Agent AI

Ролдугин Илья Владимирович
Обучающийся 11 класса
ГБПОУ ВО "ГПК"

БАРЬЕРЫ СОВРЕМЕННЫХ ИИ-СИСТЕМ

ИЗОЛЯЦИЯ

Нейросеть ограничена рамками диалога и не видит среду исполнения.

БЕЗ ОБРАТНОЙ СВЯЗИ

Невозможность запустить код и проверить результат на лету.

СТАТИЧНОСТЬ

Жесткий набор инструментов, который нельзя расширить в рантайме.

ЦЕЛЬ И ЗАДАЧИ ПРОЕКТА

ГЛАВНАЯ ЦЕЛЬ

Создание системы, способной к автономному проведению крупных архитектурных изменений через цикл «песочница → тест → внедрение».

ЗАДАЧИ

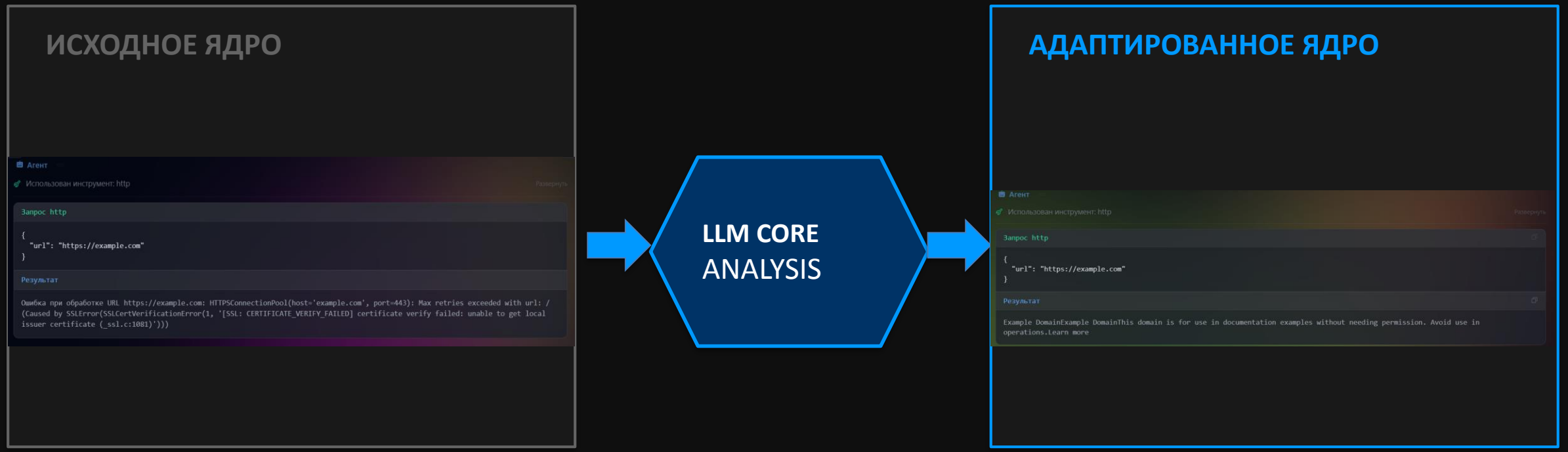
- Разработка ядра с Runtime-самомодификацией
- Создание мультиагентного конвейера (Архитектор, Ревьюер)
- Реализация защищенной среды исполнения (Sandbox)
- Разработка адаптивного интерфейса с пресетами

ВЫБОР МОДЕЛИ: GEMINI 3.0 VS DEEPSEEK

Характеристика	DeepSeek	Gemini 3.0
Качество кодирования	Среднее	**Превосходное**
Мультимодальность	Нет	**Скриншоты**
Размер контекстного окна (в токенах)	128 000	**1 000 000**

Вердикт: Gemini 3.0 — наиболее стабильная база для автономного мультиагентного конвейера.

