1페이지

그럼 저희 3조의 프로젝트인 부동산거래 정보제공 웹 제작 프로젝트에 대해 발표하겠습니다

저는 3조 '개발자국'의 발표를 맡은 OOO 입니다.

2페이지

내용 순서는 다음과 같이 구성되어있습니다.

3페이지

먼저 프로젝트 개요에 대해 설명하겠습니다.

이번 미니프로젝트에는 크게 2가지의 주제가 있었는데 저희 개발자국 팀은 1번째 주제인

'부동산거래 정보제공 웹 제작' 주제를 선택하게 됐습니다

사용한 프로그램은 저번과 동일하게 Spring, 하이디SQL, Notion이 있습니다

4페이지

이번 프로젝트에서 구현하고자 했던 목표는

지금껏 쭉 해왔던 유저 관련 기능

그리고 API를 사용하여 지도를 만들고 거기에 각 건물의 위치를 표시하는 것

마지막으로 실제 거래내역을 DB로 담아 각 건물 정보를 차트의 형태로 제작하였습니다.

5페이지

그리고 해당 페이지를 제작하기 위해서 아실이라는 사이트의 디자인 및 구성과

국토교통부의 매매 실거래 자료

주소기반산업지원서비스의 API 데이터를 사용했습니다

6페이지

두 번째론 팀 구성 및 역할인데

7페이지

역할은 이렇게 나눴습니다.

8페이지

다음으론 프로젝트 제작 과정입니다

9페이지

제작 과정은 구상, 자료 수집, 파일 제작, 기능 구현, 오류 처리, 디자인 제작으로 나뉘었습니다.

10페이지

먼저 구상 단계에서는 노션을 이용하여 어떤 기능을 만들지 정했고

아까 소개한 참고 사이트에서 데이터를 엑셀로 받아서

데이터를 데이터 선별하고, 삭제하는 정리하는 과정을 거쳤습니다.

데이터를 정리한 다음에는 제작에 필요한 파일들을 생성했고

11페이지

HTML과 스프링으로 사이트의 기능을 구축한 다음

마지막으로 사이트 사용 중 발생한 오류를 하나하나 처리하는 것과 디자인 제작을 했습니다.

12페이지

이제 프로젝트 결과입니다

13페이지

저희가 제작한 사이트의 구성은 크게 4가지로 나뉘어집니다

사용자와 관련된 유저

주소, 우편번호 등을 보여주는 건물

건물의 거래 금액과 날짜를 확인하는 거래내역

각 건물에 댓글을 달거나 즐겨찾기 기능을 추가할 수 있는 각종 유틸리티입니다

14페이지

구체적으로 객체를 어떤 부분을 어떻게 구현했는지 설명하기전에

15페이지

저희 팀에서 제작한 사이트를 시연해드리는 시간을 갖도록 하겠습니다

(시연중엔 어떤 기능을 시연할지 발표자가 간단하게 말하기

ex: 댓글 작성 중이면 '댓글 작성 기능을 보여주고 있습니다' )

--------------------------------(시연할 목록)-----------------------------------

0. 메인페이지

0-1. 회원가입

1. 로그인

2. 건물 지도 (마커 보여주기)

3. 건물 검색

4. 건물 상세정보 (마커 클릭)

5. 거래내역이 있는 건물에서 평수에 따른 거래내역 그래프 보여주기

6. 건물 즐겨찾기(좋아요)

7. 댓글 작성

8. 대댓글 작성

9. 마이페이지 (즐겨찾기한 건물 목록의 이름 클릭하면 해당 건물의 상세정보로 넘어가는 거 보여주기)

----------------------------------------------------------------------------------

16페이지

시연 장면에서 페이지가 양쪽으로 나뉘어있는 것을 보셨을 텐데

저희는 아실이라는 참고사이트에서 했던 방식대로 오른쪽에 API를 이용한 지도를 띄우고

왼쪽에는 해당 건물의 정보나 검색기능을 만드는 식으로 사이트를 제작했습니다

17페이지

그래서 실제 구현 화면은 방금 보여드린 것과 같이

왼쪽은 건물 정보, 오른쪽은 지도를 띄우는 것을 보실 수 있습니다

18페이지

먼저 지도를 띄우는 것에 대해 설명드리자면

베이스 파일에서 지도를 띄우는 프래그먼트를 제작한 다음

건물 상세 정보를 나타내는 디테일에서 호출하는 식으로 제작했습니다

19페이지

좌측 영역에 건물별로 차트를 띄우는 것은 프래그먼트로 구현하기에는 제약사항이 너무 많아서

먼저 다른 페이지에서 차트만 띄우는 것을 만든 다음

아이프레임을 사용해서 왼쪽 공간에 해당 페이지를 불러오는 식으로 제작했습니다

20페이지

이게 실제 구현된 차트 화면인데

원래는 연월일도 사실상 연속된 데이터라서 x축도 연속된 데이터로 나타내야 하나

차트.js로 구현하기는 어려워서 결국 날짜별로 데이터를 찍는 것으로 해결했습니다

그리고 건물마다 넓이가 여러 종류가 있는 것을 감안하여

평수별로 다른 그래프를 보여주기 위해 버튼을 누르면 다른 그래프를 띄울 수 있게 구현하였습니다.

21페이지

건물에 쓸 수 있는 대댓글 기능을 구현하기 위해

댓글은 코멘트(comment), 대댓글은 리플라이(reply)로 데이터를 나누고  
댓글마다 대댓글을 표시하게 하되 번호가 일치하는 대댓글만 출력하는 식으로 제작했습니다

그리고 제이쿼리를 이용하여 대댓글 창을 만들 때

기존에 있던 대댓글창은 사라지게 하는 부분을 추가하였습니다.

22페이지

마지막으로 지도에 표시할 건물의 위도 경도 정보를 구하는 것인데

처음 저희는 매번 지도를 만들 때마다 API를 이용해서 각각의 위도 경도를 불러오는 방식을 사용했었습니다

하지만 해당 방식을 사용하니 수백개의 건물의 데이터를 리스트로 처리했기 때문에 로딩이 느려지는 문제가 있었습니다

그래서 먼저 처음 리스트를 만들 때만 건물의 위도 경도를 계산하여 DB에 갱신하도록 해두고

23페이지

두 번째부터는 건물의 위도 경도가 DB에 있으므로

그냥 리스트만 받아오는 식으로 짜서 위도 경도를 구할 때 걸리는 시간을 줄였습니다

24페이지

마지막으로 자체적으로 개선해야 할 부분을 간단하게 짚고 마치겠습니다

25페이지

첫 번째로 건물에 이미지가 없어서 다소 심심하게 나오는데

좀 더 신경써서 만든다면 건물별로 이미지를 저장해두고 출력시켜야 합니다

두 번째론 저희는 DB에 미리 데이터들을 넣어두고 지도를 만들었는데

이러면 데이터가 갱신되었을 때 일일이 수동으로 데이터를 추가하거나 수정해야 하는 문제가 있습니다

제대로 만든다면 이런 데이터들도 API를 이용하여 자체적으로 갱신할 수 있게 구현하겠죠

마지막으로 시간이 부족하여 수원시를 대상으로만 데이터를 구성했고

건물이나 거래내역의 정보도 임의대로 많이 생략시켰는데

좀 더 큰 규모의 프로젝트로 해당 주제를 한다면 당연히 더 많은 데이터를 처리할 수 있어야 합니다

26페이지

지금까지 개발자국의 발표를 마치겠습니다 감사합니다