Replit Requirements - Stand Details Page

I've provided prompts to supabase to create the necessary database updates. What information, if any, would you like to know from supabase in order to best integrate? Please provide it in prompt form so i can copy/paste both the answers and the questions.

Overview

Create a detailed stand page (`/stand/[id]`) that displays comprehensive information about a firewood stand with photos, verification system, and enhanced metadata.

Technical Requirements

Framework & Integration

- **Next.js App Router** with TypeScript
- **Dynamic Route**: `/stand/[id]` (use existing Supabase integration pattern)
- **Supabase Integration**: Use existing client from `lib/supabase.ts`
- **Styling**: Tailwind CSS with shadcn/ui components
- **Brand Colors**: Forest green (#2d5d2a), warm brown (#5e4b3a), cream (#f5f1e8)
- **Icons**: Lucide React icons

Page Structure & Layout

Stand Details Page Layout:
├── Header Section (Stand name, status badge, basic info, submitted by user on date, verified by user(s), last verified date)
├── Photo Gallery (carousel/grid toggle)
├── Details Section (organized in cards/sections)
├── Location Map (small embedded map)
├── Review/Ratings Section (community verification display, comments, stars)
├── Action Buttons (directions, verify, future: save/claim, Update)
└── Navigation (breadcrumbs, back to directory)

Data Requirements

Database Query

Fetch stand details with:

- All existing `firewood_stands` fields

- New fields: `bundled_flag`, `loose_flag`, `self_serve`, `payment_type`, `location_type`, `hours_availability`, `seasonal_availability`, `contact_phone`, `delivery_available`, `submitted_by_user_id`, `last_verified_date`, `photo_urls`
- Submitter name (from `profiles` table)
- Verification statistics (total verifications, unique users, recent verifiers)
- Current user's verification status (if logged in)

Error Handling

- Handle stand not found (404)
- Handle database connection errors
- Handle image loading failures
- Handle geolocation errors for map

Photo Gallery Component

Display Modes

- **Default**: Carousel view with navigation arrows
- **Grid Toggle**: Grid view showing all photos
- **Lightbox Modal**: Full-screen photo viewing on click

Features

- **Responsive Design**: Mobile-first approach
- **Loading States**: Skeleton placeholders while images load
- **Error Handling**: Fallback for broken images
- **Performance**: Lazy loading for images
- **Navigation**: Arrow keys, swipe gestures, thumbnail navigation

Photo Gallery Requirements

- Support 1-10 photos per stand
- Aspect ratio consistency (e.g., 16:9 or 4:3)
- Image optimization (automatic sizing)
- Alt text for accessibility

Stand Information Display

Header Section

- **Stand Name** (h1, brand color)
- **Status Badge** (approved: green, pending: brown)
- **Address** with "Get Directions" button
- **Distance** from user location (if available)

Display verification community data:

- **Format**: "X users have verified this stand Y times"
- **Recent Activity**: "Last verified on [date]"

- **Verifier Names**: Show verifiers with hover tooltip for full list
- **Verify Button**: "Verify This Stand" (for authenticated users)

Key Details Cards

Organize information into clean sections:

Wood Information:

- Wood types (styled tags)
- Format: Bundled/Loose flags with icons
- Price range (prominent display)

Service Details:

- Self-serve vs. Attended
- Payment type
- Payment methods (existing icons)
- Hours of availability
- Seasonal availability

Location Information:

- Location type (with appropriate icon)
- Contact phone (if available)
- Delivery available flag

Additional Information:

- Additional details (if provided)
- On-site person available

Interactive Map Component

Map Requirements

- **Size**: Moderate size (e.g., 300-400px height)
- **Center**: Stand location with custom pin
- **Zoom**: Close zoom level showing immediate area
- **Tile Layer**: OpenStreetMap
- **Marker**: Custom forest green pin matching brand
- **Popup**: Stand name and address

Map Features

- **Get Directions**: Button opens device default map app
- **Responsive**: Adjust size for mobile
- **Loading State**: Show while map initializes

Navigation & UX

Breadcrumbs

- Format: "Home > Directory > [Stand Name]"
- Remember previous directory filters/location if going back to directory
- Remember previous map view if going back to home page or map page (future page)

Back Navigation

- "← Back to Directory" button
- "← Back to Home" button
- Preserve user's previous search/filter state/map view

