ChatGPT

Структурированное исследование и архитектурное обоснование проекта Agent Smith

1. Введение: цели проекта и контекст

Agent Smith — это инициативный проект по созданию суверенной мульти-модельной AI-платформы для госсектора, предназначенной в первую очередь для **автоматизации обработки обращений граждан**. Система интегрируется с национальной платформой электронных обращений *eOtinish* и входит в экосистему инициатив GovTech. Главные цели проекта:

- Ускорить и упростить обработку жалоб и заявлений граждан за счет применения больших языковых моделей (LLM) и вспомогательных AI-агентов.
- Повысить прозрачность и подконтрольность рассмотрения обращений, фиксируя результаты с помощью технологий **блокчейна** и ЭЦП для обеспечения юридической силы ответов.
- Обеспечить **суверенность** решения – хранение данных и выполнение моделей на отечественной инфраструктуре, с минимальной зависимостью от внешних API, во избежание утечек данных и с учётом требований национальной безопасности.
- Продемонстрировать экономический эффект: по оценкам, каждый специализированный AI-агент может ежегодно экономить государству от 1 до 10 млн тенге за счёт автоматизации рутинных операций.

Контекст. В Казахстане наблюдается огромный поток обращений граждан: за 10 месяцев государственные органы получили свыше *3,2 млн обращений* (2,1 млн – центральные органы, 1,1 млн – на местном уровне) [1]. Это стимулировало поиск новых подходов, и в конце 2024 г. Генеральная прокуратура объявила о планах применять ИИ для анализа письменных обращений [2]. По словам Комитета по правовой статистике, накопленные данные системы eOtinish вместе с NLP-технологиями позволяют **автоматически выявлять ключевые проблемы, классифицировать риски по отраслям и оперативно доводить их до ответственных органов** [3]. Проект Agent Smith нацелен реализовать этот подход на практике, став частью Национальной AI-платформы, представленной в инициативе цифрового правительства Республики Казахстан [4].

Участники. Разработку ведёт консорциум компаний *Bitmagic* и *QOSI* при поддержке технопарка Astana Hub, в сотрудничестве с государственными структурами (Министерство цифрового развития, Генеральная прокуратура и др.). Платформа будет распространяться с открытым исходным кодом (лицензии MIT и CC-BY-4.0) для стимулирования партнёрства и аудита безопасности. Пользователями системы станут государственные органы (для внутреннего ускорения обработки обращений), а в перспективе – и сами граждане, получающие более быстрый и качественный обратный ответ на свои обращения.

2. Обзор текущего состояния проекта

На основе актуальной **доски Trello** проекта можно проследить прогресс и текущие задачи Agent Smith. К настоящему моменту выполнены следующие ключевые этапы:

- **Формирование концепции и архитектуры.** Подготовлено описание общей архитектуры системы, включающее интеграцию с eOtinish, модульную структуру AI-агентов, а также использование блокчейн-слоя (*GovChain*) для фиксации действий. Определён стек технологий: среди прочего, выбраны платформы *LangChain* и *CrewAI* для оркестрации агентов, *MLflow* для управления моделями, *Hyperledger Besu* для DAO-блокчейна, *Milvus* для векторной БД, *MinIO* и *PostgreSQL* для хранения данных, *Keycloak* для единой авторизации, *Prometheus* для мониторинга, *Tyk* в качестве API-шлюза и пр.
- **Интеграция с eOtinish (MVP).** Реализован автономный модуль Agent Smith, способный подключаться к API системы eOtinish и импортировать обращения в свою базу. Настроен начальный **конвейер обработки обращения**: классификация тематики, извлечение ключевых сущностей, резюмирование содержания обращения. Разработан прототип пользовательского интерфейса (веб-приложение) с Kanban-доской для визуализации статусов обращений (входящие, в работе, выполненные, отклоненные).
- **Документация и аналитика.** Подготовлен краткий план пилотного проекта на базе eOtinish и развернута **wiki-платформа знаний** проекта. Составлен подробный отчёт по модулю «Обращения граждан» (этап пилота eOtinish) с описанием бизнес-процессов, метрик SLA и предложенными показателями эффективности пилота. Также выполнены исследования и отчёты по смежным темам: анализ опыта Республики Корея (Government24, e-People, eGovFrame) и других стран, обзор отечественных производителей серверного оборудования (для развёртывания системы on-premise), сбор лучших мировых практик по AI-агентам и технологиям (OpenAI, Anthropic, и др.). Эти материалы легли в основу требований и рекомендаций, описанных в следующих разделах.
- **Организационные шаги.** Проведены переговоры и достигнуто соглашение о сотрудничестве с операторами платформы eOtinish для запуска **совместного пилота**. Инициатива получила одобрение профильных руководителей, определены ответственные лица за пилот от каждого ведомства. Также налажены партнёрства с провайдерами **no-code** решений, чтобы в будущем облегчить интеграцию Agent Smith с другими государственными системами без существенной доработки кода.

Текущие задачи и фокус. Проект находится на стадии реализации первого пилотного запуска. Команда сосредоточена на доработке ключевых компонентов для пилота eOtinish: настройке единой авторизации через SSO, обеспечении автоматического подтягивания новых обращений в систему в режиме реального времени, доработке UI/UX для инспекторов и руководителей. Проводится углублённое **сравнительное исследование** лучших практик (подробнее в разделе 3) с целью включения дополнительных функций в следующую итерацию – таких как автозаполнение данных заявителя, отслеживание таймеров SLA с автоматической эскалацией просроченных обращений, виджеты оценки удовлетворённости ответом и публичные панели KPI. Параллельно ведётся работа над **базой знаний** для агентов: оцифровываются нормативные акты, накопленные данные предыдущих обращений, готовится загрузка этих данных в векторное хранилище для реализации механизмов RAG.

В ближайших планах – завершение Sprint 1 (авторизация, импорт обращений, канбан), после чего начнётся расширение пилота на группу реальных пользователей и отслеживание метрик. Контрольные точки и дальнейшие этапы развития подробно изложены в разделе 7 (Roadmap).

3. Международные стандарты и лучшие практики

При проектировании Agent Smith учитываются передовые международные наработки в сфере электронного правительства, искусственного интеллекта и кибербезопасности. Ниже приведён анализ ключевых идей из зарубежных источников, которые легли в основу требований к системе.

Опыт цифрового правительства Республики Корея. Система работы с обращениями граждан в Южной Корее считается «золотым стандартом». Единый портал **Government24** предоставляет гражданам единое окно доступа приблизительно к *12 000* государственным услугам и справкам [5], реализуя принцип «заполняю данные один раз – получаю услуги повсюду». Система аутентификации включает государственную PKI и мобильные ID (например, PASS), а данные пользователя подтягиваются автоматически через механизм *Public MyData* – только в 2024 г. в Корее было обработано ~560 млн транзакций автозаполнения данных【12†】. После подачи обращения через портал, оно попадает в систему **e-People** – централизованную платформу рассмотрения петиций граждан. В e-People реализована интеллектуальная маршрутизация: обращение автоматически классифицируется по теме и направляется в ответственное ведомство (из ~49 министерств) или сразу в несколько, если вопрос межведомственный. При межведомственных обращениях система создаёт *совместную задачу* с указанием ответственных от каждого органа и контролем сроков【12†】. Гражданин может в режиме онлайн отслеживать статус рассмотрения: в личном кабинете показывается таймер SLA и каждое действие чиновников по его обращению. Если сроки нарушены, e-People автоматически эскалирует обращение в надзорный орган (омбудсмен)【12†】. По результатам рассмотрения заявителю приходит ответ через портал, после чего система предлагает оценить удовлетворённость (по шкале 1–5 звёзд) и оставить отзыв; агрегированные рейтинги публикуются открыто, формируя **показатель удовлетворённости** граждан. Важная черта корейской платформы – **открытость кода**: back-end e-People построен на фреймворке **eGovFrame** (стек Spring + ~30 общих компонентов: SSO, файловый сервис, логирование, workflow и др.), который находится в открытом доступе и поддерживается государственным агентством NIA. Такой подход («open-source by default») повышает доверие и позволяет каждому ведомству запускать новые цифровые сервисы, используя уже готовые общие модули, что ускоряет разработку примерно на 60%.

