



PORTFOLIO

지원자 권준석

2022.12

CONTENTS



001 자기 소개

- 인적 사항
- 배경 지식



002 웹 프로젝트

- 프로젝트 소개
- 구현내용



003 Unity 프로젝트

- 프로젝트 소개
- 구현내용



1 About Me

게임을 사랑하는 개발자

권준석

TEL > 010 – 2235 – 6108
E_MAIL > hd3379@naver.com

EDUATION/EXPERIENCE

한국산업기술대학교 게임공학전공
삼성청년 SW아카데미

2016.03 – 2022.02
2022.07 – 2022.12

SUBJECT COMPLETOIN

BASE

운영체제
알고리즘

자료구조
데이터베이스

LANGUAGE

C#

C++

JAVA

JS

WEB

JSP/Servlet
HTML

Spring
Vue

ETC

서버프로그래밍
컴퓨터 그래픽스

멀티코어 프로그래밍
3D게임 프로그래밍

2 웹 프로젝트

2022.11 (1주일간 진행)
권준석(팀장), 전주영

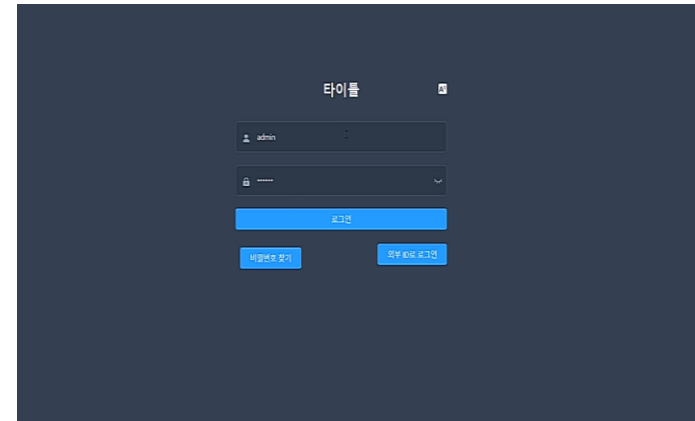
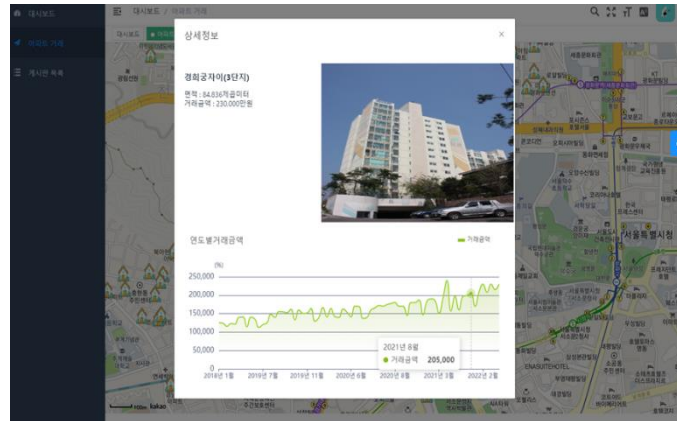
삼성 청년 SW아카데미 1학기 마무리 프로젝트

부동산 중개 앱 개발

Spring Boot, Vue, MySql 을 기반으로 제작

권준석: 부동산 관련 정보 및 부가기능 담당
전주영: 게시판 담당

시연 영상 링크: <https://youtu.be/n2ahoxZOQ70>



2 웹 프로젝트 구현내용

세부 구현 내용

>VIEW
API
DB설계

고려 사항

API와 데이터 송수신을 더 정확하고 편리하게 하기위한 TS사용

런타임 에러를 줄이기 위한 eslint,
협업시 코드 가독성을 높이기 위한
prettier 사용

◆ TypeScript를 사용해 Vue 코드를
작성한 모습

```
<script lang="ts">
import * as echarts from 'echarts'
import { Component, Prop, Watch } from 'vue-property-decorator'
import { mixins } from 'vue-class-component'
import ResizeMixin from '@/components/Charts/mixins/resize'

interface HouseInfo {
  apartmentName: string
  area: string
  dealAmount: string
  dealDay: number
  dealMonth: number
  dealYear: number
  dongcode: number
  lat: string
  lng: string
}

@Component({
  name: 'AptLineChart'
})
export default class extends mixins(ResizeMixin) {
  @Prop({ default: 'chart' }) className!: string
  @Prop({ default: 'chart' }) id!: string
  @Prop({ default: '200px' }) width!: string
  @Prop({ default: '200px' }) height!: string
  @Prop({ default: [] }) tableData!: HouseInfo[]
}
```