Action Buttons

- **Get Directions**: Opens device default map app with stand address
- **Verify This Stand**: Opens verification modal (authenticated users)
- **Save This Stand**: Placeholder button for future feature
- **This is My Stand**: Placeholder button for future feature (Receive Alerts)

User Interaction Features

Verification System

- **Verify Button**: Only show for authenticated users
- **Verification Modal**: Simple confirmation dialog
- **Prevent Duplicates**: Disable button if user verified in the last day
- **Success Feedback**: Show success message after verification
- **Real-time Updates**: Update verification count after submission

Authentication Integration

- **Logged Out**: Show login prompt for verification
- **Logged In**: Show user-specific actions (verify, save, claim)
- **User Context**: Use existing auth system

Responsive Design

Mobile Optimization

- **Photo Gallery**: Swipe gestures, touch-friendly navigation
- **Map**: Appropriate size for mobile screens
- **Buttons**: Touch-friendly sizing
- **Typography**: Readable text sizes
- **Layout**: Single column, stacked sections

Desktop Enhancement

- **Photo Gallery**: Keyboard navigation, hover effects
- **Layout**: Consider two-column layout for larger screens
- **Map**: Larger display area
- **Interactions**: Hover states, better spacing

Performance Requirements

Loading Optimization

- **Lazy Loading**: Images, map component
- **Skeleton States**: Show content structure while loading
- **Progressive Enhancement**: Core content first, enhanced features after

Image Optimization

- **Supabase Storage**: Automatic image transformations
- **Responsive Images**: Different sizes for different screen sizes
- **Caching**: Proper cache headers for images

SEO & Accessibility

SEO Optimization

- **Meta Tags**: Dynamic title, description based on stand data
- **Structured Data**: Local business schema markup
- **URLs**: Clean, descriptive URLs (`/stand/[id]`)

Accessibility

- **ARIA Labels**: Proper labeling for interactive elements
- **Keyboard Navigation**: Full keyboard accessibility
- **Screen Readers**: Proper heading hierarchy, alt text
- **Color Contrast**: Meet WCAG guidelines

Error States & Edge Cases

Handle Missing Data

- **No Photos**: Show placeholder or hide gallery
- **No Verifications**: Show "Be the first to verify" message
- **No Contact Info**: Hide contact section
- **No Additional Details**: Hide optional sections

Loading States

- **Page Loading**: Skeleton layout with placeholders
- **Image Loading**: Individual image placeholders
- **Map Loading**: Map container with loading indicator
- **Verification Loading**: Button loading state

Integration Points

Link Sources

- **Interactive Map**: Click stand name in popup
- **Directory**: Click stand name in directory cards

- **Future**: Search results, favorites, user dashboard

Data Consistency

- **Payment Icons**: Use existing implementation from directory
- **Status Badges**: Match directory styling
- **Brand Colors**: Consistent with rest of application

Future Considerations (Placeholders)

- **Reviews System**: Section for future reviews display
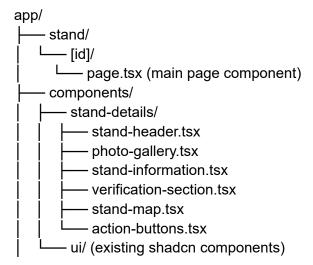- **Save Functionality**: Heart/bookmark button
- **Claim Ownership**: "This is my stand" functionality (primarily for future alerts)
- **Edit Stand**: For any authenticated users
- **Share Stand**: Social sharing buttons
- **Report Issues**: Report incorrect information

Component Structure

```
app/
├── stand/
│   └── [id]/
│       └── page.tsx (main page component)
├── components/
│   ├── stand-details/
│   │   ├── stand-header.tsx
│   │   ├── photo-gallery.tsx
│   │   ├── stand-information.tsx
│   │   ├── verification-section.tsx
│   │   ├── stand-map.tsx
│   │   └── action-buttons.tsx
│   └── ui/ (existing shadcn components)
```

Testing Requirements

- Test with existing test data (28 stands)
- Test responsive design on various screen sizes
- Test photo gallery with different numbers of photos
- Test verification system with authenticated/unauthenticated users
- Test error states (404, network errors, missing data)
- Test accessibility with screen readers
- Test performance with slow network connections