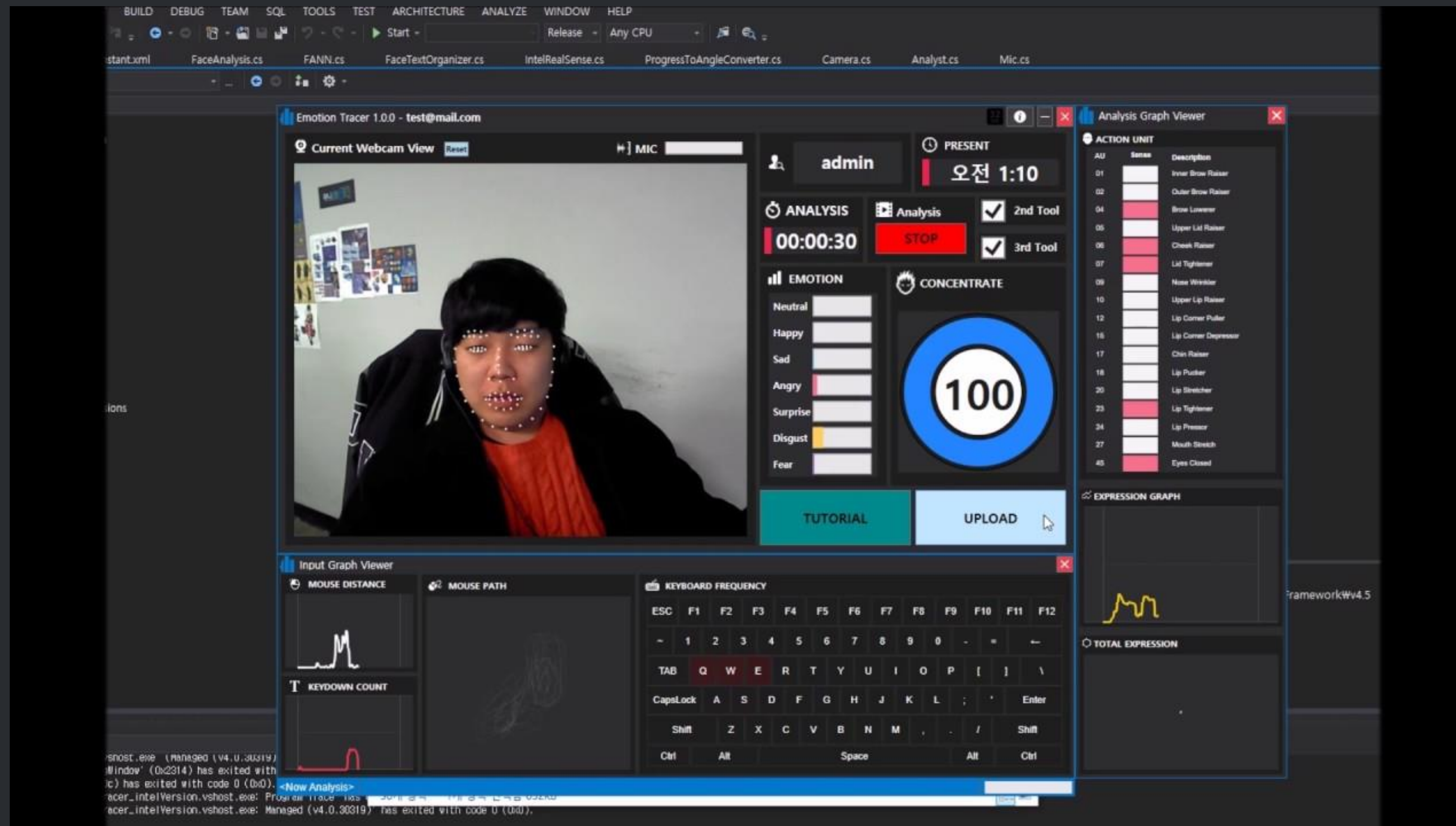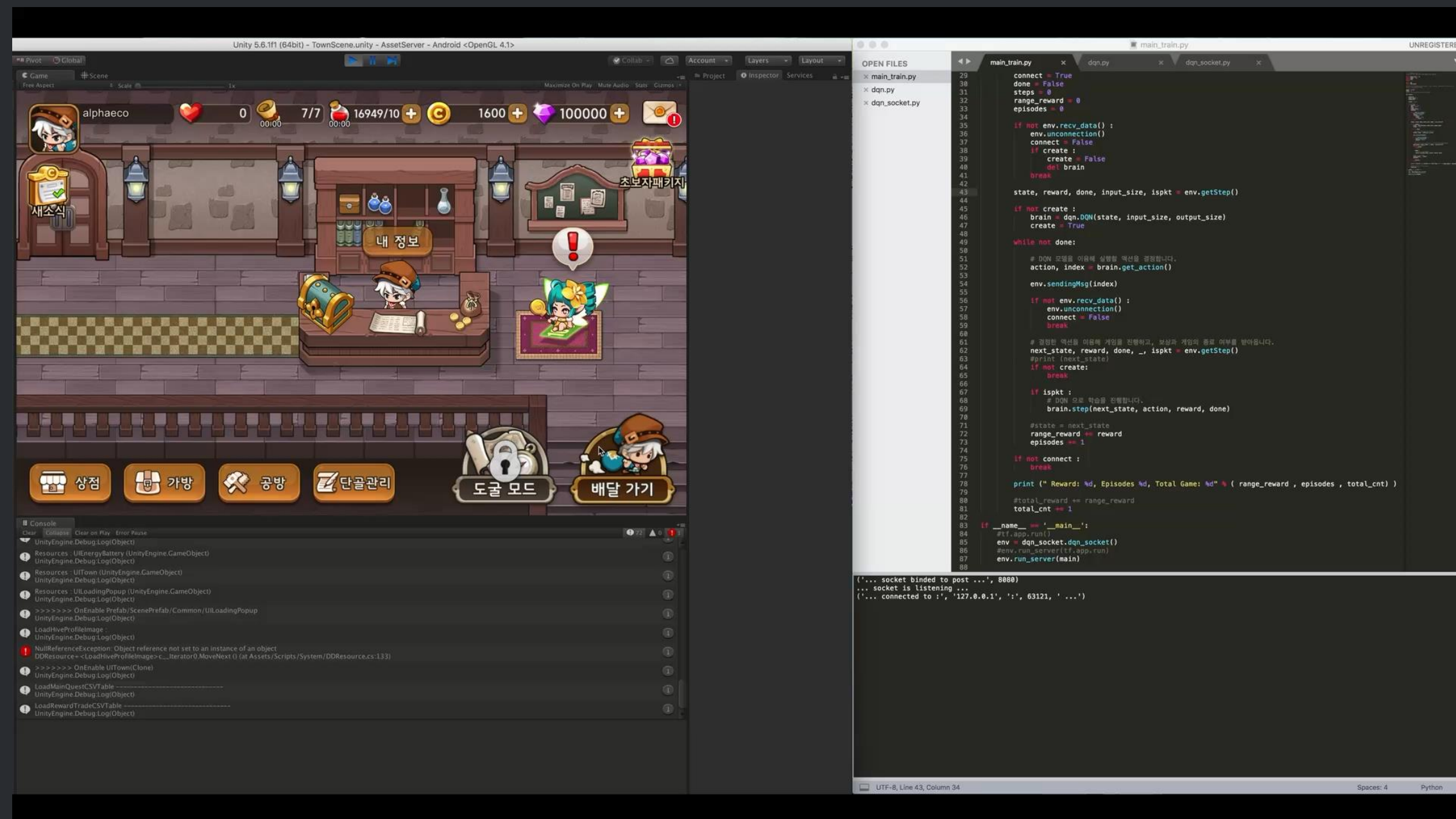# 포트폴리오

