ChatGPT

LotAnalyzer – Address-Based Lookup Implementation Spec

Version 0.9 (2025-06-10)

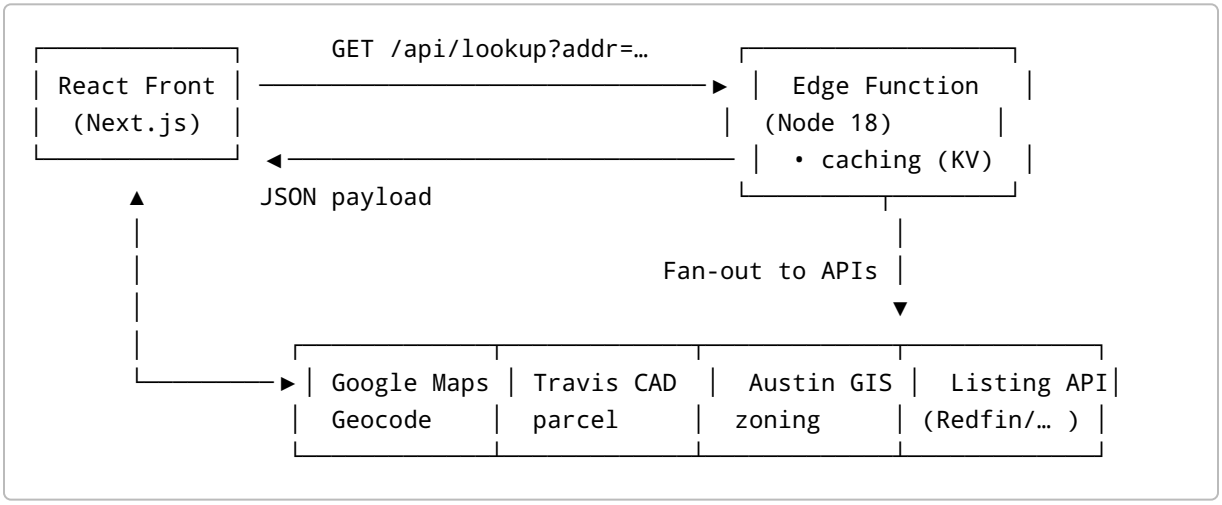
---

1 Project Goal

Convert LotAnalyzer from a bulk-CSV workflow to an **address (or Redfin/Zillow URL) search experience** that instantly tells us whether a given Austin property can be split and, if so, estimates the maximum buildable square footage for the new lot(s).

---

2 High-Level Architecture



---

3 External Services & Keys

Service	Purpose	Free-tier limits	Env Var
Google Places API	Autocomplete & geocode → lat/lng	40 000 req/mo	`GOOGLE_MAPS_KEY`
Travis CAD JSON	Parcel data (lot sq ft, legal lot, year built)	Unofficial, no key	n/a

Service	Purpose	Free-tier limits	Env Var
Austin AGOL GIS	Zoning layer + overlays	50 000 req/day	n/a
Redfin Scraper API (ZenRows)	Current listing meta (optional)	100 req/day	`ZENROWS_KEY`
KV / Redis	24 h caching of API responses	1 GB free	`KV_URL`

> **TODO:** Replace ZenRows with official MLS feed once licensed.

---

4 Backend (Edge Function) – Step-by-Step

1. **Input validation & normalization**
2. Accept `addr` (string) or `url` (Redfin/Zillow) query param.
3. Strip extra whitespace; lowercase URL host for matching.
4. **Parse property URL → address (if needed)**
5. Regex for redfin.com & zillow.com to pull the address slug.
6. **Geocode**
7. `GET https://maps.googleapis.com/maps/api/geocode/json?address={addr}`
8. Reject if precision < ROOFTOP.
9. **Parcel lookup**
10. `GET https://propertyapi.traviscad.org/property/{propId}`
    - Find `PROP_ID` by hitting Austin GIS `PropertyProfile` search with lat/lng.
11. Extract `lot_sqft`, `lot_width_ft`, `year_built`.
12. **Zoning lookup**
13. Hit Austin AGOL zoning layer (`FeatureServer/0/query`) with point geometry.
14. Return `ZONING_CLASS` (e.g., `SF-3-NP`).
15. **Split-eligibility logic**

```
const rules = zoning[zoneClass];
const canSplit = lotSqFt >= 2 * rules.min_lot_sqft &&
                 lotWidthFt >= 2 * rules.min_width_ft;
```

16. **Envelope & FAR calculations**
17. For each hypothetical half-lot: compute `max_buildable_sqft = FAR × lot_sqft`.
18. **Listing metadata (optional)**
19. If `url` provided or Redfin zpid discovered, call ZenRows → list price, DOM.
20. **Cache results**
21. Serialize JSON ↔ KV with key `addr:{sha256(address)}` exp=24 h.
22. **Respond**

```
{
  "address": "2814 Windsor Rd, Austin, TX 78703",
  "zone": "SF-3-NP",
```

```
  "lotSqFt": 12250,
  "canSplit": true,
  "halfLotMaxSqFt": 2756,
  "setbacks": {"front":25,"side":5,"rear":10},
  "listing": {"price": 1495000, "url":"…"}
}
```

---

5 Front-End Tasks

1. **Search Bar Component**
2. Google Places Autocomplete → writes selected address to state.
3. **/api/lookup Fetch Hook**
4. Handle loading / error; push result to context.
5. **Parcel Map Overlay**
6. Mapbox GL JS 0.7; draw polygon returned by GIS.
7. **Result Panel**
8. Verdict chips: ✓ *lot big enough*, ✓ *zone split-friendly*, ✗ *historic overlay*.
9. Collapsible section: Setback & FAR table.
10. **Save Candidate Button**
11. Persists to Supabase `favorites` table.

---

6 Local Dev Setup

```
pnpm i
cp .env.sample .env.local              # fill keys
npx supabase start                     # local Postgres
pnpm dev                               # next dev & functions
```

---

7 Acceptance Criteria (MVP)

-

---

8 Future Enhancements

- **Batch mode** – accept CSV of addresses, stream progress.
- **Comps module** – pull \$/sq ft comps to price new build.
- **Subdivision fee calculator** – estimate platting & utility costs.
- **National support** – swap GIS layer + zoning table per city.

---

9 Reference Links

- Austin GIS zoning FeatureServer: https://services7.arcgis.com/.../FeatureServer/0
- Travis CAD property API (unofficial): https://propertyapi.traviscad.org/property/123456
- Austin zoning code PDF (Table 3-1): internal `zoning_rules.json` build script.
- ZenRows Redfin doc: https://www.zenrows.com/documentation/redfin-api

---

Prepared by ChatGPT • 2025-06-10