**요구사항 명세서**

2019-1학기 소프트웨어 공학

3조 (산격동 코드 맛집)

전현희, 박세은, 이다연, 구윤경, 김서진

**1. 프로젝트명 : 산격동 코드 맛집**

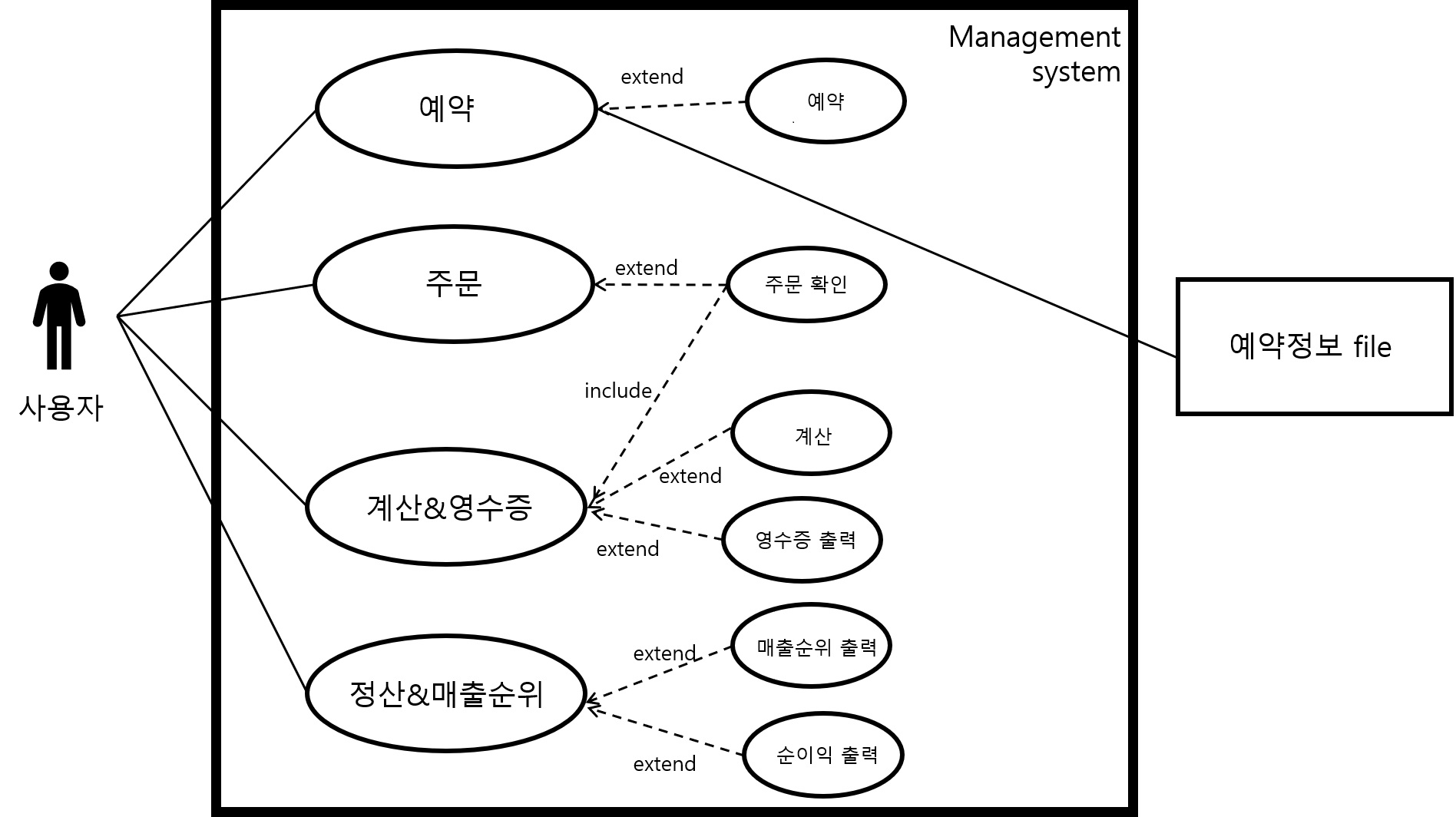
**2. 프로젝트 개요**

우리 학교 근처에는 많은 음식점이 있다. 체계적인 영업시스템을 갖춘 음식점이 있는 반면, 그렇지 않은 음식점들도 많아서 점심, 저녁시간 같은 피크타임에는 많은 사장님들이 혼란을 겪는다. 그래서 사장님들의 영업에 대한 혼동을 줄이기 위해 메뉴선택 및 계산, 영수증, 매출/ 순이익 계산, 예약/빈 자리 보기 서비스 기능을 넣은 포스기를 소프트웨어 공학을 접목하여 만들어본다.