2017.12.26 홍익대학교 EGLab,
"감성분석기술" JTBC 특허기술이전

- 얼굴 표정 분석기, Emotion Tracer
- 생체 신호 탐지기, Emotion Insole

Emotion Tracer 툴 및 GCP 기반 데이터 결과 확인 연동서비스

2017.07 ~ 2017.12 Com2us,
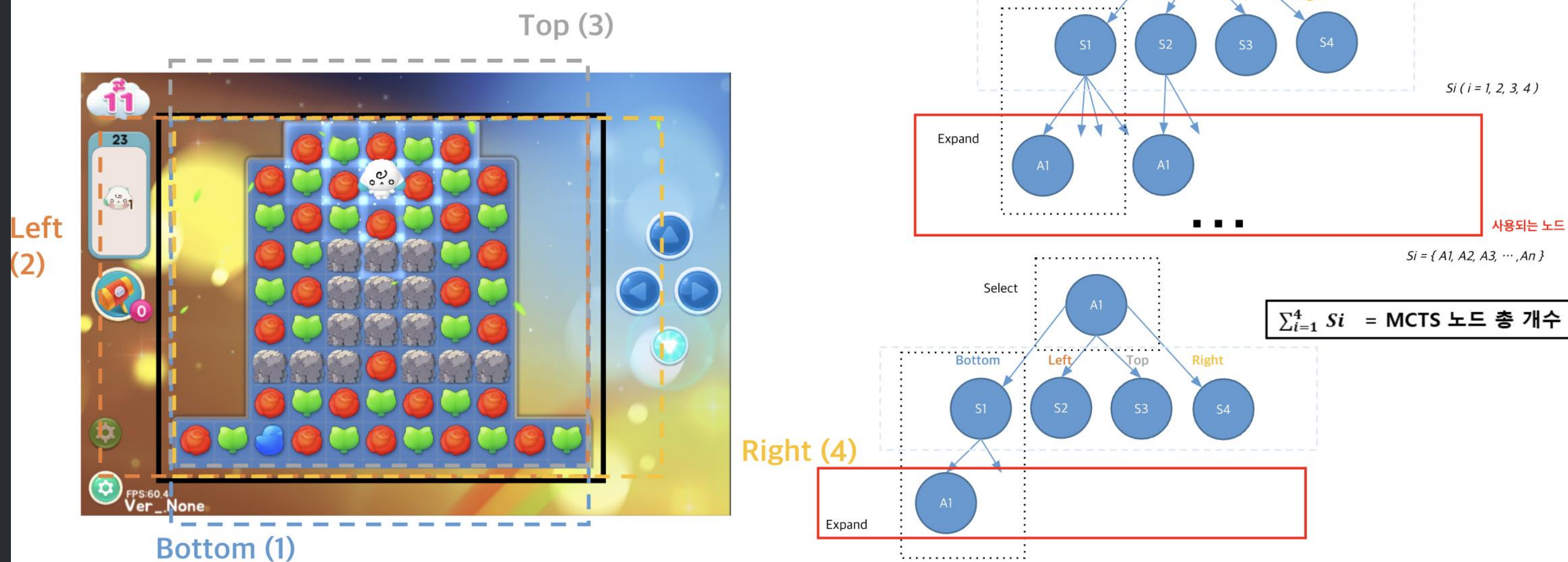## "던전 딜리버리" 강화학습 에이전트 적용 테스트

던전딜리버리 학습 시스템 구축 (Server-Client)

• 게임에 AI 기술 적용을 위한 탐색 및 가능성 타진

• Tensorflow로 작성된 강화학습 모델 설계 및 소켓 통신을 이용한 게임 적용

• Gridmap 을 이용한 환경정보, Action Decision을 위한 DQN 모델 적용

# 2018. 03 ~ 2018.09 NC SOFT, MCTS(Monte Carlo Tree Search)기반 3-Match 퍼즐 게임 난이도 평가 Agent 개발

- MCTS Node 확장 방식

Top (3)
Left (2)
Bottom (1)
Right (4)

