



BORROWED\_CAT

LG SW BOOTCAMP

# DingQ

쉽게 찾는 탐색에서 출발하여, webOS 생태계로 확장하는

**Minha**

AI/ML Engineer

**Seungkeun**

Backend Engineer

**Jihyuk**

Frontend Engineer

**모두를 위한 Dingbat 탐색 피드**

Find us @ dingq.art

### 프로젝트 요약

- 서비스명 : DingQ
- 핵심 기능 : 딤벳(Dingbat) 심볼을 손그림으로 그려
  - 1.검색하거나 2.없으면 AI 생성해주는 웹 기반 서비스
- 기존 문제 : 딤벳의 라벨을 직접 유추하여 찾아야 하나 직관적이지 않아, 원하는 심볼 찾기가 어려웠음
- 차별점 : 시각적 이미지 기반으로 딤벳 탐색이 가능함
- 사용자 : LG webOS 개발자/디자이너, 외부 디자이너
- 기대 효과 : 딤벳 탐색 시간 절감으로 생산성 향상, 신규 심볼 AI 생성 기능으로 아이디어 구체화 지원
- 접근성 : 별도 설치, 보안 없이 웹에서 바로 사용 가능
- 사용 기술 : 멀티모달 임베딩, Flutter, GCP, LLM

**i** 딤벳(Dingbat)이란?  
픽토그램 형태의 심볼로, LG webOS 구성요소중 하나



### 주요 기능 및 UI

#### 쉽게 찾는 딤벳 탐색

: 손그림으로 기존 DB에 있는 딤벳을 검색 가능함.  
유사도 기반 순위로 매칭 결과를 얻을 수 있음

【손그림 검색 사용 Scene】

【태그 기반 분류】

유사도 매칭 결과

### 서비스 배경

- 딤벳을 찾는데 불편을 겪는 사용자들 ...

- 딤벳을 빠르게 찾어야 하는 개발자 → “디자인 문서의 딤벳이 어디있는지 모르겠어요”
- 기존 딤벳을 모두 알아야 하는 디자이너 → “기존 딤벳이 어떤게 있었는지 기억이 안나요”
- webOS 앱을 개발하는 외부 디자이너 → “webOS와 잘 어울리는 신규 딤벳이 필요해요”

- 사용자들은 쉽고 빠른 딤벳 탐색을 필요로 하며, DingQ는 태그 분류, 손그림 검색, AI 생성을 통해
- 즉각적인 탐색과 창의적 아이콘 활용을 가능하게 함

