

MX Mexico Relocation Guide - Replit Implementation Brief

Project Overview

A comprehensive, interactive relocation guide for Puerto Vallarta's Zona Romántica neighborhood, designed to help potential expats evaluate the area before moving.

Current HTML File

Filename: `pv-relocation-poc` (Puerto Vallarta Relocation Guide - Proof of Concept) **Type:** Standalone HTML file with embedded CSS and JavaScript **Status:** Functional prototype with simulated data

What We Built Together

6-Tab Virtual Tour System

1. Overview Tab

- Quick snapshot of the neighborhood
- Key metrics: Cost of Living, Safety, Walkability, Expat Community
- Monthly budget breakdown
- Pros & Cons list

2. Live Cams Tab **NEEDS API INTEGRATION**

- **Current Status:** Shows clickable preview cards for webcams
- **What Needs Enabling:**
 - Beach webcam feed (Los Muertos Beach)
 - Marina webcam feed
 - Malecón (boardwalk) webcam feed
- **APIs to Integrate:**
 - Webcam sources from Windy.com, EarthCam, or local PV tourism sites
 - YouTube live streams if available
 - RTSP camera feeds (if accessible)

3. Housing Tab

- Real rental prices with data sources
- Downtown vs Outskirts comparison
- Links to rental platforms (Inmuebles24, Vivanuncios, FB Marketplace)
- Sample listings

4. Safety Tab

- Crime statistics with sources
- Safe zones and caution areas
- Emergency resources
- Links to Semáforo Delictivo (crime tracking)

5. Building Reviews Tab NEEDS DATABASE

- User-submitted reviews of buildings and landlords
- Star ratings system
- **What Needs Enabling:**
 - Database backend (Firebase, Supabase, or PostgreSQL)
 - User authentication
 - Review submission form
 - Moderation system

6. Community Tab

- Expat Facebook groups
 - Local events calendar
 - Amenities (restaurants, gyms, markets)
 - Natural attractions (beaches, charcos)
-

APIs & Services That Need Integration

1. Live Webcam Feeds

Priority: HIGH **Current Status:** Static preview images with "View Live" buttons

Implementation Options:

Option A: Embedded iframes (Easiest)

```
html

<iframe
  src="https://www.youtube.com/embed/LIVE_STREAM_ID"
  allowfullscreen>
</iframe>
```

Source URLs to find:

- Los Muertos Beach cam
- Marina Vallarta cam
- Malecón boardwalk cam

Option B: API Integration

```
javascript

// Example: Windy Webcams API
fetch('https://api.windy.com/api/webcams/v2/list/nearby=20.6063,-105.2394,50')
  .then(response => response.json())
  .then(data => {
    // Display webcam feeds
  });
```

Windy Webcams API: <https://api.windy.com/webcams/docs>

Option C: Direct RTSP Streams (Advanced)

- Requires video.js or similar library
- May need CORS proxy
- Best quality but more complex

2. Weather API Integration

Priority: MEDIUM **Current Status:** Hardcoded climate data

Recommended API: OpenWeatherMap

javascript

```
const API_KEY = 'your_key_here';
const lat = 20.6063;
const lon = -105.2394;

fetch(`https://api.openweathermap.org/data/2.5/weather?lat=${lat}&lon=${lon}&appid=${API_KEY}`)
  .then(response => response.json())
  .then(data => {
    // Update weather display
  });
```

Sign up: <https://openweathermap.org/api>

3. Interactive Map Integration

Priority: HIGH **Current Status:** Not implemented

Recommended: Mapbox or Google Maps API

Mapbox Implementation:

html

```
<script src='https://api.mapbox.com/mapbox-gl-js/v2.15.0/mapbox-gl.js'></script>
<link href='https://api.mapbox.com/mapbox-gl-js/v2.15.0/mapbox-gl.css' rel='stylesheet' />

<script>
mapboxgl.accessToken = 'your_token_here';
const map = new mapboxgl.Map({
  container: 'map',
  style: 'mapbox://styles/mapbox/streets-v12',
  center: [-105.2394, 20.6063],
  zoom: 14
});

// Add markers for safe zones, amenities, etc.
new mapboxgl.Marker()
  .setLngLat([-105.2394, 20.6063])
  .setPopup(new mapboxgl.Popup().setHTML("<h3>Zona Romántica</h3>"))
  .addTo(map);
</script>
```

Sign up: <https://www.mapbox.com/>

4. Building Reviews Database

Priority: HIGH **Current Status:** Hardcoded sample reviews

Recommended Backend: Supabase (PostgreSQL + Auth + Storage)

Database Schema:

sql

```
CREATE TABLE building_reviews (  
  id UUID PRIMARY KEY DEFAULT uuid_generate_v4(),  
  building_name TEXT NOT NULL,  
  address TEXT,  
  landlord_name TEXT,  
  rating INTEGER CHECK (rating >= 1 AND rating <= 5),  
  review_text TEXT,  
  reviewer_name TEXT,  
  reviewer_email TEXT,  
  verified BOOLEAN DEFAULT false,  
  deposit_returned BOOLEAN,  
  maintenance_quality INTEGER,  
  communication_rating INTEGER,  
  created_at TIMESTAMP DEFAULT NOW()  
);
```

Supabase Setup:

1. Sign up at <https://supabase.com>
2. Create new project
3. Run the schema above in SQL Editor
4. Enable Row Level Security
5. Get your API keys