Опираясь на корейский опыт, для Agent Smith были выделены пробелы (*gap-анализ*) и потенциальные улучшения: - **Единая точка входа.** В Казахстане портал eOtinish пока является одним из каналов (помимо писем, личных приемов и др.). Желательно движение к модели «single window», когда гражданин подает обращение через единый интерфейс (портал или приложение), откуда информация далее распределяется. В рамках пилота Agent Smith планируется реализовать *виджет обращения* на существующем eGov-портале, чтобы пользователю не пришлось переходить в отдельную систему. - **Автозаполнение данных (MyData).** Отсутствие механизма авто-подстановки персональных данных и известной информации в форму обращения приводит к лишним усилиям граждан и ошибкам. Планируется прототип интеграции с государственными реестрами (аналог корейского MyData): при авторизации заявителя основные сведения (ИИН, ФИО, контакты, адрес) будут автоматически заполняться в черновике обращения. Для реализации рассматривается безопасное хранилище профилей граждан (возможно, на базе IPFS + блокчейн-ссылок) либо прямое обращение к API «личного кабинета». - **Прозрачные SLA и эскалации.** Агент Smith возьмёт на контроль сроки выполнения по каждому обращению. На этапе пилота внедряется визуальный индикатор статуса (например, зелёный/жёлтый/красный в зависимости от близости дедлайна) для ответственных исполнителей. При нарушении срока система сможет автоматически уведомлять руководителей или надзорный орган. Будет реализован webhook или отдельная очередь событий `sla.breached` для сбора статистики по просроченным обращениям. - **Обратная связь и открытые данные.** По завершении обращения планируется запрашивать у гражданина оценку ответа. Эти данные послужат для внутренней

аналитики, а в агрегированном виде могут публиковаться (например, процент удовлетворённых ответом по ведомствам). Практика Кореи показывает, что публичность таких метрик стимулирует улучшение качества работы. В пилотной версии Agent Smith будет встроен модуль опроса и сформирован дашборд удовлетворённости на базе Grafana, доступный руководству. - **Этическая экспертиза AI.** Южная Корея ввела практику обязательной оценки влияния новых цифровых услуг на персональные данные и этику (AI Impact Assessment), результаты которой публикуются открыто. Для Agent Smith также важно проведение такой экспертизы: анализ рисков обработки обращений AI-моделями (конфиденциальность, недопустимость дискриминации, объяснимость решений и пр.). В план работ включена задача разработки полуавтоматического инструмента для **AI Impact Assessment** – генерация отчёта по чек-листу (на основе шаблона NIA) с помощью LLM и сохранение его хэша в блокчейне для подтверждения целостности.

Кроме корейского примера, проект учитывает рекомендации ведущих международных организаций. **Государства ОЭСР и ЕС** акцентируют необходимость человеко-центричного и ответственного применения ИИ в публичном секторе. Например, европейское агентство ENISA выпустило в 2023 г. многоуровневую рамку кибербезопасности для ИИ, охватывающую как базовые меры для инфраструктуры, так и специфические требования к самим AI-моделям [6] [7]. В частности, подчеркивается, что организации должны внедрять динамичный процесс управления рисками ИИ на протяжении всего жизненного цикла – от концепции до вывода из эксплуатации [8]. Также рекомендуется соблюдать профильные стандарты безопасности (например, ISO/IEC 15408, 27001) и учитывать новые законодательные требования (в ЕС – проект *AI Act* и др.). Отдельно отмечена важность учёта **этических и социальных рисков** при внедрении AI: помимо технических уязвимостей, следует анализировать вопросы biais модели, прозрачности алгоритмов, возможности объяснить решения системы, а также потенциальные последствия для общества [9]. Для Agent Smith эти принципы превращаются в конкретные практики – подробнее в разделе 6 (безопасность и суверенитет).

Наконец, проект опирается на **открытые стандарты и фреймворки**. Выбирая технологический стек, команда сверялась с опытом крупных IT-компаний. Например, решение о развертывании собственного блокчейн-слоя для фиксации решений (GovChain) согласуется с концепцией децентрализованного управления: согласно White Paper «GovChain DAO», слияние DAO-подхода с AI и блокчейном делает процессы управления более гибкими, прозрачными и подотчётными, устраняя лишних посредников [10]. Неизменяемое хранение важных записей на блокчейне позволяет гарантировать их целостность: каждая финальная резолюция или ответ могут быть **анкоровны** (захешированы) в реестре, что предотвращает их задним числом изменение [11]. Кроме того, на мировом рынке появляются референсные реализации AI-агентов для документооборота: так, концепция GovChain предполагает связку LLM-агентов для расшифровки заседаний, суммирования дискуссий и автоматического составления проектов решений, которые затем проверяются на соответствие политикам и отправляются на голосование в DAO-смарт-контракты [12]. Эти идеи подтверждают жизнеспособность архитектурных решений, заложенных в Agent Smith.

4. Архитектура системы и проектирование AI-агентов

Общая архитектура. Platforma Agent Smith построена по многоуровневому принципу, сочетая гибкость облачных AI-сервисов с безопасностью on-premise решений. Пользователь (чиновник, анализирующий обращения, либо сам гражданин через веб-интерфейс) взаимодействует с системой через фронтенд или API, запрос поступает на внутренний **маршрутизатор LLM**, который направляет его в подходящую подсистему обработки. В зависимости от характера задачи запрос может обработать либо локальная языковая модель (развёрнутая в инфраструктуре госоргана), либо вызван внешний API крупного LLM-провайдера (OpenAI,

Anthropic, Google и т.д.) – выбор осуществляется на основе политики (учёт конфиденциальности данных, стоимости, требуемого качества). AI-агенты системы выполнены в виде микросервисов, каждый из которых отвечает за свой набор *скиллов* (навыков): анализ текста, поиск информации, общение в чат-формате, работа с изображениями (CV), взаимодействие с блокчейном и пр. Эти агенты могут работать как поодиночке, так и совместно, передавая задачи друг другу. Завершающий уровень – **слой GovChain**, представляющий собой приватный блокчейн (на базе Hyperledger Besu), в который агенты записывают результаты ключевых операций (решений) вместе с электронными подписями ответственных лиц. Ниже схематично перечислены основные компоненты архитектуры:

- **Маршрутизатор запросов LLM.** Компонент, принимающий входящий запрос и определяющий, какой движок обработки задействовать. В системе настроены несколько LLM-моделей: коммерческие GPT-4, Claude 3, Google Gemini Pro (доступные через API), а также открытые модели, размещённые локально (например, LLaMA 4 от Meta, Qwen, KazLM – казахстанская модель). Маршрутизатор учитывает конфиденциальность запроса, требуемую скорость и стоимость: **только избранные запросы с обезличенными данными отправляются во внешние API**, всё остальное обрабатывается локальными моделями (это обеспечивает суверенность данных). Данный компонент реализован как gRPC-сервис, вызывающий либо внешние HTTP API, либо локальный сервер vLLM, и возвращающий унифицированный ответ.
- **Агенты и инструменты.** Каждый агент – это отдельный микросервис (контейнер в Kubernetes), который может принимать на вход задачу и последовательностью действий достичь результата. Агент может вызывать сторонние *инструменты*: например, инструмент поиска по базе данных, инструмент выполнения кода (Python-скрипты для расчётов), доступ к внешнему веб-поиску или внутренним системам. Такая концепция схожа с подходом ReAct (Reason+Act), когда LLM последовательно формирует план действий и вызывает нужные функции [13] . В нашем случае, для разных типов обращений предусмотрены специализированные агенты:
- *Агент аналитики и резюмирования* – обрабатывает длинные тексты обращений, стенограммы совещаний, способен делать конспект, выявлять задачи.
- *Агент юридической классификации* – анализирует обращения на наличие признаков нарушений, классифицирует по темам законодательства, может рекомендовать типовой ответ или ответственного исполнителя (на основе предыдущих случаев).
- *Агент вопросов-ответов (Q&A)* – отвечает на запросы сотрудников по нормативным правовым актам, используя базу знаний (например, “каким законом регулируется такой-то вопрос?”).
- *Агент CV* – способен распознавать и анализировать вложения в обращениях (фото, видео) на наличие определённых объектов, лиц, ситуаций.
- *Агент Blockchain Ops* – взаимодействует с уровнем GovChain: публикует хэши документов, инициирует голосования (если требуется коллегиальное решение), проверяет статусы транзакций.

