|  |  |
| --- | --- |
| 주제 | 은행 앱 |
| 팀 명 |  |
| 팀 원 |  |
| 개요 |  |

|  |  |
| --- | --- |
| << 팀명 >> | |
| 팀원1 이름 | 구현 가능성 조사 및 발표 |
| 팀원2 이름 | 데이터베이스 구축 |
| 팀원3 이름 | 디자인 및 프론트 엔드 전반 담당 |
| … | … |

|  |
| --- |
| 화면 #1 |
| C:\Users\ITPS\AppData\Local\Microsoft\Windows\INetCache\Content.Word\그림1.png |
| 1. 이동경로 확인 : 이동경로 확인(화면 #2) 페이지로 이동 2. 번역 : 구글 번역 서비스와 연동하여 홈페이지 내 텍스트 컨텐츠를 영어(화면#3), 중국어(화면#4), 일본어(화면#4)로 번역한 페이지로 이동 3. 메인 메뉴 : 전자민원, 정보 공개 등 메인 메뉴로 해당 메뉴 위에 마우스 커서를 이동 시킬 경우 하위 메뉴들이 나오며 하위 메뉴를 누르면 각기 알맞은 페이지로 이동 4. 날씨 정보 : 오늘의 날씨를 보여준다 (매일 오전 9시, 18시 갱신) |

스토리보드란 이 서비스가 실제로 개발됬다고 가정하고

이 화면에서는 어떤것들을 누를 수 있다거나 어떠한 동작을 한다는 등의

스토리보드를 보고 클라이언트가 “아 이 앱이 이렇게 동작하는구나 내가 의뢰하고 싶었던 서비스다” 라는 생각이들정도로 상세하게 표현해야합니다

어떤 이미지가 들어간다 버튼은 어떤 HTML 태그를 사용한다는 등의 이런 구현의 측면이 아닌 사용자의 측면에서 기능 위주로 설명을 해주고 디자인에서는 중요한 부분들(캐러셀, 슬라이드를 넣는다면 어디서 어디로 이동하고 자동 슬라이딩이되는지, 자동 슬라이딩이 된다면 몇 초에 한번씩 자동 슬라이딩이되는지 등)을 짚어주세요

ㅡㅡ

어떤 버튼을 눌렀을 때 화면을 이동한다거나 어떤 기능이 실행된다면 어떤 화면으로 이동하는지 어떤 기능이 실행되는지 설명해주세요.

“이건 당연히 알겠지” 라는 부분은 없습니다. 가령 로그인 버튼을 눌렀을 때 로그인이된다 이런 사소한 것들까지 써줘야합니다

ㅡㅡ

저는 스토리보드와 인터페이스를 만들 때는 주로 PPT로 만들고 해당 PPT를 스크린샷으로 찍어서 워드에 이미지로 붙여넣었습니다.

마찬가지로 정해진 형식은 없습니다. 먼저 스토리보드, 인터페이스 관련 자료를 조사해보는것도 좋겠죠

위의 예시는 예시이므로 실제 홈페이지를 캡쳐해서 스토리보드를 작성했지만 여러분은 실제 결과물이 없으므로 PPT나 손 등으로 실제 서비스할 프론트 엔드를 그린 뒤에 써야합니다.

|  |
| --- |
| 기능명 : 날씨 정보 크롤링 |
| URL : 없음 |
| Method : 없음 |
| 요청 파라미터 : 없음 |
| 응답코드 및 데이터 : 없음 |
| 개요 : 공공 데이터 API를 사용해 매일 9시, 18시에 부산 서구의 날씨를 갱신 |

|  |
| --- |
| 기능명 : 로그인 |
| URL : /member/login |
| Method : GET |
| 요청 파라미터 : 1. ID : 아이디  2. PW : 비밀번호 |
| 응답코드 및 데이터 : 1. 200 : 로그인 성공 2. 400 :  a. 아이디 또는 비밀번호가 전달되지 않음  b. 아이디 또는 비밀번호가 형식에 일치하지 않음 3. 404 : 아이디 또는 비밀번호가 잘못되 일치하는 회원 정보를 찾지 못했을 경우 |
| 개요 : 아이디는 세 자 이상 16자 이하 / 비밀번호는 ~~~ |

|  |
| --- |
| 기능명 : 회원가입 |
| URL : /member/join |
| Method : POST |
| 요청 파라미터 : 1. ID : 아이디  2. PW : 비밀번호 |
| 응답코드 및 데이터 : 1. 201: 회원가입 성공 2. 400 :  a. 아이디 또는 비밀번호가 전달되지 않음  b. 아이디 또는 비밀번호가 형식에 일치하지 않음 3. 409 : 이미 존재하는 아이디로 회원 가입을 신청했을 경우 |
| 개요 : 아이디는 세 자 이상 16자 이하 / 비밀번호는 ~~~ |

인터페이스는 어떤 기능을 만들지 그 목록을 나열하는 것입니다

단순히 나열만이 아닌 이 기능의 이름은 어떤것이고 이 기능을 호출(사용)하기 위해서는 어떤 URL로 접근해야하는지, 접근 방식은 어떤것인지, 접근할 때 요청 파라미터는 어떤 것이 필요한지, 응답 코드와 응답 데이터는 어떤 것이 있는지 써야합니다

인터페이스는 각자가 맡은 부분을 따로 작성할 수도 있고 한 명이 작성할 수도 있습니다.

개발할 때는 이 인터페이스를 보고 개발을 하게 됩니다.

개발자들끼리 이 인터페이스를 참고해서 각자가 맡은 부분을 개발하고 다른 사람이 만든 부분과 내가 만들 또는 만든 부분을 연결해야한다면 인터페이스 문서를 보고 요청을 하고 응답에 대한 처리를 할 수 있는 문서가 됩니다.

기능을 연결해야할 때 매번 가서 ~~씨 이거 저거 어떻게 호출해야되요? 응답은 어떻게되요?

이때는요? 저때는요? 라고 매번 물어보는 것 보다는 문서를 보고 개발하는게 물어보는 사람도 대답하는 사람도 서로 덜 번거롭겠죠

그래서 인터페이스에 작성한 그대로 개발해야합니다.

개발은 구체적인 동작이 중요한게 아닙니다. 인터페이스에 작성한대로만 동작하면 됩니다.

우리가 네이버에 로그인하기 위해서 아이디, 비밀번호를 입력하고 로그인 버튼을 누르기만할 뿐이지 어떻게 동작하는지 모르죠.

이것처럼 기능이 필요한 사용자(개발할 때는 개발자가 사용자가되고 실제 서비스할 때는 사용자가 말그대로 사용자) 입장에서는 값을 넣고 호출 했을 때 내가 원하는 결과만 나오면 됩니다.

만약 개발 중 인터페이스대로 개발이 안되겠다면 인터페이스를 수정하고 팀원들에게 꼭 알려야합니다.