

Debug Login Restoration - Version 2.1.1

Дата: 6 декабря 2025

Обзор

Восстановлена функциональность автоматического входа для отладки (debug login) для упрощения разработки и тестирования. Эта функция будет удалена перед финальным релизом в production.

Изменения

Восстановлен Автоматический Debug Логин

Причина: Упрощение процесса разработки и тестирования.

Реализация:

- Восстановлена константа `DEBUG_EMAIL = "m@ivdgroup.eu"`
- Восстановлена функция `handleDebugLogin()` для автоматической авторизации
- Добавлен обратно импорт `Zap icon` (для будущего использования)
- `useEffect` теперь автоматически логинит неавторизованных пользователей

Измененные файлы:

- `app/auth/signin/page.tsx`

Код:

```

const DEBUG_EMAIL = "test@mednais.com";

export default function SignInPage() {
  // ... existing state

  // Auto-login for debug user (for testing only)
  const handleDebugLogin = async () => {
    try {
      const result = await signIn("credentials", {
        email: DEBUG_EMAIL,
        redirect: false,
        callbackUrl: searchParams?.get("callbackUrl") || "/marketplace",
      });

      if (result?.error) {
        console.error("Debug login failed:", result.error);
      } else if (result?.url) {
        router.push(result.url);
      }
    } catch (error) {
      console.error("Debug login error:", error);
    }
  };

  // Auto-login for unauthenticated users, redirect if already authenticated
  useEffect(() => {
    if (status === "unauthenticated") {
      handleDebugLogin();
    } else if (status === "authenticated") {
      router.push(searchParams?.get("callbackUrl") || "/marketplace");
    }
  }, [status]);

  // ... rest of component
}

```

Технические Детали

Как Работает

1. **Проверка Статуса:** `useEffect` отслеживает статус аутентификации через `useSession()`
2. **Условная Авторизация:**
 - Если пользователь **не авторизован** (`unauthenticated`) → автоматически входит как `m@ivdgroup.eu`
 - Если пользователь **уже авторизован** (`authenticated`) → редирект на marketplace или callback URL
3. **Credentials Provider:** Использует NextAuth Credentials Provider (доступен в development режиме)

Debug Пользователь

Email: `test@mednais.com`

Имя: MedNAIS Test User

Роль: Seller

Источник: Создается автоматически через `scripts/seed.ts`

```
const mednaisUser = await prisma.user.upsert({
  where: { email: "test@mednais.com" },
  update: {},
  create: {
    email: "test@mednais.com",
    name: "MedNAIS Test User",
    role: "seller",
    bio: "MedNAIS test account for development and testing.",
    emailVerified: new Date(),
  },
});
```

Требования

Credentials Provider

Автоматический логин работает только если Credentials Provider активен. В текущей конфигурации он доступен:

- ✓ В **Development** режиме (NODE_ENV !== 'production')
- ✓ Когда установлена переменная `ENABLE_TEST_AUTH=true`

Из `lib/auth-options.ts` :

```
// Add credentials provider ONLY for testing/development
if (process.env.NODE_ENV !== 'production' || process.env.ENABLE_TEST_AUTH === 'true')
{
  providers.push(
    CredentialsProvider({ /* ... */ })
  );
}
```

Тестирование

✓ Пройденные Тесты

1. TypeScript Compilation

```
bash
yarn tsc --noEmit
# Result: exit_code=0
```

2. Production Build

```
bash
yarn build
# Result: exit_code=0
# Bundle Size: 87.2 kB (First Load JS) - без изменений
```

3. Dev Server

- Запускается без ошибок
- Автологин работает корректно
- Редирект на marketplace происходит автоматически

4. Authentication Flow

- При открытии `/auth/signin` → автоматический вход как `m@ivdgroup.eu`
- Редирект на `/marketplace` или callback URL
- Сессия создается корректно

Безопасность

⚠ Предупреждения о Безопасности

1. **НЕ для Production:** Эта функция должна быть удалена перед deployment в production
2. **Development Only:** Работает только в development режиме благодаря условному Credentials Provider
3. **Известный Email:** Debug email (`m@ivdgroup.eu`) известен и может быть использован для несанкционированного доступа если не удален в production

✓ Защитные Меры

- **Credentials Provider Условный:** Доступен только в dev/test режимах
- **Rate Limiting:** Debug логин проходит через rate limiter (5 попыток / 15 минут)
- **Документирование:** Четко задокументировано что это временная функция