Root
Bottom Left Top Right
S1 S2 S3 S4
S_i (i = 1, 2, 3, 4)
Expand
A1 A1
· · ·
사용되는 노드
S_i = { A1, A2, A3, ··· , An }
Select
A1
$\sum_{i=1}^{4} S_i$ = MCTS 노드 총 개수
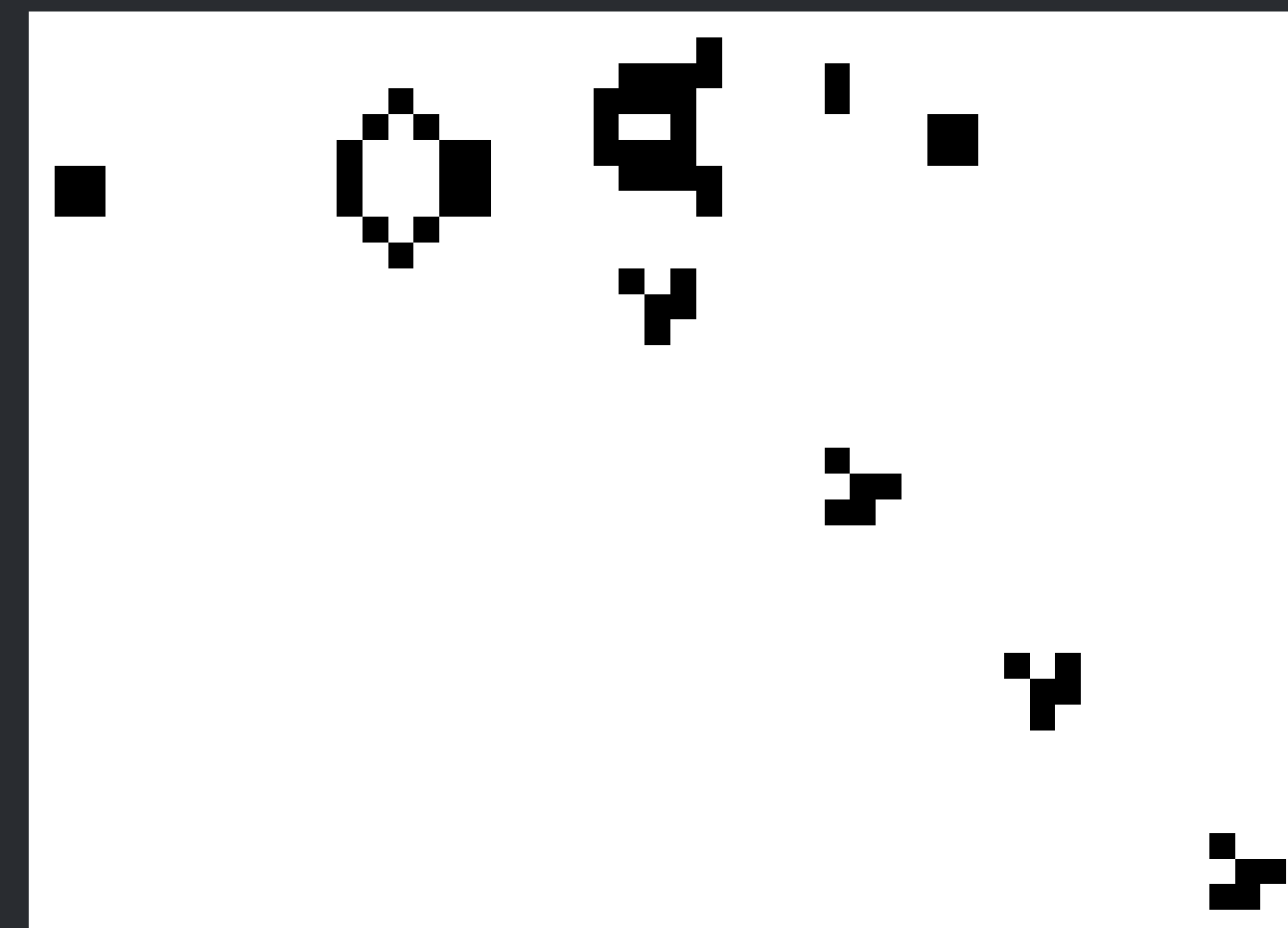Bottom Left Top Right
S1 S2 S3 S4
Expand
A1

