SuperPOS Desktop App Build Guide

🖥️ Native Offline Desktop Application

Your SuperPOS system is already configured to work as a **fully offline desktop application** using Electron. This guide shows you how to build installers for Windows, macOS, and Linux.

✅ What's Already Configured

- ✅ Electron main process (`electron/main.js`)
- ✅ Electron preload script (`electron/preload.js`)
- ✅ Build configuration (`electron-builder.yml`)
- ✅ Offline capabilities (Service Worker + IndexedDB)
- ✅ Local database support (PostgreSQL)
- ✅ PWA manifest for browser installation

📦 Building Desktop Installers

Prerequisites

1. Node.js 16+ installed
2. Yarn package manager
3. Git (for version control)

Step 1: Update package.json Scripts

Add these scripts to your `package.json` (in the `"scripts"` section):

```
"electron:build": "next build && electron-builder",
"electron:build:win": "next build && electron-builder --win",
"electron:build:mac": "next build && electron-builder --mac",
"electron:build:linux": "next build && electron-builder --linux"
```

Step 2: Install electron-builder

```
cd nextjs_space
yarn add -D electron-builder
```

Step 3: Build for Your Platform

Windows (.exe installer)

```
yarn run electron:build:win
```

Outputs:
- `dist/SuperPOS Setup X.X.X.exe` - Windows installer
- Install on any Windows machine
- Works 100% offline

macOS (.dmg installer)

```
yarn run electron:build:mac
```

Outputs:
- `dist/SuperPOS-X.X.X.dmg` - macOS disk image
- Drag to Applications folder
- Works on Mac computers

Linux (AppImage, .deb)

```
yarn run electron:build:linux
```

Outputs:
- `dist/SuperPOS-X.X.X.AppImage` - Universal Linux app
- `dist/SuperPOS_X.X.X_amd64.deb` - Debian/Ubuntu package
- Works on any Linux distribution

🚀 Installation & Distribution

For Users

1. **Download the installer** for your platform:
   - Windows: `SuperPOS Setup.exe`
   - macOS: `SuperPOS.dmg`
   - Linux: `SuperPOS.AppImage` or `.deb`

2. **Run the installer**:
   - Windows: Double-click `.exe`, follow wizard
   - macOS: Open `.dmg`, drag to Applications
   - Linux: Make executable and run, or install `.deb`

3. **Configure database** (first launch):
   - Go to Settings → Database
   - Choose Local PostgreSQL
   - Enter connection details
   - Click “Test Connection”
   - Click “Initialize Schema”

4. **Start using offline!**
   - All data stored locally
   - Works without internet
   - Sync between stores when online

🔧 Advanced Configuration

Customize App Details

Edit `electron-builder.yml`:

```
appId: com.yourcompany.superpos
productName: Your Store POS
copyright: Copyright © 2025 Your Company
```

Change App Icons

1. Create icons:
   - Windows: `build/icon.ico` (256x256)
   - macOS: `build/icon.icns` (512x512)
   - Linux: `build/icon.png` (512x512)

2. Update `electron-builder.yml`:

```
win:
  icon: build/icon.ico
mac:
  icon: build/icon.icns
linux:
  icon: build/icon.png
```

Code Signing (for production)

For Windows:

```
win:
  certificateFile: path/to/certificate.pfx
  certificatePassword: "your-password"
```

For macOS:

```
mac:
  identity: "Developer ID Application: Your Name (TEAM_ID)"
```

💾 Offline Capabilities

Your app includes:

1. Local Database

- PostgreSQL on local machine
- All data stored locally
- No internet required

2. Offline Sync

- IndexedDB for transaction queue

- Background sync when online
- Conflict resolution

3. Service Worker

- Caches app files
- Works offline after first load
- Updates automatically

4. Install Prompt

- Browser install button
- Desktop shortcut
- Start menu entry

🔄 Multi-Store Sync

For multiple locations:

1. **Central Cloud Database**:
   - Deploy app to cloud (e.g., https://superpos.abacusai.app)
   - Configure cloud database connection
   - All stores sync to central database

2. **Local Store Setup**:
   - Install desktop app on each store
   - Configure local PostgreSQL
   - Connect to cloud for sync

3. **Sync Process**:
   - Changes tracked locally
   - Push to cloud when online
   - Pull updates from other stores
   - Conflict resolution (latest wins)

📋 Distribution Options

Option 1: Direct Distribution

- Build installers for each platform
- Upload to file sharing service
- Users download and install
- Manual updates

Option 2: Auto-Update (Recommended)

Add to `electron-builder.yml`:

```
publish:
  provider: github
  owner: YourUsername
  repo: supermarket-pos-system
```

Users get automatic updates!

Option 3: Microsoft Store / Mac App Store

- Submit to app stores
- Automatic distribution
- Easier updates
- Requires developer account

🐛 Troubleshooting

Build Fails on Windows

```
# Install Windows Build Tools
npm install --global windows-build-tools

# Or use Visual Studio
# Install "Desktop development with C++" workload
```

Build Fails on macOS

```
# Install Xcode Command Line Tools
xcode-select --install
```

Build Fails on Linux

```
# Install required packages
sudo apt-get install -y rpm fakeroot dpkg
```

App Won’t Start

- Check database connection
- Verify PostgreSQL is running
- Check Settings → Database
- Review error logs

📊 File Sizes

Typical installer sizes:
- Windows: ~150-200 MB
- macOS: ~150-200 MB
- Linux: ~150-200 MB

Includes:
- Chromium engine
- Node.js runtime
- Your app code
- Dependencies

✅ Summary

1. ✅ **Build installers** using `yarn electron:build:win`
2. ✅ **Distribute** to users (file sharing or auto-update)
3. ✅ **Install** on any machine
4. ✅ **Configure** local database
5. ✅ **Use offline** - no internet needed!
6. ✅ **Sync** between stores when online

Your SuperPOS app is **production-ready** for offline deployment! 🚀