용 0 - 정책 지원금 0 = 연간 총 비용 0 11 수익 확인

Our Office

Quick Links

Business Hours

Description	
7	지원 정책 리스트
8	정책 지원금 클릭 라디오 버튼 제공
9	선택한 정책 지원금 적용 버튼
10	분석 비용 정보 제공
11	수익 확인 버튼 - 수익 확인 페이지로 이동



TURN TO FARMING

화면 설계서

수익 확인 화면

프로젝트명	MZ세대를 위한 수익 예측 기반 귀농 추천 서비스	설명	수익 확인 화면	페이지	5
화면 경로	수익분석 -> 수익 확인				

WYC

Home 수익 분석 v 정부지원 정책 커뮤니티 v 회원관리 v

수익 확인

수익 확인

매출과 비용을 고려한 수익을 확인해보세요.

원하는 경작 면적과 노동 인구를 하세요.

1 2 3 4 5 6

농작물 선택 v 지역 선택 v 경작 면적 선택 v 노동 인구 선택 v 초기화 제출

Our Office

서울시 마포구 신촌로 104 4층, 5층
그린아트

☎ 02-715-2111

🌐 <http://sinchon.greenart.co.kr/>

Quick Links

- > 매출 확인(농작물무선)
- > 매출 확인(지역무선)
- > 비용 조회
- > 수익 확인
- > 정부지원 정책들
- > 지역축제 & 문화행사

Business Hours

Monday ~ Friday
09:30 am ~ 06:30 pm

Saturday
Closed

Sunday
Closed

Description
1 농작물 선택 리스트
2 지역 선택 리스트
3 경작 면적 선택 리스트
4 노동 인구 선택 리스트
5 조건들을 초기화 해주는 버튼
6 선택한 조건(농작물, 지역, 경작 면적, 노동 인구)에 관한 매출, 비용, 정책 지원금, 수익 정보 제공 및 그래프로 시각화

프로젝트명	MZ세대를 위한 수익 예측 기반 귀농 추천 서비스	설명	수익 확인 화면	페이지	5
화면 경로	수익분석 -> 수익 확인				

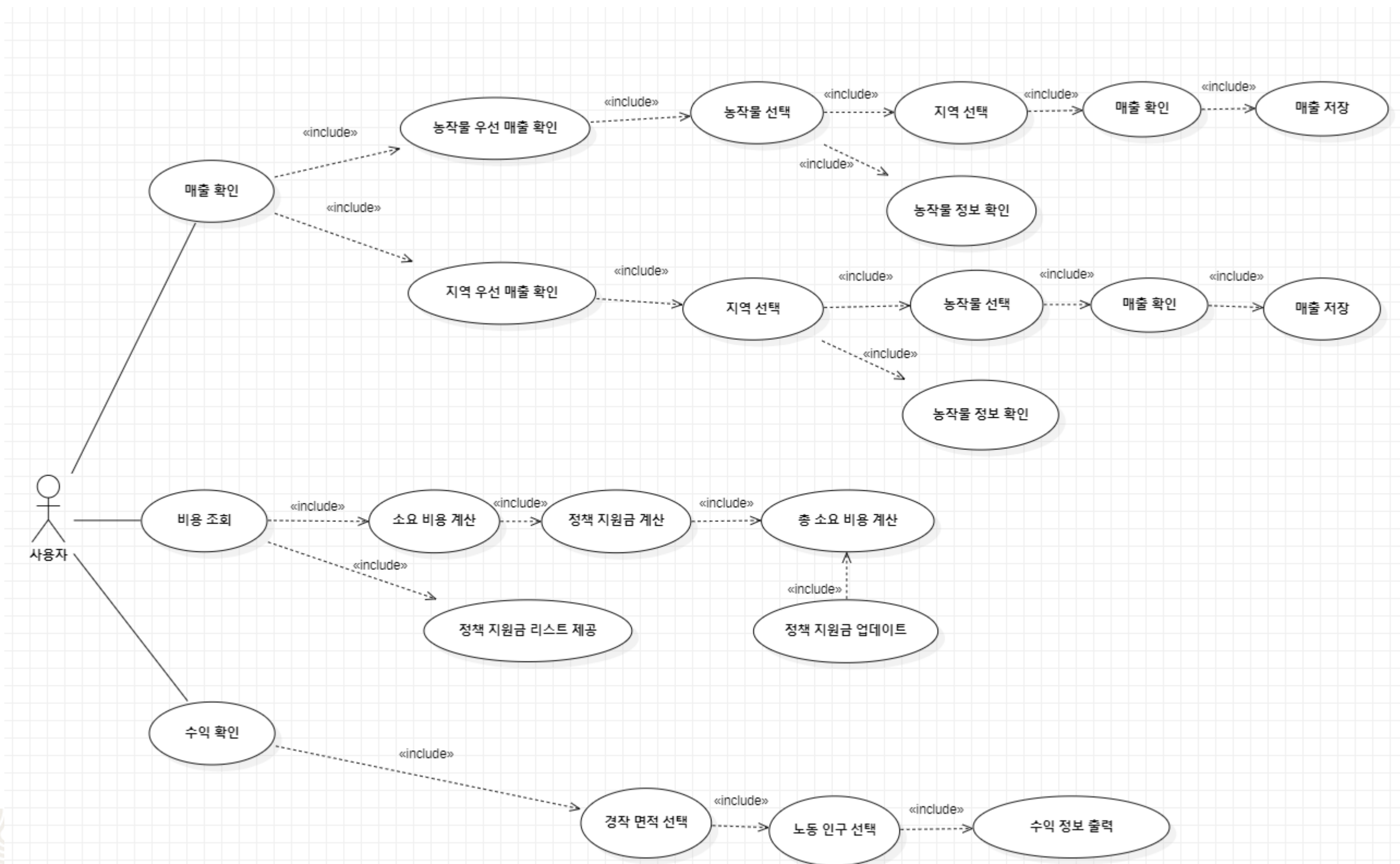


Description
7 선택한 조건(지역, 농작물, 경작 면적, 노동 인구)에 대한 수익 정보를 그래프로 시각화하여 제공
8 선택한 조건(지역, 농작물, 경작 면적, 노동 인구)에 대한 매출, 비용, 정책 지원금, 수익 정보를 제공