Агенты могут работать **параллельно и последовательно**. Например, при поступлении сложного обращения сначала срабатывает аналитический агент для резюме, затем юридический для классификации, затем BlockchainOps сохраняет результаты. Координация осуществляется с помощью очередей (Kafka) и orchestration-движка (например, нода *n8n* для определения условных маршрутов). Подход с разделением на несколько агентов предпочтителен, так как сложные процессы надёжнее выполнять по шагам, организуя цикл запросов между агентами, нежели пытаться заставить один монолитный LLM сделать всё за раз [14] . Такая *итеративная петля* с проверкой результатов на каждом шаге повышает точность и предсказуемость работы системы. - **Интеграционный слой и внешние интерфейсы.** Для взаимодействия с системами

eOtinish и другими государственными сервисами предусмотрен API-шлюз (Tyk) с набором REST/GraphQL эндпоинтов. Через них можно запрашивать, например, список новых обращений, отправлять сгенерированные ответы обратно в eOtinish, получать справочную информацию из внешних источников (реестры, классификаторы). За безопасность обмена отвечает Keycloak (OIDC провайдер) – все обращения через API аутентифицируются и авторизуются по ролям (гражданин, инспектор, администратор системы и пр.). Внутренняя коммуникация между микросервисами защищена протоколами TLS; для чувствительных данных используется шифрование на уровне полей (PGP) в хранилищах. - **Уровень GovChain.** Завершающим элементом является блокчейн-реестр, выполняющий роль журнала и механизма децентрализованного управления (DAO). Технология DAO (децентрализованной автономной организации) в контексте госуправления пока экспериментальна, однако в Agent Smith она применяется для следующих целей: (1) **Неизменяемый журнал** – каждая выдача ответа по обращению, каждая рекомендация AI фиксируются хэшем в блокчейне, что делает невозможным их незаметное исправление или удаление задним числом [15]; (2) **Коллективное принятие решений** – если решение по обращению требует согласования нескольких лиц или ведомств, можно инициировать голосование через смарт-контракт DAO (например, использование стандарта OpenZeppelin Governor для голосований); (3) **Прозрачность и доверие** – техническая возможность для внешних наблюдателей (например, общественного совета) проверять хэши записей позволяет убедиться, что предоставленные ответы соответствуют зафиксированным в блокчейне, повышая доверие к системе. Блокчейн-узлы размещены в защищённом контуре госорганов; консенсус достигается алгоритмом PoA между узлами Минцифры, Генпрокуратуры и др. Нефункционально, слой GovChain добавляет небольшую задержку (1-2 секунды на транзакцию), что приемлемо для задач документооборота.

Модели и технологии. Платформа является **мультимодельной**: одновременно используются несколько LLM с разными поставщиками и параметрами. Например, для простых задач ответ может генерировать отечественная модель KazLM или LLaMA4 (13B параметров) в локальном режиме, а для сложного юридического анализа – делегироваться на GPT-4. Такой подход позволяет балансировать качество и стоимость. Для интеграции открытых моделей применяются инструменты HuggingFace: через `huggingface_hub` загружаются обновления вроде LLaMA 4 (вышедшая в апреле 2025 г.) или специализированные модели (DeepSeek V3 для поиска, Yi/Zephyr для казахского языка и т.д.). Поддерживается подход *LoRA*-дообучений: свою узкоспециализированную модель можно обучить на доменных данных и подключить как новую ветку. Все модели регистрируются в **MLflow**, чтобы отслеживать версии, метрики качества и быстро переключаться между ними. Для выполнения inference больших моделей с максимальной производительностью используется сервер **vLLM** (оптимизация использования GPU памяти) в сочетании с NVIDIA TensorRT. Автономные агенты пишутся с использованием **LangChain** – популярного фреймворка, упрощающего реализацию шаблонов взаимодействия LLM с инструментами, хранения промежуточных состояний и т.п. Дополнительно рассматривается использование фреймворка **Haystack** для задач вопросно-ответных систем на больших датасетах.

В целом, архитектура Agent Smith соответствует лучшим практикам построения AI-систем: она модульна, масштабируема по нагрузке (благодаря микросервисам и контейнеризации), гибка к замене компонентов (можно обновлять модель или подключать новый инструмент без переделки всего приложения) и обеспечивает контроль на всех этапах обработки (через логи, блокчейн-журнал, мониторинг метрик). Далее рассматривается, как в этой архитектуре реализуется управление знаниями и памятью агентов.

5. Хранилище знаний, векторные базы и память агентов

Ограничения текущих LLM таковы, что они **не имеют длительной памяти из коробки** – модель не «помнит» прошлые обращения пользователя, если явно не предоставить ей этот контекст [16]. Для эффективной работы в сфере обращений граждан требуется, чтобы AI сохранял знания о предыдущих взаимодействиях, умел ссылаться на накопленную информацию (базу данных жалоб, нормативно-правовые акты, шаблоны ответов). В Agent Smith реализована многоуровневая модель памяти:

- **Краткосрочная память (контекст сессии).** При диалоговом взаимодействии (например, инспектор уточняет у чат-бота детали обращения) последние реплики хранятся и подаются модели вместе со следующим вопросом. Однако длина контекста у моделей ограничена (хотя новые модели позволяют >100k токенов, это не безгранично), поэтому используется подход свертывания диалогов: неключевая информация агрегируется или удаляется, сохраняются только сущности и важные факты. Также в короткосрочном контексте модель получает системные инструкции – правила форматирования ответа, политика конфиденциальности (нельзя раскрывать личные данные и пр.).
- **Долгосрочная память (семантическое хранилище).** Для хранения знаний, выходящих за рамки одного обращения, применяется векторная база данных (*Milvus*). Все релевантные документы и данные конвертируются в эмбеддинги – математические векторные представления, которые отражают смысл текста. Например, текст предыдущего похожего обращения, статьи закона, часто задаваемые вопросы – всё это хранится в виде векторов. Когда агенту нужно ответить на запрос, он сначала выполняет **поиск ближайших знаний**: на основе embedding запроса извлекаются топ-N похожих записей из базы. Эти подсказки добавляются в контекст модели, которая уже генерирует ответ с опорой на актуальные данные. Такой подход называется *Retrieval-Augmented Generation (RAG)* – генерация с дополнением из внешней базы знаний [17]. RAG существенно повышает фактологическую корректность ответов и позволяет системе быть в курсе обновлений без полной переобучения модели. В контексте Agent Smith это значит, что AI-агент всегда сможет найти, например, актуальную статью закона по теме обращения или посмотреть, как аналогичное обращение было решено ранее, прежде чем сформулировать ответ гражданину.
- **Хранилище профилей и прецедентов.** Отдельно ведётся база структурированных данных: профили заявителей, статистика по обращениям, решения по ним. Хотя основная работа AI идёт с текстами, доступ к структурированным данным позволяет улучшить персонализацию. К примеру, зная категорию заявителя (ветеран, многодетная мать и т.д.) или историю его обращений, система может точнее понять контекст нового обращения. Такие данные хранятся в реляционной БД (PostgreSQL) и по запросу могут извлекаться инструментами агентов (SQL-агент через LangChain). Также хранится массив метаданных по обращениям (темы, сроки, исполнители, исходы) – это поможет обучать модели на собственных данных и выявлять узкие места (например, часто ли определённый отдел просрочивает ответы).
- **Разделяемая память между агентами.** Если несколько агентов последовательно работают над одной задачей, им необходим единый взгляд на прогресс. Для этого внедрён механизм *общего контекста*: результат каждого шага (например, черновик ответа, список выделенных фактов) сохраняется либо в кэше Redis, либо сразу в векторном хранилище как новый «документ» с меткой текущего обращения. Другой агент, приступая к работе, делает запрос к памяти по ID обращения и получает все уже известные факты. Такая координация предотвращает ситуации, когда агенты повторяют работу друг друга или запрашивают уже найденную информацию повторно. Практика показывает, что на real-life

примерах, когда агентам явно указывают, какие инструменты/данные уже применены, они работают эффективнее [18].