◆ CSS UI framework 로
el-element 를 사용

```
<div style="position:relative">
  <el-button
    class="thirdparty-button"
    type="primary"
    @click="showDialog=true"
  >
    {{ $t('login.thirdparty') }}
  </el-button>

  <el-button
    class="passwordSearch-button"
    type="primary"
    @click="showPWDialog=true"
  >
    {{ $t('login.SearchPassword') }}
  </el-button>
</div>
</el-form>

<el-dialog
  :title="$t('login.SearchPassword')"
  :visible.sync="showPWDialog"
>
  <search-password/>
</el-dialog>
```

2 웹 프로젝트 구현내용

세부 구현 내용

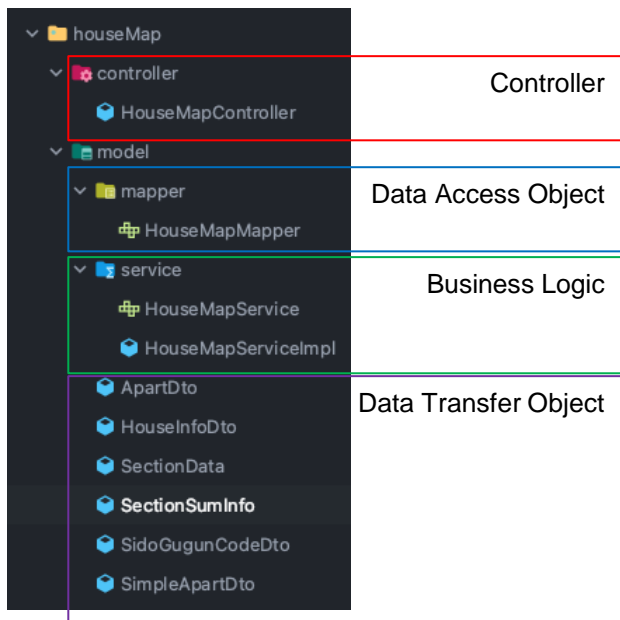
VIEW
>API
DB설계

구현 내용

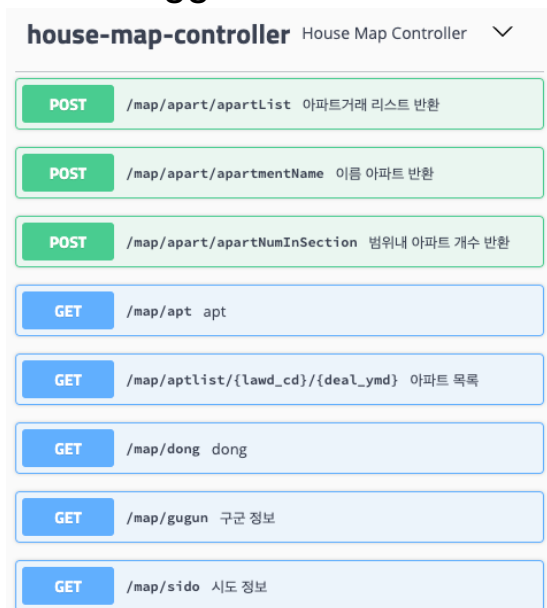
핵심 구현인 House-API 구성 모습

SpringFramework 사용,
DB연동을 위한 Mybatis 사용

◆ 기능별로 class를 정리



◆ 문서화 및 테스트를 위해 Swagger를 사용



2 웹 프로젝트 구현내용

세부 구현 내용

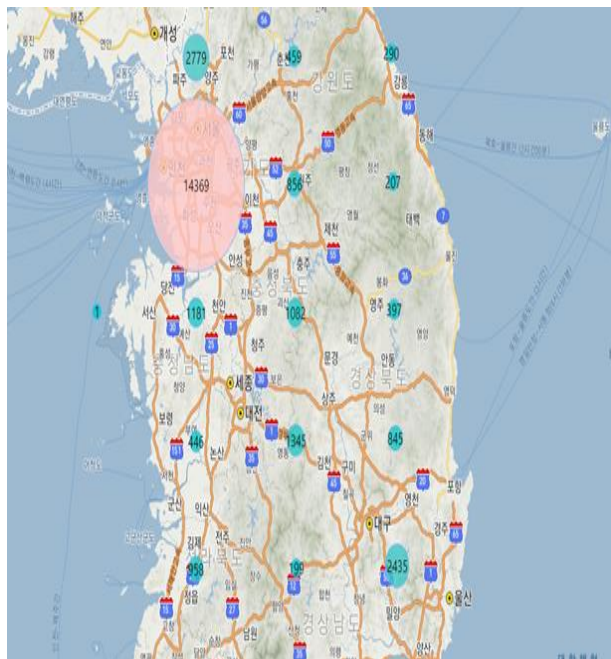
VIEW
>API
DB설계

구현 내용

부동산 거래 정보 최적화

대한민국 좌표를 1000 * 1000 의 구역으로 나눠 지도 정보에 해당 구역 정보만 가져올 수 있게 구현

◆ 탐색 범위가 넓어지면 숫자와 원 색으로 해당 부동산 수만을 표시



◆ 구역을 나누는 핵심 코드

//aptSection에 사용할 데이터들

```
private final double startLat = 34.271572, startLng = 125.790695;  
private final double endLat = 38.537473, endLng = 129.486041;  
private final double latBlock = (endLat - startLat) / 1000.0;  
private final double lngBlock = (endLng - startLng) / 1000.0;  
private SectionData[][] aptSection;
```

```
for(SimpleApartDto aptInfo: aptInfos) {  
    if(aptInfo == null) {  
        continue;  
    }  
    int latIndex = (int) ((endLat - aptInfo.getLat()) / latBlock);  
    int lngIndex = (int) ((endLng - aptInfo.getLng()) / lngBlock);  
    if(latIndex < 0 || lngIndex < 0 || latIndex >= 1000 || lngIndex >= 1000) {  
        continue;  
    }  
    aptSection[latIndex][lngIndex].add(aptInfo);  
}
```