TURN TO FARMING

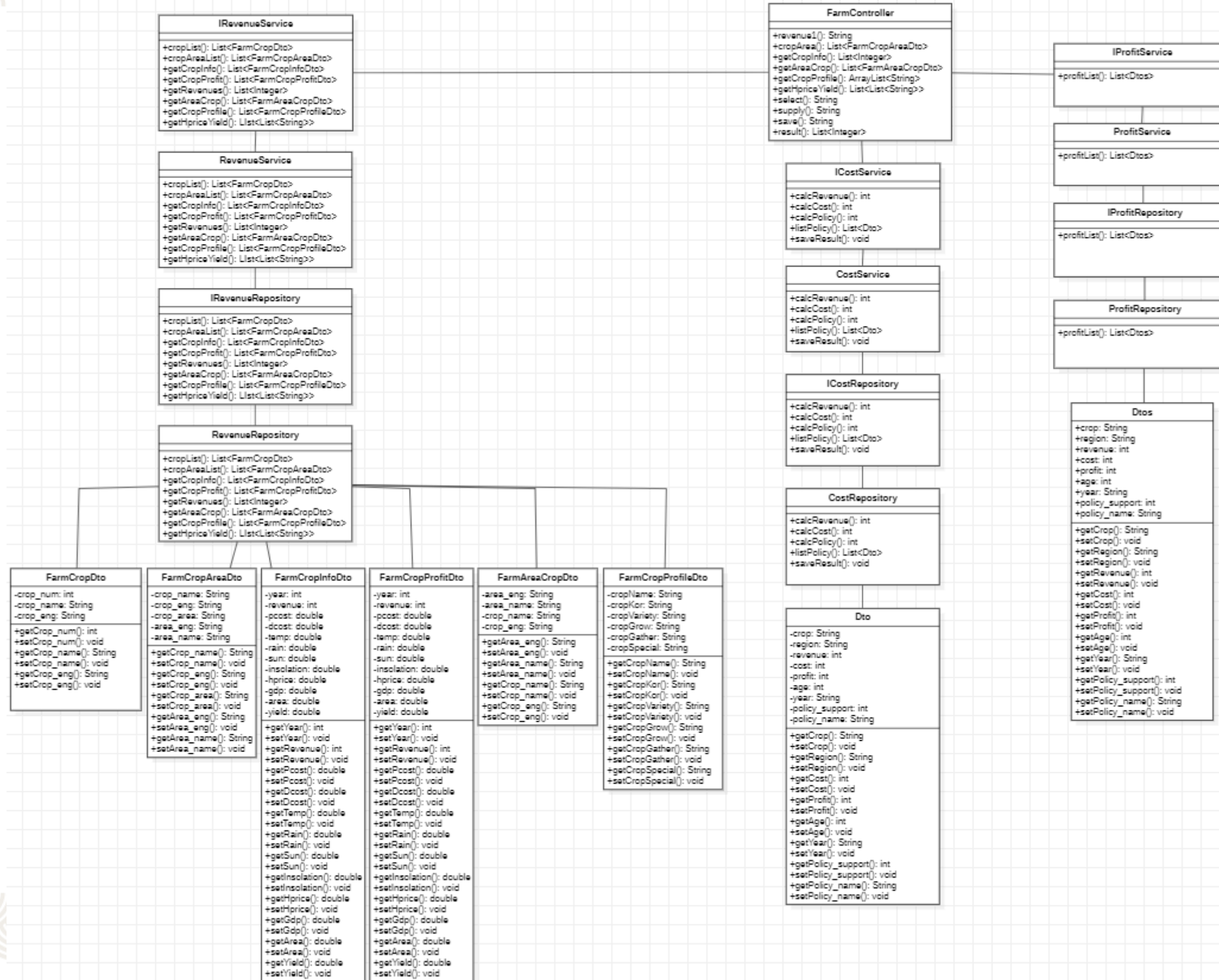
Usecase Diagram





TURN TO FARMING

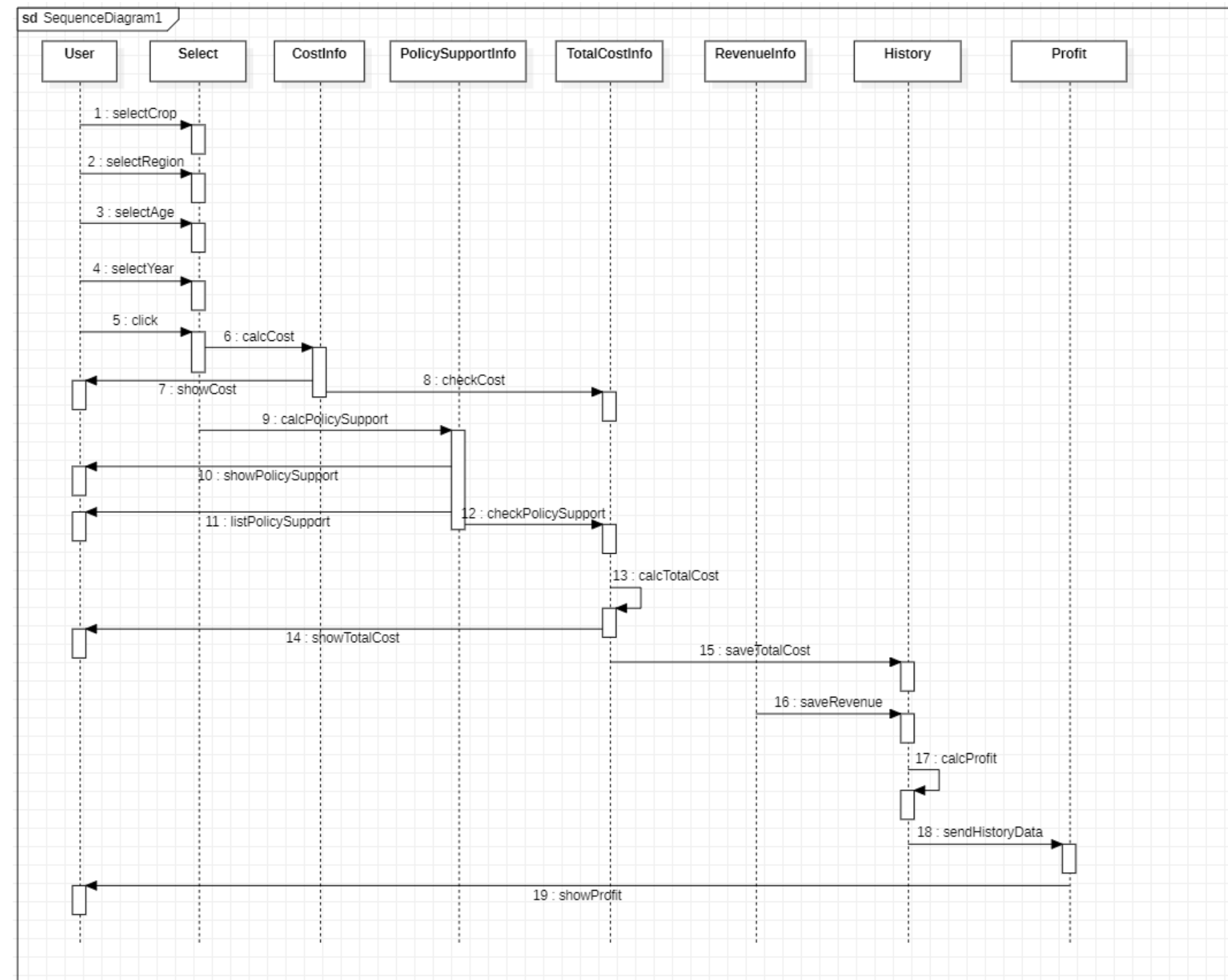
Class Diagram





TURN TO FARMING

Sequence Diagram





TURN TO FARMING ER Diagram

SCOTT.AREA_CROPS	
AREA_CODE	NUMBER (38)
AREA_ENG	VARCHAR2 (26 BYTE)
AREA_NAME	VARCHAR2 (26 BYTE)
CROP_NAME	VARCHAR2 (26 BYTE)
CROP_ENG	VARCHAR2 (26 BYTE)