Организация знаний. Наполнение базы знаний Agent Smith – это непрерывный процесс. На старте туда загрузят массив нормативных документов (Конституция, кодексы, типовые регламенты услуг и т.д.), а также исторические данные обращений за несколько последних лет (обезличенные). Далее, по мере работы системы, каждое новое обращение и сформированный по нему ответ будут добавляться в базу: таким образом, агент как бы учится на реальных кейсах. Предусмотрены задачи по *очистке и актуализации* знаний: устаревшие записи помечаются, противоречивые – выявляются и передаются специалисту для разрешения (человек может откорректировать или удалить). Для эффективного поиска по базе используются современные методы векторного семантического поиска, обладающие контекстностью (например, **Hybrid Search** – комбинация классического ключевого поиска и sematic search). Это значит, что агент найдёт нужный документ даже если запрос задан не буквально теми же словами, а синонимично.

Модель памяти AI-агентов спроектирована с учётом масштабируемости: Milvus способен хранить миллиард+ эмбеддингов, а его шардирование позволяет распределить нагрузку между серверами. Таким образом, по мере подключения новых источников данных (например, обращений из других систем, результатов мониторинга соцсетей и т.д.) архитектура хранения выдержит рост. В то же время реализована концепция *data locality* – чувствительные данные остаются внутри периметра организации. Если для ответа нужно обратиться к внешнему знанию (например, новость на сайте), агент делает это через прокси-сервис, фильтрующий результаты, чтобы не допустить утечки.

В итоге, сочетание краткосрочной и долгосрочной памяти позволяет Agent Smith выступать действительно интеллектуальным помощником, **помнящим контекст** и обладающим корпоративной памятью организации. Это соответствует лучшим практикам построения AI-ассистентов: добавление внешнего хранилища знаний устраняет «склероз» LLM и даёт системе долгосрочную устойчивость в эксплуатации.

6. Принципы безопасности, приватности и суверенитета

При внедрении AI-системы в государственном секторе критически важно соблюсти требования информационной безопасности, защиты персональных данных и технологического суверенитета. Архитектура Agent Smith изначально спроектирована с учётом этих принципов:

Безопасность данных и инфраструктуры. Базовый уровень безопасности (Layer I по классификации ENISA [19]) обеспечивает защиту ИТ-инфраструктуры, на которой работает система. Все компоненты Agent Smith развёрнуты в сертифицированном государственном ЦОД с необходимыми сетевыми экранирующими средствами. Внутренние коммуникации защищены шифрованием, конфиденциальные данные хранятся в зашифрованном виде. Проводится регулярный анализ уязвимостей и управление патчами. Согласно рекомендациям, внедрён **процесс управления рисками безопасности ИИ** – на этапе проектирования выполнен анализ угроз (например, перехват токенов API LLM, атаки внедрения подсказок prompt injection, утечки через ответы модели и др.). Для каждой категории рисков определены меры: ограничение прав сервисных аккаунтов, фильтрация пользовательского ввода, логирование и мониторинг аномалий. ENISA рекомендует двухэтапный процесс security management (анализ и управление рисками) как динамичный цикл на всём протяжении эксплуатации AI [8] – в проекте предусмотрено, что политика безопасности будет регулярно пересматриваться по мере появления новых угроз и по мере эволюции самой модели (например, при обновлении LLM до новой версии проводится повторное тестирование на уязвимости).

Особое внимание уделяется **API-вызовам внешних LLM**. Поскольку часть запросов может направляться во внешние облака (OpenAI, Anthropic и т.д.), действует строгая политика: персональные данные граждан **никогда не отправляются во внешние сервисы в открытом виде**. Перед передачей текста обращения во внешнюю модель все поля с персональными идентификаторами (ФИО, ИИН, адрес и т.д.) обезличиваются или заменяются токенами. Обратно полученный ответ проходит пост-обработку, где реальные данные подставляются на места токенов. Таким образом, внешние LLM «видят» только обезличенные обращения. Кроме того, с каждым облачным провайдером заключается соглашение (DPA) об обработке данных, гарантирующее неиспользование предоставленных данных в чужих целях. Для критичных же сценариев (например, обращение содержит гостайну или конфиденциальную информацию) система маршрутизирует запрос исключительно на локальную модель.

Приватность и соответствие законодательству. Agent Smith полностью соответствует закону РК «О персональных данных и их защите». Граждане информируются, что их обращение будет обработано автоматизированно; при необходимости реализованы механизмы отзыва согласия на автоматизированную обработку. Все персональные данные хранятся внутри страны. Реализован принцип минимизации данных: агент оперирует только той информацией, которая необходима для выполнения задачи. Например, модуль классификации обращений не видит личных данных заявителя – только текст обращения. Это снижает риск компрометации приватности при потенциальной утечке отдельных компонентов. При разработке также учтён международный опыт: ориентируемся на GDPR в части прав субъектов данных (право быть забытым – в системе предусмотрено полное удаление обращения по законному требованию, что включает удаление из векторной базы и блокчейн-записи с помощью механизма криптографического стирания). Логи доступа и действий агентов ведутся и могут быть аудитированы уполномоченными лицами, что создаёт дополнительный слой ответственности.

Суверенитет и независимость. Под суверенностью понимается способность государства автономно управлять системой без критической зависимости от иностранных компаний. В архитектуре Agent Smith этот принцип реализуется через максимальное использование **отечественных и открытых технологий**. Ключевые компоненты (база данных, векторное хранилище, blockchain) – с открытым исходным кодом, что позволит при необходимости аудит и модификацию под национальные стандарты. Модели LLM, обученные на языках РК, интегрированы наряду с зарубежными: например, KazLM, собранная на национальном датацентре, может обрабатывать обращения на государственном языке без отправки их за рубеж. Это также решает задачу локализации – иностранные модели не всегда хорошо знают контекст Казахстана (законы, географию и др.), тогда как местные модели, обученные на отечественных данных, восполняют этот пробел.

Суверенность включает и аппаратную составляющую: рассматривается возможность развёртывания части инфраструктуры на отечественном серверном оборудовании (произведённом в РК). Это не только стимулирует локальную индустрию, но и снижает риски закладок на уровне аппаратного обеспечения. Конечно, полностью избежать импорта технологий невозможно (те же GPU для обучения моделей – зарубежные), но принцип диверсификации поставщиков соблюдается.

Мониторинг и контроль качества AI. Поскольку Agent Smith будет участвовать в принятии решений, важна **надежность и беспристрастность** его работы. Встроенные метрики и мониторинг (система Prometheus+Grafana) отслеживают ключевые показатели: долю обращений, обработанных автоматически, среднее время ответа, процент ошибок или отклонённых агентом обращений (когда AI не уверен и передаёт человеку). Настраиваются алерты на аномалии –

например, если вдруг агент начинает чаще обычного давать сбои или ответы получают низкую оценку граждан, это будет сигналом для команды поддержки.

Также реализован механизм *«человек в цикле»*: на пилотном этапе все ответы, сгенерированные AI, будут проверяться ответственными сотрудниками перед отправкой заявителю. Автономность системы будет наращиваться постепенно, и на каждом этапе будут введены **ограничители**: например, агент не сможет сам окончательно ответить на обращение, относящееся к категории высокой важности (обращения к Президенту, массовые петиции и т.п.) – он лишь подготовит проект ответа для должностного лица. Такой подход соответствует мировым принципам осторожного внедрения AI в управление: автоматизируем рутинное, но оставляем человеку последнее слово в спорных или важных вопросах.

Надёжность и защита от ошибок. Будучи сложной распределённой системой, Agent Smith проектируется с запасом по отказоустойчивости. Кластеры Kubernetes распределены по двум зонам доступности, данные в PostgreSQL реплицируются, снапшоты важнейших баз (MinIO, Milvus) создаются ежедневно. Смарт-контракты в блокчейне прошли аудит на отсутствие уязвимостей (например, переполнения, возможности неавторизованных голосований). Если какой-то из внешних сервисов LLM недоступен, маршрутизатор автоматически переключится на альтернативный (пусть менее точный, но доступный) движок, чтобы система продолжала работу. Таким образом достигается **устойчивость к сбоям**.

В итоге, принципы безопасности, приватности и суверенитета в Agent Smith не являются второстепенным аспектом, а пронизывают всю архитектуру. Соблюдение этих принципов подкреплено как техническими решениями (шифрование, изоляция, блокчейн-аудит), так и организационными мерами (политики, регламенты взаимодействия человека и AI). Такая комбинация обеспечивает доверие к системе со стороны и госорганов, и общества, что критически важно для успешного внедрения. International best practices confirm this approach: the system is designed *secure-by-design* and *privacy-by-design*, aligning with the evolving regulatory landscape for AI.

