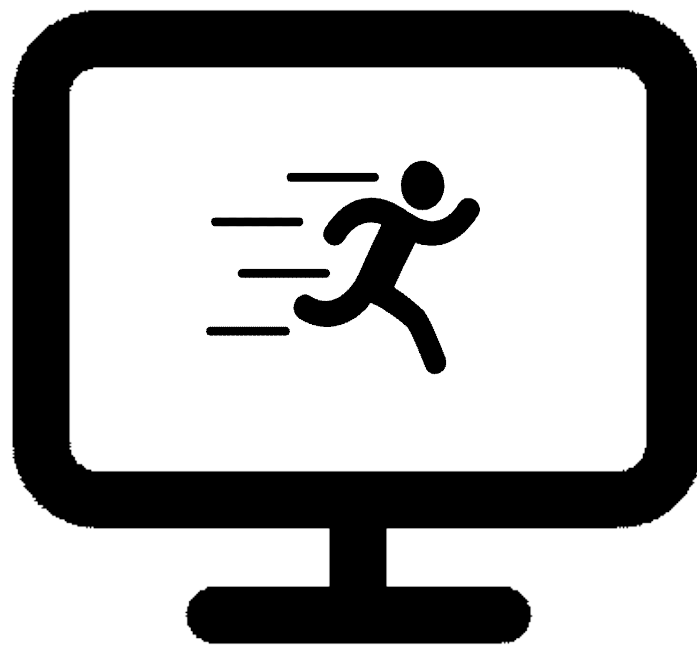


Customer Tracking



22311947

김서현

canna7610@gmail.com

[Revision history]

Revision date	Version #	Description	Author
03/20/2025	1.00	First Writing	김서현
05/03/2025	1.01	Use case 수정 : View customer 수정	김서현
05/05/2025	1.11	Use case 수정 : View hour, View age, View gender 삭제 View statistics 수정 Choose criterion 추가	김서현

= Contents =

1. Business purpose	4
2. System context diagram	5
3. Use case list	6
4. Concept of operation	8
5. Problem statement	11
6. Glossary	13
7. References	14

1. Business purpose

자영업을 하시는 부모님은 항상 손님이 오는 것에 많은 신경을 쓰신다. 식사 시간을 가지거나 잠시 가게를 비울 때도 최대한 손님이 덜 오는 시간을 골라 일을 처리하기 위해 노력하신다. 하지만 손님이 오는 시간이 정확히 언제라고 정해져 있지도 않을뿐더러 점주의 입장에서 그것을 예측하기란 쉽지 않다. 매년 그런 고민을 거듭하던 중 문득 ‘손님을 관리해 주는 프로그램이 있으면 어떨까?’라는 생각이 들었다. 손님이 자주 오는 시간대, 매장에 머무는 기간, 구입한 물건의 유형 등을 기록해뒀다 필요한 순간에 사용할 수 있다면 보다 더 편리한 가게 운영이 가능하지 않을까?

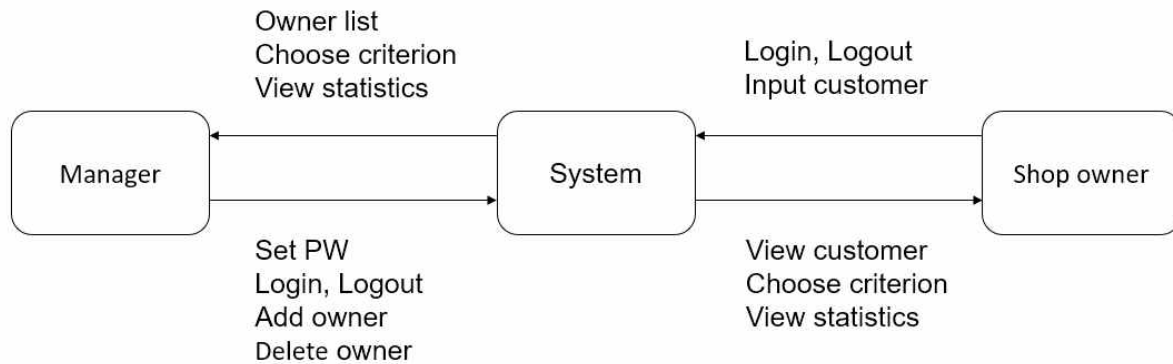
때문에 나는 이번 프로젝트에서 자영업자를 위한 고객 추적 프로그램을 만들기로 결심했다. 이번 프로젝트에서 만들게 될 Customer Tracking 시스템은 가게에 들어오는 손님의 정보를 기록해 뒀다가 필요한 순간에 확인할 수 있도록 서비스를 제공한다. 점주는 자신의 가게를 방문한 손님의 나이, 성별, 매장에 들어온 시간, 매장에 머문 시간, 구입한 물건의 가격대 등을 기록하고 시스템에서는 그 데이터의 통계를 내 적절한 결과를 제공한다.