SCOTT.CROPAREA	
CROP_NUM	NUMBER (38)
CROP_NAME	VARCHAR2 (26 BYTE)
CROP_ENG	VARCHAR2 (26 BYTE)
CROP_AREA1	VARCHAR2 (26 BYTE)
CROP_AREA2	VARCHAR2 (26 BYTE)
CROP_AREA3	VARCHAR2 (26 BYTE)
CROP_AREA4	VARCHAR2 (26 BYTE)
CROP_AREA5	VARCHAR2 (26 BYTE)
CROP_AREA6	VARCHAR2 (26 BYTE)
CROP_AREA7	VARCHAR2 (26 BYTE)
CROP_AREA8	VARCHAR2 (26 BYTE)

SCOTT.CROPLIST	
CROP_NUM	NUMBER (38)
CROP_NAME	VARCHAR2 (26 BYTE)
CROP_ENG	VARCHAR2 (26 BYTE)

SCOTT.CROPPREDICTION_MATRIX	
CROP	VARCHAR2 (26 BYTE)
REVENUE	NUMBER (38)
INTEREPT	NUMBER (38)
PCOST	NUMBER (38,4)
DCOST	NUMBER (38)
HPRICE	NUMBER (38,2)
TEMP	NUMBER (38)
RAIN	NUMBER (38)
SUN	NUMBER (38)
INSOLATION	NUMBER (38)
GDP	NUMBER (38,4)
AREA	NUMBER (38,2)
YIELD	NUMBER (38,3)

SCOTT.CROP_PROFILE	
CROPNAME	VARCHAR2 (26 BYTE)
CROPKOR	VARCHAR2 (26 BYTE)
CROPVARIETY	VARCHAR2 (100 BYTE)
CROPGROW	VARCHAR2 (100 BYTE)
CROPGATHER	VARCHAR2 (26 BYTE)
CROPSPECIAL	VARCHAR2 (500 BYTE)

SCOTT.TURN_TO_FARM_POLICY	
CROP	VARCHAR2 (26 BYTE)
REGION	VARCHAR2 (26 BYTE)
AGE	NUMBER (38)
YEAR	VARCHAR2 (26 BYTE)
POLICY_SUPPORT	NUMBER (38)
POLICY_NAME	VARCHAR2 (128 BYTE)

SCOTT.TURN_TO_FARM_REVENUE_COST	
CROP	VARCHAR2 (26 BYTE)
REGION	VARCHAR2 (26 BYTE)
REVENUE	NUMBER (38)
COST	NUMBER (38)
PROFIT	NUMBER (38)

SCOTT.CUCUMBER_MATRIX	
YEAR	NUMBER (38)
REVENUE	NUMBER (38)
PCOST	NUMBER (38)
DCOST	NUMBER (38,1)
HPRICE	NUMBER (38)
TEMP	NUMBER (38,2)
RAIN	NUMBER (38,2)
SUN	NUMBER (38,2)
INSOLATION	NUMBER (38,2)
GDP	NUMBER (38)
AREA	NUMBER (38)
YIELD	NUMBER (38)

SCOTT.GARLIC_MATRIX	
YEAR	NUMBER (38)
REVENUE	NUMBER (38)
PCOST	NUMBER (38)
DCOST	NUMBER (38,1)
HPRICE	NUMBER (38)
TEMP	NUMBER (38,2)
RAIN	NUMBER (38,2)
SUN	NUMBER (38,2)
INSOLATION	NUMBER (38,2)
GDP	NUMBER (38)
AREA	NUMBER (38)
YIELD	NUMBER (38)

SCOTT.GRAPE_MATRIX	
YEAR	NUMBER (38)
REVENUE	NUMBER (38)
PCOST	NUMBER (38)
DCOST	NUMBER (38,1)
HPRICE	NUMBER (38)
TEMP	NUMBER (38,2)
RAIN	NUMBER (38,2)
SUN	NUMBER (38,2)
INSOLATION	NUMBER (38,1)
GDP	NUMBER (38)
AREA	NUMBER (38)
YIELD	NUMBER (38)

SCOTT.GREENONION_MATRIX	
YEAR	NUMBER (38)
REVENUE	NUMBER (38)
PCOST	NUMBER (38)
DCOST	NUMBER (38,1)
HPRICE	NUMBER (38)
TEMP	NUMBER (38,2)
RAIN	NUMBER (38,2)
SUN	NUMBER (38,2)
INSOLATION	NUMBER (38,2)
GDP	NUMBER (38)
AREA	NUMBER (38)
YIELD	NUMBER (38)

SCOTT.KIWI_MATRIX	
YEAR	NUMBER (38)
REVENUE	NUMBER (38)
PCOST	NUMBER (38)
DCOST	NUMBER (38)
HPRICE	NUMBER (38)
TEMP	NUMBER (38,2)
RAIN	NUMBER (38,2)
SUN	NUMBER (38,2)
INSOLATION	NUMBER (38,2)
GDP	NUMBER (38)
AREA	NUMBER (38)
YIELD	NUMBER (38)

SCOTT.ONION_MATRIX	
YEAR	NUMBER (38)
REVENUE	NUMBER (38)
PCOST	NUMBER (38)
DCOST	NUMBER (38,1)
HPRICE	NUMBER (38)
TEMP	NUMBER (38,2)
RAIN	NUMBER (38,2)
SUN	NUMBER (38,2)
INSOLATION	NUMBER (38,2)
GDP	NUMBER (38)
AREA	NUMBER (38)
YIELD	NUMBER (38)

SCOTT.CHIVES_MATRIX	
YEAR	NUMBER (38)
REVENUE	NUMBER (38)
PCOST	NUMBER (38)
DCOST	NUMBER (38)
HPRICE	NUMBER (38)
TEMP	NUMBER (38,2)
RAIN	NUMBER (38,2)
SUN	NUMBER (38,2)
INSOLATION	NUMBER (38,2)
GDP	NUMBER (38)
AREA	NUMBER (38)
YIELD	NUMBER (38)

SCOTT.PEACH_MATRIX	
YEAR	NUMBER (38)
REVENUE	NUMBER (38)
PCOST	NUMBER (38)
DCOST	NUMBER (38,1)
HPRICE	NUMBER (38)
TEMP	NUMBER (38,2)
RAIN	NUMBER (38,2)
SUN	NUMBER (38,2)
INSOLATION	NUMBER (38,2)
GDP	NUMBER (38)
AREA	NUMBER (38)
YIELD	NUMBER (38)

SCOTT.PEPPER_MATRIX	
YEAR	NUMBER (38)
REVENUE	NUMBER (38)
PCOST	NUMBER (38)
DCOST	NUMBER (38,1)
HPRICE	NUMBER (38)
TEMP	NUMBER (38,2)
RAIN	NUMBER (38,2)
SUN	NUMBER (38,2)
INSOLATION	NUMBER (38,2)
GDP	NUMBER (38)
AREA	NUMBER (38)
YIELD	NUMBER (38)

SCOTT.RICE_MATRIX	
YEAR	NUMBER (38)
REVENUE	NUMBER (38)
PCOST	NUMBER (38)
DCOST	NUMBER (38,1)
HPRICE	NUMBER (38)
TEMP	NUMBER (38,2)
RAIN	NUMBER (38,2)
SUN	NUMBER (38,2)
INSOLATION	NUMBER (38,2)
GDP	NUMBER (38)
AREA	NUMBER (38)
YIELD	NUMBER (38)