7. Roadmap: этапы, вехи и контрольные точки

Для успешной реализации проекта Agent Smith разработана поэтапная дорожная карта. Каждый этап соответствует определённым целям и результатам, позволяющим поэтапно нарастить функциональность системы и охват пользователей. Ниже представлена краткая дорожная карта на ближайшие ~6 месяцев с указанием основных Milestones:

- **Этап 0: Проектирование и подготовка (март – апрель 2025).** В эти сроки были уточнены требования, собрана команда, разработана архитектура (см. выше), выполнены необходимые исследования. Результатом этапа стали: архитектурный документ, согласованный с ИТ-службами госорганов; прототипы основных модулей (agent routing, базовый AI-модель, каркас веб-интерфейса); план пилотного внедрения, одобренный руководством eOtinish. *Контрольные точки:* утверждение ТЗ, готовность тестового контура для пилота.
- **Этап 1: Пилотная интеграция с eOtinish (май – июнь 2025).** Фокус на базовой функциональности для обработки обращений. План спринтов:
- **Недели 1–2:** Настройка авторизации и доступа (SSO через Keycloak для пользователей пилота), запуск коннектора к API eOtinish для импорта обращений. К концу 2 недели система должна автоматически подтягивать новые обращения в внутреннюю БД и

размещать их на канбан-доске для инспекторов. *KPI:* время импорта обращения < 5 с, все участники пилота имеют доступ к интерфейсу.

- **Недели 3–4:** Внедрение базового AI-агента для анализа обращений. Модель должна уметь определять тему обращения, критичность, рекомендовать исполнителя. Подключение векторной базы знаний с начальными данными и реализация RAG при классификации. Также – прототип автозаполнения профиля заявителя (MyData-lite) на тестовом примере. *KPI:* точность тематической классификации > 80%, авто-подстановка хотя бы 3 полей в форме обращения, время автозаполнения < 1 с【12†】.
- **Недели 5–6:** Реализация мониторинга сроков (SLA-трекер). Настройка фонового процесса (cron-задача или поток) для проверки времени нахождения обращения в статусе. Разработка механизма эскалации: по событию просрочки триггерится уведомление в eOtinish или email ответственному. Параллельно – интеграция блока GovChain: развертывание приватного блокчейна, запись первых транзакций (например, хэша полученного обращения). *KPI:* ни одно обращение пилота не проходит без внимания после истечения SLA (нулевой уровень просрочек без уведомления), задержка между выявлением просрочки и нотификацией < 5 минут.
- **Недели 7–8:** Модуль обратной связи и дашборды. В интерфейс eOtinish внедряется виджет оценки ответа (звёзды и комментарий). Агент Smith собирает эти оценки, аггрегирует и публикует метрики для пилотных органов: средняя удовлетворённость, процент закрытых обращений с оценкой и т.д. Разворачивается публичный (для участников пилота) Grafana-дэшборд с этими показателями в реальном времени. *KPI:* >=70% обработанных пилотом обращений имеют выставленную оценку; дашборд обновляется онлайн при каждом новом ответе.
- **Недели 9–10:** Автоматизация отчётности по AI (AI Impact Assessment). Запуск агента, который по заданному шаблону генерирует PDF-отчёт о влиянии использования AI на основе данных пилота: фиксирует, не было ли инцидентов утечки, оценку корректности решений AI и прочие пункты этического чек-листа. Хэш отчёта сохраняется в блокчейне (Besu). *KPI:* формирование отчёта занимает < 60 с; отчёт проверен и подписан ответственным; смарт-контракт зафиксировал хэш документа.

- **Недели 11–12:** Итоговое улучшение и документация. На основе результатов пилота (метрик KPI, отзывов пользователей) дорабатываются модели и правила. Готовится **white-paper по проекту («KZ-GovFrame»)** – публичный документ, описывающий архитектуру, достигнутые показатели, сравнение с международными аналогами. Публикация white-paper на GitHub, рассылка заинтересованным сторонам. *KPI:* не менее 100 звёзд на репозитории GitHub (показатель интереса сообщества); прицельная презентация проекта для упоминания в отчётах NIA или профильных ведомств (попадание в поле зрения международных коллег).

- **Этап 2: Масштабирование и расширение (3–4 квартал 2025).** После успешного пилота планируется расширение охвата Agent Smith на большее число госорганов и типов обращений. Это потребует доработки некоторых модулей с учётом специфики новых данных. Вехи этого этапа: интеграция с другими системами (например, *e-Нотария* для обработки нотариальных запросов, *eLicense* для лицензий и т.д., используя накопленный опыт); обучение дополнительных доменных моделей (например, модель на казахском языке для соцблока). Одновременно будут усиливаться требования по отказоустойчивости – возможно, поднятие резервных копий системы в альтернативном датацентре. *Контрольные точки:* официальное принятие Agent Smith в эксплуатацию на уровне Минцифры; внесение системы в реестр госуслуг как подсистемы; подготовка методических рекомендаций для других ведомств по подключению к платформе.

- **Этап 3: Эксплуатация и автономизация (2026 г.).** На этом этапе Agent Smith станет штатным инструментом работы с обращениями по всей стране. Фокус сместится на оптимизацию и поддержку. Планируется достичь большей автономности агентов: поэтапно увеличивать долю обращений, которые AI обрабатывает самостоятельно, без ручной доработки. Параллельно будет расширяться функциональность – возможно, появятся **прогнозные функции** (анализ трендов обращений, предупреждение социальных рисков) и интеграция с проактивными сервисами (если AI выявил частую проблему, инициировать рассылку разъяснений гражданам и т.п.). Этот этап будет открывать новые горизонты применения Agent Smith за пределами обработки обращений – как платформы для внедрения ИИ-агентов и в других процессах госуправления.

Данный roadmap остаётся гибким: по мере прогресса пилота и появления новых технологий (например, выход более продвинутых моделей, новых версий фреймворков) план может корректироваться. Важнейшим механизмом контроля является **ретроспектива по Milestones**: после каждого крупного этапа команда и стейкхолдеры оценивают, достигнуты ли KPI, что можно улучшить, и только затем переходят к следующему этапу. Такой адаптивный подход обеспечит последовательное и контролируемое развитие системы, минимизируя риски срывов и неуспешного масштабирования.

8. Рекомендации по оргструктуре и пилотированию

Успешное внедрение Agent Smith зависит не только от технической реализации, но и от правильно выстроенной организационной структуры управления проектом и процесса пилотирования. Ниже представлены рекомендации, основанные на опыте аналогичных проектов и спецификой данной инициативы:

1. Межведомственная рабочая группа. Поскольку система затрагивает несколько государственных органов (разработчики, владельцы данных eOtinish, пользователи-исполнители из разных министерств, надзорные органы), целесообразно создать единую рабочую группу или координационный совет. В её состав должны войти: представитель Минцифры (владелец продукта от государства), представители ключевых ведомств-участников пилота (для обратной связи и учета требований), технический лидер от команды разработки (Bitmagic/QOSI), эксперт по безопасности, а также представитель гражданского общества (например, из офиса Омбудсмена, чтобы учитывать интересы заявителей). Такая группа будет регулярно собираться для обзора хода проекта, решения спорных вопросов и определения приоритетов. Практика показывает, что высокий уровень вовлечённости заказчика и конечных пользователей на этапе разработки существенно повышает шансы на успех проекта.

2. Команда разработки и эксплуатации. Рекомендуется чётко разделить роли внутри технической команды. Необходимы **ML-инженеры** (обучение и тонкая настройка моделей, мониторинг качества ответов), **backend-разработчики** (интеграция с eOtinish, разработка микросервисов агентов), **DevOps-инженеры** (поддержка инфраструктуры Kubernetes, CI/CD, обеспечение масштабируемости), **специалисты по безопасности** (пентесты, настройка IAM, анализ соответствия требованиям закона о данных). Кроме того, нужен **аналитик предметной области**, знакомый с процессами обработки обращений – он поможет правильно интерпретировать результаты, настроить бизнес-правила (например, маршрутизацию по ведомствам) и сформулировать данные для обучения модели. На этапе пилота важно также иметь **службу поддержки** (хотя бы 1–2 человека), которая будет оперативно помогать пользователям пилота, собирать от них проблемы и пожелания. В перспективе, по мере масштабирования, поддержкой может заняться штат ИТ-отделов самих госорганов, но на старте необходима централизованная команда.

