

# Multi-Brand SaaS Platform - Мастер документации

**Создать 4 идентичные SaaS-платформы с единым backend и уникальным брандингом:**

 **Nimbus** (Blue) - базовая версия для отладки

 **Velocity** (Green) - фокус на скорость

 **Aurora** (Purple) - фокус на дизайн

 **Zenith** (Orange) - фокус на масштаб

**Подход:** Начинаем с Nimbus, доводим до совершенства, затем реплицируем на остальные 3.

---

## ▼ 1. Процесс работы

### Этап 1: Nimbus - базовая версия

#### ▼ 1.1 Оценка проекта

Что проверяем:

- Текущее состояние кода и структура файлов
- Наличие всех необходимых страниц
- Функционирующие компоненты
- Конфигурация ESLint product quality
- Статус деплоя на Vercel

**Локация:** `/Users/rentamac/Documents/repos/repos/saas`

**GitHub:** <https://github.com/souldevsoul/accelerator-saas>

#### ▼ 1.2 ESLint Product Quality Setup

**Правила для Nimbus:**

Allowed colors: blue-\*<sup>\*</sup>, sky-\*<sup>\*</sup>, cyan-\*<sup>\*</sup>, gray-\*<sup>\*</sup>, slate-\*<sup>\*</sup>

Company: "Nimbus"

Email: "support@nimbusdev.com"

Payment: "stripe"

### Команды проверки:

```
cd /Users/rentamac/Documents/repos/repos/saas  
npx eslint --config eslint.config.product.mjs "**/*.{ts,tsx}"
```

### Что исправляем автоматически:

- Цвета не из палитры → blue/sky/cyan
- Некорректные email → support@nimbusdev.com
- Кнопки без type → добавить type="button"
- Битые ссылки → создать страницы

## ▼ 1.3 Маркетинговые страницы - ПЕРВЫЙ ПРИОРИТЕТ

### Обязательные страницы (проверяем каждую):

#### / (Homepage)

- Hero с ясным value proposition
- Демо AI-генерации (скриншот или видео)
- "Как это работает" - 3 простых шага
- Pricing превью
- FAQ секция
- Final CTA
- НЕТ фейковых отзывов
- НЕТ фейковых метрик ("10,000+ users")

#### /about

- О платформе и миссии

- Команда (реальная или честно указать "indie maker")
- Контакты: [support@nimbusdev.com](mailto:support@nimbusdev.com)
- НЕТ фейковых сертификатов

## /pricing

- 3 тарифа: Free (\$0), Pro (\$29), Business (\$99)
- Сравнение функций в таблице
- Честные цены без манипуляций
- FAQ по ценам
- CTA для каждого плана

## /how-it-works

- Пошаговое объяснение процесса
- Скриншоты реального продукта (не stock)
- Примеры проектов
- CTA внизу

## /contact

- Форма связи (работающая)
- Email: [support@nimbusdev.com](mailto:support@nimbusdev.com)
- Время ответа

## /experts

- Marketplace разработчиков
- Поиск по навыкам
- CTA "Become an expert"

## Политики (обязательные):

-  /privacy - заполнена, не template
-  /terms - заполнена, не template
-  /cookies - описывает реальное использование

-  /acceptable-use - ясные правила

## ▼ 1.4 Чеклист качества маркетинга

### КРИТИЧНО - Честность контента:

- НЕТ фейковых отзывов - только реальные или без них
- НЕТ фейковых цифр - "10,000+ users" только если правда
- НЕТ stock фото - уникальные скриншоты продукта
- Все email корректные - [support@nimbusdev.com](mailto:support@nimbusdev.com)
- Нет битых ссылок - каждая ведет куда надо

### Стилистика:

- Все цвета из палитры - только blue/sky/cyan
- Шрифты загружаются - проверить в DevTools
- Иконки уникальные - не generic stock
- Градиенты консистентные - from-sky-600 to-blue-700
- Тени правильные - голубой оттенок

### Функциональность:

- Все кнопки кликабельные
- Формы работают
- Navigation responsive
- Footer полный с ссылками

## ▼ 1.5 Деплоймент на Vercel

### Процесс:

```
cd /Users/rentamac/Documents/repos/repos/saas  
pnpm run build # Проверка локально  
vercel --prod # Deploy
```

