

Fix "Not Found" Error After OAuth Implementation

Tell the Replit AI Agent:

"The OAuth fix is implemented but now I'm getting 'Not Found' errors on homepage and dashboard. Please check and fix these issues:"

1. Check Route Files Exist

Make sure these files exist in your project:

```
app/  
├── page.tsx (or page.js) - Homepage  
├── dashboard/  
│   └── page.tsx - Dashboard page  
└── auth/  
    ├── callback/  
    └── route.ts - Auth callback
```

2. Fix Homepage Route

Check `app/page.tsx`:

```
typescript  
  
// app/page.tsx  
export default function HomePage() {  
  return (  
    <div>  
      <h1>Welcome to TitleTesterPro</h1>  
      { /* Your homepage content */ }  
    </div>  
  )  
}
```

3. Fix Dashboard Route

Check `app/dashboard/page.tsx`:

typescript

```
// app/dashboard/page.tsx
import { redirect } from 'next/navigation'
import { createServerComponentClient } from '@supabase/auth-helpers-nextjs'
import { cookies } from 'next/headers'

export default async function DashboardPage() {
  const supabase = createServerComponentClient({ cookies })
  const { data: { session } } = await supabase.auth.getSession()

  // Redirect if not authenticated
  if (!session) {
    redirect('/login')
  }

  return (
    <div>
      <h1>Dashboard</h1>
      { /* Your dashboard content */ }
    </div>
  )
}
```

4. Check Middleware Issues

If you have `middleware.ts`, it might be blocking routes:

typescript

```
// middleware.ts
import { createMiddlewareClient } from '@supabase/auth-helpers-nextjs'
import { NextResponse } from 'next/server'
import type { NextRequest } from 'next/server'

export async function middleware(req: NextRequest) {
  const res = NextResponse.next()
  const supabase = createMiddlewareClient({ req, res })

  await supabase.auth.getSession()
  return res
}

// Make sure matcher doesn't block your routes
export const config = {
  matcher: [
    '/(?!_next/static|_next/image|favicon.ico|public|api).*',
  ]
}
```

5. Check for Redirect Loops

In your auth callback, make sure redirects are correct:

typescript

```
// app/auth/callback/route.ts
export async function GET(request: Request) {
  const requestUrl = new URL(request.url)
  const code = requestUrl.searchParams.get('code')

  if (code) {
    // ... auth code ...

    // Make sure these paths exist:
    return NextResponse.redirect(new URL('/dashboard', requestUrl.origin))
  }

  // Fallback redirect
  return NextResponse.redirect(new URL('/', requestUrl.origin))
}
```

6. Quick Debug Steps

Add this temporary debug page:

```
typescript
```

```
// app/debug/page.tsx
```

```
import { headers } from 'next/headers'
```

```
export default function DebugPage() {
```

```
  const headersList = headers()
```

```
  const host = headersList.get('host')
```

```
  return (
```

```
    <div>
```

```
      <h1>Debug Info</h1>
```

```
      <p>Host: {host}</p>
```

```
      <p>If you see this, routing works!</p>
```

```
      <a href="/">Go to Homepage</a>
```

```
      <br />
```

```
      <a href="/dashboard">Go to Dashboard</a>
```

```
    </div>
```

```
  )
```

```
}
```

Then try visiting `/debug` to see if any routes work.

7. Common Fixes

1. Clear Next.js cache:

```
bash
```

```
rm -rf .next
```

```
npm run dev
```

2. Check for typos in file names (case-sensitive on deployment)

3. Ensure all imports are correct after agent changes

4. Check console for errors - might show what's failing

If Still Broken:

Tell the agent: "Routes are returning 404. Please check:

1. All page.tsx files exist in correct folders

2. No middleware is blocking routes
3. All imports are valid after changes
4. No redirect loops in auth flow"