

# Next.js Route Handler(API) 사양

## 개요

- 프로덕션 배포는 단일 Next.js 앱 내부의 Route Handler( `/api/generate` )를 사용합니다.
- Vercel 서버리스 함수로 실행되며 자동 확장/로그/보호가 적용됩니다.

## 파일 위치

- `apps/web/app/api/generate/route.ts`

## 엔드포인트

- `POST /api/generate`
- 요청 바디(JSON)

```
{
  "prompt": "텍스트 프롬프트",
  "images": [{ "slot": 1, "dataUrl": "data:image/jpeg;base64,..." }],
  "modelId": "gemini-2.5-flash-image-preview",
  "options": { "concurrency": 2 }
}
```

- 응답(JSON)

```
{
  "results": [ { "slot": 1, "image": "data:image/png;base64,..." }, ... ],
  "provider": "google-gemini"
}
```

## 동작 포인트

- 입력 검증: Zod 스키마로 프롬프트/이미지 구조 검증
- 이미지 처리: Data URL → `inlineData` 변환 후 모델 호출
- 모델: `@google/genai` SDK, 기본 모델은 `gemini-2.5-flash-image-preview`
- 재시도/동시성: 슬롯별 최대 3회 재시도, 기본 동시성 2
- 데모 모드: `DEMO_MODE=true` 이면 입력 이미지를 그대로 반환(키 없이 UI 테스트)

## 환경 변수

- `GEMINI_API_KEY` 또는 `GOOGLE_API_KEY`
- `GEMINI_MODEL_ID` (기본: `gemini-2.5-flash-image-preview`)
- `DEMO_MODE` (선택)

## 오류 처리

- 400: 유효성 오류(Zod)
- 500: 내부 오류/키 누락( `{ "error": "GEMINI_API_KEY missing" }` )

## 로그/디버깅

- Vercel → Project → Deployments → 해당 배포 → Functions Logs 에서 확인