### Проверка после деплоя:

- Сайт открывается по production URL
- Homepage загружается < 3 сек
- Нет ошибок в console (F12)
- Все страницы доступны
- Стили применяются корректно
- Изображения загружаются
- Шрифты видны

## ▼ 1.6 Dashboard и user flows - ВТОРОЙ ПРИОРИТЕТ

После того как маркетинг готов:

### /login

- Email + Password форма
- OAuth кнопки (Google, GitHub)
- "Forgot password" ссылка
- Валидация формы

### /dashboard

- Список проектов пользователя
- Баланс кредитов
- Кнопка "New Project"
- Статистика

### /dashboard/projects/new

- Поле описания проекта
- Выбор AI модели
- Кнопка "Generate"
- Индикатор баланса кредитов

### /dashboard/projects/[id]

- Код генерации

- Live preview в iframe
- Список файлов
- Кнопка "Deploy to Vercel"

#### **/dashboard/wallet**

- Текущий баланс
- Пакеты кредитов для покупки
- История транзакций
- Stripe checkout интеграция

#### **/dashboard/settings**

- Профиль пользователя
- API ключи
- Управление подпиской
- Опасная зона (удаление аккаунта)

### ▼ 1.7 Тестирование user flows

#### **Happy path - Регистрация:**

1. Открыть [/login](#)
2. Кликнуть "Sign Up"
3. Ввести email и пароль
4. Submit форму
5. Проверить email для верификации
6. Подтвердить email
7. Перенаправление на [/dashboard](#)
8. Проверить 50 приветственных кредитов

#### **Happy path - Создание проекта:**

1. В dashboard кликнуть "New Project"
2. Ввести: "Landing page for SaaS product"

3. Выбрать модель: Claude
4. Кликнуть "Generate"
5. Дождаться генерации (1-3 мин)
6. Увидеть live preview
7. Проверить что код работает
8. Кликнуть "Deploy to Vercel"
9. Получить production URL
10. Проверить списание кредитов

#### **Happy path - Покупка кредитов:**

1. Dashboard → Wallet
2. Выбрать пакет (500 credits / \$29)
3. Кликнуть "Buy"
4. Stripe checkout открывается
5. Ввести тестовую карту: 4242 4242 4242 4242
6. Завершить оплату
7. Webhook обрабатывается
8. Кредиты добавлены на баланс
9. Транзакция в истории

---

## ▼ 2. Параллельная очистка других проектов

Пока тестируем **Nimbus**, запускаем **ESLint** на остальных:

### ▼ 2.1 Velocity (Green)

**Локация:** `/Users/rentamac/Documents/repos/repos/batch-3/velocity`

**GitHub:** <https://github.com/souldevsoul/velocity>

**ESLint правила:**

Allowed colors: emerald-\*<sub>1</sub>, green-\*<sub>1</sub>, teal-\*<sub>1</sub>

Email: support@velocitydev.com

Tagline: "Ship Faster, Scale Smarter"

### Команда:

```
cd /Users/rentamac/Documents/repos/repos/batch-3/velocity
npx eslint --config eslint.config.product.mjs "**/*.{ts,tsx}" --fix
pnpm run build
vercel --prod
```

### Статус:

- ESLint: 0 errors, 210 warnings
- Build: Success
- Git: Committed (a7d6c12) and pushed

## ▼ 2.2 Aurora (Purple)

Локация: [/Users/rentamac/Documents/repos/repos/batch-3/aurora](#)

GitHub: <https://github.com/souldevsoul/aurora>

### ESLint правила:

Allowed colors: purple-\*<sub>1</sub>, violet-\*<sub>1</sub>, pink-\*<sub>1</sub>, fuchsia-\*<sub>1</sub>

Email: support@auroradev.com

Tagline: "Illuminate Your Ideas"

### Команда:

```
cd /Users/rentamac/Documents/repos/repos/batch-3/aurora
npx eslint --config eslint.config.product.mjs "**/*.{ts,tsx}" --fix
pnpm run build
vercel --prod
```

### Статус:

- ESLint: 0 errors, 314 warnings
- Build: Success
- Git: Committed (ed7c63b) and pushed
- Production: <https://aurora-2e5zap5oh-soultosoul.vercel.app>

## ▼ 2.3 Zenith (Orange)

**Локация:** /Users/rentamac/Documents/repos/repos/batch-3/zenith

**GitHub:** <https://github.com/souldevsoul/zenith>

**ESLint правила:**

Allowed colors: orange-\*<sup>\*</sup>, amber-\*<sup>\*</sup>, yellow-\*<sup>\*</sup>

