대본

(제목)

안녕하세요. 지금부터 ㅇㅇㅇㅇ에 대해 발표하게 될 지하보다 지상팀의 발표자 ㅇㅇㅇ입니다.

(개발 목표)

앞선 발표에서 많이 말씀드렸기 때문에 개발 목표에 대해서는 간단하게 발표하겠습니다.

DC 스마트그리드는 효율성이 좋은 DC 전력망을 사용하고, 신재생 에너지와 P2P 전력 거래 등과 같은 기능들을 도입하여, 환경 문제와 에너지 효율성을 증대시키고자 하였습니다.

(시스템 구조)

시스템 구조에 대해서 설명 드리겠습니다.

(구상도)

다음은 전체 구상도와 로컬 구상도입니다. 전체 구상도는 로컬과 로컬 간의 전력 거래를 컨트롤하는 모듈로 구성되며 여기서 해당되는 로컬은 오른쪽 그림에서 보다시피 이와 같이 구상 됨을 알 수 있습니다.

(좀 쉬고)

(상세기능 -> **천천히**)

상세 기능으로는 신재생 에너지 충전 제어, 소비량, 배터리 잔량 측정 및 표시, 원격 ON/OFF기능, P2P 전기 거래가 있으며, 각각 아두이노와 라즈베리파이로 제어가 되며 웹서버를 통해 데이터를 주고받게 됩니다.

가장 중점적으로 본 기능은 P2P 전기 거래 기능이므로 이에 대해 (더) 설명하겠습니다.

(Usecase Diagram )

P2P 전력거래의 Usecase Diagram은 Buyer, Seller, Web Server의 Actor가 있는데, 판매자가 판매글을 올리면, 그에 대해 구매자가 구매 요청을 하게 되고, 웹 서버가 구매자와 판매자의 거래를 중개하고 유효성 검사 등의 역할을 수행합니다.

( Sequence Diagram)

밑의 그림은 Usecase의 흐름을 토대로 작성한 Sequence Diagram입니다.

보시다시피 기본흐름은 T로 나타내었고, 대안흐름은 S로 나타내었습니다. 대안 상황에 걸리는 경우 빨간색 화살표를 따라가게 되고, 아니라면 파란색 흐름에 맞추어 검은색 흐름으로 넘어갑니다.

(Class Diagram)

다음은 Class Diagram인데, 저번에는 연결관계가 명확하지 않아 일부 수정하였습니다.

(ER Diagram)

ER Diagram은 다음과 같이 사용자 정보를 저장할 User와

기기 정보를 저장할 Device,

거래글 정보를 저장할 Transaction으로 구성되어 있습니다.

(팀원별 역할 분담).

저희가 개발 추진 계획을 크게 문서작업과 FrontEnd, BackEnd, 그리고 하드웨어로 작업을 나누었기 때문에 다음과 같이 역할 분담을 하였습니다.

(제안서 대비 수정 내용)

초기 제안서를 작성한 후 중간 발표를 하기까지 수정 된 내용에 대해 말씀드리겠습니다.

큰 변경점은 없었지만 구매량 지정 방식에 변경이 있었습니다.

수정 전에는 판매자가 업로드한 전력량 전부를 구매해야 하는 방식에서 구매자가 원하는 만큼 전력량을 정하여 거래할 수 있는 방식으로 변경하였습니다.

(개발 추진 계획)

개발 추진 계획과 현재 진행상황은 다음과 같습니다.

<질문 시 : 아직 하드웨어를 받지 못하여 하드웨어 쪽은 진행률이 미미하지만 현재 프론트엔드 쪽과 백엔드 쪽은 하드웨어만 연결되면 테스트는 가능할 정도로 개발이 진행되었습니다.>

(현재 구현 내용)

지금까지 개발한 내용에 대해 설명 드리겠습니다.

(1. 뼈대 파일 작성 및 각각의 UI 기본 구상)

(1,2 사진)

가장 먼저, 구상한 것들을 가시화하고, 서버에 병합시킬 때 좀더 용이하게 하기 위해, 뼈대 프론트엔드 파일들을 만들었습니다.

(2. MVC패턴을 적용한 기본 서버 구성)

(3번사진)

서버 작성 시 구조적으로 잘 정돈하여 추후에 디버깅하기 용이하게 MVC패턴을 적용하였습니다.

(3. DB 설계 및 스키마 생성)

(4, 5, 6번 사진)

앞서 정의한 ER 다이어그램에 맞게 스키마를 생성하였습니다.

저희의 경우, mongo DB를 사용하였는데 앞의 ER 다이어그램과 조금 차이가 나는 이유는 자료형에 대해서 float형과 int형에 대해 구분을 안 하기 때문에 number형이라 하였습니다.

(4. Integration)

(7번사진)

앞서 만든 기본 HTML 파일들을 바탕으로 백엔드에서 다음과 같이 라우팅 하였습니다.

실행시, 가장 먼저 로그인 페이지가 렌더링 되며, 로그인 하지 않을 경우, main 라우트를 포함한 하위 라우트에 접근하지 못하도록 설계하였습니다.

( 9 , 10 ~~)

이미지와 같이 각 라우트에 맞게끔 파일들이 렌더링 되는 것을 볼 수 있습니다.

이상으로 발표 마치겠숨돠!