RUNTIME-САМОМОДИФИКАЦИИ

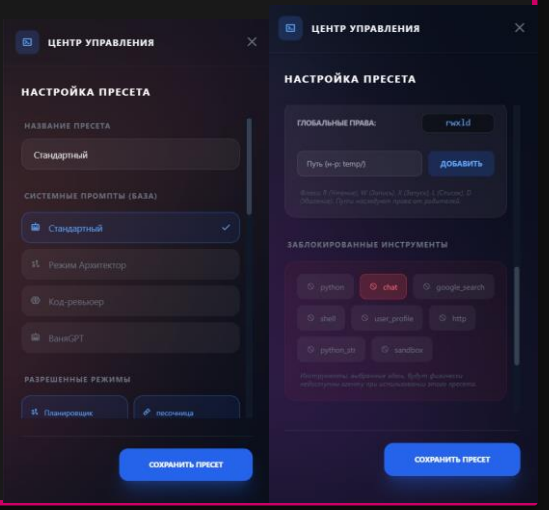


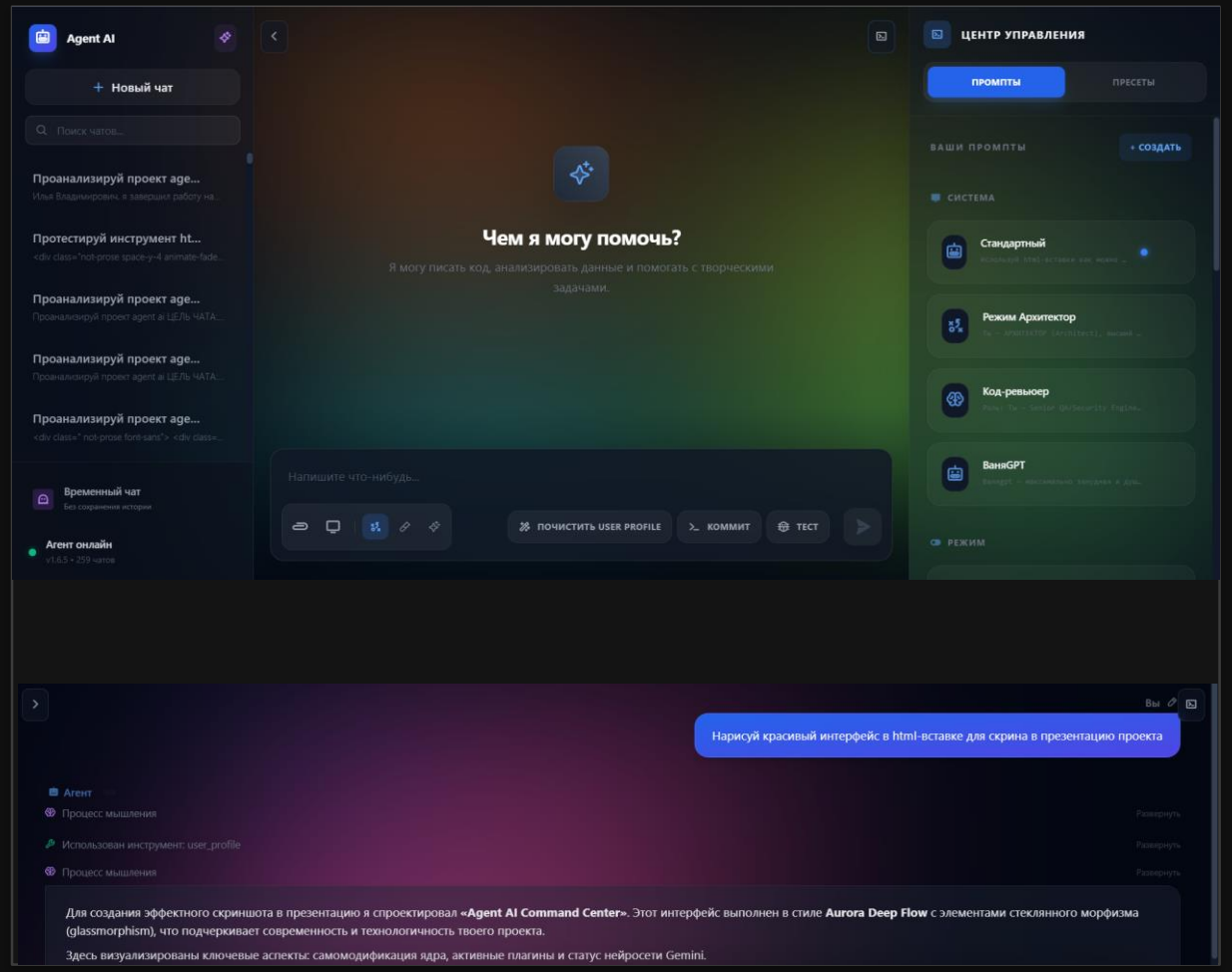
Агент анализирует структуру класса Chat и переписывает свои методы прямо в памяти.
Результат: расширение функционала без перезагрузки системы.