SCOTT.SPINACH_MATRIX	
YEAR	NUMBER (38)
REVENUE	NUMBER (38)
PCOST	NUMBER (38)
DCOST	NUMBER (38)
HPRICE	NUMBER (38)
TEMP	NUMBER (38,2)
RAIN	NUMBER (38,2)
SUN	NUMBER (38,2)
INSOLATION	NUMBER (38,2)
GDP	NUMBER (38)
AREA	NUMBER (38)
YIELD	NUMBER (38)

SCOTT.SPOTATO_MATRIX	
YEAR	NUMBER (38)
REVENUE	NUMBER (38)
PCOST	NUMBER (38)
DCOST	NUMBER (38,1)
TEMP	NUMBER (38,2)
RAIN	NUMBER (38,2)
SUN	NUMBER (38,2)
INSOLATION	NUMBER (38,2)
HPRICE	NUMBER (38)
GDP	NUMBER (38)
AREA	NUMBER (38)
YIELD	NUMBER (38)

SCOTT.STBERRY_MATRIX	
YEAR	NUMBER (38)
REVENUE	NUMBER (38)
PCOST	NUMBER (38)
DCOST	NUMBER (38,1)
HPRICE	NUMBER (38)
TEMP	NUMBER (38,2)
RAIN	NUMBER (38,2)
SUN	NUMBER (38,2)
INSOLATION	NUMBER (38,2)
GDP	NUMBER (38)
AREA	NUMBER (38)
YIELD	NUMBER (38)

SCOTT.TOMATO_MATRIX	
YEAR	NUMBER (38)
REVENUE	NUMBER (38)
PCOST	NUMBER (38)
DCOST	NUMBER (38)
HPRICE	NUMBER (38)
TEMP	NUMBER (38,2)
RAIN	NUMBER (38,2)
SUN	NUMBER (38,2)
INSOLATION	NUMBER (38,2)
GDP	NUMBER (38)
AREA	NUMBER (38)
YIELD	NUMBER (38)



TURN TO FARMING

주요 서비스 기능

농작물 매출 분석 서비스


추천 농작물 리스트

- ☒ 고구마
- ☐ 고추
- ☐ 대파
- ☐ 딸기
- ☐ 마늘
- ☐ 복숭아
- ☐ 시금치
- ☐ 양파
- ☐ 오이
- ☐ 정국
- ☐ 쪽파
- ☐ 참다래
- ☐ 토마토
- ☐ 포도

재배 가능 지역

무안군
영광군
영암군
해남군

고구마 상세 정보



- 이름 : 고구마
- 종 분류 : 매꽃과의 여러해살이 뿌리채소
- 재배 시기 : 5월 상순~중순
- 특이사항 : 고구마재배에 있어서 비닐로 멀칭(피복)을 하면 보온 효과/보습 효과(토양수분유지)/토양유실방지/잡초발생억제 등의 효과가있다.

경매 단가 확인

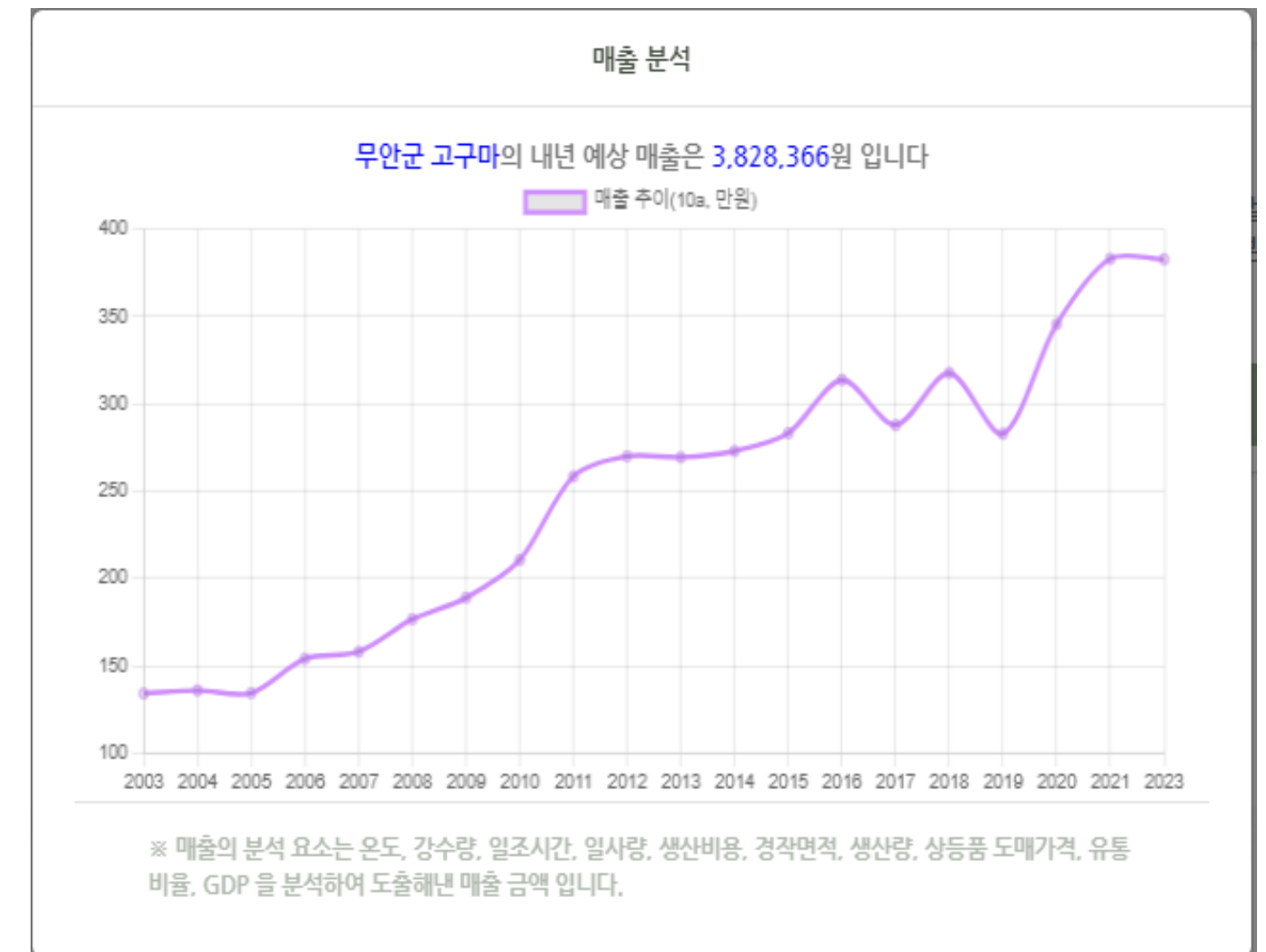
생산량 확인

매출 확인

비용 분석

- ✓ 농작물과 지역을 선택하여 매출 추이 변화 확인 가능
- ✓ 농작물의 상세 정보, 경매 단가, 생산량 확인 가능

농작물 매출 분석 및 예측



- ✓ 선택한 농작물과 지역 조건에 대한 매출을 확인 가능
- ✓ 마지막 년도는 다중 선형 회귀 분석을 사용하여 예측한 매출



TURN TO FARMING

주요 서비스 기능

비용 조회 서비스

농작물별 비용과 정책 지원금 계산하기

농작물 ▼

정책 조건 ▼

초기화

검색

지역 검색

광역시/도	시군구	나이	귀농 경력
전라남도	무안군	20~30살	1년차(예비귀농인 포함)
	영광군	30~40살	2년차
	영암군	40~50살	3년차
	해남군	50~60살	4년차 이상
		60살 이상	

고구마

무안군

30~40살

1년차(예비귀농인 포함)

※ 아래 정부 지원 정책에 맞는 정책지원금 항목을 클릭하고 정책지원금 적용 버튼을 눌러주세요.