**3. 개발 및 시스템 환경**

java

**4. 유스케이스 다이어그램**



**5. 유스케이스 목록 (유스케이스별 numbering)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| # | Name | Primary Actor | Scope | Complexity | Priority |
| 1 | 예약 | 사장님, 고객 | out | Low | 1 |
| 2 | 주문확인 | 사장님 | in | Med | 2 |
| 3 | 영수증 출력 | 사장님, 고객 | in | Med | 2 |
| 4 | 메뉴 계산 | 사장님 | in | High | 3 |
| 5 | 매출순위 출력 | 사장님 | in | High | 4 |
| 6 | 순이익 출력 | 사장님 | in | High | 4 |

**6. 유스케이스 시나리오 (번호 당 한 페이지씩 표로 작성)**

**6-1 예약**

|  |  |  |
| --- | --- | --- |
| **번호 / 이름** | 1. 예약 | |
| **개요** | 고객이 음식점에 오기 전에 자리를 예약하기 위해 예약시스템을 이용한다. | |
| **액터** | 주액터 | 사장님 |
| 보조액터 | 시스템 |
| **시작상태** | -인원수가 입력된 고객이 보낸 정보를 받아야 한다.  -시스템과 예약을 하려는 고객의 네트워크 연결이 정상적이어야 한다. | |
| **일반상황** | 1. 사장님이 예약/ 자리확인 버튼을 누른다. 2. 시스템이 손님의 파일을 연다. 3. 시스템이 손님의 파일에 쓰인 정보인 예약자 이름과 숫자를 읽는다. 4. 사장님은 좌석 창에서 이미 표시되지 않은 자리 중 하나를 선택한다. 5. 만약 지정할 자리가 남아있다면, 6. 시스템은 사장님이 선택한 좌석의 번호를 저장한다. 7. 시스템은 좌석 창에 해당되는 번호를 가진 블록을 다른 색으로 바꾼다. 8. 시스템은 예약명단 창의 <성함> 칸 아래에는 파일에서 읽은 예약자 이름을 입력한다. 9. 시스템은 예약 명단 창의 <인원수> 칸 아래에는 파일에서 읽은 숫자를 입력한다. 10. 시스템은 예약명단 창의 <배정> 칸 아래에 시스템에 저장된 좌석 번호를 입력한다. 11. <뒤로가기> 블록을 눌러 초기화면으로 돌아간다. 12. 만약 지정할 자리가 남아있지 않다면, 13. <뒤로가기> 블록을 눌러 초기화면으로 돌아간다. | |
| **예외상황** | 2a. 손님의 파일을 읽어 들일 수 없는 경우  2a.1. 예약자 명단이 없으므로 닫기 버튼을 눌러 초기화면으로 돌아간다.  3a. 읽어 들인 파일의 정보가 부족한 경우  3a.1. 이름/ 인원수가 부족합니다.’라는 출력 창을 출력한다. | |
| **완료상태** | * 배정된 좌석의 변경된 GUI 출력화면 | |
| **입력값** | - 예약정보 파일  - 선택된 좌석의 정보 | |
| **출력값** | * 고객의 성함 * 방문할 사람 수 * 배정된 좌석번호 | |