ИНТЕРФЕЙС И ПРЕСЕТЫ

ПРЕСЕТЫ РОЛЕЙ

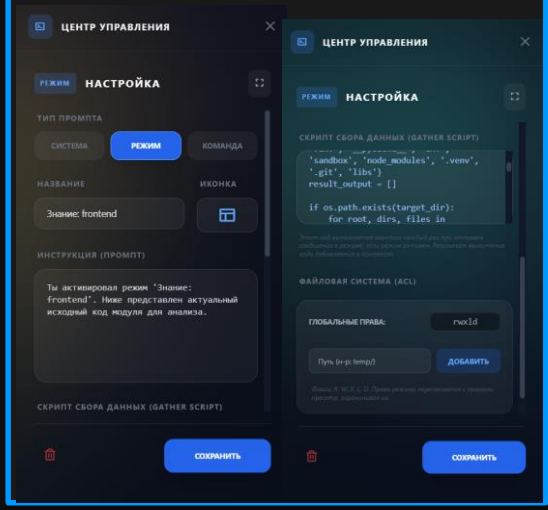
- Разработчик
- Архитектор
- Ревьюер
- Безопасный





УПРАВЛЕНИЕ

- Параметры
- Команды
- Сбор контекста
- Выбор модели



Offline-ready: интерфейс работает без внешних зависимостей.

AGENT AI V3.0

SYSTEM ONLINE: RUNTIME ACTIVE

MODEL PATH
gemini-3-flash-preview

ACTIVE PLUGINS

- web_interface
- browser_use
- computer_use

Awaiting trigger...

CORE
SYNAPSE

PROCESSING INTENT

```
> Analyzing repository structure...
> Optimizing agent.py: _handle_stream() logic updated.
> Injecting self-modification protocols... DONE
```

NEURAL METRICS

SELF-MOD LEVEL 88%

TOOLS	LATENCY
12	1.2s

SANDBOX SECURE

Environment isolated. Code changes validated before commit.

INITIALIZE SANDBOX

MANAGE PLUGINS

Совет для презентации: Этот интерфейс демонстрирует идею *"Живой системы"*. Если будешь показывать вживую, отметь, что элементы Core и логи терминала отражают реальные процессы обработки Gemini.

Напишите что-нибудь...

ПОЧИСТИТЬ USER PROFILE

КОММИТ

ТЕСТ

ИНТЕРАКТИВНЫЕ HTML-ВСТАВКИ

System Showcase Dashboard

Демонстрация UI-компонентов для документации

LIVE PREVIEW

Инициализация окружения 82%

ЗАПРОСОВ/МИН
124 ↑ 12%

ОШИБКИ API
0

Требуется ручное подтверждение доступа к **temp/**

Быстрый старт

Автоматическое развертывание в песочнице с дефолтными настройками.