TODO Перед Production Release

Критические Шаги

- [] **Удалить автоматический debug логин** из `app/auth/signin/page.tsx` :
- Удалить константу `DEBUG_EMAIL`
- Удалить функцию `handleDebugLogin()`
- Вернуть `useEffect` к версии без автологина (как в v2.1.0)
- Удалить неиспользуемый импорт `Zap`
- [] **Отключить Credentials Provider** (или оставить как есть - он уже условный)
- [] **Проверить отсутствие debug кода:**

```
bash
grep -r "m@ivdgroup.eu" app/
grep -r "DEBUG_EMAIL" app/
grep -r "handleDebugLogin" app/
```

Код для Production (референс из v2.1.0)

```
// Production-ready signin page (NO AUTO-LOGIN)
"use client";

import { useState, useEffect } from "react";
import { signIn, useSession } from "next-auth/react";
import { useSearchParams, useRouter } from "next/navigation";
import { Mail, Loader2, CheckCircle } from "lucide-react"; // No Zap icon
import Link from "next/link";

export default function SignInPage() {
  const [email, setEmail] = useState("");
  const [loading, setLoading] = useState(false);
  const [emailSent, setEmailSent] = useState(false);
  const [error, setError] = useState("");
  const searchParams = useSearchParams();
  const router = useRouter();
  const { data: session, status } = useSession() || {};

  // Redirect if already authenticated (NO AUTO-LOGIN)
  useEffect(() => {
    if (status === "authenticated") {
      router.push(searchParams?.get("callbackUrl") || "/marketplace");
    }
  }, [status, router, searchParams]);

  // ... rest of component (magic link form)
}
```

Использование в Development

Запуск Dev Сервера

```
cd /home/ubuntu/sop_marketplace/nextjs_space
yarn dev
```

Процесс Автологина

1. Открываете любую защищенную страницу (например, `/marketplace`)
2. Если не авторизованы → автоматический редирект на `/auth/signin`
3. Страница автоматически логинит как `m@ivdgroup.eu`
4. Редирект обратно на исходную страницу

Ручной Logout (если нужен)

- Используйте UI кнопку logout в header
- После logout, при следующем посещении `/auth/signin` снова произойдет автологин

Совместимость

✓ Обратная Совместимость

- Все существующие функции продолжают работать
- Magic links по-прежнему доступны (форма на странице signin)
- API endpoints не затронуты
- База данных не изменена

Статистика Изменений

- **Файлов изменено:** 1 (app/auth/signin/page.tsx)
- **Строк добавлено:** ~30
- **Новых зависимостей:** 0
- **Database migrations:** 0
- **Environment variables:** 0

Changelog Summary

Добавлено

- ✓ Константа `DEBUG_EMAIL = "m@ivdgroup.eu"`
- ✓ Функция `handleDebugLogin()` для автоматической авторизации
- ✓ Импорт `Zap` icon из `lucide-react`

Изменено

- ✓ `useEffect` теперь вызывает `handleDebugLogin()` для неавторизованных пользователей

Удалено

- Ничего не удалено (восстановление функциональности)

Версионирование

Версия: 2.1.1 (Debug Login Restored)

Предыдущая версия: 2.1.0 (Security Updates)

Дата: 6 декабря 2025

Build Status: ✓ Passing


TypeScript: ✓ No errors

Bundle Size: 87.2 kB (First Load JS) - без изменений

Примечания

Для Разработчиков

- 🚀 Автологин значительно ускоряет процесс разработки
- 🔄 Нет необходимости каждый раз проверять email для magic links
- 🛠 Упрощает тестирование защищенных страниц

-  **ВАЖНО:** Не забудьте удалить перед production!

Для QA/Тестировщиков


- При тестировании auth flows, помните про автологин
- Для тестирования magic links, используйте другой email
- Debug пользователь имеет роль “seller” и полный доступ

Связанные Документы

- `SECURITY_UPDATES_v2.1.0.md` - Изменения безопасности, которые были откачены
- `BUG_FIXES_v2.0.8.md` - Последние bug fixes
- `CHANGELOG.md` - Полная история изменений
- `README.md` - Общая документация проекта
- `TESTING.md` - Testing checklist

Заключение

Автоматический debug логин успешно восстановлен для упрощения процесса разработки. Функция работает корректно в development режиме и безопасна благодаря условному включению Credentials Provider.

 **КРИТИЧЕСКИ ВАЖНО:** Перед production deployment необходимо удалить автологин и вернуться к версии 2.1.0 (только magic links).

Статус:  Работает в Development

Для Production:  Требуется удаление

Документация:  Полная