**6-2 주문확인**

|  |  |  |
| --- | --- | --- |
| **번호/**  **이름** | 2. 주문확인 | |
| **액터** | 주 액터 | 사장님 |
| 보조 액터 | 고객 |
| **개요** | 사장이 가게에 오는 손님들의 주문을 입력/출력/수정할 수 있다. 그리고 각 주문들의 합계를 계산버튼을 누르면 알 수 있다. | |
| **입력값** | 클릭 된 메뉴정보 | |
| **출력값** | 주문 목록 | |
| **시작상태** | - 프로그램이 정상적으로 작동해야한다.  - 대기중인 정보가 없어야 한다. | |
| **일반상황** | 1. 사장은 고객이 주문한 메뉴에 해당하는 버튼을 누른다.  2. 버튼에 저장되어 있는 정보(메뉴명, 메뉴별 인덱스 값, 가격, 수량 등)들을 읽어서 저장소에 저장한다.  3. 저장소의 정보들을 읽어 온다.  4. 읽어온 정보를 주문 목록에 출력한다.  5. 만약 수정할 필요가 있으면  6. 모든 작동을 멈춘다.  7. 사장은 주문 목록에서 고치고 싶은 주문을 클릭한다.  8. 클릭된 정보를 저장소에서 삭제한다.  9. 주문 목록의 출력을 초기화한다.  10. 저장소의 정보들을 주문목록에 출력한다.  11. 만약 메뉴를 여러개 입력하고 싶으면  12. 주문 버튼을 여러번 누른다.  13. 정보가 저장소에 저장되고 같은 인덱스 값을 가진 정보는 수량을 늘려준다.  14. 주문이 모두 끝났으면 사장은 결제버튼을 누른다. | |
| **예외상황** | 4a. 누른 버튼의 정보와 출력된 정보의 값이 맞지 않는 경우  4a.1. 작동을 멈추고 저장소를 초기화 시킨 뒤 재출력한다. | |
| **완료상태** | - 유스케이스 수행 후 주문목록 = 클릭한 주문버튼들 | |

**6-3 계산**

|  |  |
| --- | --- |
| **번호 / 이름** | 3. 계산 |
| **개요** | 고객이 한 주문을 계산한다. |
| **액터** | 가게 사장 |
| **시작상태** | * 주문 목록에 더 이상 수정이나 삭제가 없어야 한다. * 사장님이 결제 버튼을 클릭해야 한다. |
| **일반상황** | 1. 시스템은 주문 목록의 메뉴명과 수량을 읽어들여서 최종 주문으로 확정한다.  2. 시스템은 최종 주문된 메뉴의 메뉴별 가격 정보를 불러온다.  3. 시스템은 모든 주문된 메뉴의 메뉴별 가격과 메뉴별 수량을 곱해서 총 금액에 저장한다.  4. 시스템은 총 금액을 총 금액 창에 표시한다.  5. 시스템은 최종 주문 정보(메뉴명, 수량)를 모든 주문을 저장하는 총 주문 목록에 저장한다. |
| **예외상황** | 1.a 주문 목록에 메뉴가 0개인 경우  1.a.1 시스템은 ‘결제할 주문이 없습니다’ 라고 표시한다.  2.a 시스템이 메뉴별 가격 정보를 불러올 수 없는 경우  2.a.1 시스템은 ‘메뉴 정보를 불러올 수 없음’이라고 표시한다. |
| **완료상태** | * 총 금액 창에 총 금액이 표시된다. * 총 주문 목록에 새로운 주문 정보가 들어온다. |
| **입력값** | 없음 |
| **출력값** | 총 주문 금액(이하 ‘총금액’) |

**6-4 영수증 출력**

|  |  |
| --- | --- |
| **유스케이스**  **번호 및 이름** | 4. 영수증 출력 |
| **개요** | 고객이 주문한 내용을 영수증에 출력한다. |
| **액터** | 가게 사장 |
| **입력값** | 없음 |
| **출력값** | 가게명, 주문한 메뉴, 메뉴별 금액, 총 주문 금액 |
| **시작상태** | -고객이 결제를 끝내야 한다.  -영수증 출력 버튼이 클릭 되어야 한다. |
| **일반상황** | 1. 시스템은 결제 완료된 주문 정보(이하 ‘주문 정보’)를 읽어온다. 2. 시스템은 결제 완료된 총 주문 금액(이하 ‘총 주문 금액’)을 읽어온다. 3. 시스템은 영수증 상에 가게명, 항목명(메뉴/수량/금액)을 표시한다. 4. 시스템은 영수증 창에 주문 정보를 메뉴별로 한 줄씩 표시한다. 5. 시스템은 영수증 창 하단에 총 주문 금액을 표시한다. |
| **예외상황** | 1a. 결제 완료된 주문 정보가 없을 경우  1a.1 시스템은 영수증 창에 ‘출력 내용 없음’이라고 표시한다.  2a. 결제 완료된 총 주문 금액 정보가 없을 경우  2a.1 시스템은 영수증 창에 ‘출력 내용 없음’이라고 표시한다. |
| **완료상태** | -영수증 창에 가게명, 항목명, 주문정보, 총 주문 금액이 표시된다. |