2 웹 프로젝트 구현내용

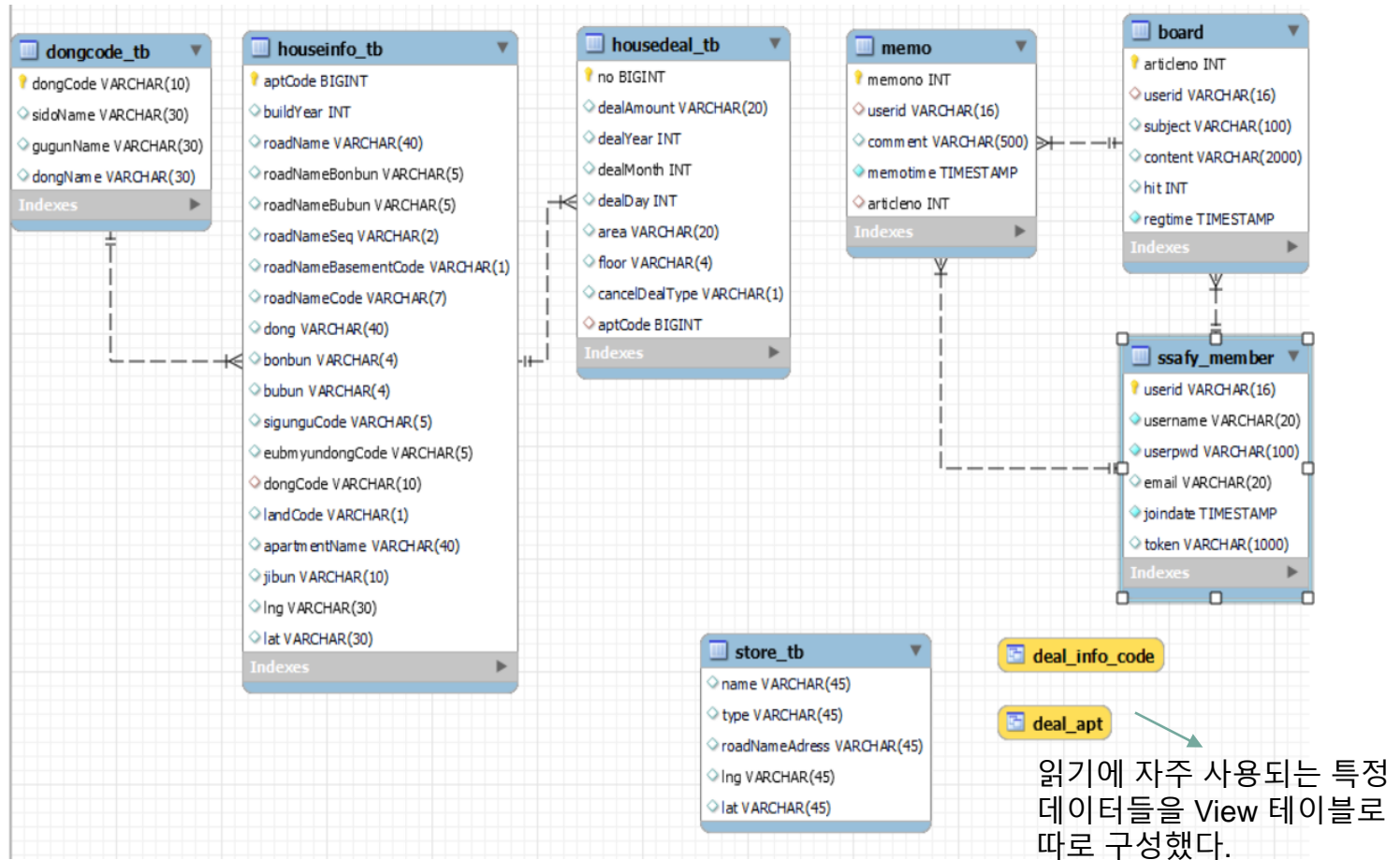
세부 구현 내용

VIEW
API
>DB설계

고려 사항

설계한 DB에 대한 ERD 모습이다.

houseinfo 관련 테이블은 앱 사용 중 업데이트 없이 읽기만 요구돼 추가적인 정규화 없이 데이터 추가에 편한 형태로 사용하였다.



3 Unity 프로젝트

2021.01 ~ 2021.08

권준석(팀장), 박건웅, 오선재

한국산업기술대학교 졸업작품

Oculus Quest2 연동 VR 어드벤처 게임

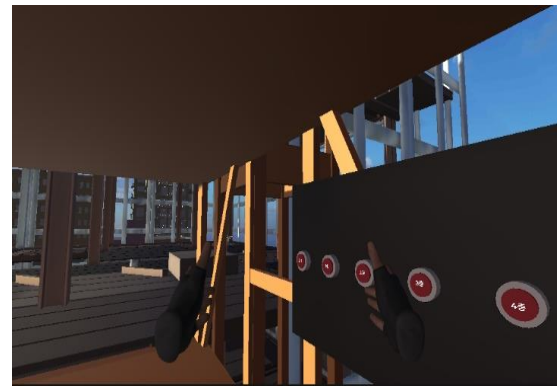
Unity 기반으로 개발

권준석: 클라이언트 담당

박건웅: VR연동, 기획, 클라이언트보조

오선재: 리소스 제작 담당

시연 영상 링크: <https://youtu.be/yn-UPbUfDMM>



3 Unity 프로젝트 구현내용

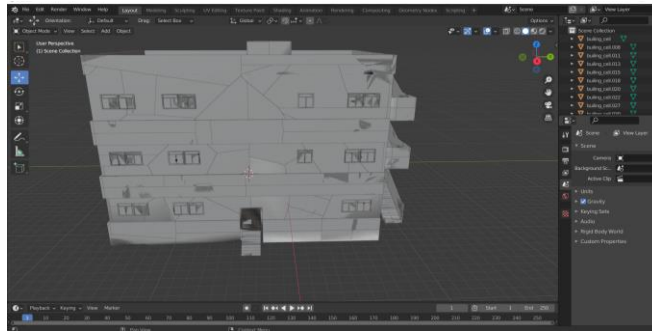
세부 구현 내용

>이팩트
조명
컷신

구현 효과

오브젝트들이 잘게 부서진 후 다시
조립되는 장면

◆ Blender의 celfracture 기능으로
조각난 오브젝트를 준비했다.



◆ 일정 거리만큼 차례로 흩어졌다