지역	정부 지원 정책	정책 지원금
무안군	귀농 창업 활성화 지원(창업 컨설팅)	2,000,000원
무안군	귀농 창업 활성화 지원(예비 창업실행비)	2,000,000원
무안군	귀농인 농가주택수리비 지원 사업	3,000,000원
무안군	귀농 농업창업 지원사업	3,000,000원
무안군	귀농 주택구입 지원사업	4,000,000원
무안군	귀농인 정착 지원 사업	5,000,000원

정책 지원금 적용

※ 해당 소요 비용은 경작면적 10a(1000㎡)기준으로 분석하여 도출해낸 소요 비용 금액입니다.

소요 비용	정책 지원금		연간 총 비용	수익 확인
1,480,633원	-	2,000,000원	= -519,367원	

- ✓ 농작물, 지역, 나이, 귀농 경력에 따른 비용 및 정책 지원금 확인 가능
- ✓ 사용자가 선택한 정부 지원 정책에 따라 정책 지원금 변경

수익 확인 서비스

원하는 경작 면적과 노동 인구를 선택하세요.

고구마

▼ 무안군

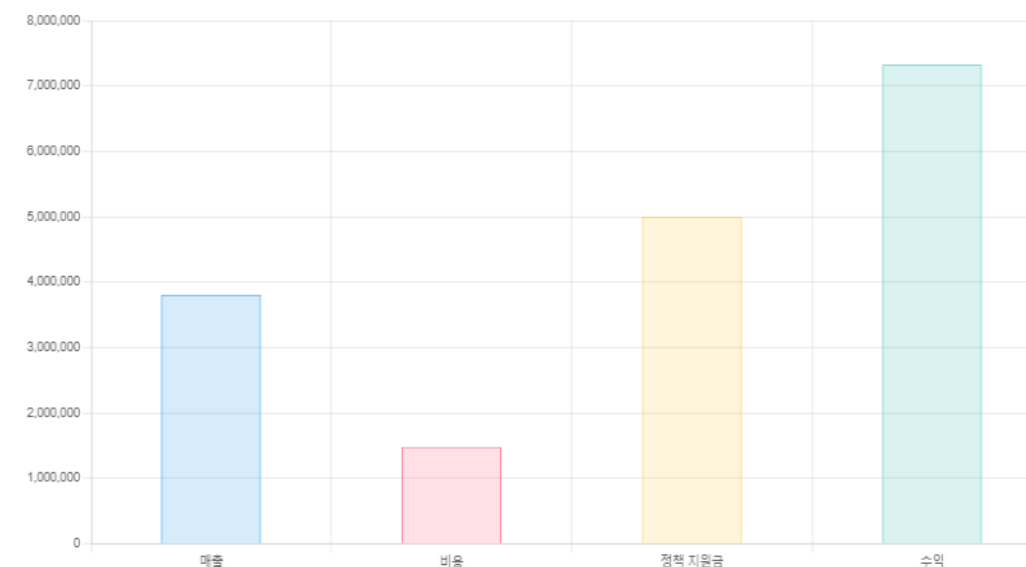
▼ 10a

▼ 1명

▼

초기화

검색



※ 해당 비용은 농작물, 지역, 경작 면적, 노동 인구를 종합한 비용입니다.