하루에 손님이 몇 명이나 오는지, 손님이 가장 많이 찾아오는 시간대는 언제인지, 내 가게에 자주 오는 손님의 유형은 어떤지, 각 손님이 매장에 머무는 평균 시간은 어떤지, 그리고 이러한 것들이 물건을 구매하는 소비력에 어떤 영향을 끼치는지, 점주는 자신이 입력한 데이터를 바탕으로 필요한 정보를 선택적으로 확인할 수 있다.

또한 프로젝트의 아이디어를 구상하던 중 이러한 Customer Tracking 시스템이 개인 사업체는 물론, 가맹점을 여럿 거느린 프랜차이즈 매장에서도 유용하게 사용될 수 있을 것이라는 확신을 느꼈다. 따라서 단순히 한 명의 점주가 데이터를 작성하는 것에 그치지 않고, 데이터를 입력하는 ‘가맹점주’와 가맹점주들을 관리하는 ‘매니저’로 나뉘어 Customer Tracking 시스템을 구상하였다.

고객 추적 프로그램인 Customer Tracking 시스템은 옷이나 가구 등을 판매하는 매니저와 점주들에게 필요한 데이터를 제공할 것이며 사용자는 이 데이터를 바탕으로 손님을 관리하고 매출을 올리는 것에 직접적인 도움을 받게 될 것이다.

2. System context diagram



- Set PW
- Login
- Logout
- Add owner
- Delete owner
- Owner list
- Input customer
- View customer
- Choose criterion
- View statistics

비밀번호 등록
로그인
로그아웃
가맹점주 추가하기
가맹점주 삭제하기
가맹점주 목록 보기
고객 정보 추가하기
고객 정보 확인하기
고객 통계 확인하기
기준에 따른 고객 통계 확인하기

3. Use case list

1) Set PW

Actor	Manager
Description	시스템 시작 시 매니저가 자신의 PW를 등록한다.

2) Login

Actor	Manager, Shop owner
Description	매니저와 점주가 각자 자신의 아이디로 로그인한다.

3) Logout

Actor	Manager, Shop owner
Description	매니저와 점주가 각자 자신의 아이디에서 로그아웃한다.

4) Add owner

Actor	Manager
Description	매니저가 새로운 점주를 추가한다.

5) Delete owner

Actor	Manager
Description	매니저가 기존의 점주를 삭제한다.

6) Owner list

Actor	Manager
Description	매니저가 현재 존재하는 점주들의 정보를 확인한다.

7) Input customer

Actor	Shop owner
Description	점주가 가게에 다녀간 고객의 정보(나이, 성별, 매장에 들어온 시간, 매장에 머문 시간, 구입한 물건의 가격대)를 등록한다.

8) View customer

Actor	Shop owner
Description	점주가 등록된 고객의 정보를 확인한다.

9) Choose criterion

Actor	Manager, Shop owner
Description	매니저와 점주가 등록된 고객 정보의 통계를 확인하기 위해 통계의 기준을 선택한다. 통계의 기준은 고객의 입장 시간별/나이별/성별 3가지로 나뉘어진다.

