Property Sunday Code Base:

```
// PropertySunday - Real Estate Listings Platform
// Next.js with TailwindCSS and MongoDB
import React, { useState, useEffect } from 'react';
import Head from 'next/head';
import Link from 'next/link';
import { useRouter } from 'next/router';

export default function Home() {
  const [listings, setListings] = useState([]);
  const [search, setSearch] = useState("");
  const [filters, setFilters] = useState({ minPrice: '', maxPrice: '', bedrooms: '' });
  const router = useRouter();

  useEffect(() => {
    fetch('/api/listings')
      .then(res => res.json())
      .then(data => setListings(data));
  }, []);

  const handleListingClick = (id) => {
    router.push(`/listing/${id}`);
  };

  const filteredListings = listings.filter(listing => {
    return (
      listing.title.toLowerCase().includes(search.toLowerCase()) &&
      (!filters.minPrice || listing.price >= filters.minPrice) &&
      (!filters.maxPrice || listing.price <= filters.maxPrice) &&
      (!filters.bedrooms || listing.bedrooms == filters.bedrooms)
    );
  });

  return (
    <div className="min-h-screen bg-gray-100 p-6">
      <Head>
        <title>PropertySunday</title>
      </Head>
      <header className="bg-white shadow-md p-4 mb-6 flex flex-col md:flex-row justify-between">
        <h1 className="text-4xl font-bold">PropertySunday Listings</h1>
        <div className="flex space-x-4 mt-4 md:mt-0">
          <input
            type="text"
            placeholder="Search properties..."
            className="border p-2 rounded"
            value={search}
            onChange={(e) => setSearch(e.target.value)}
          />
          <input
            type="number"
            placeholder="Min Price"
            className="border p-2 rounded"
            value={filters.minPrice}
            onChange={(e) => setFilters({ ...filters, minPrice: e.target.value })}
          />
          <input
            type="number"
            placeholder="Max Price"
```

```jsx
          className="border p-2 rounded"
          value={filters.maxPrice}
          onChange={(e) => setFilters({ ...filters, maxPrice: e.target.value })}
        />
        <input
          type="number"
          placeholder="Bedrooms"
          className="border p-2 rounded"
          value={filters.bedrooms}
          onChange={(e) => setFilters({ ...filters, bedrooms: e.target.value })}
        />
      </div>
    </header>
    <div className="container mx-auto mt--6">
      <div className="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-3 gap-6">
        {filteredListings.map(listing => (
          <div
            key={listing.id}
            className="bg-white p-4 rounded-lg shadow-md hover:shadow-lg transition cursor-pointer"
            onClick={() => handleListingClick(listing.id)}
          >
            <h2 className="text-2xl font-semibold">{listing.title}</h2>
            <p className="text-lg text-gray-600">Price: ${listing.price}</p>
            <p className="text-lg text-gray-600">Bedrooms: {listing.bedrooms}</p>
            <span className="text-blue-500 hover:underline mt-2 block">View Details</span>
          </div>
        ))}
      </div>
    </div>
  </div>
);
}
```

Here's a detailed plan to build **PropertySunday.com** (Zillow-style) and **OnShowMap.com** (Opendoor-style) tailored for South Africa, including style guides, API integrations, and a roadmap:

---

1. Style Guides
PropertySunday.com (Zillow-Inspired)
Color Palette:
- **Primary**: Terracotta Red (`#A84448`) – Reflects South African earth tones, warmth, trust.
- **Secondary**: Sand Beige (`#D7C9AA`) – Neutrality, simplicity.
- **Accent**: Olive Green (`#6B8E23`) – Growth, nature.

Typography:
- **Headings**: Playfair Display (elegant, authoritative).
- **Body**: Open Sans (clean, mobile-friendly).

UI Elements:
- Listing cards with hover effects and "Instant Bid" buttons.
- Search filters with sliders for price, bedrooms, and location.
- Mobile-first design with a "Save to Favorites" heart icon.