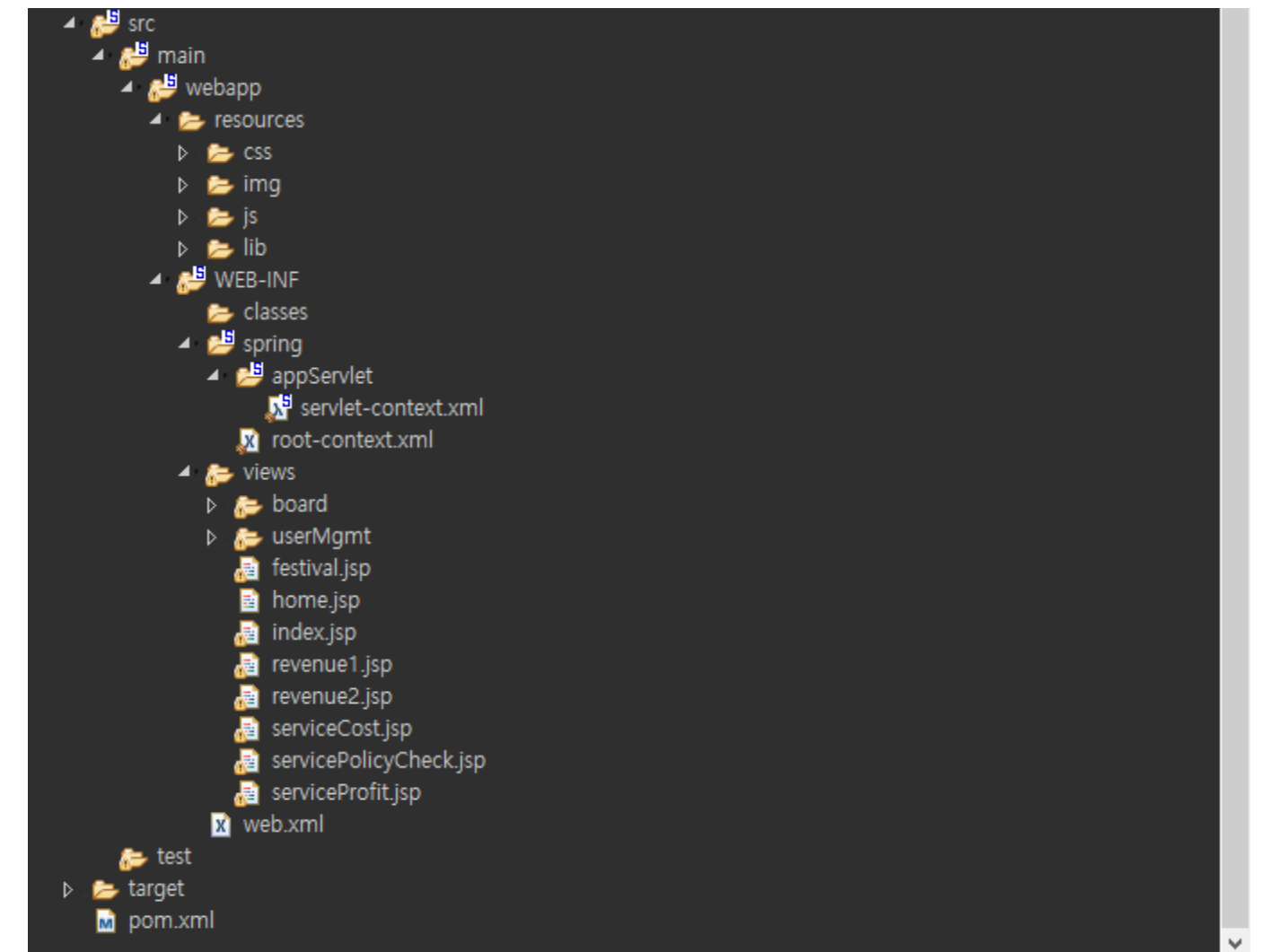
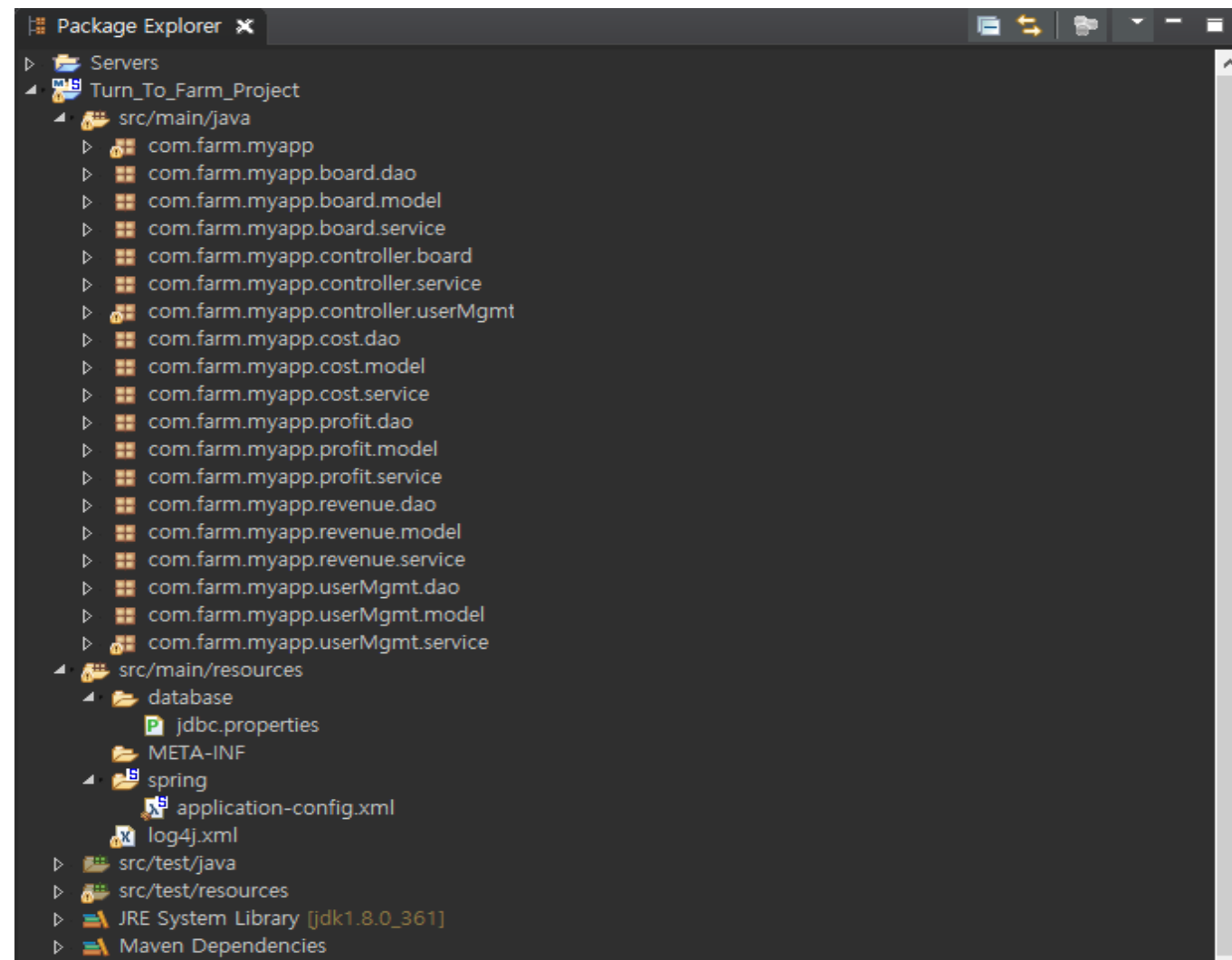
매출	비용	정책 지원금	수익
3,809,900원	1,480,633원	5,000,000원	7,329,267원

- ✓ 경작 면적과 노동 인구에 따른 매출, 비용, 수익 변화
- ✓ 예상 매출, 비용, 정책 지원금, 수익 확인 가능

TURN TO FARMING

소스 코드

프로젝트 구조 및 패키지 구성



- ✓ 3가지 주요 기능을 위한 컨트롤러 구성(서비스, 게시판, 회원 관리)
- ✓ 각 기능에 따라 서로 다른 패키지로 구성
(revenue, cost, profit, board, userMgmt)



TURN TO FARMING

소스 코드

SPRING WEB MVC(Controller)

```
@Controller
public class FarmController {

    @Autowired
    IRevenueService revenueService;

    @RequestMapping(value="/")
    public String home(Model model) {
        return "/index";
    }

    @RequestMapping(value="/index.do") // 메인 페이지로 이동
    public String index(Model model) {
        return "/index";
    }

    @RequestMapping(value="/revenue1.do") //농작물 우선 매출 페이지로 이동, 농작물 리스트 표시
    public String revenue1(Model model) {
        List<FarmCropDto> cropList = revenueService.cropList();
        model.addAttribute("cropList", cropList);
        return "/revenue1";
    }

    @RequestMapping(value="/cropArea.do") //재배가능지역 출력
    public @ResponseBody List<FarmCropAreaDto> cropArea(@RequestParam(value="crop", required=false, defaultValue="0")
String selectedCrop, Model model) {
        List<FarmCropAreaDto> cropAreaList = revenueService.cropAreaList(selectedCrop);
        return cropAreaList;
    }
}
```

```
@Autowired
ICostService costService;

@RequestMapping(value="/select.do") // 비용 조회
public String select(@RequestParam("crop_policy_checkbox") String crop, @RequestParam("select_area_radio_2")
String region, HttpServletRequest request, HttpSession session, Model model) {
    int calcRevenue = costService.calcRevenue(crop, region);

    String strCrop = crop;
    String strRegion = region;
    int revenue = calcRevenue;

    session.setAttribute("crop", strCrop);
    session.setAttribute("region", strRegion);
    session.setAttribute("revenue", revenue);

    int calcCost = costService.calcCost(crop, region);

    int cost = calcCost;

    session.setAttribute("cost", cost);

    int calcPolicy = costService.calcPolicy(region);

    List<Dto> listPolicy = costService.listPolicy(region);

    model.addAttribute("region", strRegion);
    model.addAttribute("cost", cost);
    model.addAttribute("policy_support", calcPolicy);
    model.addAttribute("list", listPolicy);

    return "/serviceCost";
}
```

- ✓ FarmController.java 소스 코드
- ✓ 클라이언트 요청을 받아 해당 요청을 처리하고 필요한 비즈니스 로직을 실행하거나 생성하여 뷰(View)로 전달