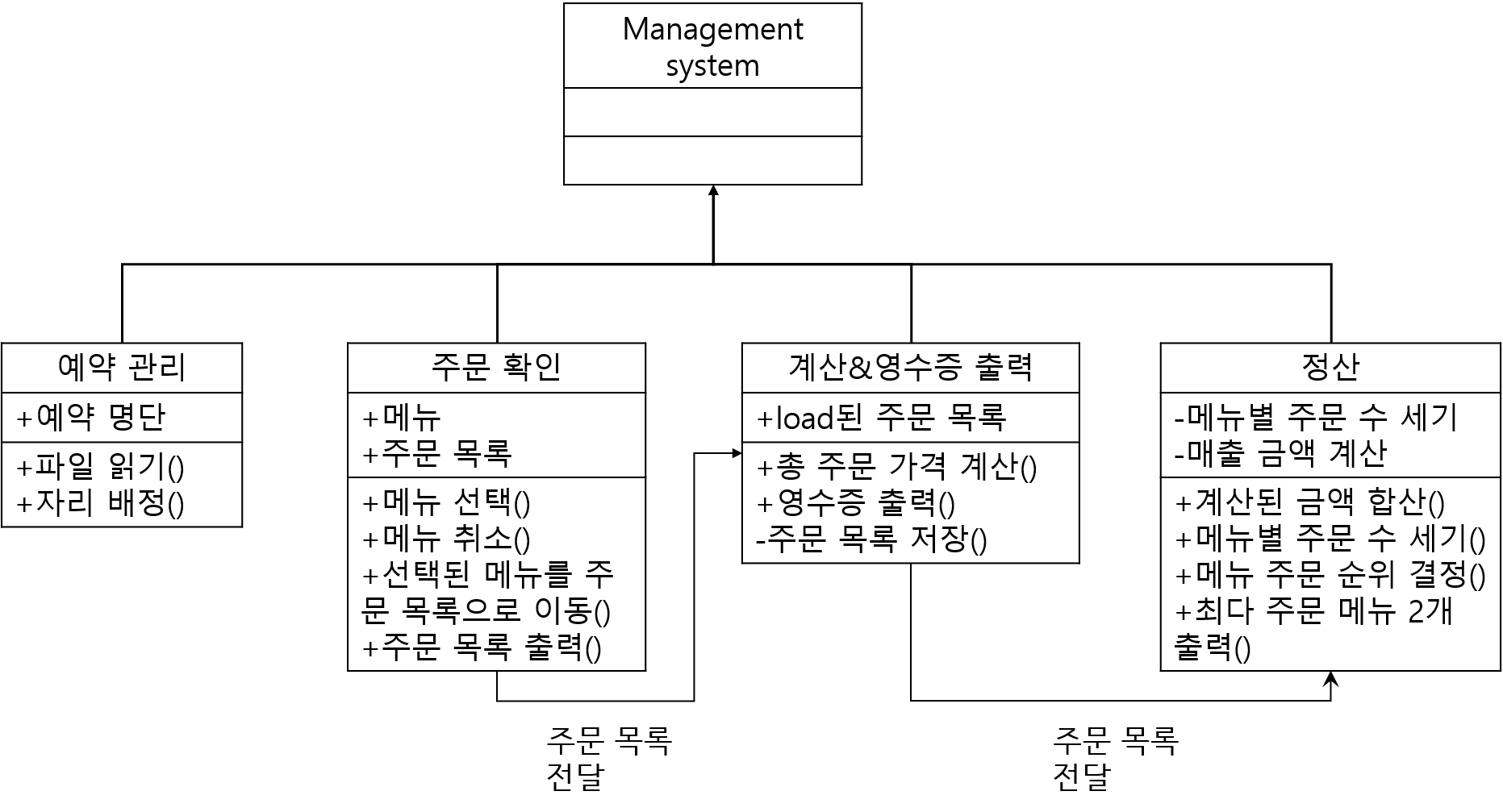
**6-5 매출 내역 및 순이익 계산**

|  |  |
| --- | --- |
| **번호 / 이름** | 5. 매출 내역 출력 및 순이익 계산 |
| **개요** | 가게 사장이 지금까지의 매출을 확인 할 수 있도록 한다. |
| **액터** | 가게 사장 |
| **시작상태** | * ‘계산’에는 총 주문 목록이 저장되어 있다. * ‘준비금액’에 인건비 및 재료비용, 기타 비용이 저장되어 있다. |
| **일반상황** | 1. 가게 사장(사용자)가 정산 버튼을 누른다.  2. 정산 버튼을 누르면 ‘매출’ 버튼과 ‘판매순위’ 버튼이 뜬다.  3. 가게 사장(사용자)이 ‘매출’ 버튼을 누른다.  4. 시스템은 ‘계산’으로부터 총 주문 목록을 받는다.  5. 총 주문 목록에 저장된 판매된 음식과 각 수량을 읽는다.  6, ‘전체 매출액’을 계산하여 시스템에 저장한다.  7. ‘전체 매출액’에서 기존에 저장되어 있는 ‘준비금액’을 빼서 ‘순이익’을  계산하여 시스템에 저장한다.  8. 시스템 창에 주문 목록을 출력한다.  9. 목록의 제일 아래에 ‘전체 매출액’과 ‘순이익’을 출력한다. |
| **예외상황** | 4.a. 받은 총 주문 목록에 데이터가 없을 경우 (매출이 발생하지 않았을 시)  4.a.1. ‘매출이 없습니다!’라는 팝업 창 혹은 문구를 출력한다. |
| **완료상태** | 정산 화면에 주문 목록 및 전체 매출액과 순이익이 표시된다. |