- 3-Match 게임의 난이도를 시뮬레이션 기반으로 평가하기 위한 Agent 개발

- MCTS 기능 구현 및 게임 로직에 Agent 적용

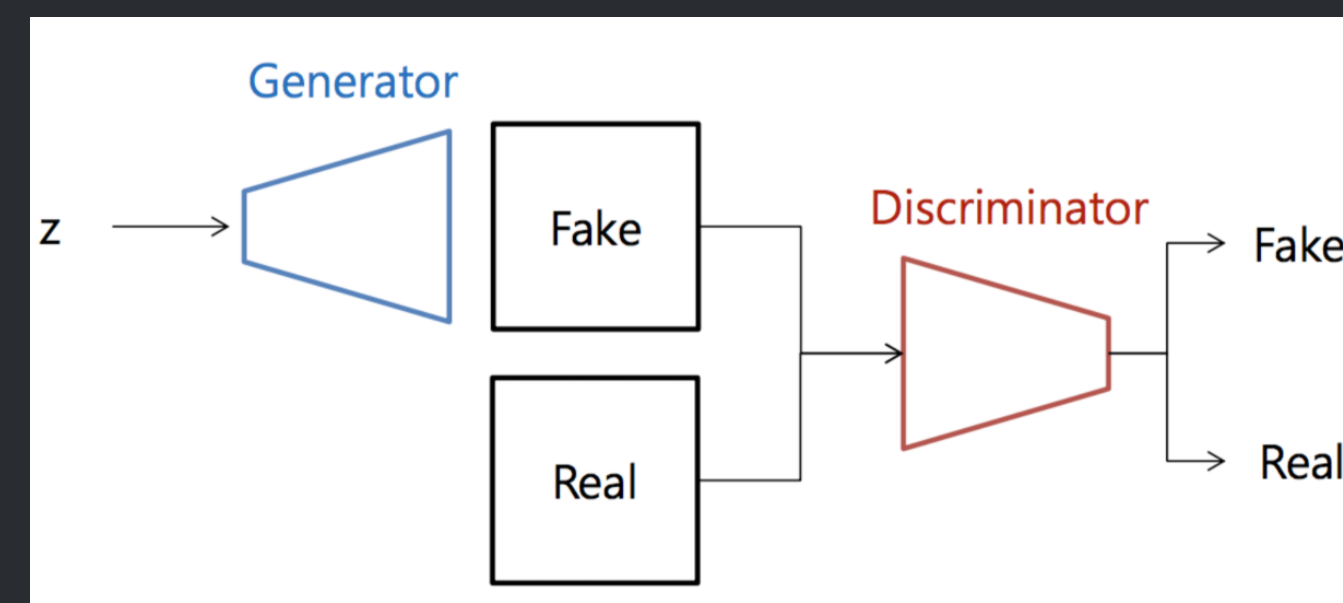- 가속화 및 난이도 평가 결과 도출

2018. 10 ~ 2021. 03 NC SOFT,
CA(Celluala Automata)기반 3-Match 퍼즐게임 난이도 생성기 개발
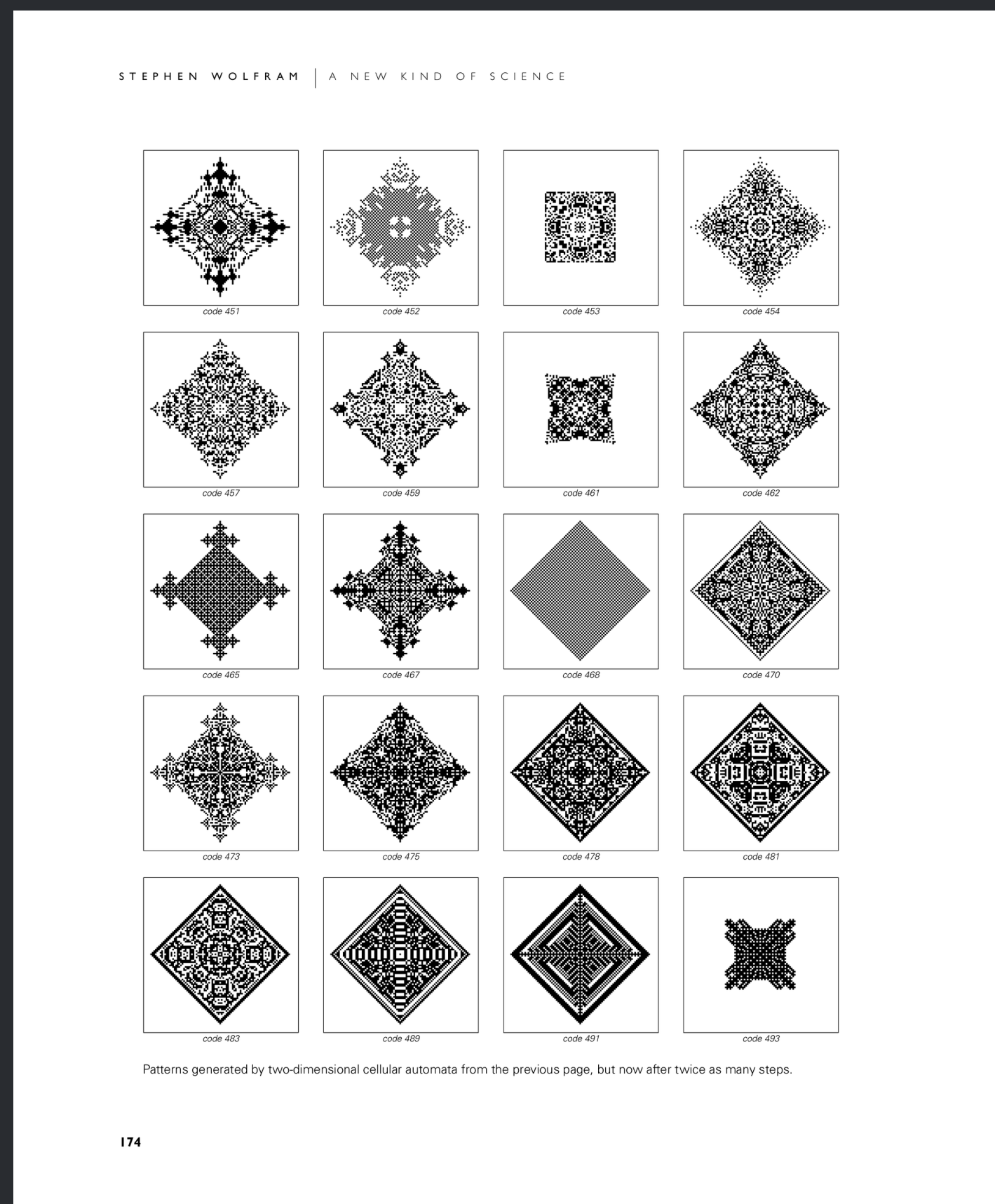
Patterns by CA

