

# AI-CODING

**Создание AI интернет-магазина**

>\_ Введение в AI-программирование и новые профессии



по Сергей Стародубцев

1 урок курса





# **AI-Coding: Новый взгляд на программирование**

**Вас приветствует курс  
«AI Coding: Создание интернет-магазина»**

Этот курс — ваш первый шаг к профессиям будущего, где искусственный интеллект (ИИ) становится незаменимым партнёром в создании современных веб-приложений.

## **Сегодня вы узнаете:**

- что такое AI-программирование?
- что дает парное программирование с ИИ?
- какие новые профессии появляются с внедрением ИИ?
- какие инструменты понадобятся на пути?

\*ИИ - искусственный интеллект, AI - artificial intelligence

# Что такое AI Coding?

AI Coding — это программирование в связке с современными инструментами искусственного интеллекта: Cursor AI, Copilot, Cloude Desktop, OpenAI Agents SDK и т.д. Основная идея заключается в использовании AI-агентов, которые могут автономно выполнять отдельные этапы или даже целые процессы в разработке программного обеспечения.

## Ключевые особенности:



### Программирование на естественном языке

Возможность создавать код, используя естественный язык



### Мгновенная генерация кода

Быстрое создание кода по описанию задачи



### Итеративная доработка

Улучшение кода через диалог с искусственным интеллектом



### Фокус на бизнес-задачах

Концентрация на решении проблем, а не на написании кода

Главная идея: Разработчик + ИИ = команда, дополняющая сильные стороны друг друга.





## AI Coding на примере Cursor AI: принципы работы агента



### Понимание контекста

Анализирует существующий код, изменения и комментарии разработчика.



### Выполнение запросов

Обрабатывает задачи, сформулированные на обычном языке.



### Планирование решения

Разбивает сложные задачи на логические этапы.



### Генерация кода

Самостоятельно внедряет изменения и обновляет документацию.



### Диалоговый режим

Уточняет детали, предлагает альтернативные решения.



### Валидация

Запускает тесты, находит ошибки, проводит статический анализ

Агент работает как "цифровой напарник", берущий на себя рутинные задачи и ускоряющий разработку.

# Vibe-coding: Программирование «по ощущениям»

Vibe-coding — это программирование «по вайбу», когда разработчик полагается на AI-ассистента для генерации большей части кода.

## Характеристики:

 <b>Минимум понимания</b>	 <b>Быстрое прототипирование</b>	 <b>Поверхностный подход</b>
Разработчик не вникает глубоко в алгоритмы и структуру кода.	Создание работающих прототипов без детального планирования.	Фокус на результате, а не на качестве и понимании процесса.

## Особенности и риски:

 <b>Суть подхода</b>	 <b>Процесс работы</b>	 <b>Риски метода</b>
<ul style="list-style-type: none"><li><input type="checkbox"/> Описывает задачу обычными словами, ИИ генерирует готовый код.</li><li><input type="checkbox"/> Вместо синтаксиса — интуиция и "вайб".</li></ul>	<ul style="list-style-type: none"><li><input type="checkbox"/> Описание задачи на естественном языке</li><li><input type="checkbox"/> Получение кода от ИИ</li><li><input type="checkbox"/> Тестирование и уточнение запроса</li></ul>	<ul style="list-style-type: none"><li><input type="checkbox"/> "Чёрный ящик" — непонимание внутренней логики</li><li><input type="checkbox"/> Проблемы с масштабированием решений</li><li><input type="checkbox"/> Ложная уверенность у новичков</li></ul>

Автор концепции: Андрей Карпати (бывший директора по ИИ и зрению автопилота Tesla и сооснователь OpenAI), 2025 год

Vibe-coding — мощный инструмент для быстрой разработки прототипов, но не замена фундаментальным знаниям программирования.

# AI-coding ≠ Vibe-coding

## Vibe-coding

- Быстрые прототипы и эксперименты
- Не требует глубоких технических знаний и навыков программирования
- Vibe-coder использует AI как замену профессиональным навыкам
- Хрупкая архитектура
- Неструктурированный код с неочевидными ошибками и уязвимостями
- Работающий прототип за пару часов, но с ограничениями.

## Настоящий AI-coding

- Человек: архитектура, планирование, стратегия
- ИИ: реализация, оптимизация, кодогенерация
- Vibe-coder использует AI как инструмент для усиления своих компетенций
- Масштабируемая система на годы вперёд.



Для серьёзных проектов нужен симбиоз человека и ИИ, а не просто "вайб"

# Преимущества парного программирования с AI



## Мгновенный доступ к знаниям

AI знает тысячи технологий и паттернов



## Ускорение разработки

Генерация кода, архитектуры и контента в реальном времени



## Меньше ошибок

AI помогает найти и исправить баги



## Креативность

Предлагает альтернативные решения



## Непрерывное обучение

Изучаем новое во время работы



## Автоматизация рутинны

Внесение быстрых правок, документация кода или API, написание тестов

Скорость разработки ↑, рутинна ↓, креативность ↑.