| **입력값** | 총 주문 목록 |
| **출력값** | 총 주문 목록, 전체 매출액, 순이익 |

**6-6 최다 주문 메뉴 출력**

|  |  |
| --- | --- |
| **번호 / 이름** | 6. 최다 주문 메뉴 출력 |
| **개요** | 판매가 잘 되는 잘 팔리는 메뉴를 확인하기 위해 메뉴의 매출 순위를 알아본다. |
| **액터** | 가게 사장 |
| **시작상태** | * ‘계산’에는 주문 목록이 저장되어 있다. |
| **일반상황** | 1. 가게 사장(사용자)가 정산 버튼을 누른다.  2. 정산 버튼을 누르면 ‘매출’ 버튼과 ‘판매순위’ 버튼이 뜬다.  3. 가게 사장(사용자)이 ‘판매순위’ 버튼을 누른다.  4. 시스템은 ‘계산’으로부터 총 주문 목록을 받는다.  5. 총 주문 목록에 저장된 판매된 음식과 각 수량을 읽는다.  6. 주 메뉴/사이드 메뉴/음료로 분류한다.  7. 항목별로 판매된 수량이 많은 만큼 순위를 매긴다.  8. 항목별로 최다 주문 메뉴 및 수량을 2가지씩 출력한다. |
| **예외상황** | 7.a. 판매된 음식이 없는 항목이 있는 경우  7.a.1. 그 항목은 출력하지 않는다.  8.a. 항목 안의 판매된 메뉴가 2가지 미만일 경우  8.a.1. 한 가지만 출력한다. |
| **완료상태** | 정산 화면에 주 메뉴, 사이드 메뉴, 음료별로 최다 주문 메뉴를 2가지씩 표시한다. |
| **입력값** | 주문 목록 |
| **출력값** | 항목별 최다 주문 메뉴 |