Game of Life

GAN

- CA 기능 구현 및 CA 기반 Generator 개발

- CA Generative Combination 을 제안하여 Co-relationship pattern 개선

- GAN (Generative Adversarial Networks) 모델 학습데이터로 활용

2024.07 ~ 2024. 08 개인프로젝트,
추리게임용 온 디바이스 SLM Chatbot 경량화 및 런타임 최적화

```
PeftModelForCausalLM(
    (base_model): LoraModel(
      (model): GPTNeoXForCausalLM(
        (gpt_neox): GPTNeoXModel(
          (embed_in): Embedding(30080, 2048)
          (emb_dropout): Dropout(p=0.0, inplace=False)
          (layers): ModuleList(
            (0-23): 24 x GPTNeoXLayer(
              (input_layernorm): LayerNorm((2048,), eps=1e-05, elementwise_affine=True)
              (post_attention_layernorm): LayerNorm((2048,), eps=1e-05, elementwise_affine=True)
              (post_attention_dropout): Dropout(p=0.0, inplace=False)
              (post_mlp_dropout): Dropout(p=0.0, inplace=False)
              (attention): GPTNeoXSdpaAttention(
                (rotary_emb): GPTNeoXRotaryEmbedding()
                (query_key_value): lora.Linear(
                  (base_layer): Linear(in_features=2048, out_features=6144, bias=True)
                  (lora_dropout): ModuleDict(
                    (default): Dropout(p=0.1, inplace=False)
                  )
                  (lora_A): ModuleDict(
                    (default): Linear(in_features=2048, out_features=8, bias=False)
                  )
                  (lora_B): ModuleDict(
                    (default): Linear(in_features=8, out_features=6144, bias=False)
                  )
                  (lora_embedding_A): ParameterDict()