TURN TO FARMING

소스 코드

SPRING WEB MVC(Model)

```
@Repository
public class RevenueRepository implements IRevenueRepository{

    @Autowired
    JdbcTemplate jdbcTemplate;

    public class CropListMapper implements RowMapper<FarmCropDto>{

        @Override
        public FarmCropDto mapRow(ResultSet rs, int count) throws SQLException {
            FarmCropDto dtos = new FarmCropDto();
            dtos.setCrop_num(rs.getInt("crop_num"));
            dtos.setCrop_name(rs.getString("crop_name"));
            dtos.setCrop_eng(rs.getString("crop_eng"));
            return dtos;
        }
    }

    public class CropAreaListMapper implements RowMapper<FarmCropAreaDto>{

        @Override
        public FarmCropAreaDto mapRow(ResultSet rs, int count) throws SQLException {
            FarmCropAreaDto dtos = new FarmCropAreaDto();
            dtos.setArea_eng(rs.getString("area_eng"));
            dtos.setArea_name(rs.getString("area_name"));
            return dtos;
        }
    }
}
```

```
@Override
public List<FarmCropDto> cropList() {
    System.out.println("cropList() 실행");
    String sql = "SELECT * FROM croplist ORDER BY crop_name";

    return jdbcTemplate.query(sql, new RowMapper<FarmCropDto>() {
        @Override
        public FarmCropDto mapRow(ResultSet rs, int count) throws SQLException {
            FarmCropDto dtos = new FarmCropDto();
            dtos.setCrop_num(rs.getInt("crop_num"));
            dtos.setCrop_name(rs.getString("crop_name"));
            dtos.setCrop_eng(rs.getString("crop_eng"));
            return dtos;
        }
    });
}

@Override
public List<FarmCropAreaDto> cropAreaList(String crop){
    System.out.println("cropAreaList() 실행");
    String sql = "SELECT * FROM area_Crops WHERE crop_eng=? ORDER BY area_name";

    return jdbcTemplate.query(sql, new Object[] {crop}, new RowMapper<FarmCropAreaDto>() {
        @Override
        public FarmCropAreaDto mapRow(ResultSet rs, int count) throws SQLException {
            FarmCropAreaDto dtos = new FarmCropAreaDto();
            dtos.setArea_eng(rs.getString("area_eng"));
            dtos.setArea_name(rs.getString("area_name"));
            return dtos;
        }
    });
}
```

- ✓ RevenueRepository.java 소스 코드
- ✓ 데이터베이스와 연결을 설정하고 접근하여 데이터를 읽고 쓰는 역할



TURN TO FARMING

소스 코드

MyBatis

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"
    "http://mybatis.org/dtd/mybatis-3-mapper.dtd">
<mapper namespace = "com.farm.myapp.revenue.dao.IRevenueRepository">

    <resultMap id="cropList" type="com.farm.myapp.revenue.model.FarmCropDto">
        <result property="crop_num" column="crop_num"/>
        <result property="crop_name" column="crop_name"/>
        <result property="crop_eng" column="crop_eng"/>
    </resultMap>

    <resultMap id="cropArealist" type="com.farm.myapp.revenue.model.FarmCropAreaDto">
        <result property="area_eng" column="area_eng"/>
        <result property="area_name" column="area_name"/>
        <result property="crop_name" column="crop_name"/>
        <result property="crop_eng" column="crop_eng"/>
    </resultMap>

    <resultMap id="getCropInfo" type="com.farm.myapp.revenue.model.FarmCropInfoDto">
        <result property="year" column="year"/>
        <result property="revenue" column="revenue"/>
        <result property="pcost" column="pcost"/>
        <result property="dcost" column="dcost"/>
        <result property="temp" column="temp"/>
        <result property="rain" column="rain"/>
        <result property="sun" column="sun"/>
        <result property="insolation" column="insolation"/>
        <result property="hprice" column="hprice"/>
        <result property="gdp" column="gdp"/>
        <result property="area" column="area"/>
        <result property="yield" column="yield"/>
    </resultMap>

</mapper>
```

```
<select id="cropList" resultType="com.farm.myapp.revenue.model.FarmCropDto">
    SELECT * FROM cropList ORDER BY crop_name
</select>

<select id="cropArealist" parameterType="String" resultType="com.farm.myapp.revenue.model.FarmCropAreaDto">
    SELECT * FROM area_Crops WHERE crop_eng=#{crop_eng} ORDER BY area_name
</select>

<select id="getCropInfo" parameterType="String" resultType="com.farm.myapp.revenue.model.FarmCropInfoDto">
    SELECT * FROM ${selectCrop}_matrix WHERE year=2021
</select>

<select id="getCropProfit" parameterType="String" resultType="com.farm.myapp.revenue.model.FarmCropProfitDto">
    SELECT * FROM cropPrediction_matrix WHERE crop='${selectCrop}'
</select>

<select id="getRevenues" parameterType="String" resultType="int">
    SELECT revenue FROM ${selectCrop}_matrix ORDER BY year
</select>

<select id="getCropProfile" parameterType="String" resultType="com.farm.myapp.revenue.model.FarmCropProfileDto">
    SELECT * FROM crop_profile WHERE cropName=#{cropName}
</select>

<select id="getAreaCrop" parameterType="String" resultType="com.farm.myapp.revenue.model.FarmAreaCropDto">
    SELECT * FROM area_crops WHERE area_eng=#{crop_eng}
</select>

<select id="getHpriceYield" parameterType="String" resultType="com.farm.myapp.revenue.model.FarmHpriceYieldDto">
    SELECT year, hPrice, yield FROM ${selectCrop}_matrix WHERE year IN (2017, 2018, 2019, 2020, 2021)
</select>

</mapper>
```

- ✓ RevenueMapper.xml 소스 코드
- ✓ 앞 페이지 RevenueRepository.java 파일을 MyBatis로 전환



TURN TO FARMING

소스 코드

RStudio를 사용한 데이터 분석

```
1 #디렉토리 위치 설정
2 setwd("C:/Temp/R_data/matrix")
3 #csv 파일을 읽음
4 pepper <- read.csv("pepper_matrix.csv", header=T)
5 head(pepper)
6 pepper
7
8 #선형 회귀 분석을 위해 'lm()'함수를 사용
9 mod <- lm(revenue~Pcost+Dcost+Hprice+temp+rain+sun+insolation+gdp+area+yield, data=pepper)
10 mod
11 summary(mod)
12
13 #단계적 선택법을 사용한 회귀 분석
14 library(MASS)
15 mod2 <- stepAIC(mod,direction = "both")
16 mod2
17 summary(mod2)
18
19 #23년도 매출을 다른변수 조건을 마지막 년도와 조건이 동일하다고 생각하고 매출 예상
20 Pcost <- c(pepper[19,3])
21 Hprice <- c(pepper[19,5])
22 sun <- c(pepper[19,8])
23 area <- c(pepper[19,11])
24 result <- 7.991e+06 + 1.886e+00*Pcost -1.553e+01*Hprice -1.472e+04*sun -2.270e+02*area
25 result
26
27 #년도별 매출 추이를 분석하기 위해 값 입력
28 revenue <- c((pepper$revenue)/10000, result/10000)
29 revenue
30 year <- c((pepper$year), 2023)
31 year
32
33 #년도별 매출 추이 그래프 생성
34 plot(year, revenue, main="고추 매출 변화", type="o", lty=1, lwd=2, col="red", xlab="년도", ylab="10a당 매출(만원)")
```

```
Call:
lm(formula = revenue ~ Pcost + Hprice + sun + area, data = pepper)

Residuals:
    Min       1Q   Median       3Q      Max
-903745 -193363  -23362   299450   850184