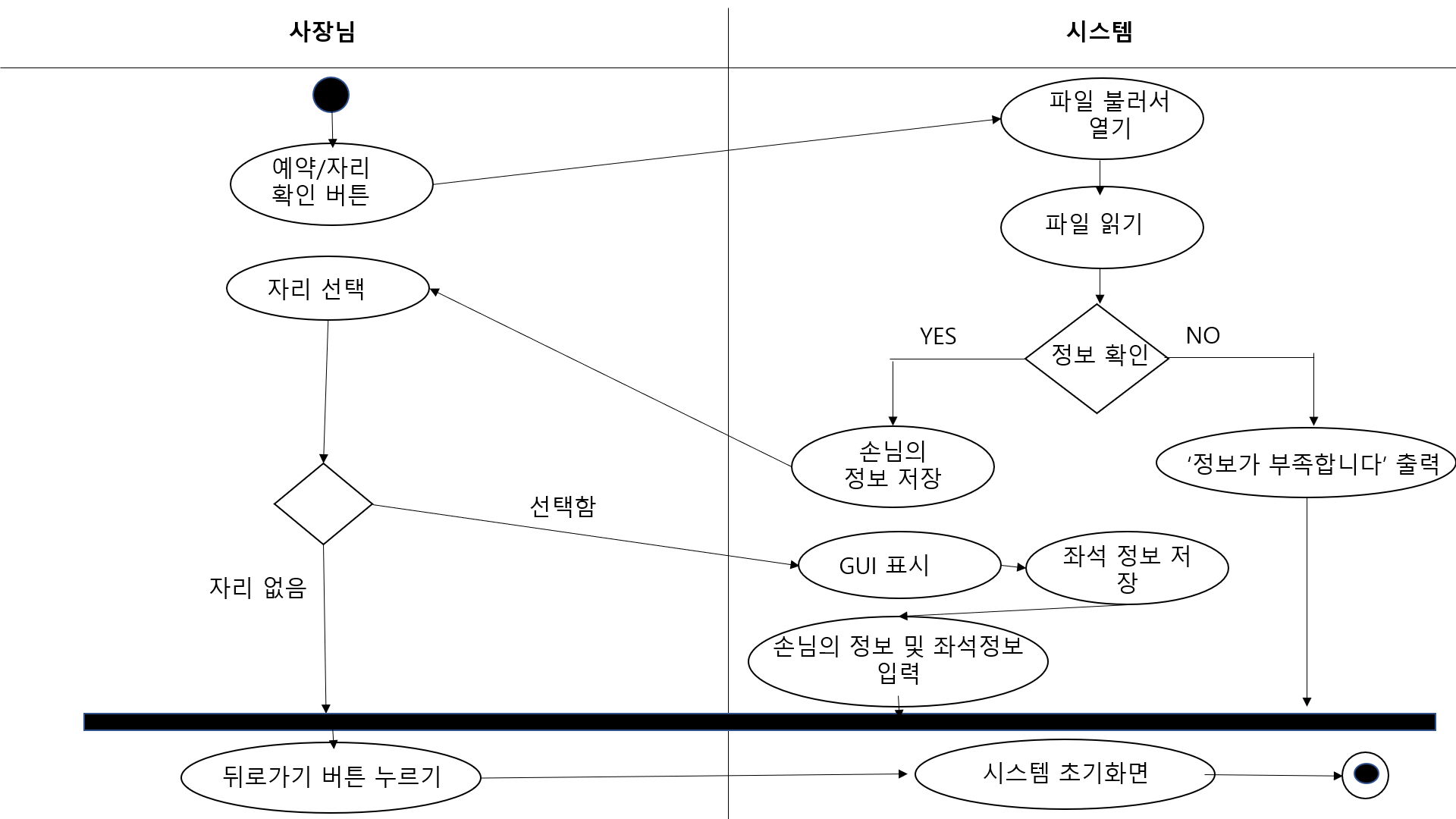
**7. 시스템 아키텍처 다이어그램 (클래스, 컴포넌트 등 내부 구조 및 관계 확인용)**