다시 원위치.



3 Unity 프로젝트 구현내용

세부 구현 내용

>이팩트
조명
컷신

구현 효과

시간 충돌의 표현으로 화면 왜곡
효과

```
vertexOutput vert(vertexInput input)
{
    vertexOutput output;

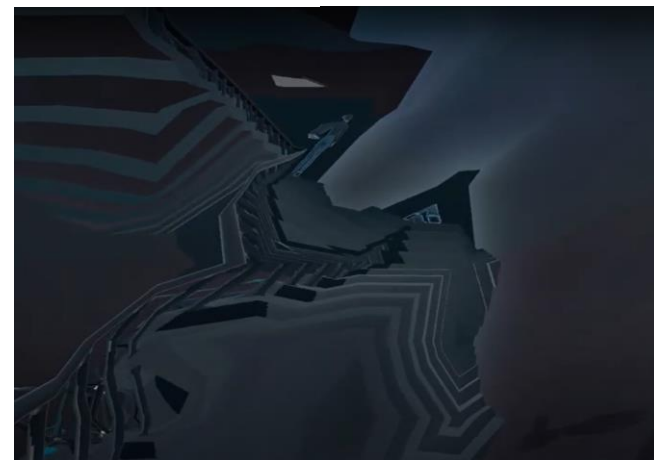
    output.pos = UnityObjectToClipPos(input.vertex);

    output.grabPos = ComputeGrabScreenPos(output.pos);

    float noise = tex2Dlod(_Noise, float4(input.texCoord, 0)).rgb;
    float3 filt = tex2Dlod(_StrengthFilter, float4(input.texCoord, 0)).rgb;
    output.grabPos.x += cos(noise * _Time.x * _Speed) * filt * _Strength;
    output.grabPos.y += sin(noise * _Time.x * _Speed) * filt * _Strength;

    return output;
}
```