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  7.991e+06   2.071e+06   3.858 0.001739 **
Pcost        1.886e+00   4.353e-01   4.332 0.000689 ***
Hprice       -1.553e+01   4.664e+00  -3.330 0.004959 **
sun          -1.472e+04   5.895e+03  -2.497 0.025595 *
area         -2.270e+02   1.285e+02  -1.767 0.098925 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 503500 on 14 degrees of freedom
Multiple R-squared:  0.7909,    Adjusted R-squared:  0.7311
F-statistic: 13.24 on 4 and 14 DF,  p-value: 0.0001143
```

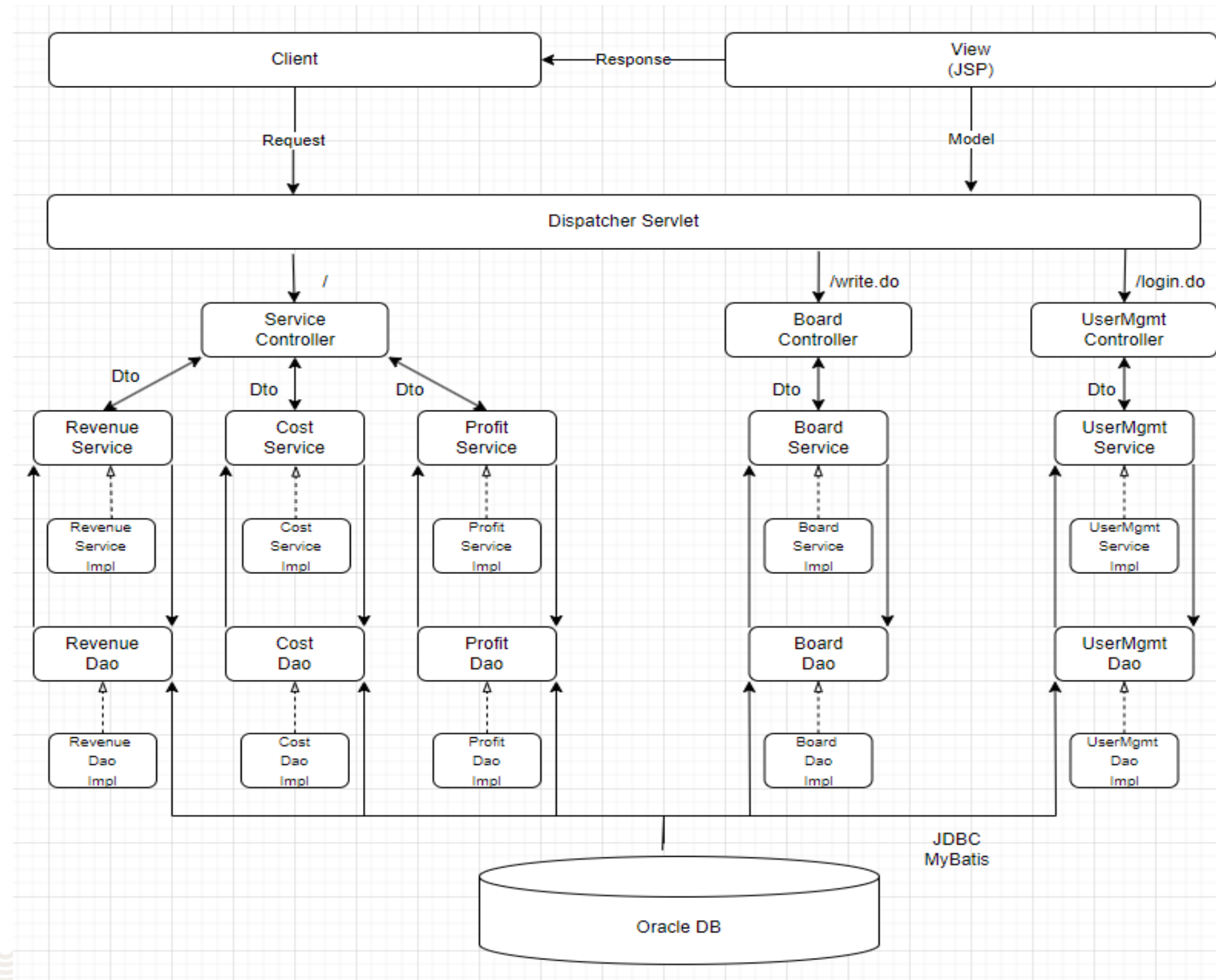
- ✓ 선형 회귀 분석을 위해 lm()함수를 사용
- ✓ 회귀 분석을 위한 변수 선택법으로 단계적 선택법을 사용
(StepAIC(mod, direction="both"))
- ✓ 연도별 매출 추이 그래프 생성을 위해 plot()함수를 사용

- ✓ 단계적 선택법을 사용한 선형 회귀 분석 결과



TURN TO FARMING

Software Architecture



TURN TO FARMING

시연



[Home](#) [수익 분석 ▼](#) [지원 정책](#) [커뮤니티 ▼](#) [로그인](#) [회원가입](#)

당신이 선호하는 농작물의 지역과 매출을 확인해보세요!

Welcome to Jeonnam!

매출 분석(농작물 우선)



TURN TO FARMING

향후 계획

- ✓ 프로젝트를 Spring Boot로 전환
 - 더욱 현대적이고 효율적인 어플리케이션으로 전환
- ✓ 보다 현대적인 프론트엔드 기술 도입
 - React 또는 Vue.js 적용
- ✓ 선형 회귀 분석 모델 성능 향상 기법 도입
 - 특성공학(feature engineering)
 - 규제(regularization)
 - 확률적 경사 하강법(Stochastic Gradient Descent)
- ✓ 데이터 전처리 및 정제 강화
 - 이상치 처리
 - 결측치 처리



TURN TO FARMING

프로젝트 수행 소감

- ✓ 비동기 데이터 입력과 Ajax 활용으로 극복한 개발 과정
 - 프로젝트를 수행하면서, 웹 페이지의 데이터 입력을 위해 form 태그를 활용하면서 비동기적으로 데이터를 처리하고자 Ajax를 사용하는 과정에서 어려움을 마주하게 되었습니다. 초기에는 이러한 비동기 처리 방식이 익숙하지 않아서 어떻게 해결해야 할지 막막한 순간이 있었지만, 끈기와 노력으로 이 어려움을 극복해냈다는 자신감을 얻게 되었습니다.
- ✓ Bean설정과 DI의 역할과 중요성
 - 이번 프로젝트를 통해 Spring Framework의 강력한 기능과 유연성에 대해 깊이 이해할 수 있었습니다. Bean 설정 과 의존성 주입(Dependency Injection)의 개념은 처음에는 이해하기 어려웠지만, 프로젝트를 진행하면서 실제로 적용하고 나니 코드의 모듈성과 유지보수성이 크게 향상되는 것을 깨달았습니다.
- ✓ 다양한 Spring 모듈 활용으로 개발 능력 강화
 - Spring의 다양한 모듈을 활용하여 개발하는 방법을 익힐 수 있었습니다. Spring Web MVC를 활용하여 웹 페이지와 컨트롤러를 구성하고, Spring JDBC를 활용하여 데이터베이스 연동을 수행하는 과정에서 데이터를 효율적으로 관리하고 조작하는 방법을 배울 수 있었습니다.



감사합니다.