Email: support@zenithdev.com

Tagline: "Reach New Heights"

**Команда:**

```
cd /Users/rentamac/Documents/repos/repos/batch-3/zenith
npx eslint --config eslint.config.product.mjs "**/*.{ts,tsx}" --fix
pnpm run build
vercel --prod
```

**Статус:**

- ESLint: 0 errors, 286 warnings
- Build: Success
- Git: Committed (b3eb770) and pushed

# ▼ 3. Чеклист маркетинговых страниц

**Проверяем для КАЖДОГО проекта:**

## ▼ Homepage (/)

- Hero секция с уникальным headline
- Value proposition ясен за 5 секунд

- CTA кнопка работает (ведет на /login или /dashboard)
- Демо или скриншот продукта (уникальный, не stock)
- "Как это работает" - 3 простых шага
- Pricing превью с ссылкой на /pricing
- FAQ секция (5-8 вопросов)
- Footer со всеми ссылками
- НЕТ фейковых отзывов
- НЕТ фейковых метрик ("10,000+ users")

## ▼ About (/about)

- О платформе - что решаем
- Миссия - зачем создали
- Команда - реальная или честно "indie maker"
- Контакты - support@[project].com
- НЕТ фейковых сертификатов
- НЕТ фейковых партнеров

## ▼ Pricing (/pricing)

- 3 тарифа: Free, Pro, Business
- Цены реалистичные
- Сравнение функций в таблице
- FAQ по ценам
- CTA для каждого плана
- НЕТ "Most Popular" без данных

## ▼ How It Works (/how-it-works)

- Пошаговое объяснение
- Скриншоты каждого шага
- Видео демо (опционально)

- СТА внизу страницы

## ▼ Contact (/contact)

- Форма связи работает
- Email корректный
- Социальные сети (если есть)
- Время ответа указано

## ▼ Experts (/experts)

- Список экспертов или "Coming soon"
- Поиск работает (если есть эксперты)
- Карточки экспертов
- СТА "Become an expert"

## ▼ Политики

- `/privacy` - заполнена, не template
- `/terms` - заполнена, не template
- `/cookies` - описывает реальное использование
- `/acceptable-use` - ясные правила

---

# ▼ 4. Уникальность стилей

Каждый проект должен выглядеть уникально:

## ▼ Nimbus (Blue)

Цвета:

- Primary: `sky-600` (#0284c7)
- Gradient: `from-sky-600 to-blue-700`
- Accent: `cyan-500`
- Shadows: `shadow-sky-500/50`

Стиль:

- Чистый, профессиональный
- Технологичный, облачный
- Минималистичный

#### **Personality:**

- Надежный, стабильный
- Современный fintech vibe
- Профессиональные команды

### ▼ Velocity (Green)

#### **Цвета:**

- Primary: `emerald-600` (#059669)
- Gradient: `from-emerald-600 to-green-700`
- Accent: `teal-500`
- Shadows: `shadow-emerald-500/50`

#### **Стиль:**

- Энергичный, быстрый
- Акцент на скорость
- Динамичные анимации

#### **Personality:**

- Fast-moving, momentum
- Growth-oriented
- Speed-first startups

### ▼ Aurora (Purple)

#### **Цвета:**

- Primary: `purple-600` (#9333ea)
- Gradient: `from-purple-600 to-violet-700`
- Accent: `pink-500`

- Shadows: `shadow-purple-500/50`

#### Стиль:

- Креативный, магический
- Акцент на дизайне
- Плавные transitions

#### Personality:

- Creative, innovative
- Design-focused
- Agencies and designers

### ▼ Zenith (Orange)

#### Цвета:

- Primary: `orange-600` (#ea580c)
- Gradient: `from-orange-600 to-amber-700`
- Accent: `yellow-500`
- Shadows: `shadow-orange-500/50`

#### Стиль:

- Амбициозный, мощный
- Акцент на масштабе
- Энергичные цвета

#### Personality:

- Ambitious, peak performance
- Scale-focused
- Enterprise teams

## ▼ 5. User Flows - Обязательные для MVP

Тестируем каждый flow вручную:

## ▼ Flow 1: Регистрация нового пользователя

**Шаги:**

1. Открыть homepage
2. Кликнуть "Sign Up" или "Get Started"
3. Перенаправление на `/login`
4. Ввести email: `test@example.com`
5. Ввести пароль: TestPass123!
6. Submit форму
7. Проверить email (development: console log)
8. Кликнуть ссылку верификации
9. Перенаправление на `/dashboard`
10. Проверить 50 стартовых кредитов

**Что проверяем:**

- Форма валидирует некорректный email
- Слабый пароль отклоняется
- Email отправляется
- Кредиты начисляются
- Session создается корректно

## ▼ Flow 2: Создание первого проекта

**Шаги:**

1. В dashboard кликнуть "New Project"
2. Ввести название: "E-commerce landing"
3. Ввести описание: "Landing page for online store with product showcase"
4. Выбрать модель: Claude Sonnet
5. Кликнуть "Generate Project"

6. Дождаться генерации (loading indicator)
7. Redirect на `/dashboard/projects/[id]`
8. Увидеть сгенерированный код
9. Live preview загружается в iframe
10. Проверить структуру файлов

**Что проверяем:**

- AI генерация работает
- Код валидный TypeScript/React
- Preview отображается без ошибок
- Кредиты списались (50 → 40 например)
- Проект сохранился в БД

▼ **Flow 3: Деплоймент проекта**

**Шаги:**

1. На странице проекта кликнуть "Deploy"
2. Выбрать Vercel
3. Настроить env variables (если нужно)
4. Кликнуть "Deploy to Production"
5. Дождаться деплоя (1-2 мин)
6. Получить production URL
7. Кликнуть на URL
8. Проверить что сайт работает

**Что проверяем:**

- Vercel интеграция работает
- Деплоймент завершается успешно
- URL доступен
- Сайт функционирует

## ▼ Flow 4: Покупка кредитов

**Шаги:**

1. Dashboard → Wallet
2. Текущий баланс отображается
3. Выбрать пакет: "500 credits - \$29"
4. Кликнуть "Purchase"
5. Stripe Checkout открывается
6. Ввести тестовые данные карты
7. Submit payment
8. Redirect обратно на /dashboard/wallet
9. Баланс обновился (+500)
10. Транзакция в истории

**Что проверяем:**

- Stripe интеграция настроена
- Webhook получен и обработан
- Баланс обновился корректно
- История транзакций работает

## ▼ Flow 5: Marketplace экспертов

**Шаги:**

1. Открыть [/experts](#)
2. Просмотр списка экспертов
3. Кликнуть на карточку эксперта
4. Просмотр портфолио
5. Кликнуть "Hire" или "Assign Task"
6. Создать задачу с описанием

7. Указать стоимость
8. Submit
9. Задача назначена эксперту
10. Кредиты зарезервированы

**Что проверяем:**

- Список экспертов загружается
- Создание задачи работает
- Назначение функционирует
- Система оплаты корректна

---

## ▼ 6. Технический стек (идентичный для всех 4)

### ▼ Frontend

- **Framework:** Next.js 15.4.3 + App Router
- **Language:** TypeScript (строгая типизация)
- **UI:** React 19.1.0
- **Styling:** Tailwind CSS 4
- **Components:** Radix UI + shadcn/ui
- **Animations:** Framer Motion
- **Icons:** Lucide React, Tabler Icons

### ▼ Backend

- **Database:** PostgreSQL (Neon)
- **ORM:** Prisma 5.4.2
- **Auth:** NextAuth 5.0.0-beta.30
- **Storage:** Vercel Blob

- **Payments:** Stripe

## ▼ AI сервисы

- **Anthropic:** Claude (Sonnet 4.5, Opus)
- **OpenAI:** GPT-4, GPT-4o
- **Google:** Gemini Pro
- **Groq:** Llama models (быстрые)
- **Firecrawl:** Web scraping для анализа референсов

## ▼ DevOps

- **VCS:** Git + GitHub
- **Hosting:** Vercel (auto CI/CD)
- **Sandbox:** Vercel Sandbox / E2B
- **Monitoring:** Vercel Analytics

# ▼ 7. ESLint Product Quality

Автоматизированные проверки качества для всех проектов:

## ▼ Что проверяет ESLint

**Брендинг:**

- `use-styleguide-colors-only` - только разрешенные цвета
- `consistent-company-info` - корректные email адреса
- `consistent-payment-provider` - только Stripe

**Функциональность:**

- `no-button-without-handler` - кнопки с type или onClick
- `no-broken-internal-links` - все ссылки работают
- `require-empty-state` - UI для пустых списков