10) View statistics

Actor	Manager, Shop owner
Description	매니저와 점주가 등록된 고객 정보의 기준별 통계를 확인한다. 각 기준에 따라 몇 명의 고객이 가게에 방문했는지, 평균 얼마만큼의 시간을 가게에 머물렀는지, 평균적으로 어떤 가격대의 물건을 샀는지를 확인할 수 있다.

4. Concept of operation

1) Set PW

Purpose	시스템 시작 시 매니저가 자신의 PW를 등록
Approach	사용자가 시스템을 실행할 시 가장 먼저 매니저의 PW를 등록하게 된다. Set PW는 시스템을 처음 실행하는 순간에 단 한 번만 사용된다.
Dynamics	시스템 최초 실행 시 매니저의 PW를 등록할 경우
Goals	시스템의 첫 실행에서 매니저의 PW를 등록해 권한을 가진 매니저만이 시스템에 로그인할 수 있도록 한다.

2) Login

Purpose	시스템을 사용하기 위해 등록된 사용자인지 확인
Approach	사용자가 시스템을 실행할 시 본인이 매니저인지 점주인지를 선택한다. 사용자가 매니저를 선택한 경우, PW를 입력하고 등록된 PW와 입력한 PW가 일치할 때 시스템에 로그인한다. 사용자가 점주를 선택한 경우, ID와 PW를 입력한다. 이후 등록된 점주 ID가 존재하고 입력된 PW가 일치한다면 로그인한다.
Dynamics	시스템 실행 시 로그인할 경우
Goals	로그인 기능을 구현한다.

3) Logout

Purpose	시스템을 종료
Approach	사용자가 시스템을 종료하길 원할 시 매니저 혹은 점주에서 로그아웃한다. 사용자는 기능을 선택하는 메인 페이지에서 로그아웃이 가능하다.
Dynamics	시스템 종료 시 로그아웃할 경우
Goals	로그아웃 기능을 구현한다.

4) Add owner

Purpose	새 점주를 등록
Approach	매니저가 새 점주를 등록할 시 등록할 점주의 ID와 PW를 입력한다. 만약 동일 ID를 가진 기존 점주가 존재하지 않는다면 입력한 ID와 PW를 가진 새 점주를 등록한다.
Dynamics	매니저가 새 점주 등록을 원할 경우
Goals	매니저가 원할 때 점주를 등록해 여러 명의 점주를 관리할 수 있다.

5) Delete owner

Purpose	기존 점주를 삭제
Approach	매니저가 기존 점주의 삭제를 원할 시 해당 점주의 ID와 PW를 확인하고 일치한다면 해당 점주를 삭제한다.
Dynamics	매니저가 기존 점주의 삭제를 원할 경우
Goals	더는 관리가 필요하지 않게 된 가게의 점주를 삭제해 매니저의 점주 관리를 돕는다.

6) Owner list

Purpose	매니저가 관리 중인 점주 리스트를 출력
Approach	현재 매니저가 관리하고 있는 모든 점주들의 ID를 등록 순서대로 정렬해 보여준다.
Dynamics	매니저가 등록된 점주들의 정보를 확인하길 원할 경우
Goals	매니저가 등록된 점주들을 확인함으로써 새 점주를 추가하거나 기존 점주를 삭제하는 것에 도움을 얻는다.

7) Input customer

Purpose	새로운 손님 정보 등록
Approach	점주가 자신의 가게에 방문한 손님의 정보를 입력한다. 점주는 손님의 나이, 성별, 매장에 들어온 시간, 매장에 머문 시간, 구입한 물건의 가격대를 순서대로 입력한다. 이때 입력 순서에 따라 손님의 정보에는 자동으로 CID(Customer ID)가 붙여진다. 해당 CID는 입력 순서에 따라 정수로 부여된다.
Dynamics	점주가 손님 정보를 등록하길 원할 경우
Goals	고객의 정보를 받아 이후 사용될 고객 정보 통계를 낼 수 있도록 돕는다.