# Что изменилось в мире программирования?

 Скорость разработки выросла в 10 раз

 Фокус сместился с синтаксиса на архитектуру и логику

 AI стал полноценным партнером в разработке

 Новые роли и профессии в IT

# AI Coding открывает новые профессии

Мир разработки меняется — появляются принципиально новые профессии:

## </> Вайб-кодер (Vibe Coder)

Разработчик, управляющий AI через контекст и промпты. Уровень Junior-разработчика.



## Промпт-инженер (Prompt Engineer)

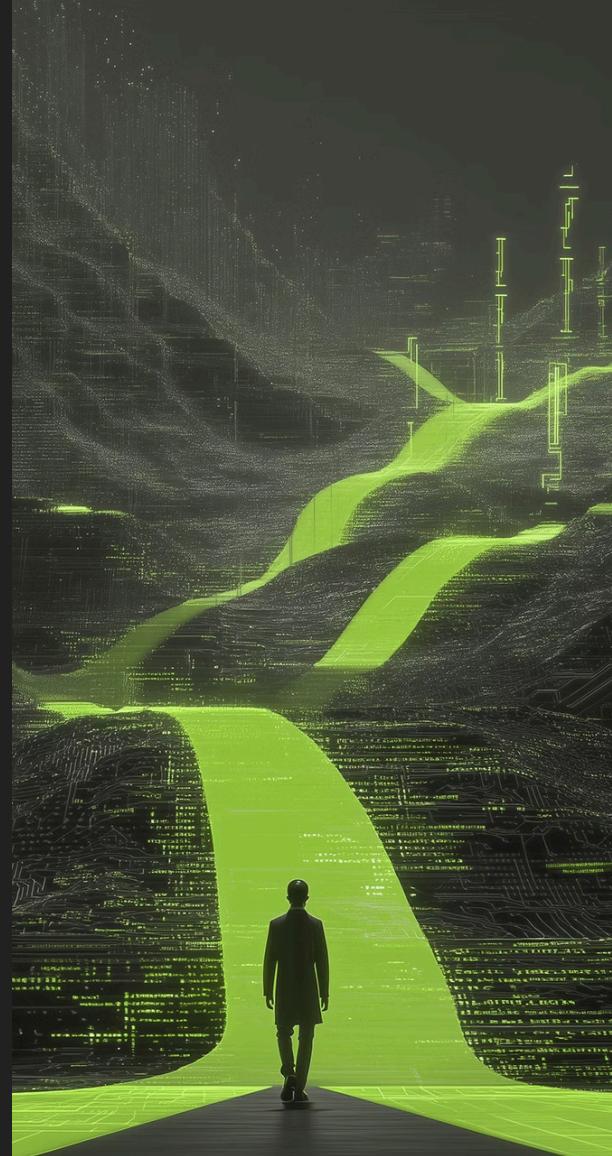
Специалист по эффективному взаимодействию с AI. Уровень Middle-разработчика.



## AI-архитектор (AI Architect)

Проектировщик систем, интегратор AI-систем. Уровень Senior-разработчика.

Они не просто пишут код — строят интерфейс общения между человеком и ИИ, задают тон работы искусственного интеллекта в командах.





# Вайб-кодер [Vibe Coder]

Роль - работает в потоке:

- Придумывает концептуальные решения "на ходу"
- Творчески взаимодействует с ИИ для поиска нестандартных подходов
- Часто работает над прототипами и первыми версиями продуктов

Главная задача: Создание атмосферы и пользовательского опыта

## Ключевые навыки:

- Понимание UX/UI принципов
- Работа с дизайн-системами
- Создание интерактивных интерфейсов
- Передача эмоций через код

## Зоны ответственности:

- Frontend-разработка с фокусом на пользователя
- Анимации и микровзаимодействия
- Адаптивность и доступность
- А/В тестирование интерфейсов

# Промпт-инженер [Prompt Engineer]

Роль — специалист по работе с языковыми моделями:

- Пишет эффективные промпты для искусственного интеллекта
- Получает оптимальные ответы для разработки кода, архитектуры, тестирования
- Ключевая фигура в команде AI-разработки

Главная задача: Эффективное взаимодействие с AI-системами

## Ключевые навыки:

-  Формулирование точных запросов к AI
-  Итеративная доработка промптов
-  Понимание возможностей и ограничений LLM
-  Анализ качества AI-ответов

## Зоны ответственности:

- Создание библиотеки промптов
- Настройка AI-инструментов под задачи
- Обучение команды работе с AI
- Контроль качества AI-генерируемого кода





# AI-архитектор [AI Architect]

Роль - проектирует инфраструктуру новых цифровых продуктов:

- Определяет, где и как использовать ИИ
- Проектирует архитектуру решений
- Проектирует базы данных
- Интегрирует AI в продукт
- Анализирует и оптимизирует связку микросервисов и моделей
- Работает на стыке бизнеса, данных и технологий

Главная задача: Проектирование систем

## Ключевые навыки:

-  Системное мышление и архитектурные паттерны
-  Интеграция AI/ML компонентов
-  Масштабирование AI-решений
-  Балансировка производительности и качества