**Качество:**

- `require-try-catch-fetch` - обработка ошибок API

- `require-aria-label-on-icon-buttons` - accessibility
- Image optimization warnings

## ▼ Запуск проверок

### Локально:

```
npx eslint --config eslint.config.product.mjs "**/*.{ts,tsx}"
```

### Автофикс (где возможно):

```
npx eslint --config eslint.config.product.mjs "**/*.{ts,tsx}" --fix
```

### Цель:

- 0 errors (блокируют деплой)
- < 300 warnings (best practices, не критично)

## ▼ Результаты по проектам

### Nimbus:

- Before: 363 problems (19 errors, 344 warnings)
- After: 291 problems (0 errors, 291 warnings)
- Fixed: 72 issues

### Velocity:

- Before: 253 problems (19 errors, 234 warnings)
- After: 210 problems (0 errors, 210 warnings)
- Fixed: 43 issues

### Aurora:

- Before: 373 problems (19 errors, 354 warnings)
- After: 314 problems (0 errors, 314 warnings)
- Fixed: 59 issues

### Zenith:

- Before: 373 problems (19 errors, 354 warnings)
- After: 286 problems (0 errors, 286 warnings)
- Fixed: 87 issues

## ▼ 8. Деплоймент процесс

Для каждого проекта:

### ▼ Pre-deployment проверки

```
pnpm run build      # Локальная сборка  
npx tsc --noEmit    # TypeScript проверка  
npm run lint        # Стандартный ESLint  
npm run lint:product # Product quality
```

Все должны пройти без errors.

### ▼ Коммит изменений

```
git add -A  
git commit -m "Fix: Apply ESLint product quality fixes"  
git push origin main
```

### ▼ Deploy на Vercel

```
vercel --prod
```

Или автоматически: Push в main → Vercel auto-deploys

### ▼ Проверка после деплоя

1. Открыть production URL
2. Номерей < 3 сек загрузка
3. Пройти happy path регистрации
4. Создать тестовый проект
5. Проверить все маркетинговые страницы

6. Проверить все policy страницы
  7. Нет ошибок в console (F12)
  8. Lighthouse score > 90
- 

## ▼ 9. Критерии готовности MVP

Проект готов когда:

### ▼ Маркетинг (Tier 1 - критично)

- Все 6 основных страниц работают
- Все 4 policy страницы заполнены
- Нет битых ссылок
- Нет фейковых отзывов/метрик
- Уникальные скриншоты
- Правильные email адреса
- Стили консистентные
- Footer полный
- Шрифты загружаются

### ▼ Core функционал (Tier 2 - важно)

- Регистрация работает
- Login работает
- Dashboard загружается
- Создание проекта работает
- AI генерация функционирует
- Preview отображается
- Система кредитов работает

### ▼ Advanced функции (Tier 3 - желательно)

- Stripe payments работают

- ○ Деплоймент на Vercel работает
- ○ Marketplace экспертов доступен
- ○ Управление подписками

## ▼ Качество кода (Tier 1 - критично)

- ✓ TypeScript без ошибок
- ✓ ESLint без errors
- ✓ Build успешный
- ✓ Нет console errors на фронте

---

# ▼ 10. Workflow для параллелизации

Эффективная работа команды:

## ▼ Параллельный процесс

### Dev 1: Nimbus - Testing & Fixes

1. Тестирует все маркетинговые страницы
2. Проверяет user flows
3. Фиксит баги по мере нахождения
4. Коммитит в Nimbus repo

### Dev 2: Velocity - ESLint Cleanup

1. Запускает ESLint --fix
2. Проверяет что стили green
3. Коммитит изменения
4. Deploy на Vercel
5. Проверяет production URL

### Dev 3: Aurora - ESLint Cleanup

1. Запускает ESLint --fix
2. Проверяет что стили purple

3. Коммитит изменения
4. Deploy на Vercel
5. Проверяет production URL

#### **Dev 4: Zenith - ESLint Cleanup**

1. Запускает ESLint --fix
2. Проверяет что стили orange
3. Коммитит изменения
4. Deploy на Vercel
5. Проверяет production URL

### ▼ Синхронизация после очистки

#### **После параллельной работы:**

1. Все багфиксы из Nimbus → копируем в другие 3
2. Проверяем что у каждого уникальный стиль
3. Проверяем что backend идентичный
4. Deploy все 4 одновременно
5. Тестируем все 4 production URLs