3. Обучение и user onboarding. Даже лучшая AI-система не принесёт пользы, если пользователи не будут ей доверять или не понимать, как она работает. Поэтому параллельно с техническим внедрением следует организовать **обучающие сессии** для государственных служащих, участвующих в пилоте. Нужно объяснить, какие задачи решает Agent Smith, какие у него ограничения (например, может ошибаться и зачем нужна проверка), как интерпретировать результаты, что делать при нестандартных ситуациях. Полезно создать *FAQ для пользователей* и методичку. Кроме того, стоит внедрить механизм сбора обратной связи: например, в интерфейсе для инспектора добавить кнопку «Сообщить о неточности AI» – чтобы можно было быстро отметить случаи, где модель ошиблась или выдала неполезный ответ. Этот фидбек затем анализируется командой ML для доработки.

4. Постепенное наращивание автономности. В организационном плане важно определить *границы ответственности* AI и человека на каждом этапе. На первом пил# Структурированное исследование и архитектурное обоснование проекта Agent Smith

1. Введение: цели проекта и контекст

Agent Smith — это инициативный проект по созданию суверенной мульти-модельной AI-платформы для госсектора, предназначенной в первую очередь для **автоматизации обработки обращений граждан**. Система интегрируется с национальной платформой электронных обращений *eOtinish* и входит в экосистему инициатив GovTech. Главные цели проекта:

- Ускорить и упростить обработку жалоб и заявлений граждан за счет применения больших языковых моделей (LLM) и вспомогательных AI-агентов.
- Повысить прозрачность и подконтрольность рассмотрения обращений, фиксируя результаты с помощью технологий **блокчейна** и ЭЦП для обеспечения юридической силы ответов.
- Обеспечить **суверенность** решения – хранение данных и выполнение моделей на отечественной инфраструктуре, с минимальной зависимостью от внешних API, во избежание утечек данных и с учётом требований национальной безопасности.
- Продемонстрировать экономический эффект: по оценкам, каждый специализированный AI-агент может ежегодно экономить государству от 1 до 10 млн тенге за счёт автоматизации рутинных операций.

Контекст. В Казахстане наблюдается огромный поток обращений граждан: за 10 месяцев государственные органы получили свыше *3,2 млн обращений* (2,1 млн – центральные органы, 1,1 млн – на местном уровне) [1]. Это стимулировало поиск новых подходов, и в конце 2024 г. Генеральная прокуратура объявила о планах применять ИИ для анализа письменных обращений [2]. По словам Комитета по правовой статистике, накопленные данные системы eOtinish вместе с NLP-технологиями позволяют **автоматически выявлять ключевые проблемы, классифицировать риски по отраслям и оперативно доводить их до ответственных органов** [3]. Проект Agent Smith нацелен реализовать этот подход на практике, став частью Национальной AI-платформы, представленной в инициативе цифрового правительства Республики Казахстан [4].

Участники. Разработку ведёт консорциум компаний *Bitmagic* и *QOSI* при поддержке технопарка Astana Hub, в сотрудничестве с государственными структурами (Министерство цифрового развития, Генеральная прокуратура и др.). Платформа будет распространяться с открытым исходным кодом (лицензии MIT и CC-BY-4.0) для стимулирования партнёрства и аудита безопасности. Пользователями системы станут государственные органы (для внутреннего

ускорения обработки обращений), а в перспективе – и сами граждане, получающие более быстрый и качественный обратный ответ на свои обращения.

2. Обзор текущего состояния проекта

На основе актуальной **доски Trello** проекта можно проследить прогресс и текущие задачи Agent Smith. К настоящему моменту выполнены следующие ключевые этапы:

- **Формирование концепции и архитектуры.** Подготовлено описание общей архитектуры системы, включающее интеграцию с eOtinish, модульную структуру AI-агентов, а также использование блокчейн-слоя (*GovChain*) для фиксации действий. Определён стек технологий: среди прочего, выбраны платформы *LangChain* и *CrewAI* для оркестрации агентов, *MLflow* для управления моделями, *Hyperledger Besu* для DAO-блокчейна, *Milvus* для векторной БД, *MinIO* и *PostgreSQL* для хранения данных, *Keycloak* для единой авторизации, *Prometheus* для мониторинга, *Tyk* в качестве API-шлюза и пр.
- **Интеграция с eOtinish (MVP).** Реализован автономный модуль Agent Smith, способный подключаться к API системы eOtinish и импортировать обращения в свою базу. Настроен начальный **конвейер обработки обращения**: классификация тематики, извлечение ключевых сущностей, резюмирование содержания обращения. Разработан прототип пользовательского интерфейса (веб-приложение) с Kanban-доской для визуализации статусов обращений (входящие, в работе, выполненные, отклоненные).
- **Документация и аналитика.** Подготовлен краткий план пилотного проекта на базе eOtinish и развернута **wiki-платформа знаний** проекта. Составлен подробный отчёт по модулю «Обращения граждан» (этап пилота eOtinish) с описанием бизнес-процессов, метрик SLA и предложенными показателями эффективности пилота. Также выполнены исследования и отчёты по смежным темам: анализ опыта Республики Корея (Government24, e-People, eGovFrame) и других стран, обзор отечественных производителей серверного оборудования (для развёртывания системы on-premise), сбор лучших мировых практик по AI-агентам и технологиям (OpenAI, Anthropic, и др.). Эти материалы легли в основу требований и рекомендаций, описанных в следующих разделах.
- **Организационные шаги.** Проведены переговоры и достигнуто соглашение о сотрудничестве с операторами платформы eOtinish для запуска **совместного пилота**. Инициатива получила одобрение профильных руководителей, определены ответственные лица за пилот от каждого ведомства. Также налажены партнёрства с провайдерами **no-code** решений, чтобы в будущем облегчить интеграцию Agent Smith с другими государственными системами без существенной доработки кода.

Текущие задачи и фокус. Проект находится на стадии реализации первого пилотного запуска. Команда сосредоточена на доработке ключевых компонентов для пилота eOtinish: настройке единой авторизации через SSO, обеспечении автоматического подтягивания новых обращений в систему в режиме реального времени, доработке UI/UX для инспекторов и руководителей. Проводится углублённое **сравнительное исследование** лучших практик (подробнее в разделе 3) с целью включения дополнительных функций в следующую итерацию – таких как автозаполнение данных заявителя, отслеживание таймеров SLA с автоматической эскалацией просроченных обращений, виджеты оценки удовлетворённости ответом и публичные панели KPI. Параллельно ведётся работа над **базой знаний** для агентов: оцифровываются нормативные акты, накопленные данные предыдущих обращений, готовится загрузка этих данных в векторное хранилище для реализации механизмов RAG.

В ближайших планах – завершение Sprint 1 (авторизация, импорт обращений, канбан), после чего начнётся расширение пилота на группу реальных пользователей и отслеживание метрик. Контрольные точки и дальнейшие этапы развития подробно изложены в разделе 7 (Roadmap).

3. Международные стандарты и лучшие практики

При проектировании Agent Smith учитываются передовые международные наработки в сфере электронного правительства, искусственного интеллекта и кибербезопасности. Ниже приведён анализ ключевых идей из зарубежных источников, которые легли в основу требований к системе.

