Vercel Deployment Configuration Fix

Issue Summary

The Vercel deployment was failing due to the Root Directory setting being incorrectly set to `/app` in the Vercel project settings, causing path conflicts with Next.js output and preventing the generation of `routes-manifest.json`.

Solution Implemented

1. Updated vercel.json Configuration

COMPLETED: Updated `/vercel.json` to remove hardcoded `/app` path references:

Before:

```
{
  "functions": {
    "app/api/**/*.ts": {
      "runtime": "nodejs18.x"
    }
  }
}
```

After:

```
{
  "functions": {
    "api/**/*.ts": {
      "runtime": "nodejs18.x"
    }
  }
}
```

2. Verified Build Configuration

TESTED: Confirmed that the build process works correctly from the root directory:
- Next.js recognizes the correct project structure
- Build process initiates properly with `npx next build`
- API routes are correctly mapped without `/app` prefix

Manual Steps Required (Sam to Complete)

Step 1: Update Vercel Project Settings

1. Log into Vercel dashboard
2. Navigate to the mySafePlay project
3. Go to **Settings → General**
4. Find **"Root Directory"** setting
5. **REMOVE** the `/app` value (clear the field completely)

6. **SAVE** the changes

Step 2: Add Missing Build Scripts to package.json

The root `package.json` is missing essential build scripts. Add these to the "scripts" section:

```json
{
  "scripts": {
    "dev": "next dev -p 3000",
    "build": "next build",
    "start": "next start",
    "lint": "next lint"
  }
}
```

Step 3: Trigger New Deployment

After updating the Vercel settings:

1. Push any small change to trigger a new deployment, OR
2. Use the "Redeploy" button in Vercel dashboard

Expected Results

- Vercel will use the project root as the build directory
- `routes-manifest.json` should generate correctly
- API routes will be properly mapped without path conflicts
- Deployment should complete successfully

Project Structure Verification

The current project structure is correct for Next.js 13+ App Router:

```
/home/ubuntu/safeplay-staging/        # ↵ Root directory (Vercel should use this)
    app/                              # ↵ Next.js App Router directory
    components/                       # ↵ React components
    lib/                              # ↵ Utility libraries
    package.json                      # ↵ Build scripts and dependencies
    next.config.js                    # ↵ Next.js configuration
    vercel.json                       # ↵ Updated Vercel configuration
    ...
```

Verification Steps

After deployment:

1. Check that the build logs show proper API route detection
2. Verify that `routes-manifest.json` is generated in the build output
3. Test that API endpoints respond correctly
4. Confirm the application loads without path-related errors

Additional Notes

- The `/app` directory setting was causing Vercel to `cd app` before building

- This created conflicts with Next.js's expected project structure
- All configuration files are now properly aligned for root-level deployment

---

Status: Configuration files updated
Next Action: Sam needs to update Vercel project settings manually
Priority: High - Required to resolve deployment failure