#### **Критично:**

- Backend ИДЕНТИЧНЫЙ (100%)
- Frontend styling УНИКАЛЬНЫЙ для каждого
- Marketing content УНИКАЛЬНЫЙ для каждого

## ▼ 11. Что идентично, что уникально

### ▼ Идентично (100%) для всех 4

#### **Backend:**

- `/lib/*` – вся бизнес-логика

- `/prisma/schema.prisma` - схема БД
- `/app/api/*` - все API routes
- `/auth.ts` - NextAuth config
- `/middleware.ts` - маршрутизация
- `/hooks/*` - React hooks
- `/utils/*` - helper functions

### **Dashboard components:**

- Layout структура
- Admin components
- Wallet components
- Project components
- Settings components
- Task management

### **Конфигурация:**

- `package.json` (кроме name)
- `tsconfig.json`
- `next.config.ts`
- База `tailwind.config.ts`

## **▼ Уникально для каждого**

### **Брендинг:**

- `/lib/brand.ts` - цвета, имя, tagline
- Tailwind theme colors (primary/secondary)
- Logo colors
- Favicon
- OG images

### **Marketing:**

- `/app/(marketing)/page.tsx` - homepage content
- `/app/(marketing)/about/page.tsx` - О КОМПАНИИ
- `/app/(marketing)/pricing/page.tsx` - тарифы (цены те же, стили разные)
- `/app/(marketing)/how-it-works/page.tsx` - процесс
- `/app/(marketing)/contact/page.tsx` - контакты
- Весь marketing копирайт

#### **Визуальные элементы:**

- Градиенты в компонентах
- Цвета кнопок и акцентов
- Тени (shadow-[color]-500/50)
- Hover effects
- Feature icons colors

---

## ▼ 12. Деплоймент статус

### ▼ Текущие production URLs

#### **Nimbus (Blue):**

- URL: <https://saas-j1ip5504s-soultosoul.vercel.app>
- Status: Ready (21h ago)
- Note: Requires Vercel auth (SSO enabled)

#### **Velocity (Green):**

- URL: <https://velocity-ias5fnyp-p-soultosoul.vercel.app>
- Status: Ready (3d ago - needs redeploy)
- Note: Requires Vercel auth

#### **Aurora (Purple):**

- URL: <https://aurora-2e5zap5oh-soultosoul.vercel.app>

- Status: Ready (21h ago)
- Note: Requires Vercel auth

#### **Zenith (Orange):**

- URL: <https://zenith-51mefi8fe-soultosoul.vercel.app>
- Status: Ready (3d ago - needs redeploy)
- Note: Requires Vercel auth

### ▼ Проблема: Vercel SSO Auth

**Все сайты защищены Vercel authentication.**

**Решение:**

1. Отключить Vercel SSO в настройках проекта
2. Или логиниться в Vercel для доступа
3. Или настроить правильную аутентификацию через NextAuth

## ▼ 13. Следующие шаги

### ▼ Приоритет 1: Маркетинг Nimbus

1. Пройти чеклист всех маркетинговых страниц
2. Проверить каждую ссылку
3. Убрать фейковый контент (если есть)
4. Добавить реальные скриншоты
5. Проверить все email адреса
6. Deploy и проверить на production

**Цель:** 100% готовый маркетинг

### ▼ Приоритет 2: User flows Nimbus

1. Тестировать регистрацию
2. Тестировать создание проекта

3. Тестировать AI генерацию
4. Тестировать покупку кредитов
5. Тестировать marketplace
6. Фиксить баги по мере нахождения

**Цель:** Все flows работают

### ▼ Приоритет 3: Репликация

1. Копировать все багфиксы в Velocity/Aurora/Zenith
2. Проверить уникальность стилей каждого
3. Запустить ESLint на каждом
4. Deploy все 4
5. Финальное тестирование

**Цель:** 4 идентичных по функциям, уникальных по виду

---

#### Проекты:

-  Nimbus: <https://github.com/souldevsoul/accelerator-saas>
-  Velocity: <https://github.com/souldevsoul/velocity>
-  Aurora: <https://github.com/souldevsoul/aurora>
-  Zenith: <https://github.com/souldevsoul/zenith>

**Общий статус:**  All fixed, built, committed, pushed

**Следующий шаг:** Отключить Vercel SSO и протестировать маркетинг