![PropertySunday Style Guide](https://via.placeholder.com/600x400.png/000000?text=PropertySunday+UI+Example)

OnShowMap.com (Opendoor-Inspired)
Color Palette:
- **Primary**: Deep Blue (`#2A3F54`) – Trust, professionalism.
- **Secondary**: Gold (`#FFD700`) – Luxury, urgency.
- **Accent**: Light Gray (`#F5F5F5`) – Modernity.

Typography:
- **Headings**: Montserrat (bold, tech-savvy).
- **Body**: Lato (readable, neutral).

UI Elements:
- Instant offer calculator with a sliding scale for property value.
- Interactive heatmaps for high-demand areas in Cape Town/Johannesburg.
- "Sell Now" CTA buttons with gold accents.

![OnShowMap Style Guide](https://via.placeholder.com/600x400.png/000000?text=OnShowMap+UI+Example)

---

2. API Integrations for South Africa
Core APIs
1. **Property Listings**:
   - **Property24 API**: Pull live listings (price, location, images).
   - **Private Property API**: Access 100k+ South African residential/commercial properties.
   - **Google Maps API**: Display properties with traffic/pricing layers.

2. **Valuation & Offers (OnShowMap)**:
   - **Prop Data API**: Automated property valuations using SA market data.
   - **Lightstone API**: Historical sales data for instant offers.

3. **Agent Integration**:
   - **RealNet**: Connect with 15k+ registered SA real estate agents.
   - **Lead Generation**: Zapier webhooks to sync agent CRM systems (e.g., Salesforce).

Sample API Code
```javascript
// Fetch Property24 Listings for PropertySunday.com
const fetchListings = async () => {
  const response = await fetch('https://api.property24.com/listings', {
    headers: {
      'Authorization': 'Bearer YOUR_API_KEY',
      'Content-Type': 'application/json'
    }
  });
  const data = await response.json();
  return data.listings;
};

// OnShowMap Instant Offer Calculator (Prop Data API)
const calculateOffer = async (propertyId) => {
  const response = await fetch(`https://api.propdata.co.za/valuations/${propertyId}`);
  const valuation = await response.json();
  return valuation.instantOffer * 0.95; // 5% Opendoor-style service fee
};
```

---

3. Roadmap for South African Market

Phase 1: MVP Launch (0–3 Months)
- **PropertySunday**:
  - Integrate Property24/Private Property APIs to display 10k+ live listings.
  - Launch in Cape Town with 50+ agents.
- **OnShowMap**:
  - Build instant offer tool using Prop Data API (Johannesburg pilot).
  - Partner with 10+ agencies for direct sales.

Phase 2: Growth (4–6 Months)
- Add **virtual tours** via SA-based platform Seeff Virtual Tours.
- Integrate **PayFast** for seamless transaction payments.
- Launch mobile apps with offline search for low-connectivity areas.

Phase 3: Scaling (7–12 Months)
- Expand to Durban and Pretoria.
- Add **AI chatbot** for Zulu/Xhosa/English support.
- Partner with SA Home Loans for mortgage integrations.

---

4. Local Market Validation
1. Property24
- **Why It Works**: 80% of SA buyers use Property24; 200k+ listings.
- **Gap**: PropertySunday's auction-style "Sunday Deals" create urgency missing in SA.

2. Seeff Properties
- **Why It Works**: 60+ years in SA; dominates luxury market.
- **Gap**: OnShowMap's instant offers compete with slow traditional sales.

3. PayProp
- **Why It Works**: 70% of SA rental agencies use PayProp for payments.
- **Gap**: PropertySunday can integrate rentals with bidding features.

---

5. Key Stats for SA Market
- **Online property searches** grew by 65% post-COVID (Lightstone).
- **Cape Town** has the highest demand (R1.8M avg home price).
- **60% of buyers** under 35 prefer mobile-first platforms (SA Real Estate Board).

---

6. Next Steps
1. Hire a **local dev team** familiar with SA APIs (Python/Django or Node.js).
2. Test MVP with **Gauteng-based agents** (highest transaction volume).
3. Use **Google Ads** to target keywords like "buy property Johannesburg" or "sell house fast SA."

Need help with wireframes or API documentation? Let's refine this! 🏘️🚀