Опыт цифрового правительства Республики Корея. Система работы с обращениями граждан в Южной Корее считается «золотым стандартом». Единый портал **Government24** предоставляет гражданам единое окно доступа приблизительно к *12 000* государственным услугам и справкам [5], реализуя принцип «заполняю данные один раз – получаю услуги повсюду». Система аутентификации включает государственную PKI и мобильные ID (например, PASS), а данные пользователя подтягиваются автоматически через механизм *Public MyData* – только в 2024 г. в Корее было обработано ~560 млн транзакций автозаполнения данных【12†】. После подачи обращения через портал, оно попадает в систему **e-People** – централизованную платформу рассмотрения петиций граждан. В e-People реализована интеллектуальная маршрутизация: обращение автоматически классифицируется по теме и направляется в ответственное ведомство (из ~49 министерств) или сразу в несколько, если вопрос межведомственный. При межведомственных обращениях система создаёт *совместную задачу* с указанием ответственных от каждого органа и контролем сроков【12†】. Гражданин может в режиме онлайн отслеживать статус рассмотрения: в личном кабинете показывается таймер SLA и каждое действие чиновников по его обращению. Если сроки нарушены, e-People автоматически эскалирует обращение в надзорный орган (омбудсмен)【12†】. По результатам рассмотрения заявителю приходит ответ через портал, после чего система предлагает оценить удовлетворённость (по шкале 1–5 звёзд) и оставить отзыв; агрегированные рейтинги публикуются открыто, формируя **показатель удовлетворённости** граждан. Важная черта корейской платформы – **открытость кода**: back-end e-People построен на фреймворке **eGovFrame** (стек Spring + ~30 общих компонентов: SSO, файловый сервис, логирование, workflow и др.), который находится в открытом доступе и поддерживается государственным агентством NIA. Такой подход («open-source by default») повышает доверие и позволяет каждому ведомству запускать новые цифровые сервисы, используя уже готовые общие модули, что ускоряет разработку примерно на 60%.

Опираясь на корейский опыт, для Agent Smith были выделены пробелы (*gap-анализ*) и потенциальные улучшения:

- **Единая точка входа.** В Казахстане портал eOtinish пока является одним из каналов (помимо писем, личных приемов и др.). Желательно движение к модели «single window», когда гражданин подает обращение через единый интерфейс (портал или приложение), откуда информация далее распределяется. В рамках пилота Agent Smith планируется реализовать *виджет обращения* на существующем eGov-портале, чтобы пользователю не пришлось переходить в отдельную систему.
- **Автозаполнение данных (MyData).** Отсутствие механизма авто-подстановки персональных данных и известной информации в форму обращения приводит к лишним усилиям граждан и ошибкам. Планируется прототип интеграции с государственными реестрами (аналог корейского MyData): при авторизации заявителя основные сведения (ИИН, ФИО, контакты, адрес) будут автоматически заполняться в черновике обращения.

Для реализации рассматривается безопасное хранилище профилей граждан (возможно, на базе IPFS + блокчейн-ссылок) либо прямое обращение к API «личного кабинета».

- **Прозрачные SLA и эскалации.** Агент Smith возьмёт на контроль сроки выполнения по каждому обращению. На этапе пилота внедряется визуальный индикатор статуса (например, зелёный/жёлтый/красный в зависимости от близости дедлайна) для ответственных исполнителей. При нарушении срока система сможет автоматически уведомлять руководителей или надзорный орган. Будет реализован webhook или отдельная очередь событий `sla.breached` для сбора статистики по просроченным обращениям.
- **Обратная связь и открытые данные.** По завершении обращения планируется запрашивать у гражданина оценку ответа. Эти данные послужат для внутренней аналитики, а в агрегированном виде могут публиковаться (например, процент удовлетворённых ответом по ведомствам). Практика Кореи показывает, что публичность таких метрик стимулирует улучшение качества работы. В пилотной версии Agent Smith будет встроен модуль опроса и сформирован дашборд удовлетворённости на базе Grafana, доступный руководству.
- **Этическая экспертиза AI.** Южная Корея ввела практику обязательной оценки влияния новых цифровых услуг на персональные данные и этику (AI Impact Assessment), результаты которой публикуются открыто. Для Agent Smith также важно проведение такой экспертизы: анализ рисков обработки обращений AI-моделями (конфиденциальность, недопустимость дискриминации, объяснимость решений и пр.). В план работ включена задача разработки полуавтоматического инструмента для **AI Impact Assessment** – генерация отчёта по чек-листу (на основе шаблона NIA) с помощью LLM и сохранение его хэша в блокчейне для подтверждения целостности.

Кроме корейского примера, проект учитывает рекомендации ведущих международных организаций. **Государства ОЭСР и ЕС** акцентируют необходимость человеко-центричного и ответственного применения ИИ в публичном секторе. Например, европейское агентство ENISA выпустило в 2023 г. многоуровневую рамку кибербезопасности для ИИ, охватывающую как базовые меры для инфраструктуры, так и специфические требования к самим AI-моделям [6] [7]. В частности, подчеркивается, что организации должны внедрять динамичный процесс управления рисками ИИ на протяжении всего жизненного цикла – от концепции до вывода из эксплуатации AI [8]. Также рекомендуется соблюдать профильные стандарты безопасности (например, ISO/IEC 15408, 27001) и учитывать новые законодательные требования (в ЕС – проект *AI Act* и др.). Отдельно отмечена важность учёта **этических и социальных рисков** при внедрении AI: помимо технических уязвимостей, следует анализировать вопросы.bias модели, прозрачности алгоритмов, возможности объяснить решения системы, а также потенциальные последствия для общества [9]. Для Agent Smith эти принципы превращаются в конкретные практики – подробнее в разделе 6 (безопасность и суверенитет).

Наконец, проект опирается на **открытые стандарты и фреймворки**. Выбирая технологический стек, команда сверялась с опытом крупных IT-компаний. Например, решение о развертывании собственного блокчейн-слоя для фиксации решений (GovChain) согласуется с концепцией децентрализованного управления: согласно White Paper «GovChain DAO», слияние DAO-подхода с AI и блокчейном делает процессы управления более гибкими, прозрачными и подотчётными, устраняя лишних посредников [10]. Неизменяемое хранение важных записей на блокчейне позволяет гарантировать их целостность: каждая финальная резолюция или ответ могут быть **анкоровны** (захешированы) в реестре, что предотвращает их задним числом изменение [11]. Кроме того, на мировом рынке появляются референсные реализации AI-агентов для документооборота: так, концепция GovChain предполагает связку LLM-агентов для расшифровки заседаний, суммирования дискуссий и автоматического составления проектов решений, которые

затем проверяются на соответствие политикам и отправляются на голосование в DAO-смарт-контракты [12]. Эти идеи подтверждают жизнеспособность архитектурных решений, заложенных в Agent Smith.

4. Архитектура системы и проектирование AI-агентов

Общая архитектура. Платформа Agent Smith построена по многоуровневому принципу, сочетая гибкость облачных AI-сервисов с безопасностью on-premise решений. Пользователь (чиновник, анализирующий обращения, либо сам гражданин через веб-интерфейс) взаимодействует с системой через фронтенд или API, запрос поступает на внутренний **маршрутизатор LLM**, который направляет его в подходящую подсистему обработки. В зависимости от характера задачи запрос может обработать либо локальная языковая модель (развёрнутая в инфраструктуре госоргана), либо вызван внешний API крупного LLM-провайдера (OpenAI, Anthropic, Google и т.д.) – выбор осуществляется на основе политики (учёт конфиденциальности данных, требуемой скорости и стоимости). AI-агенты системы выполнены в виде микросервисов, каждый из которых отвечает за свой набор *скиллов* (навыков): анализ текста, поиск информации, общение в чат-формате, работа с изображениями (CV), взаимодействие с блокчейном и пр. Эти агенты могут работать как поодиночке, так и совместно, передавая задачи друг другу. Завершающий уровень – **слой GovChain**, представляющий собой приватный блокчейн (на базе Hyperledger Besu), в который агенты записывают результаты ключевых операций (решений) вместе с электронными подписями ответственных лиц. Ниже схематично перечислены основные компоненты архитектуры:

- **Маршрутизатор запросов LLM.** Компонент, принимающий входящий запрос и определяющий, какой движок обработки задействовать. В системе настроены несколько LLM-моделей: коммерческие GPT-4, Claude 3, Google Gemini Pro (доступные через API), а также открытые модели, размещённые локально (например, LLaMA 4 от Meta, Qwen, KazLM – казахстанская модель). Маршрутизатор учитывает конфиденциальность запроса, требуемую скорость и стоимость: **только избранные запросы с обезличенными данными отправляются во внешние API**, всё остальное обрабатывается локальными моделями (это обеспечивает суверенность данных). Данный компонент реализован как gRPC-сервис, вызывающий либо внешние HTTP API, либо локальный сервер vLLM, и возвращающий унифицированный ответ.
- **Агенты и инструменты.** Каждый агент – это отдельный микросервис (контейнер в Kubernetes), который может принимать на вход задачу и последовательностью действий достичь результата. Агент может вызывать сторонние *инструменты*: например, инструмент поиска по базе данных, инструмент выполнения кода (Python-скрипты для расчётов), доступ к внешнему веб-поиску или внутренним системам. Такая концепция схожа с подходом ReAct (Reason+Act), когда LLM последовательно формирует план действий и вызывает нужные функции [13]. В нашем случае, для разных типов обращений предусмотрены специализированные агенты:
- *Агент аналитики и резюмирования* – обрабатывает длинные тексты обращений, стенограммы совещаний, способен делать конспект, выявлять задачи.
- *Агент юридической классификации* – анализирует обращения на наличие признаков нарушений, классифицирует по темам законодательства, может рекомендовать типовой ответ или ответственного исполнителя (на основе предыдущих случаев).
- *Агент вопросов-ответов (Q&A)* – отвечает на запросы сотрудников по нормативным правовым актам, используя базу знаний (например, “каким законом регулируется такой-то вопрос?”).
- *Агент CV* – способен распознавать и анализировать вложения в обращениях (фото, видео) на наличие определённых объектов, лиц, ситуаций.