◆ 위 코드는 `_Strength`만큼
의 왜곡 강도와 `_Speed`만큼
의 속도로 왜곡을 일으키는
셰이더 코드 일부이다.



3 Unity 프로젝트 구현내용

세부 구현 내용

이팩트
>조명
컷신

고려 사항

성능 > 용량
frame rate 가 유지되는 것을 최우선

Light mode = Shadow mask

Shadow mask 특징 :

1. 라이트맵 베이킹 시 그림자 영역을 별도의 텍스처로 따로 저장
2. 동적인 오브젝트에 대해서만 쉐도우맵 연산으로 성능 향상
3. 라이트맵 외에도 별도의 텍스처를 저장하기에 용량이 큼

건물 내부 조명?

건물 내부에서 형광등이 켜지거나 손전등을 비추는 표현이 나와 조금 더 자연스러운 조명 반사 효과가 필요했다. 이를 위해 리플렉션 프로브를 적용해 건물 내부 영역 안에 환경 반사 이미지를 별도로 사용했다.

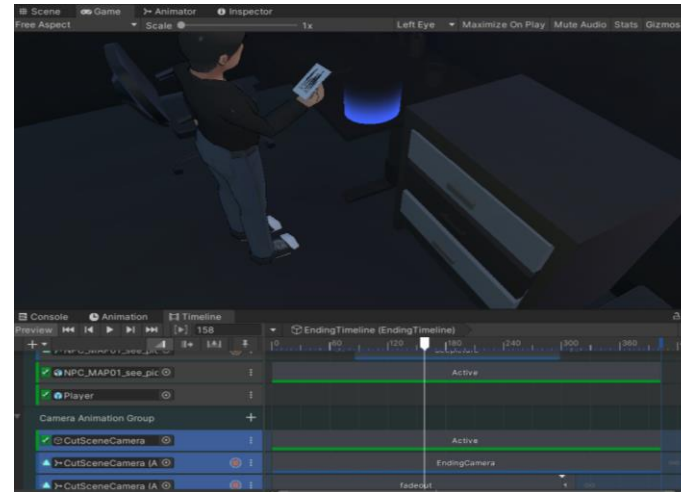
Unity 프로젝트 구현내용

세부 구현 내용

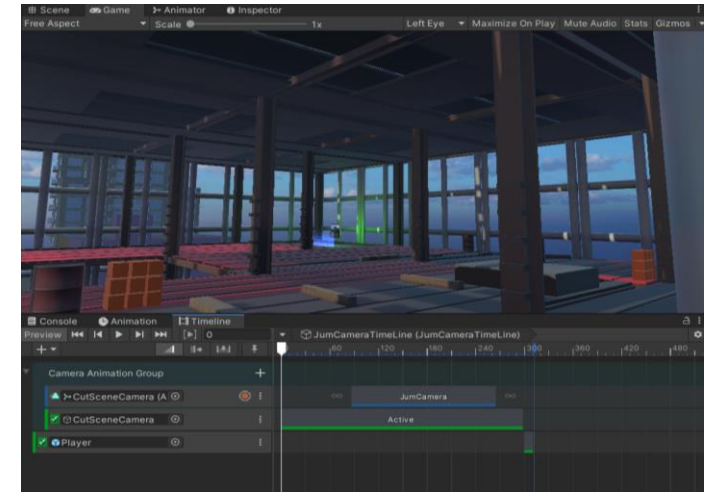
이팩트
조명
>컷신

구현 내용

이벤트 발생 및 스토리 진행에
컷신을 사용



◆ 유니티의 Timeline 기능을
사용한 컷신 제작



◆ 목표물을 알려주기 위한
카메라 줌에 사용

감사합니다

지원자 권준석