**Sub group1 user story (requirements-analysis-document)**

**[Analysis 단계 -2 ]**

Sub group1 담당자 : 고은서, 오예원

횟수 : 5

회의 날짜 : 2021.04.08(목)

📌 오늘 할 일 📌

1. User Interface 방식으로 requirements 작성하기

1-1) 유저가 앱에 접속하여 회원인증 하는것에서부터 주문 완료후 결제하는 과정까지 UI화면을 구상하며 그려보기

1-2) 고객과 의논하며 가장 적합한 UI를 그려나간다는 생각으로 analysis작성

1-3) UI는 직관적이고 간단하게 그리기

2. functional 방식으로 requirements 작성하기

3. non-functional 방식으로 requirements 작성하기

☑️ User Interface Requirements

① 회원인증 및 로그인 화면

|  |  |
| --- | --- |
| **Initial** | **Analysis**   * 식당이용자가 학생 또는 교직원일 경우에는 로그인기능을 사용할 수 있으므로 별도의 회원가입을 통해 회원인증을 거칠 수 있어야 한다! * 비회원은 로그인이 아닌, 바로 주문을 하러 갈 수 있어야 한다 * 회원로그인을 할 경우 전 메뉴 5% 할인됨을 명시해준다 * 학생은 학생증을, 교직원은 직원카드를 통해 회원인증을 진행한다. |
| **Re-designed** | |

② 학식당 건물 리스트 선택 화면

|  |  |
| --- | --- |
| **Initial** | **Analysis**   * 식당이용자라면 식당 리스트를 지도 상으로(GPS)으로 보기 보단 식당리스트가 목록화된, 그리고 메뉴가 간단히 나열된 모습이 메인화면이어야 앱을 사용하는데 더욱 편리할 것이다. * 허나, 자신의 위치를 기준으로 학식당이 어디있는지 알고 싶은 경우도 충분히 고려할 만 하므로 옵션기능으로 옮기도록한다 (돋보기 버튼을 누르면 볼 수 있도록) |
| **Re-designed** | |

③ 오늘 제공되는 학식당의 판매 메뉴 조회 화면 (자세히 보기)

|  |  |
| --- | --- |
| **Initial** | **Analysis**   * 학식당에서 메뉴를 선택하고 주문하기 위해선 먼저 현재시간에 가능한 식사메뉴가 있는지 확인하는게 우선이 되어야 할것이다. * 따라서 특정 학식당을 누르면 오늘의 조식/중식/석식에 대한 모든 메뉴를 확인할 수 있고, 아래에 ‘주문하러가기’버튼을 눌러야 주문을 시작할 수 있다. |
| **Re-designed** | |

④ 현재 시각에 결제 가능한 메뉴 선택 (장바구니) + 총 결제 금액 화면

|  |  |
| --- | --- |
| **Initial** | **Analysis**   * 이전 UI에서 주문하러가기 버튼을 누르면 현재 시간에 가능한 식사의 메뉴만 보여지고 담기버튼을 누르면 메뉴가 장바구니에 들어가게 된다 * 장바구니 아이콘을 클릭하면 +,-버튼으로 수량을 조절할 수 있고 총금액, 할인금액을 통한 최종 결제금액도 확인할 수 있다 * 신용카드 버튼을 누르면 카드결제를 할 수 있다. |
| **Re-designed** | |

⑤ 주문리스트 재확인 및 QR코드 발급 화면

|  |  |
| --- | --- |
| **Initial** | **Re-designed** |

< 최종 설계된 화면별 UI 모음 >



☑️ Functional Requirements

* 재학중인 학생과 교직원은 해당 앱을 사용하여 메뉴를 결제할 경우 중앙대 할인을 받을 수 있다.
* 식당 이용자는 중앙대 내 학식당 건물 리스트와 학식당의 판매 메뉴를 조회할 수 있다.
* 식당 이용자는 앱 접속시간에 판매중인 메뉴의 음식재고를 수치상으로 확인할 수 있다.
* 식당 이용자는 장바구니 기능을 이용하여 원하는 메뉴를 추가하고 개수를 조절할 수 있다.
* 식당 이용자는 모든 주문을 완료하면 합산된 금액을 알 수 있고, 주문에 대한 QR코드를 발급 받을 수 있다.
* 식당 이용자는 카드 결제 방식을 활용할 수 있다.

☑️ Non-Functional Requirements

* F ( Functionality(security)

: 회원 정보의 유출을 방지하기 위해 앱 내 개인정보 보호에 대한 대응책을 항시 마련해야 한다.

: 사용자 트래픽이 급증할 경우에 서버가 꺼지지 않도록 대응책을 마련해야한다.

* U ( Usability)

: 각 화면의 UI를 생성할 때마다 사용자의 입장에서 어떤식으로 작동해야 편의할지 계속해서 test해야한다.

: 예) 개수추가/삭제 기능에 관하여 슬라이드 방식을 사용할지, 숫자를 기입하는 형식을 사용할지, + - 버튼을 사용할지 등

* R (Reliability)

: 음식재고량에 대한 표시를 오차없이 정확하게 보여줌으로써 재고가 없을땐 반드시 주문이 성사되지 않도록 한다.

: 사용자의 회원 정보 유출을 방지하기 위해 앱 내 개인정보 보호에 대한 대응책을 항시 마련해야한다.

* S (Supportability)

: 사용자 트래픽이 급증할 경우에 서버가 꺼지지 않도록 대응책을 마련해야한다