Deployment Guide: ATS Application on Enhance Control Panel

This guide details the deployment process for the ATS Application (React + Node.js) on an existing Enhance Control Panel website. It assumes the Enhance panel is installed and the website entry has already been created.

1. Prerequisites & Access Checklist

Before starting, ensure you have the following from your Enhance Control Panel:

- **Website User**: The system user assigned to your website (e.g., `ats_user`).
- **Server IP**: The IP address of the App Server hosting the website.
- **SSH Access**: Ensure "Shell Access" is enabled for this website user in the Enhance Panel.
- **Database Access**: Ability to create databases and users via Enhance or phpMyAdmin.

---

2. Database Configuration

Even if the website is created, the database needs to be set up for the application.

1. **Create Database**:
    - Log in to Enhance Panel > **Websites** > Select your website.
    - Go to **Databases** > **Add Database**.
    - **Name**: `ats_DB` (recommended).
    - **User**: Create a new database user (e.g., `ats_db_user`) and generate a strong password. **Save these credentials**.
2. **Import Schema**:
    - Click **phpMyAdmin** in the database list.
    - Select your new database (`ats_DB`) from the sidebar.
    - Go to the **Import** tab.
    - Upload the `server/db/schema.sql` file from your local project.
    - *Note: If you have migration scripts (`server/migrations/*.sql`), import them in order or run the migration script later.*

---

3. Application Deployment

Step 3.1: File Upload (SFTP)

Upload the application code to the server.

1. **Connect via SFTP**:
    - **Host**: Server IP
    - **User**: Your Website User (e.g., `ats_user`)
    - **Password**: The Website User's password
    - **Port**: 22
2. **Navigate & Clean**:
    - Go to `public_html/` (or your document root).
    - Delete the default `index.html` created by Enhance.
3. **Upload Files**:
    - Upload the **entire project folder contents** into `public_html/`.
    - **Exclude**: `node_modules` (we will install these on the server).
    - **Critical Files to Include**:
        - `server/` directory
        - `src/` directory
        - `public/` directory
        - `package.json`
        - `ecosystem.config.js`

Step 3.2: Node.js Environment Setup (SSH)

Enhance website containers are isolated. You need to set up the Node.js environment specifically for this user.

1. **SSH into the Website**:

   ```
   ssh website_user@your_server_ip
   ```

2. **Install Node.js (via NVM)**:
   Since this is a fresh user environment, install NVM to manage Node.js:

   ```
   curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.7/install.sh | bash
   export NVM_DIR="$HOME/.nvm"
   [ -s "$NVM_DIR/nvm.sh" ] && \. "$NVM_DIR/nvm.sh"

   nvm install 18
   nvm use 18
   nvm alias default 18
   ```

3. **Verify Installation**:

   ```
   node -v
   npm -v
   ```

Step 3.3: Install Dependencies & Build

1. **Navigate to Document Root**:

   ```
   cd public_html
   ```

2. **Install Dependencies**:

   ```
   npm install
   ```

3. **Build React Frontend**:

   ```
   npm run build
   ```

   This compiles the React app into the `build/` directory.

Step 3.4: Configuration (.env)

1. **Create Environment File**:

   ```
   nano .env
   ```

2. **Paste Configuration**:
   Update the values to match your created database credentials.

   ```
   NODE_ENV=production
   PORT=5000
   DB_HOST=127.0.0.1
   DB_USER=ats_db_user
   DB_PASSWORD=your_db_password
   DB_NAME=ats_DB
   JWT_SECRET=your_secure_random_string
   ```

   Note: Use `127.0.0.1` for DB_HOST to ensure it uses the local TCP connection or check if Enhance requires a specific socket path.

3. **Save & Exit**: Press `Ctrl+O`, `Enter`, `Ctrl+X`.

Step 3.5: Start Backend with PM2

Use PM2 to keep the Node.js server running in the background.

1. **Install PM2**:

   ```
   npm install -g pm2
   ```

2. **Start Application**:

   ```
   pm2 start ecosystem.config.js --env production
   ```

3. **Save Process List**:
   Ensure the app restarts if the container restarts.

```
pm2 save
pm2 startup
```

(Run the command output by `pm2 startup` if instructed).

---

4. Web Server Configuration (Reverse Proxy)

Since Enhance serves the website on Port 80/443 via Apache/OpenLiteSpeed, we must proxy traffic to the Node.js app running on Port 5000.

1. **Edit .htaccess**:
   In the `public_html` directory:

   ```
   nano .htaccess
   ```

2. **Add Proxy Rules**:
   Replace existing content with:

   ```
   DirectoryIndex disabled
   RewriteEngine On

   # Proxy all requests to Node.js app on port 5000
   RewriteRule ^$ http://127.0.0.1:5000/ [P,L]
   RewriteCond %{REQUEST_FILENAME} !-f
   RewriteCond %{REQUEST_FILENAME} !-d
   RewriteRule ^(.*)$ http://127.0.0.1:5000/$1 [P,L]
   ```

---

5. Verification & Post-Deployment

5.1 Verify Deployment

1. **Browser Test**: Open `https://affinitytaxservices.com`.
   - You should see the ATS Application home page.
2. **API Test**: Open `https://affinitytaxservices.com/health`.
   - Expected JSON response: `{"status":"ok", ...}`.

5.2 Troubleshooting Common Issues

- **503 Service Unavailable / 500 Error**:
  - **Check Node App**: Run `pm2 status`. If stopped, check logs: `pm2 logs ats-backend`.
  - **Check .htaccess**: Ensure the syntax is correct and `mod_proxy` is enabled on the server.
  - **Check Port**: Ensure `.env` has `PORT=5000` and the app is actually listening on it (`netstat -tulpn | grep 5000`).
- **Database Connection Refused**:
  - Verify `DB_HOST`, `DB_USER`, and `DB_PASSWORD` in `.env`.
  - Try `localhost` instead of `127.0.0.1` if using a socket, or vice versa.

5.3 Maintenance Commands

Run these via SSH as the website user:

Action	Command
View Logs	`pm2 logs`
Restart App	`pm2 restart ats-backend`
Stop App	`pm2 stop ats-backend`
Update App	Upload files > `npm install` > `npm run build` > `pm2 restart ats-backend`