8) View customer

Purpose	등록된 고객의 정보 확인
Approach	점주가 이미 등록된 손님의 정보를 확인한다. 손님의 정보는 CID 순으로 정렬되어 나오며 해당 고객의 CID값과 입력했던 모든 정보(나이, 성별, 매장에 들어온 시간, 매장에 머문 시간, 구입한 물건의 가격대)를 포함한다. 점주는 자신의 매장에 등록된 손님 정보만을 확인할 수 있다.
Dynamics	점주가 등록된 손님 정보를 확인하길 원할 경우
Goals	손님 정보를 제공해 고객 정보의 관리를 돕고 필요하다면 구체적인 고객 정보를 확인할 수 있도록 한다.

9) Choose criterion

Purpose	등록된 고객 정보의 통계를 확인하기 위한 기준 선택
Approach	매니저와 점주는 등록된 고객 정보의 통계를 확인하기 위해 기준을 선택한다. 매니저와 점주는 입장 시간대/나이/성별 3가지의 기준 중 어떤 기준의 통계 정보를 볼 것인지를 선택할 수 있다.
Dynamics	매니저와 점주가 등록된 고객의 정보를 통계로써 확인하기 위해 기준을 선택할 경우
Goals	통계의 기준을 선택할 수 있게 함으로써 점주와 매니저의 고객 관리 및 분석을 돕는다.

10) View statistics

Purpose	등록된 고객 정보의 기준별 통계를 확인
Approach	매니저와 점주가 등록된 고객 정보의 기준별 통계를 확인한다. 해당 기준은 위의 Choose criterion에서 선택된다. 매니저와 점주는 각 기준에 따라 몇 명의 고객이 가게를 방문했었는지, 평균 얼마만큼의 시간을 가게에 머물렀었는지, 평균적으로 어떤 가격대의 물건을 샀는지를 확인할 수 있다. 특히 매니저의 경우 확인하길 바라는 점주의 ID를 선택해 각 점주별로 이 통계를 확인할 수 있다.
Dynamics	매니저와 점주가 등록된 고객의 정보를 기준별 통계로써 확인하길 원할 경우
Goals	기준 별로 가게에 방문한 손님의 정보를 제공함으로써 손님의 정보를 한 눈에 확인할 수 있다. 예를 들어 시간별 분석의 경우, 어떤 시간대에 손님이 물리고 어떤 시간대에 손님이 방문하지 않는지, 어떤 시간대에 온 손님이 가장 오랜 시간을 가게에 머무는지, 어떤 시간대에 온 손님이 가장 높은 가격대의 물건을 구입하는지를 알 수 있다.

5. Problem statement

이 프로젝트의 Customer Tracking 시스템은 매니저와 점주가 자신의 가게를 방문한 손님의 정보를 확인하고 이를 통해 가게의 관리 및 운영에 도움을 받을 수 있도록 만들어졌다. 특히 1명의 매니저가 여러 점주를 관리할 수 있는 Customer Tracking 시스템만의 특징을 잘 이용한다면, 각 프랜차이즈들은 더 넓은 데이터를 바탕으로 고객의 동향을 분석할 수 있을 것이다.

그러나 이러한 이점에도 불구하고 본격적으로 Customer Tracking 시스템이 잘 작동되기 위해서는 아래에서 이야기될 여러 조건들을 잘 갖추어야 한다.

Problem 1)

이 시스템은 오로지 점주들이 입력한 고객 정보를 바탕으로 운영된다. 때문에 시스템의 적용 직후, 초반 데이터를 쌓는 과정이 필수적으로 요구된다. 일정량 이상의 데이터가 쌓이기 전의 통계는 고객을 분석하는 데 있어 크게 도움이 되지 않을 것이다. 따라서 이 시스템을 사용하려는 매니저들은 시스템의 사용 초기, 눈에 보이는 결과가 없더라도 점주들이 꾸준히 고객 정보를 등록할 수 있도록 많은 주의를 기울여야 한다.