## Зоны ответственности:

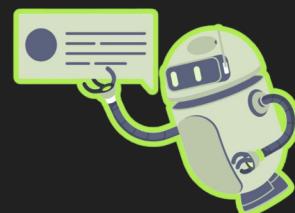
- Выбор AI-стека технологий
- Проектирование микросервисной архитектуры
- Планирование нагрузки на AI-сервисы
- Обеспечение безопасности AI-компонентов

# Отличия и зоны ответственности

Роль	Фокус	Задачи	Результат
Вайб-кодер	Пользователь	Написание и рефакторинг кода	Интерфейсы и опыт
Промпт-инженер	AI-инструмент	Оптимизация запросов к AI	Качество AI-решений
AI-архитектор	Система	Проектирование системы, разработка API, интеграция AI в проект	Архитектура и масштаб

## Пересечения ролей:

- Все работают с AI-инструментами
- Общие принципы качества кода
- Командная работа и коммуникация



# Почему эти навыки важны сейчас?

## Тренды рынка:

**85%**

### Компаний

планируют внедрить  
AI в разработку к 2026 году

**40%**

### Рост зарплат

AI-специалистов  
за последний год

**3x**

### Превышение спроса

Спрос на AI-навыки превышает  
предложение в 3 раза

## Конкурентные преимущества:

- Эффективность разработки
- Качество продукта
- Скорость выхода на рынок
- Инновационность решений

# Инструменты и технологии

В данном курсе мы коснемся следующих инструментов:



LLM



beget

## Cursor AI

IDE с AI-ассистентом. Это наша рабочая лошадка.

## LLM

GPT, Claude, Gemini и др. – для генерации и анализа. Доступ к инструменту мы получим через API OpenRouter.AI, рассмотрим генерацию изображений через SYNTX AI бота.

## Figma

За основу проекта мы возьмем проект из Figma и выполним на базе нее прототип.

## Beget

Готовый проект мы загрузим на хостинг Beget.com.

Инструменты по ролям:

Профессия	Инструменты
Вайб-кодер	Cursor AI, Figma
Промпт-инженер	Openrouter.ai, SYNTX AI, Cursor AI
AI-архитектор	API Openrouter.ai, Cursor AI, Beget.com, базы-данных

## Что вас ждет дальше?

Установка Cursor AI  
и первый проект

Освоение промпт-инжиниринга

Погружение в микросервисную  
архитектуру

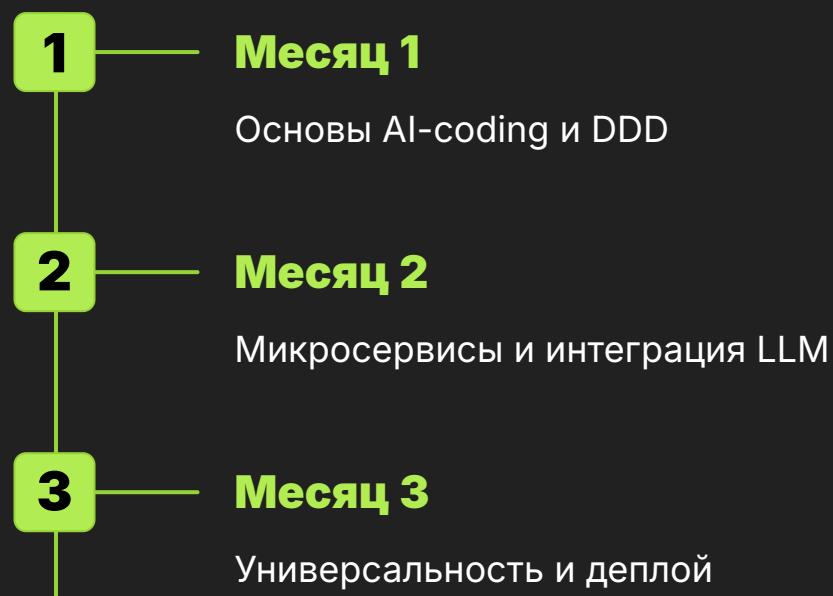
DDD-подход к разработке

Новые soft & hard навыки AI-  
разработчика

Создание своего интернет-  
магазина

Будущее начинается прямо сейчас — давайте создавать его вместе!

# Ваш путь в курсе



## К концу курса вы:

- Освоите базовые (начальные) навыки для всех трех ролей
- Создадите собственный интернет-магазин
- Получите портфолио для новой карьеры
- Будете готовы к AI-ориентированному рынку

# Итоги урока

- Разработчик + ИИ = команда, дополняющая сильные стороны друг друга.
- Агент работает как "цифровой напарник", берущий на себя рутинные задачи и ускоряющий разработку.
- Vibe-coding — мощный инструмент для быстрой разработки прототипов, но не замена фундаментальным знаниям программирования.
- AI-coding ≠ Vibe-coding.
- Фокус в разработке сместился с синтаксиса на архитектуру и логику.
- Мир разработки меняется — появляются принципиально новые профессии.
- Новые роли требуют навыков работы с промптами и архитектурой.

## Вопросы для размышления:

- Какая из трех ролей вам ближе?
- Какой тип интернет-магазина хотите создать?
- Готовы ли изменить подход к программированию?