****

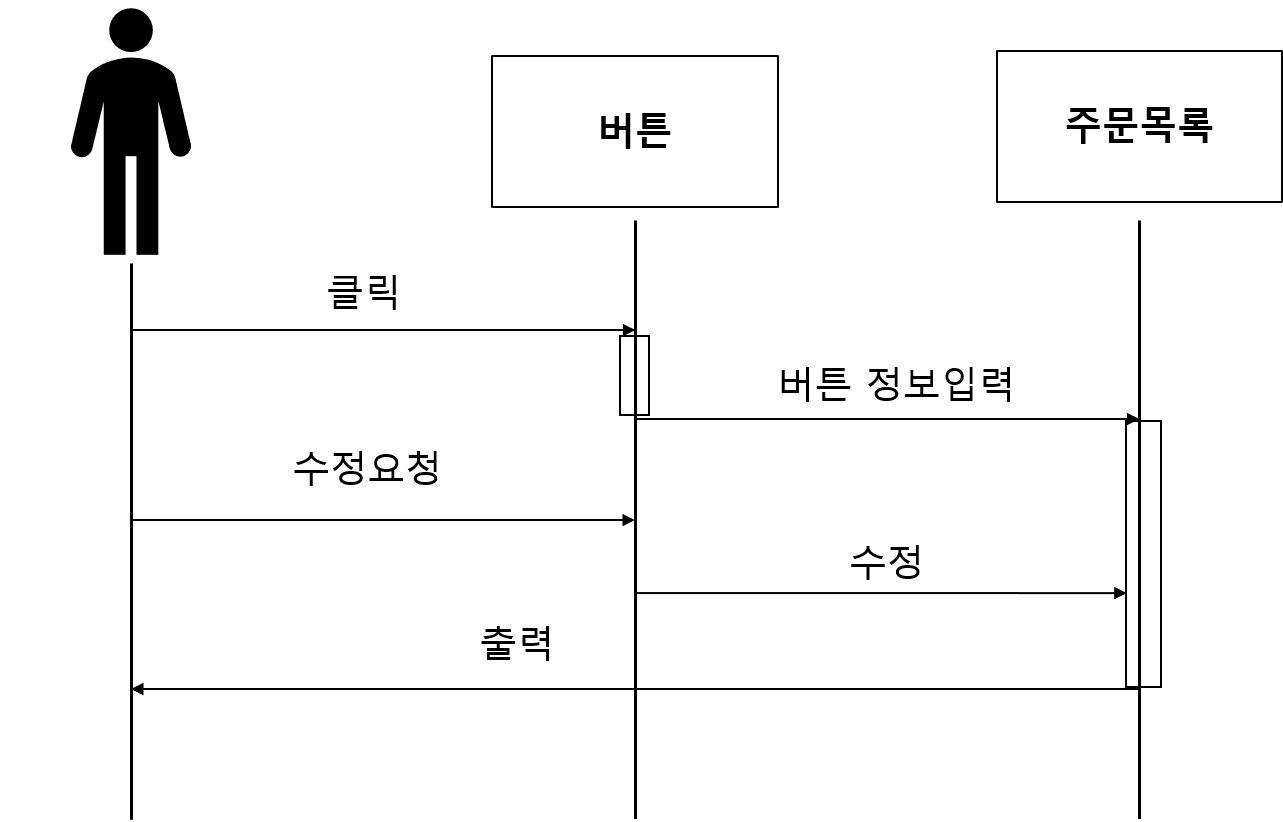
**8. 유스케이스별 시스템 모델 다이어그램**

**(상태, 시퀀스 액티비티 등 기타 필요하다고 생각되는 다이어그램들)**

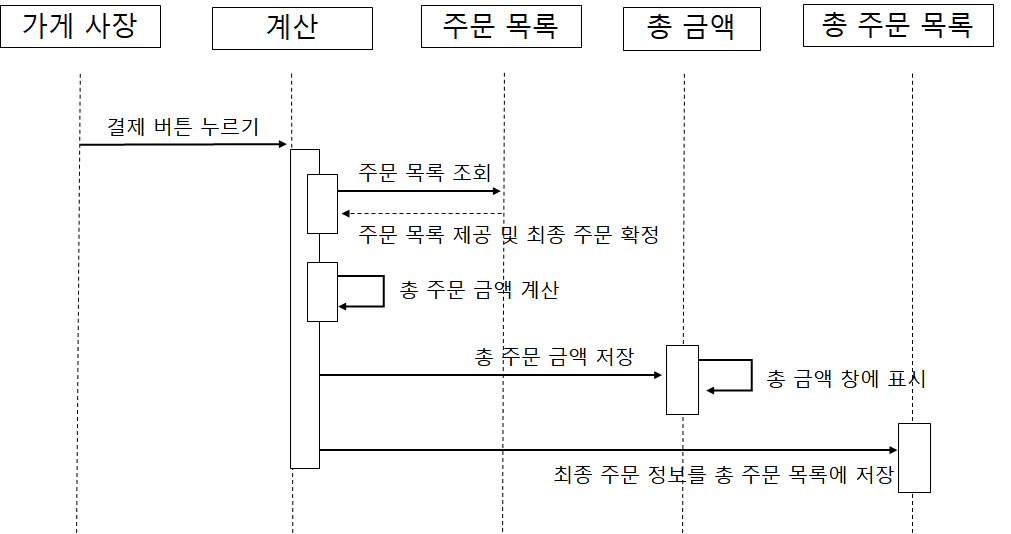