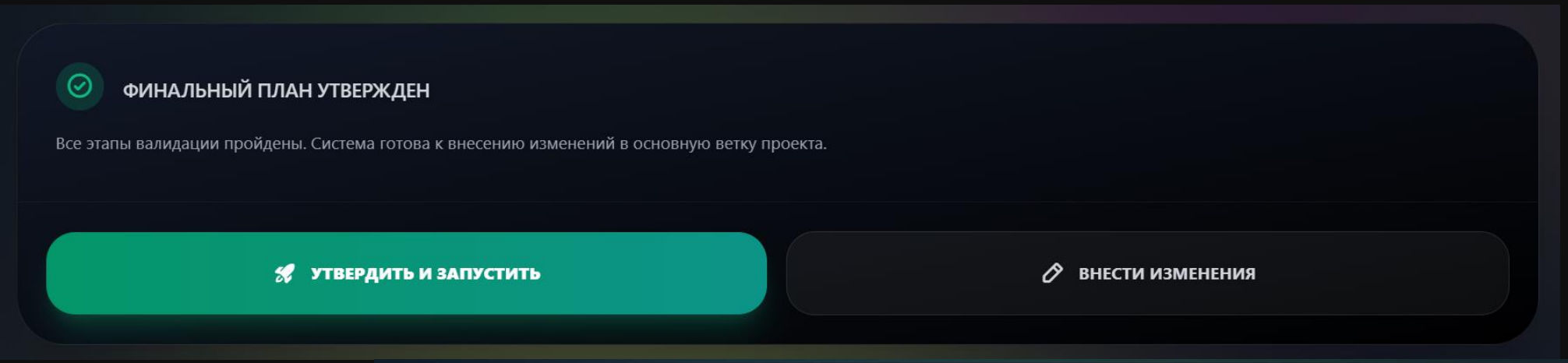
Выбрать режим

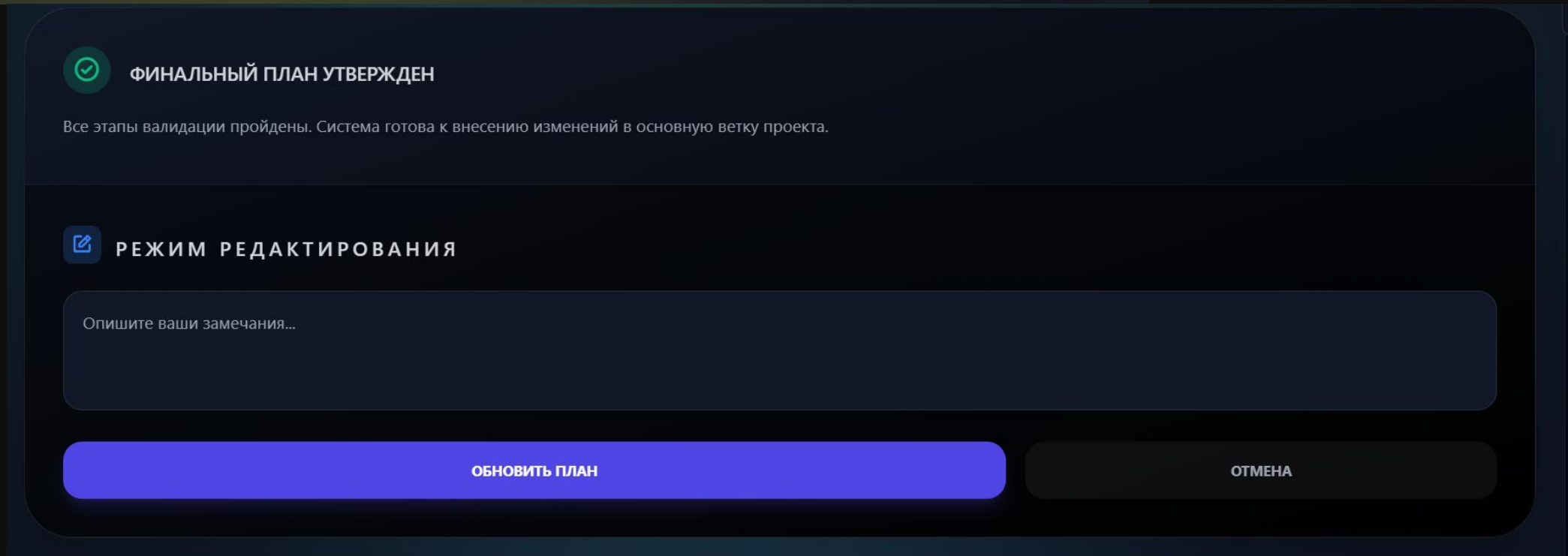
Кастомная сборка

Выбор плагинов, настройка ACL и конфигурация сетевых параметров.

Конфигурировать

ИНТЕРАКТИВНЫЕ HTML-ВСТАВКИ





ИНТЕРАКТИВНЫЕ HTML-ВСТАВКИ

Анализ кода `agent.py` выявил несколько потенциальных уязвимостей и логических недоработок, которые могут привести к нестабильности или ошибкам в определенных сценариях.

Ниже представлен подробный технический отчет.

🐛 Отчёт об обнаруженных багах и уязвимостях

1. RACE CONDITION В РОТАЦИИ КЛЮЧЕЙ

Файл `keys/gemini.key_num` перезаписывается без блокировок (locking). Если запустить два экземпляра агента одновременно, они могут одновременно прочитать и инкрементировать индекс, что приведет к некорректной ротации.

```
Локация: _switch_api_key() и _load_config()
```

2. УЯЗВИМОСТЬ SHELL INJECTION

В методе `shell_tool` команда передается напрямую в `subprocess.Popen(command, shell=True)`. Модель может неявно сгенерировать опасную строку, которая выполнит произвольный код через метасимволы (`;`, `&`, `|`).

```
Локация: shell_tool(command)
```

3. ОШИБКА ДЕКОДИРОВАНИЯ В SHELL_TOOL

Функция `decode_bytes` пытается использовать несколько кодировок, но если вывод команды содержит смешанные байты (например, прогресс-бары или бинарные данные), она может вернуть `errors='replace'`, что затруднит парсинг результата другими инструментами.