                  (lora_embedding_B): ParameterDict()
                  (lora_magnitude_vector): ModuleDict()
                )
                (dense): Linear(in_features=2048, out_features=2048, bias=True)
                (attention_dropout): Dropout(p=0.0, inplace=False)
              )
              (mlp): GPTNeoXMLP(
                (dense_h_to_4h): Linear(in_features=2048, out_features=8192, bias=True)
                (dense_4h_to_h): Linear(in_features=8192, out_features=2048, bias=True)
                (act): GELUActivation()
              )
            )
          )
          (final_layer_norm): LayerNorm((2048,), eps=1e-05, elementwise_affine=True)
        )
        (embed_out): Linear(in_features=2048, out_features=30080, bias=False)
      )
    )
)
```

PEFT LoRA 학습결과

```
model loading...
/Users/jangyoung/Documents/Github/LiarHeart/Assets/StreamingAsse
Special tokens have been added in the vocabulary, make sure the
model completely loaded
실행 시간 : 11.57147 초
클라이언트와 연결을 대기합니다 ...
클라이언트가 연결되었습니다. 주소 : ('127.0.0.1', 61251)
<socket.socket fd=5, family=2, type=1, proto=0, laddr=('127.0.0.
받은 답변 :  Shinyooyoung!@#$$#@!신유영 오후 6시 이후로 뭘 하고 있
답변 생성중
character: Shinyooyoung
questionText:  신유영 오후 6시 이후로 뭘 하고 있었지 ?
<startoftext>이름 : 신유영
질문 : 신유영 오후 6시 이후로 뭘 하고 있었지 ?
답변 :
생성된 전체 텍스트 :  <startoftext>이름 : 신유영
질문 : 신유영 오후 6시 이후로 뭘 하고 있었지 ?
답변 : 집에서 ...... 쉬고 있었어요 ...... 심란했어요 .......
감정 : 감정없음

생성된 답변 : 집에서 ...... 쉬고 있었어요 ...... 심란했어요 .......
생성된 감정 : 감정없음
실행 시간 : 12.23388 초
```
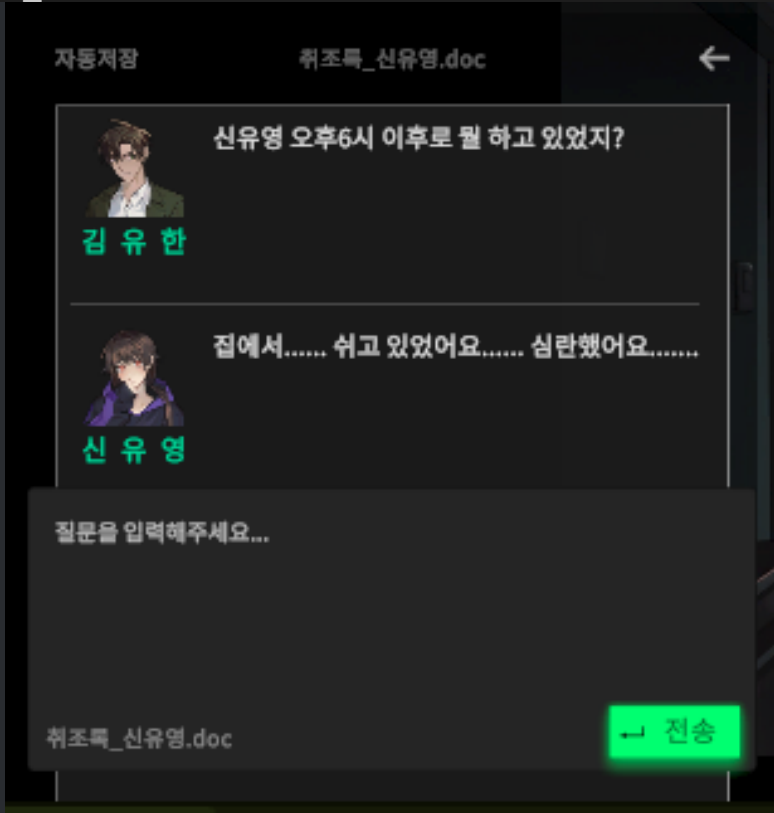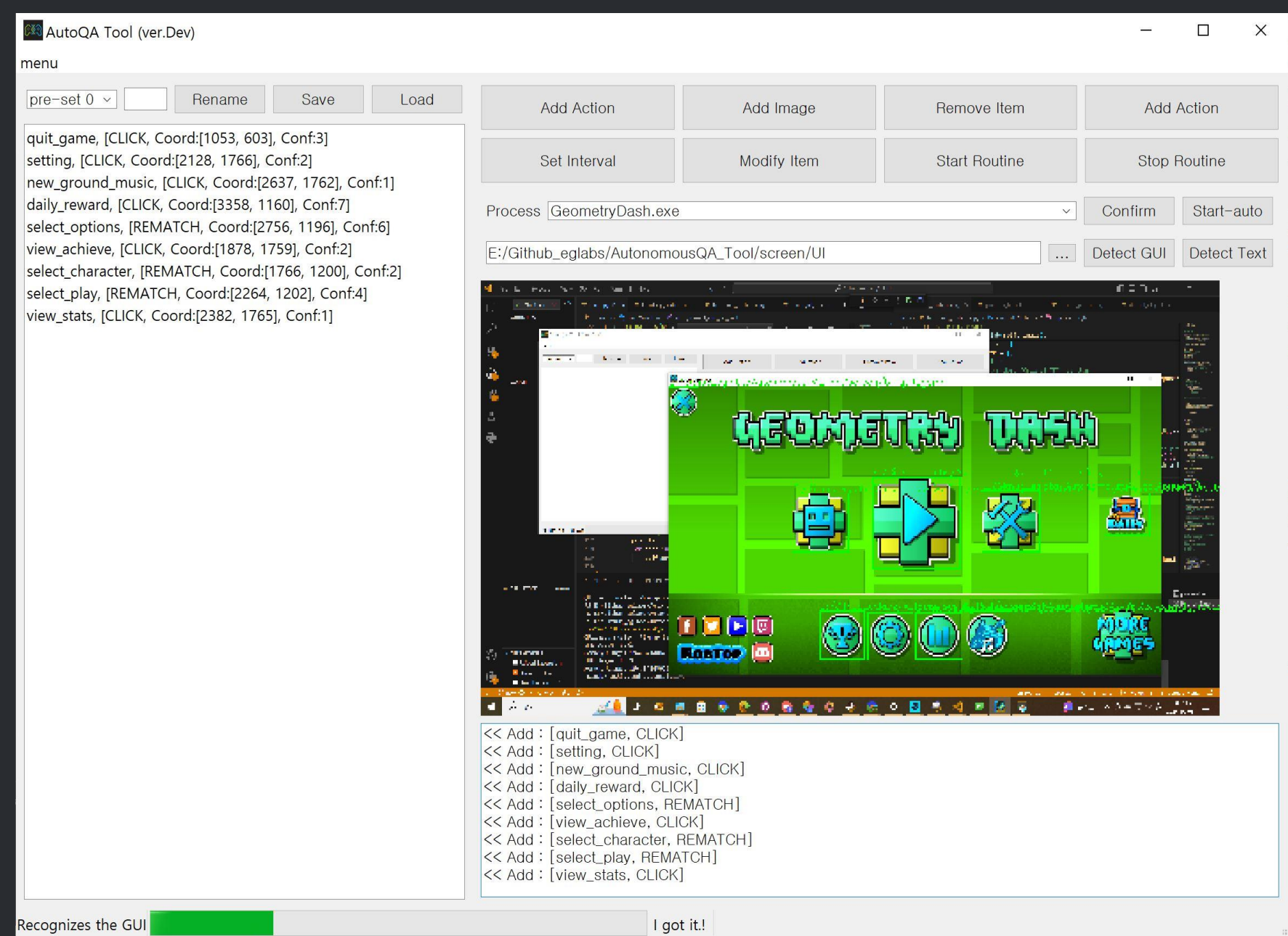
추리게임에 적용된 화면

