Windows Development Environment Setup for Saree Shopping App

Prerequisites Installation

1. Node.js & npm

- Download from: https://nodejs.org/en/download/
- Install LTS version (v18 or v20)
- Verify: `node --version` and `npm --version`

2. Git

- Download from: https://git-scm.com/download/win
- Install with default settings
- Verify: `git --version`

3. PostgreSQL

- Download from: https://www.postgresql.org/download/windows/
- Install PostgreSQL 15+
- Remember your password for 'postgres' user
- Default port: 5432
- Verify: Open pgAdmin or run `psql --version`

4. Redis (for Windows)

- Download from: https://github.com/microsoftarchive/redis/releases
- Install Redis-x64-3.0.504.msi
- Start Redis service
- Verify: `redis-cli ping` (should return PONG)

5. React Native Development Environment

Install React Native CLI

bash

```
npm install -g react-native-cli
npm install -g @react-native-community/cli
```

Android Development (for testing)

1. **Install Android Studio**
   - Download from: https://developer.android.com/studio
   - Install with default settings
   - Open Android Studio → More Actions → SDK Manager
   - Install Android SDK Platform 33 (Android 13)

2. **Set Environment Variables**
   - Add to Windows Environment Variables:

   ```
   ANDROID_HOME = C:\Users\[YourUsername]\AppData\Local\Android\Sdk
   ```

   - Add to PATH:

   ```
   %ANDROID_HOME%\platform-tools
   %ANDROID_HOME%\tools
   %ANDROID_HOME%\tools\bin
   ```

3. **Create Android Virtual Device (AVD)**
   - Open Android Studio → Tools → AVD Manager
   - Create Virtual Device → Choose Pixel 4 → Android 13 (API 33)

6. Code Editor

- **VS Code** (Recommended)
  - Download from: https://code.visualstudio.com/
  - Install essential extensions:
    - ES7+ React/Redux/React-Native snippets
    - Prettier - Code formatter
    - Auto Rename Tag
    - Bracket Pair Colorizer
    - Thunder Client (for API testing)

7. Additional Tools

bash

```bash
# Install global packages
npm install -g nodemon
npm install -g expo-cli
npm install -g yarn
```

Project Structure Setup

Create Main Project Directory

bash

```bash
mkdir saree-shopping-app
cd saree-shopping-app
```

Initialize Backend (Node.js + Express)

bash

```bash
mkdir backend
cd backend
npm init -y
npm install express cors dotenv bcryptjs jsonwebtoken
npm install pg redis multer cloudinary razorpay
npm install -D nodemon
```

Initialize Mobile App (React Native)

bash

```bash
cd ..
npx react-native init SareeShopApp
cd SareeShopApp
npm install @react-navigation/native @react-navigation/stack
npm install react-native-screens react-native-safe-area-context
npm install react-native-vector-icons react-native-image-picker
npm install @react-native-async-storage/async-storage
```

Initialize Admin Panel (React.js)

bash

```bash
cd ..
npx create-react-app admin-panel
cd admin-panel
npm install axios react-router-dom
npm install @mui/material @emotion/react @emotion/styled
```

Development Workflow

Daily Development Routine

1. **Start PostgreSQL** (usually auto-starts)

2. **Start Redis** (Windows service)

3. **Backend Development:**

   bash

   ```bash
   cd backend
   npm run dev  # (we'll set this up)
   ```

4. **Mobile App Testing:**

   bash

   ```bash
   cd SareeShopApp
   npx react-native run-android  # or run-ios
   ```

5. **Admin Panel:**

   bash

   ```bash
   cd admin-panel
   npm start
   ```

Environment Variables Template

Backend `.env` file:

```
# Database
DB_HOST=localhost
DB_PORT=5432
DB_NAME=saree_shop
DB_USER=postgres
DB_PASSWORD=your_password

# Redis
REDIS_HOST=localhost
REDIS_PORT=6379

# JWT
JWT_SECRET=your_jwt_secret_key

# Cloudinary
CLOUDINARY_CLOUD_NAME=your_cloud_name
CLOUDINARY_API_KEY=your_api_key
CLOUDINARY_API_SECRET=your_api_secret

# Razorpay
RAZORPAY_KEY_ID=your_razorpay_key
RAZORPAY_KEY_SECRET=your_razorpay_secret

# Server
PORT=3000
NODE_ENV=development
```

Verification Steps

After setup, verify everything works:

1. **Node.js**: `node --version` (should show v18+)

2. **PostgreSQL**: Open pgAdmin, connect to local server

3. **Redis**: `redis-cli ping` (should return PONG)

4. **React Native**: `npx react-native --version`

5. **Android**: Launch AVD from Android Studio

Next Steps

Once environment is ready:

1. ✅ Database schema design
2. ✅ Backend API structure
3. ✅ React Native navigation setup
4. ✅ Basic authentication flow

Troubleshooting Common Issues

PostgreSQL Connection Issues

- Ensure PostgreSQL service is running
- Check firewall settings
- Verify username/password

React Native Android Issues

- Ensure USB debugging is enabled
- Check AVD is running
- Clean and rebuild: `npx react-native run-android --reset-cache`

Port Conflicts

- Backend: 3000
- React Admin: 3001
- React Native Metro: 8081
- PostgreSQL: 5432
- Redis: 6379

Estimated Setup Time: 2-3 hours

Ready to proceed with database design once setup is complete!