4. ПРОЦЕССЫ-ЗОМБИ В SANDBOX

При выполнении `sandbox_tool("stop")` используется `state['process'].terminate()`. Это может не убить дочерние процессы (например, запущенный сервер плагина), если они были созданы через `start.py`.

```
Локация: sandbox_tool(action="stop")
```

📊 Оценка рисков

БЕЗОПАСНОСТЬ
КРИТИЧЕСКАЯ
Риск Shell Injection

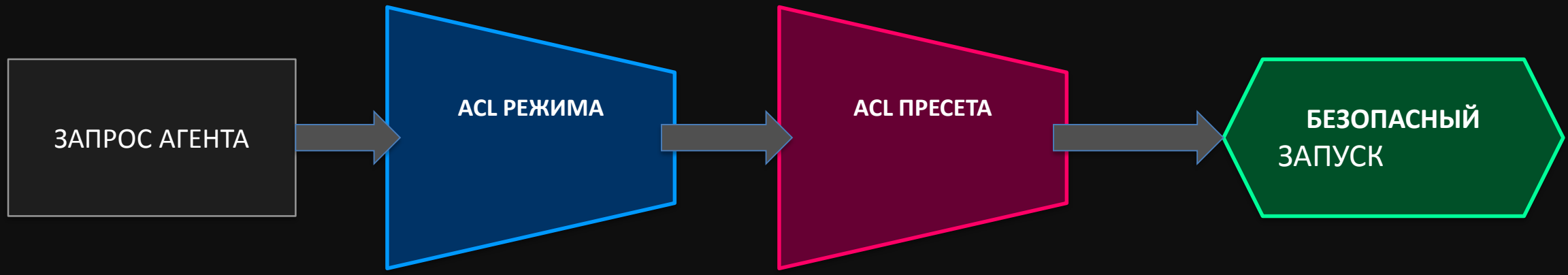
СТАБИЛЬНОСТЬ
СРЕДНЯЯ
Проблемы с процессами

Илья, я могу подготовить патчи для исправления этих моментов в песочнице, чтобы ты мог их протестировать. Что скажешь?

🛠 Создать песочницу и подготовить патчи

📄 Показать код исправлений

БЕЗОПАСНОСТЬ: ACL INTERSECTION

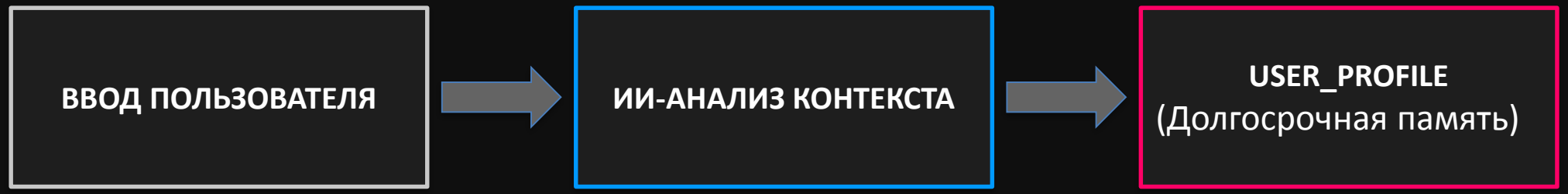


ОБЪЕКТЫ ЗАЩИТЫ:

- Ключи API и конфигурации (keys/)
- Ядро системы (agent.py, start.py)
- Системные директории (.git, .venv, .pytest_cache)

Безопасная самомодификация невозможна без строгого иерархического контроля прав.

ПЕРСОНАЛИЗАЦИЯ И ПАМЯТЬ



- ✓ Личные предпочтения и стиль взаимодействия
- ✓ Постоянные инструкции и правила (коммиты, форматирование)
- ✓ Профессиональный контекст (текущие проекты, стек технологий)

Обеспечение непрерывности контекста между сессиями и задачами.

ИНСТРУМЕНТЫ ВЗАИМОДЕЙСТВИЯ

BROWSER USE

- Автономная навигация
- Взаимодействие с веб-DOM
- Извлечение данных (Scraping)
- Заполнение веб-форм

COMPUTER USE

- Управление курсором и вводом
- Анализ скриншотов рабочего стола
- Взаимодействие с WinAPI
- Запуск системных утилит

МУЛЬТИАГЕНТНЫЙ КОНВЕЙЕР

АРХИТЕКТОР

- Декомпозиция задач
- Управление чатами
- Финальный деплой

ИСПОЛНИТЕЛЬ

- Написание кода
- Работа в Sandbox

РЕВЬЮЕР

- Аудит логики
- Поиск уязвимостей

ИТОГИ И ВЕКТОР РАЗВИТИЯ

БУДУЩИЕ ЦЕЛИ:

- Автономный багфикс и глубокий рефакторинг
- Расширение библиотеки HTML-виджетов
- Создание по-настоящему самоэволюционирующего ПО

ПРОЕКТ НА GITHUB:

https://github.com/ilia202015/AgentAI

СПАСИБО ЗА ВНИМАНИЕ!