#### 사용자 생성 플레이그라운드

: 기존 DB에 없는 심볼인 경우 사용자 손그림으로 LG 스타일 딤벳을 생성하고, 플레이그라운드에 적재됨

#### 【손그림 AI 생성 사용 Scene】

1. 사용자 손그림 & 간단 설명

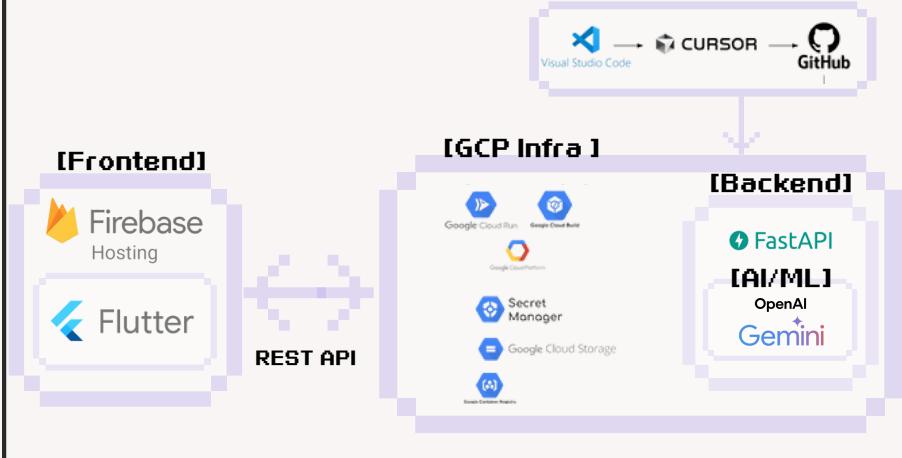
2. LG 딤벳 스타일 참고 프롬프팅

3. LG 스타일 딤벳 생성 및 다운로드

4. 사용자 생성 플레이그라운드 적재

## 시스템 아키텍처

- Flutter와 Firebase를 기반한 웹 애플리케이션으로, 프론트엔드는 Flutter로 구현해 실시간 드로잉과 반응형 UI를 제공하고, 백엔드는 GCP, FastAPI 기반으로 AI 모델과 연동하여 결과를 반환하는 구조로 설계함. 전체 시스템은 클라우드 기반 배포해 안정성과 확장성을 확보함



## 기대효과

### 1. 드로잉 기능으로 사용성 향상 🎉

사용자가 직접 스케치하여 아이디어를 시각적으로 표현할 수 있도록 실시간 드로잉 캔버스를 제공하여, 사용자 경험 개선

### 2. 생성형 AI를 통한 아이콘 생성 💡

사용자의 드로잉을 기반으로 맞춤형 아이콘을 생성하여, 데이터베이스에 없는 아이콘도 LG 디啵 스타일로 빠르게 제공

### 3. LG webOS 생태계 확장 ❤️

외부 디자이너와 개발자의 참여를 이끌어내는 시작점이자, LG webOS 플랫폼을 성장시키는 생태계 확장의 첫걸음



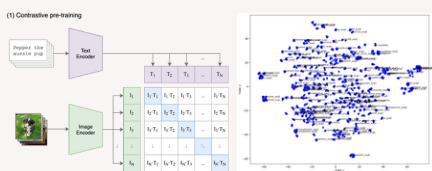
디啵 탐색 시간 절감으로 생산성 향상  
AI 생성 기능으로 아이디어 구체화 지원  
LG webOS 생태계 확장의 기반 마련

## 기술 소개

### 멀티모달 임베딩

#### 의미 기반 매칭 ⭐

CLIP으로 이미지와 텍스트를 동일한 벡터 공간에 임베딩하여, 서로 다른 형태 데이터를 의미 수준에서 직접 비교함. 별도의 추가 학습 없이도 다양한 도메인에 적용 가능한 범용성과 일반화 성능을 가짐



#### 사전 계산 임베딩 고속 검색 ⭐

모든 레퍼런스 이미지의 임베딩 벡터를 사전 계산하여 저장함으로써, 쿼리 입력 시 실시간 추론은 최소화되고, 코사인 유사도 계산만으로 빠른 검색이 가능함. 실제로 평균 검색 속도는 약 138MS(CPU 기준)로, 경량 시스템에서도 실시간에 준하는 사용자 경험을 제공

### Google Cloud Platform

#### 서비스 아키텍처 🚀

GCP의 서비스 아키텍처를 통하여, 사용자 급증시 0개에서 10개 서버로 자동확장, 서울(northeast-3) 리전 배치로 한국 사용자 대상 50ms 이하 초저지연 응답 보장, 월 100만 요청까지 안정적 처리 구현



#### 효율적인 개발/운영 🎵

Google Secret Manager로 API 키 보안 관리, Cloud Build - Cloud Run 의 흐름을 통한 편리한 배포 가능

### Flutter

#### Custom Painter 🎨

인터랙티브 캔버스를 구현. Flutter의 CustomPainter 위젯을 활용하여 사용자의 자유로운 드로잉을 부드럽게 처리, RepaintBoundary를 통해 드로잉 결과물을 캡처 및 크롭하여 정확하게 서버로 전송하도록 구성



#### 유지보수 가능한 코드 설계 🌲

프로젝트 전반에 걸쳐 상태 관리 라이브러리 Riverpod와 클린 아키텍처 패턴을 채택. Riverpod의 간결하고 선언적인 상태 관리를 통해 데이터 흐름을 명확히 하고, Domain, Data, Application, Presentation 레이어로 구분되는 클린 아키텍처를 적용하여 각 계층의 역할을 분리