Problem 2)

해당 시스템은 JDK21을 기반으로 구현된다. 이 시스템에서는 점주들에 의해 각 고객의 정보를 다량으로 입력받게 되므로 입력받은 고객의 정보를 어떻게 정리하고 관리하느냐가 무엇보다 중요하다. 정보를 효율적으로 운용하기 위한 적절한 방법을 찾는 것에 많은 시간 투자가 필요할 것이다.

Problem 3)

사용자에게 정보를 제공하는 방법은 많다. 특히 여러 기준의 통계 자료를 제공하는 Customer Tracking 시스템의 특성상 제공되는 통계를 어떤 방식으로 보여줄지를 고려해 봐야 한다. 같은 정보라도 어떤 방식으로 보여지느냐에 따라 사용자의 만족도가 크게 달라질 것이다. 필요에 따라서는 관련 라이브러리를 이용해 보기 쉬운 데이터를 제공해야 한다.

여기까지가 프로젝트 진행 중 고려해야 할 문제 상황들이다. 아래로는 시스템이 동작하는 방식과 품질에 대한 NFRs(Non-Functional Requirements)를 서술해 두었다.

NFRs)

- ① 시스템은 한 명의 매니저가 여러 점주를 관리할 수 있도록 설계되어야 한다.
- ② 매니저와 점주의 권한을 완전히 분리하여, 매니저는 모든 매장의 데이터를 보고 점

주는 자신의 매장 데이터만 볼 수 있도록 해야 한다.

③ 데이터의 무결성을 지키기 위해 입력된 데이터가 임의로 변경되거나 삭제되지 않도록 해야 한다.

④ 점주가 고객 정보를 빠르고 간편하게 입력할 수 있도록 직관적이고 편리한 UI를 제공해야 한다. 이 시스템에서는 Swing을 이용해 이를 구현한다.

⑤ 통계 데이터는 그래프, 차트 등을 활용해 한눈에 보기 쉽게 표현해야 한다. 이 시스템에서는 Swing과 JFreeChart 등의 라이브러리를 활용해 이를 구현한다.

6. Glossary

Term	Description
Manager	여러 명의 Shop owner를 관리하는 관리자. 프랜차이즈 매장의 경우, 본사에 해당한다.
Shop owner	자신의 가게 안에서 물건을 판매하고 방문한 손님에 대한 정보를 기록한다. 모든 Shop owner는 Manager의 아래에 존재한다.
Customer	Shop owner의 가게에 와 물건을 구매하는 구매자.
ID	Shop owner를 식별하기 위한 아이디. 각 Shop owner를 식별하기 위한 아이디이므로 중복된 아이디는 존재할 수 없다.
PW	Manager와 Shop owner를 확인하기 위한 패스워드. 중복된 패스워드는 존재할 수 있다.
CID	등록된 고객 정보를 관리하기 위한 고객 아이디. Shop owner가 고객의 정보를 입력할 때 자동으로 부여된다. 첫 번째로 입력된 고객의 CID는 0번. 이후 입력 순서에 따라 양의 정수값을 순서대로 갖는다.
JDK	JDK(Java Development Kit)는 Java 프로그램을 개발하기 위한 소프트웨어 개발 환경이다. Java 프로그램을 작성, 컴파일, 실행하고 디버깅하는데 필요한 도구, 실행 환경, 라이브러리 및 API를 포함하고 있다. 이 프로젝트에서는 많은 버전 중 JDK 21을 사용한다.
Swing	Java에서 GUI(그래픽 사용자 인터페이스)를 개발하기 위해 사용되는 표준 라이브러리이다. AWT(Abstract Window Toolkit)를 기반으로 하되 더 많은 기능과 유연성을 제공한다.
JFreeChart	Java 애플리케이션에서 전문적인 품질의 차트를 쉽게 생성할 수 있게 해주는 오픈 소스 차트 라이브러리이다. 다양한 종류의 차트(막대, 파이, 선, 산점도 등)를 지원하며 Swing과의 통합이 쉽다는 장점이 있다.

7. References

1) Icons and Images

<https://www.flaticon.com/>