- Polyglot-kr-1.3B 을 PEFT LoRA 학습

- LLM Python server 경량화를 위한 batch 로 분리 및 구조 설계

- Application service 를 위한 CPU 추론 모델 ONNX exporting

# 2024. 08 ~ 2024. 09 개인 프로젝트,
이미지 기반 시스템 독립적 AutoQA Tool

AutoQA Tool 동작 화면

- OpenCV 기반 Multi-Scale Template Matching 알고리즘 적용

- Thread Pool (GIL) 이용한 속도 최적화 및 Window Handler 적용한 Macro 기능

- Keras OCR 적용

- Fine-tune DETR 적용

- Fine-tune Yolov7 적용

- (~현재) VLM 모델을 이용한 Context 오류 탐지 리포트 기능추가

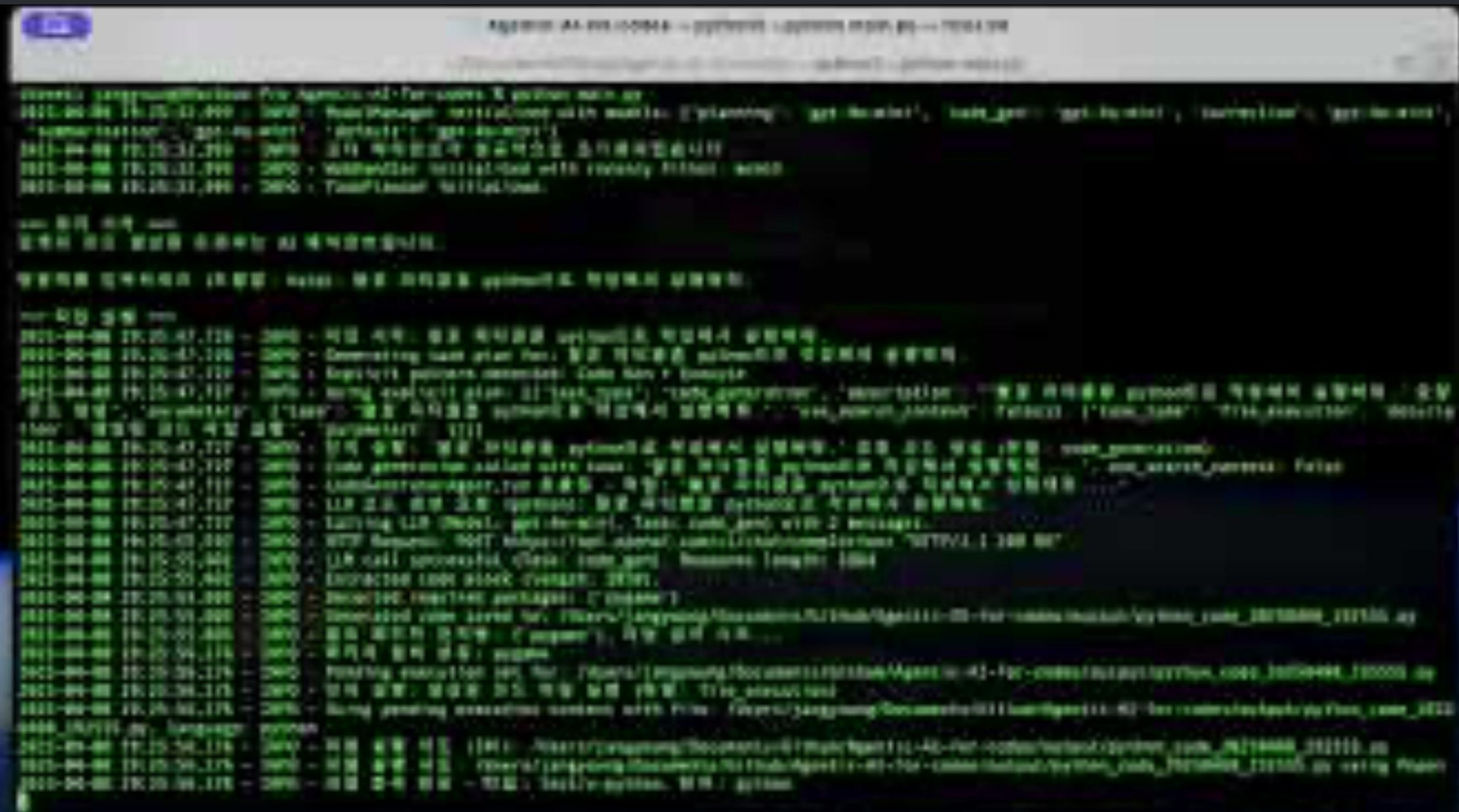# 2025. 01 ~ 2025.02. 개인프로젝트,
SLM 기반 온 디바이스 에이전트 시스템 구축 프로젝트

Flutter 채팅 기능 구현 및 모델 테스트
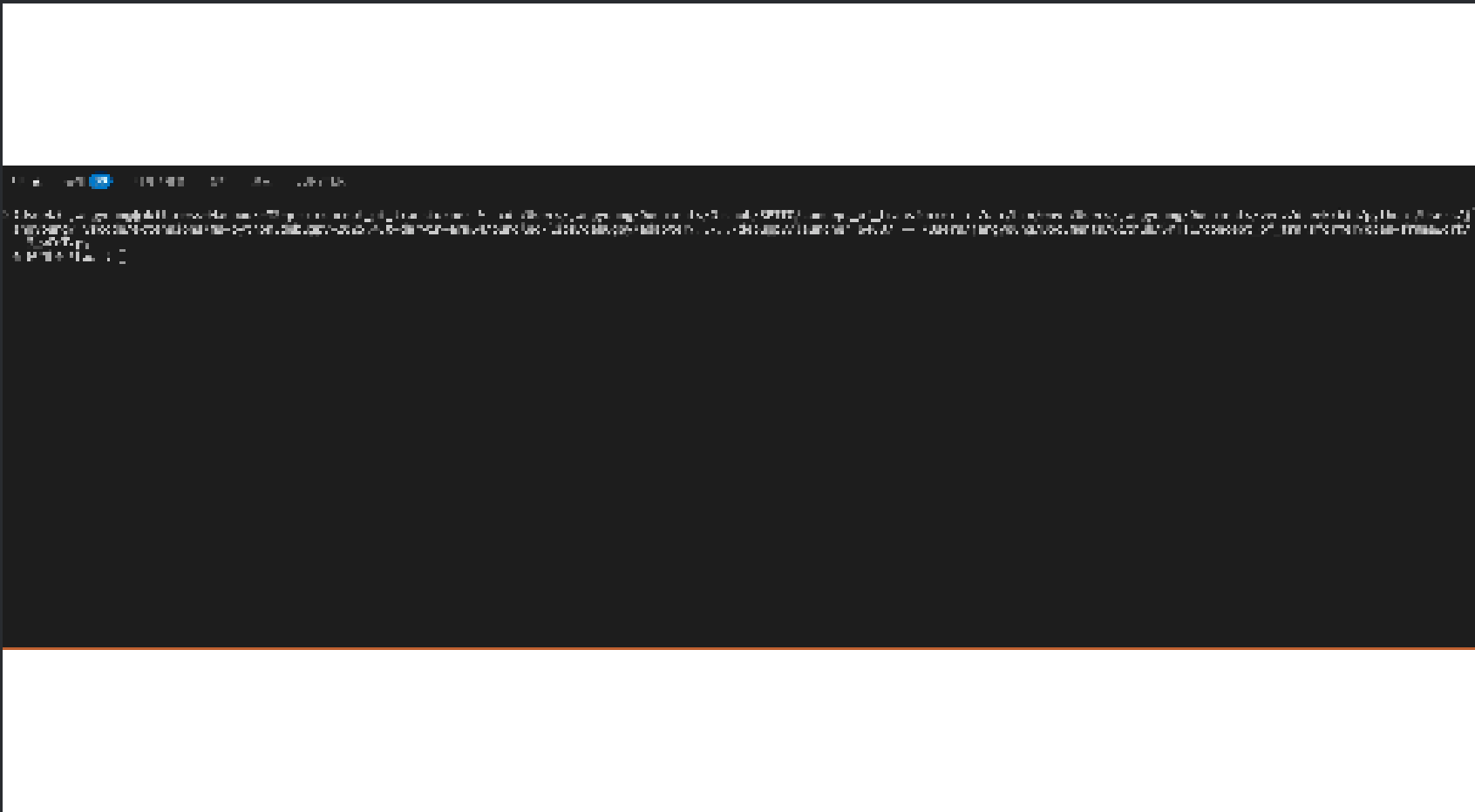
- DeepSeek R1 Distil Qwen 1.5B, TinyLlama, DistilBERT 등 모델 및 LlamaIndex, LangChain, LangGraph , CoT 적용 테스트 완료