JavaScript Integration:

javascript

```

import { createClient } from '@supabase/supabase-js';

const supabase = createClient(
  'YOUR_PROJECT_URL',
  'YOUR_ANON_KEY'
);

// Submit review
async function submitReview(reviewData) {
  const { data, error } = await supabase
    .from('building_reviews')
    .insert([reviewData]);

  return { data, error };
}

// Fetch reviews
async function getReviews(buildingName) {
  const { data, error } = await supabase
    .from('building_reviews')
    .select('*')
    .eq('building_name', buildingName)
    .order('created_at', { ascending: false });

  return { data, error };
}

```

5. Real Estate Listings API (Optional)

Priority: MEDIUM **API:** Inmuebles24 or similar (may require scraping)

6. Safety Data API

Priority: LOW (Manual updates sufficient) **Source:** Semáforo Delictivo **Note:** No official API; consider manual updates or web scraping

File Structure for Replit

```

mexico-relocation-guide/
├── index.html      # Main HTML file (current pv-relocation-poc)
├── css/
│   └── styles.css  # External stylesheet

```

```
├── js/
│   ├── main.js      # Core functionality
│   ├── webcams.js   # Webcam integration
│   ├── maps.js      # Map functionality
│   ├── reviews.js   # Review system
│   └── config.js     # API keys (gitignore this!)
├── data/
│   └── neighborhoods.json # City/neighborhood data
├── .env              # Environment variables
├── .gitignore        # Keep API keys private
└── README.md         # Documentation
```

Environment Variables Needed (.env file)

```
env

# Webcams
WINDY_API_KEY=your_key_here

# Weather
OPENWEATHER_API_KEY=your_key_here

# Maps
MAPBOX_TOKEN=your_token_here

# Database
SUPABASE_URL=your_project_url
SUPABASE_ANON_KEY=your_anon_key

# Optional
GOOGLE_MAPS_API_KEY=your_key_here
```

Step-by-Step Implementation Guide

Phase 1: Setup (30 minutes)

1. Create new Replit project (HTML/CSS/JS)
2. Copy current HTML file as `index.html`
3. Sign up for required services:

- Windy Webcams API
- OpenWeatherMap
- Mapbox
- Supabase

Phase 2: Webcam Integration (1-2 hours)

1. Find live webcam URLs for Puerto Vallarta
2. Test iframe embeds
3. Add fallback for offline cams
4. Implement "View Live" button functionality

Phase 3: Map Integration (2-3 hours)

1. Add Mapbox GL JS library
2. Create map container in HTML
3. Add markers for key locations
4. Implement custom popups with info

Phase 4: Database Setup (2-3 hours)

1. Create Supabase project
2. Set up reviews table
3. Implement submission form
4. Add review display functionality
5. Enable basic moderation

Phase 5: Weather API (1 hour)

1. Replace hardcoded weather data
2. Add live temperature/conditions
3. Implement refresh functionality

Phase 6: Polish & Deploy (1-2 hours)

1. Mobile responsiveness testing

2. Error handling
 3. Loading states
 4. Deploy to Replit public URL
-

Testing Checklist

- ☐ All 6 tabs load without errors
 - ☐ Live webcams display properly
 - ☐ Map shows correct location with markers
 - ☐ Weather data updates from API
 - ☐ Review submission works
 - ☐ Reviews display correctly
 - ☐ Mobile responsive (test on phone)
 - ☐ All external links work
 - ☐ No console errors
 - ☐ Loading states show during API calls
-

Known Limitations & Future Enhancements

Current Limitations:

- No user authentication (reviews are anonymous)
- No photo uploads for reviews
- Limited to Puerto Vallarta only
- Weather updates require manual refresh

Future Enhancements:

- User accounts with Google/Facebook login
- Photo uploads for building reviews
- Multiple cities (Mérida, Playa del Carmen, etc.)
- Neighborhood comparison tool
- Cost of living calculator
- Expat forum integration

- Real-time chat with current residents
 - Video tours
 - 360° street view integration
-

Questions for Replit Agent

1. **Webcams:** "Can you help me integrate live webcam feeds from Windy API or find YouTube live streams of Puerto Vallarta beaches?"
 2. **Database:** "Can you set up a Supabase database for building reviews with user submission and display functionality?"
 3. **Maps:** "Can you add a Mapbox interactive map showing safe zones, amenities, and points of interest in Zona Romántica?"
 4. **API Keys:** "Can you help me set up environment variables to keep API keys secure?"
 5. **Mobile:** "Can you make sure all tabs work smoothly on mobile devices?"
-

Success Criteria

Minimum Viable Product (MVP):

- All 6 tabs functional
- At least 1 working live webcam
- Interactive map with key markers
- Review submission and display working
- Mobile responsive

Ideal Launch Product:

- 3+ live webcams with fallbacks
- Real-time weather integration
- Full review system with moderation
- Multiple neighborhoods
- Photo uploads
- Search functionality

Contact & Support

- **Original HTML File:** `pv-relocation-poc`
- **Project Type:** Single-page web application
- **Tech Stack:** HTML5, CSS3, Vanilla JavaScript
- **Target Audience:** English-speaking expats considering Puerto Vallarta relocation

Ready to build? Let's make this the ultimate Mexico relocation tool! 🌮🌊