

## Дизайн-концепции (Mood & Principles)

Я предлагаю два направления. Выбери одно, чтобы мы зафиксировали стиль.

### Концепция 1: "The Systematic Minimalist" (Рекомендуемая)

- **Mood:** Строгий, чистый, "Swiss Style". Много воздуха, крупная типографика, минимум декора. Акцент на сетке и вертикальном ритме.
- **Почему подходит:** Идеально отражает твой тезис «*Semantic, accessible markup*». Дизайн сам по себе выглядит как хорошо структурированный документ.
- **Референсы:** Vercel Design System, Geist, Notion.
- **Шрифт:** Sans-serif (Inter или Geist Sans) + Mono для технических деталей.

### Концепция 2: "The Modern Craftsman"

- **Mood:** Мягкий, с использованием карточек, легких теней и blur-эффектов. Более дружелюбный, но технологичный.
- **Почему подходит:** Отражает тезис «*high-quality web interfaces*» через визуальную полировку (UI polish).
- **Референсы:** Linear, современные SaaS лэндинги.
- **Шрифт:** Sans-serif (Satoshi или General Sans) — более геометричный.

(Далее я буду описывать систему, базируясь на Концепции 1, так как она лучше всего "продает" компетенции архитектора дизайн-систем, но она легко адаптируется под вторую).

---

## ЭТАП 2: Design Tokens (Основа системы)

Мы не рисуем "на глаз". Мы определяем переменные. Это готовая структура для твоего tailwind.config.ts.

### 1. Color Palette (Semantic)

Вместо hex-кодов мы используем семантические имена (совместимо с Shadcn/ui).

- background: Pure White (#ffffff) / Dark Grey (#0a0a0a)
- foreground: Black (#09090b) / White (#fafafa) — для основного текста.
- muted: Light Grey (#f4f4f5) — для фонов карточек/секций.
- muted-foreground: Medium Grey (#71717a) — для описаний и второстепенного текста.
- primary: Deep Indigo или Pure Black — для кнопок CTA и акцентов.
- border: Light Grey (#e4e4e7) — тонкие границы.

## 2. Typography (Scale)

- **Font Family:** Geist Sans (или Inter) для UI; Geist Mono для тегов и кода.
- **H1 (Hero):** 48px - 60px (Bold, tight tracking).
- **H2 (Section):** 30px - 36px (SemiBold).
- **H3 (Project Title):** 24px (Medium).
- **Body:** 16px (Regular, line-height 1.6 — для читаемости контента ).
- **Small/Label:** 14px (Medium, uppercase optional).

## 3. Spacing & Radius

- **Base Unit:** 4px.
  - **Container:** Max-width 1200px (centered).
  - **Section Gap:** 80px - 120px (много воздуха).
  - **Radius:** 0.5rem (8px) — сбалансировано (не слишком квадратно, не слишком кругло).
- 

## ЭТАП 3: UI Kit (Компоненты)

Набор компонентов, необходимый для сборки страниц.

### 1. Button:

- *Primary:* Solid background, white text. (Для "View Projects" ).
- *Outline:* Border only. (Для "Contact Me" ).
- *Ghost:* Text only, hover effect. (Для навигации).

### 2. Project Card:

- *Layout:* Сетка или Flex.
- *Elements:* Thumbnail (optional), Title (H3), One-line summary (Muted foreground), Stack Tags (Badge component).
- *State:* Hover вызывает легкий сдвиг вверх или подсветку рамки.

### 3. Badge / Tag:

- Small, rounded-full, subtle background (e.g., bg-slate-100 text-slate-900).  
Используется для стека: "React", "TypeScript".

### 4. Content Section (Case Study):

- Заголовок слева (sticky) или сверху.
- Текстовый блок справа или по центру (max-width 65ch для читаемости).

---

## ЭТАП 4: Wireframes (Ключевые экраны)

Здесь мы раскладываем структуру контента по сетке.

## 1. Home Page (Landing)

- **Header:** Лого (Твое Имя) слева. Навигация (About, Projects, Resume, Contact) справа.
- **Hero Section:**
  - Центрированный блок или Split-screen.
  - **H1:** "Front-End Developer focused on content-rich, high-quality web interfaces.".
  - **Subtext:** "UI developer with 2+ years building content-heavy CMS platforms...".
  - **Actions:** Две кнопки в ряд: [View Projects] [Get in Touch].
- **Featured Projects (Grid):**
  - Заголовок "Selected Work".
  - Сетка из 2-3 карточек (Stripe Mini App, Kanban Board).
  - Каждая карточка кликабельна.

## 2. Project Detail (Case Study)

Структура критически важна для демонстрации "thought process".

- **Project Header:**
  - Breadcrumbs: Home > Projects > Kanban Board.
  - **Title:** "Kanban Board App".
  - **One-liner:** "A task management board with draggable columns...".
  - **Metadata Row:** Role (Frontend), Stack (Next.js, Redux), Year, Links (Live Demo / GitHub).
- **Media:** Большой скриншот или видео демо.
- **Content Body (Чередование текста и кода/картинок):**
  - *Section 1: Context.* Зачем создано?.
  - *Section 2: Tech Stack Deep Dive.* Не просто список, а объяснение выбора (RTK Query for caching).
  - *Section 3: Challenges & Solutions.* Пример кода или схема решения проблемы

drag-and-drop.

- **Next Project:** Ссылка на следующий кейс внизу.

### 3. Resume Page

- Визуально имитирует документ А4, но сверстанный в вебе.
- Кнопка "Download PDF" закреплена сверху справа.
- Четкая иерархия: Experience, Skills (сгруппированные по категориям: Front-end, Testing, Tools).

---

## ЭТАП 5: Правила адаптива (Responsive Rules)

Твой тезис — "*fully responsive layouts*", поэтому адаптив должен быть безупречным.

1. **Mobile (< 768px):**
  - **Nav:** Превращается в "Гамбургер" или упрощенное меню.
  - **Grid:** Проекты выстраиваются в 1 колонку.
  - **Padding:** Уменьшенные отступы (16px - 20px) по краям.
  - **Typography:** H1 уменьшается с 60px до 36px.
2. **Tablet (768px - 1024px):**
  - **Grid:** Проекты в 2 колонки.
  - **Nav:** Полная навигация, если помещается.
3. **Desktop (> 1024px):**
  - **Container:** Фиксированная ширина (например, 1024px или 1200px) для читаемости текста. Не растягивать текст на всю ширину экрана!

---

## ЭТАП 6: Критерии приемки (Acceptance Criteria)

Перед началом кодинга проверь, что дизайн отвечает требованиям:

1.  **Tokens Defined:** Есть список цветов и размеров шрифтов (в уме или Figma).
  2.  **Component Reusability:** Кнопки и карточки на странице Projects и Home выглядят одинаково.
  3.  **Accessibility:** Контраст текста соответствует WCAG AA. (Серый текст на белом фоне должен быть читаем).
  4.  **Content First:** В макете используется реальный текст из PDF, а не Lorem Ipsum. Это доказывает подход "Content-rich".
-