- MCP(Model Context Protocol) 을 활용한 Agentic AI 기술 테스트

- Flutter 챗팅 Client 환경 구성
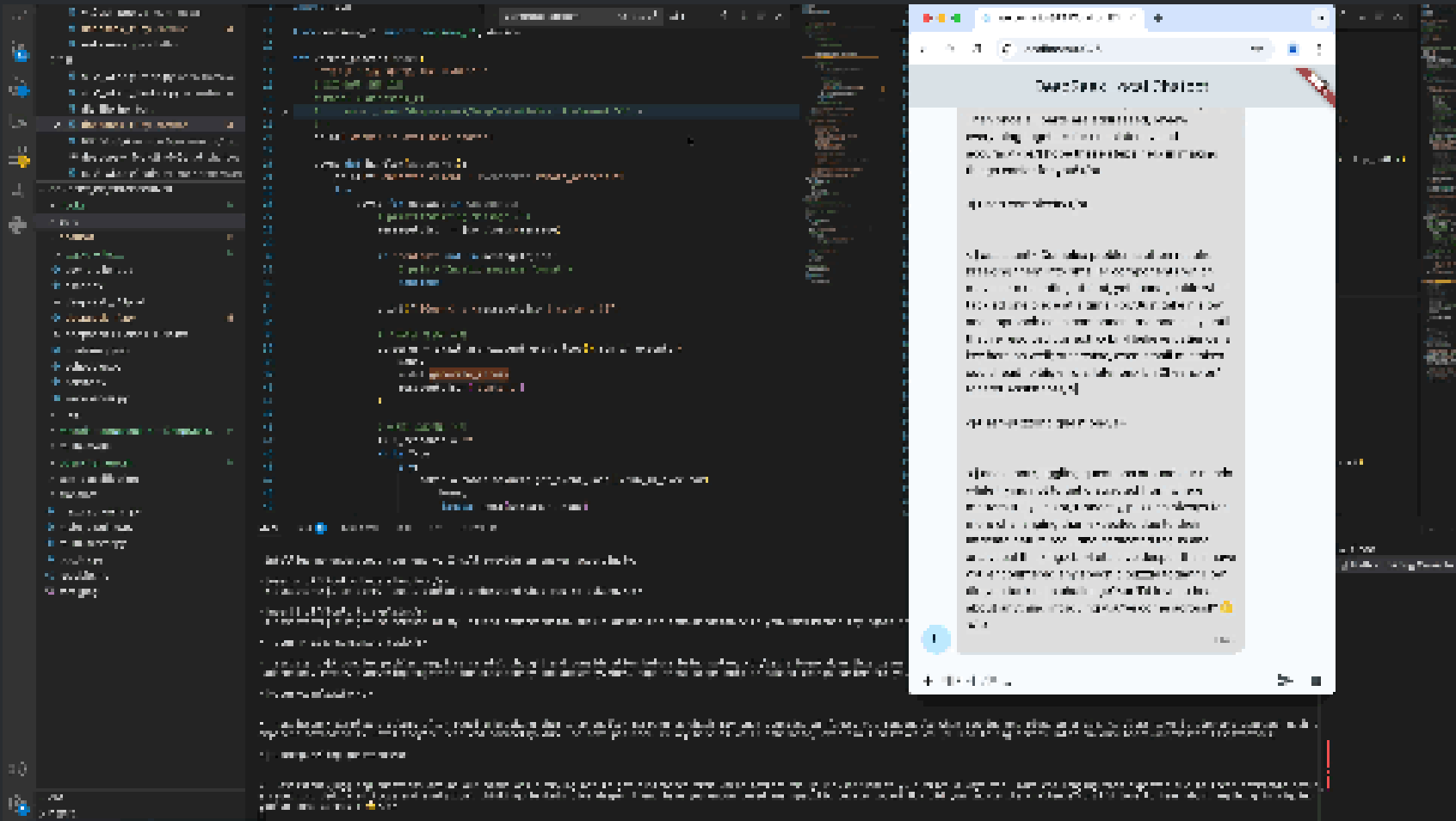
2025. 02 ~ 2025.03. 개인프로젝트,
LLM 기반 Agentic AI – 코딩&웹 검색 : 바이브 코딩

코드&웹 검색 에이전틱 AI

- LLM 기반의 Smol Agent 구조 활용 (OpenAI GPT API)

- Agentic AI의 Agent 기능 구현

- 언어별 코딩 파일 생성 및 실행 기능 구현, 구글 검색 기능 구현

2025. 03 ~ 2025.03. 개인프로젝트, LLM 기반 Game Design Agent 개발

- ChatGPT API 를 활용한 Vive Game Design Agent 개발

- ColBERT 기반의 Embedding 를 활용한 Score 기반 Chunk, Qdrant VectorDB

- Task Planning 을 통한 Task Role (기획,시스템,밸런스,이벤트,시나리오) 동작

- Document(pdf,word,excel,pdf, pptx), Crawling 의 Function Calling

시나리오 데이터(링크,문서)로부터 게임기획서(컨텐츠,시스템,밸런스,이벤트,시나리오) 제작 에이전트

2025. 04 ~ 2025.05. 개인프로젝트,
지능형 Multi-Agentic AI framework with Ollama, LangGraph & PraisonAI

- 온 디바이스 Ollama 서버, praisonaiagens 를 이용한 multi-agents Workflow process

- Knowledge Graph 기반의 Sementic Routing

- MCP(Model Context Protocol), Function Calling(Crawling) Handler

지능형 멀티 에이전트 워크플로우 프레임워크 구축