**8-1 예약**



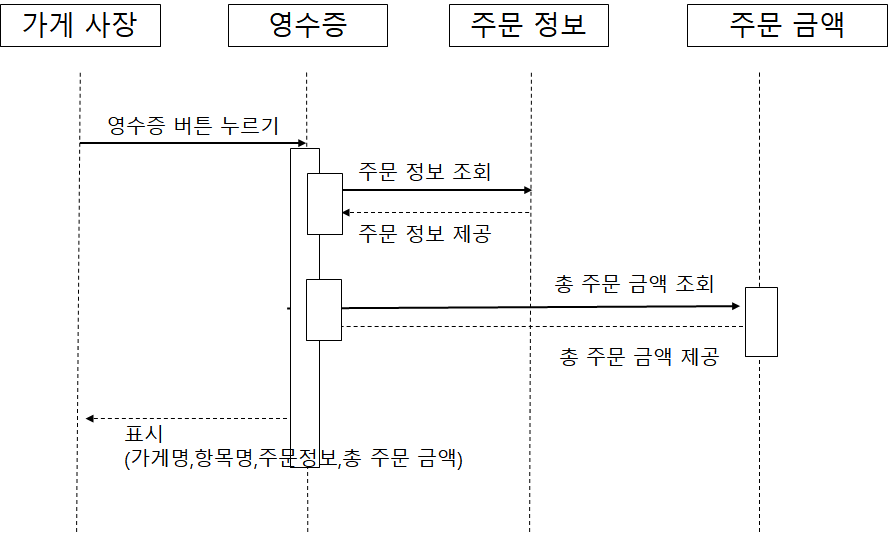
**8-2 주문확인**



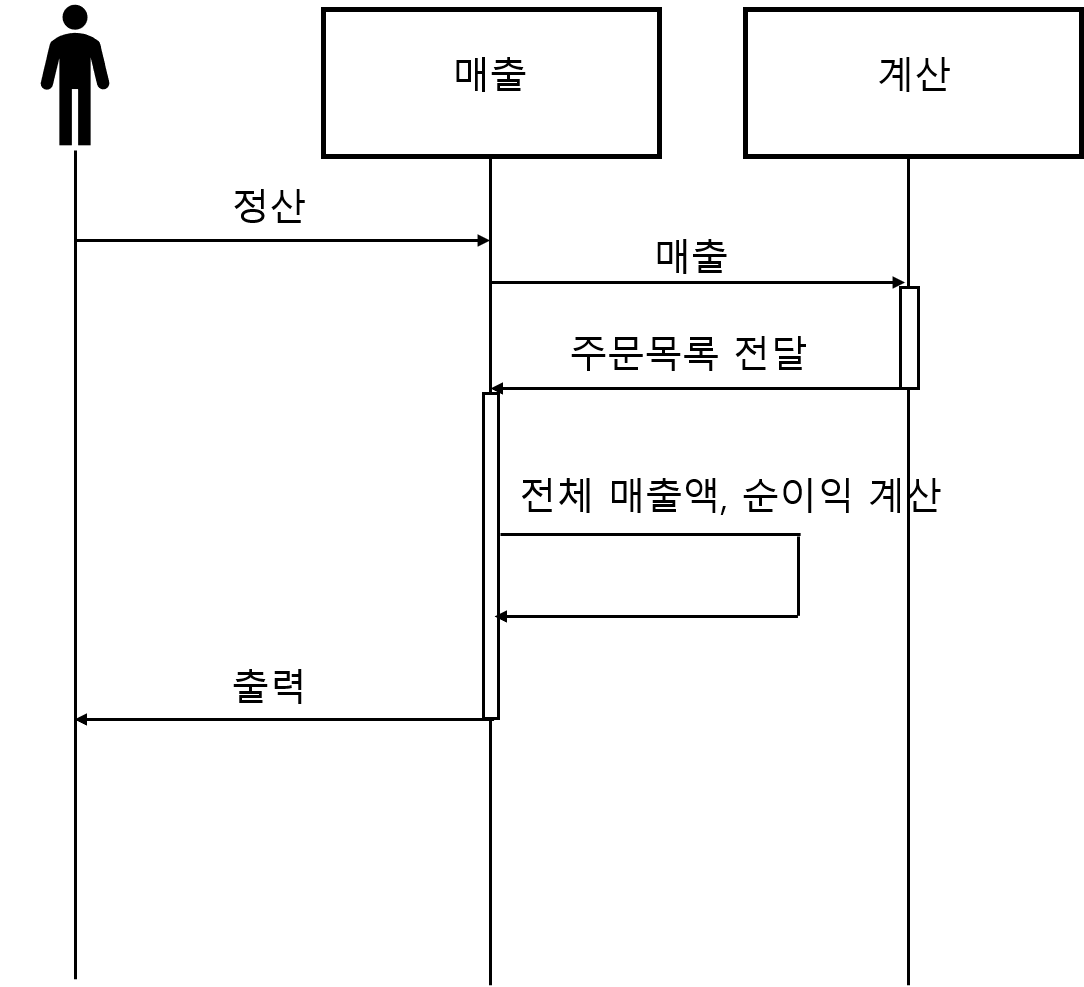
**8-3 계산**



**8-4 영수증 출력**



**8-5 매출 내역 및 순이익 계산**



**8-6 최다 주문 메뉴 출력**