- *Агент Blockchain Ops* – взаимодействует с уровнем GovChain: публикует хэши документов, инициирует голосования (если требуется коллегиальное решение), проверяет статусы транзакций.

Агенты могут работать **параллельно и последовательно**. Например, при поступлении сложного обращения сначала срабатывает аналитический агент для резюме, затем юридический для классификации, затем BlockchainOps сохраняет результаты. Координация осуществляется с помощью очередей (Kafka) и orchestration-движка (например, нода *n8n* для определения условных маршрутов). Подход с разделением на несколько агентов предпочтителен, так как сложные процессы надёжнее выполнять по шагам, организуя цикл запросов между агентами, нежели пытаться заставить один монолитный LLM сделать всё за раз [14]. Такая *итеративная петля* с проверкой результатов на каждом шаге повышает точность и предсказуемость работы системы. - **Интеграционный слой и внешние интерфейсы.** Для взаимодействия с системами eOtinish и другими государственными сервисами предусмотрен API-шлюз (Tyk) с набором REST/GraphQL эндпоинтов. Через них можно запрашивать, например, список новых обращений, отправлять сгенерированные ответы обратно в eOtinish, получать справочную информацию из внешних источников (реестры, классификаторы). За безопасность обмена отвечает Keycloak (OIDC провайдер) – все обращения через API аутентифицируются и авторизуются по ролям (гражданин, инспектор, администратор системы и пр.). Внутренняя коммуникация между микросервисами защищена протоколами TLS; для чувствительных данных используется шифрование на уровне полей (PGP) в хранилищах. - **Уровень GovChain.** Завершающим элементом является блокчейн-реестр, выполняющий роль журнала и механизма децентрализованного управления (DAO). Технология DAO (децентрализованной автономной организации) в контексте госуправления пока экспериментальна, однако в Agent Smith она применяется для следующих целей: (1) **Неизменяемый журнал** – каждая выдача ответа по обращению, каждая рекомендация AI фиксируются хэшем в блокчейне, что делает невозможным их незаметное исправление или удаление задним числом [15]; (2) **Коллективное принятие решений** – если решение по обращению требует согласования нескольких лиц или ведомств, можно инициировать голосование через смарт-контракт DAO (например, использование стандарта OpenZeppelin Governor для голосований); (3) **Прозрачность и доверие** – техническая возможность для внешних наблюдателей (например, общественного совета) проверять хэши записей позволяет убедиться, что предоставленные ответы соответствуют зафиксированным в блокчейне, повышая доверие к системе. Блокчейн-узлы размещены в защищённом контуре госорганов; консенсус достигается алгоритмом PoA между узлами Минцифры, Генпрокуратуры и др. Нефункционально, слой GovChain добавляет небольшую задержку (1-2 секунды на транзакцию), что приемлемо для задач документооборота.

Модели и технологии. Платформа является **мультимодельной**: одновременно используются несколько LLM с разными поставщиками и параметрами. Например, для простых задач ответ может генерировать отечественная модель KazLM или LLaMA4 (13B параметров) в локальном режиме, а для сложного юридического анализа – делегироваться на GPT-4. Такой подход позволяет балансировать качество и стоимость. Для интеграции открытых моделей применяются инструменты HuggingFace: через `huggingface_hub` загружаются обновления вроде LLaMA 4 (вышедшая в апреле 2025 г.) или специализированные модели (DeepSeek V3 для поиска, Yi/Zephyr для казахского языка и т.д.). Поддерживается подход *LoRA*-дообучений: свою узкоспециализированную модель можно обучить на доменных данных и подключить как новую ветку. Все модели регистрируются в **MLflow**, чтобы отслеживать версии, метрики качества и быстро переключаться между ними. Для выполнения inference больших моделей с максимальной производительностью используется сервер **vLLM** (оптимизация использования GPU памяти) в сочетании с NVIDIA TensorRT. Автономные агенты пишутся с использованием **LangChain** – популярного фреймворка, упрощающего реализацию шаблонов взаимодействия LLM с

инструментами, хранения промежуточных состояний и т.п. Дополнительно рассматривается использование фреймворка **Haystack** для задач вопросно-ответных систем на больших датасетах.

В целом, архитектура Agent Smith соответствует лучшим практикам построения AI-систем: она модульна, масштабируема по нагрузке (благодаря микросервисам и контейнеризации), гибка к замене компонентов (можно обновлять модель или подключать новый инструмент без переделки всего приложения) и обеспечивает контроль на всех этапах обработки (через логи, блокчейн-журнал, мониторинг метрик). Далее рассматривается, как в этой архитектуре реализуется управление знаниями и памятью агентов.

5. Хранилище знаний, векторные базы и память агентов

Ограничения текущих LLM таковы, что они **не имеют длительной памяти из коробки** – модель не «помнит» прошлые обращения пользователя, если явно не предоставить ей этот контекст [16]. Для эффективной работы в сфере обращений граждан требуется, чтобы AI сохранял знания о предыдущих взаимодействиях, умел ссылаться на накопленную информацию (базу данных жалоб, нормативно-правовые акты, шаблоны ответов). В Agent Smith реализована многоуровневая модель памяти:

- **Краткосрочная память (контекст сессии).** При диалоговом взаимодействии (например, инспектор уточняет у чат-бота детали обращения) последние реплики хранятся и подаются модели вместе со следующим вопросом. Однако длина контекста у моделей ограничена (хотя новые модели позволяют >100k токенов, это не безгранично), поэтому используется подход свертывания диалогов: неключевая информация агрегируется или удаляется, сохраняются только сущности и важные факты. Также в короткосрочном контексте модель получает системные инструкции – правила форматирования ответа, политика конфиденциальности (нельзя раскрывать личные данные и пр.).
- **Долгосрочная память (семантическое хранилище).** Для хранения знаний, выходящих за рамки одного обращения, применяется векторная база данных (*Milvus*). Все релевантные документы и данные конвертируются в эмбеддинги – математические векторные представления, которые отражают смысл текста. Например, текст предыдущего похожего обращения, статьи закона, часто задаваемые вопросы – всё это хранится в виде векторов. Когда агенту нужно ответить на запрос, он сначала выполняет **поиск ближайших знаний**: на основе embedding запроса извлекаются топ-N похожих записей из базы. Эти подсказки добавляются в контекст модели, которая уже генерирует ответ с опорой на актуальные данные. Такой подход называется *Retrieval-Augmented Generation (RAG)* – генерация с дополнением из внешней базы знаний [17]. RAG существенно повышает фактологическую корректность ответов и позволяет системе быть в курсе обновлений без полной переобучения модели. В контексте Agent Smith это значит, что AI-агент всегда сможет найти, например, актуальную статью закона по теме обращения или посмотреть, как аналогичное обращение было решено ранее, прежде чем сформулировать ответ гражданину.
- **Хранилище профилей и прецедентов.** Отдельно ведётся база структурированных данных: профили заявителей, статистика по обращениям, решения по ним. Хотя основная работа AI идёт с текстами, доступ к структурированным данным позволяет улучшить персонализацию. К примеру, зная категорию заявителя (ветеран, многодетная мать и т.д.) или историю его обращений, система может точнее понять контекст нового обращения. Такие данные хранятся в реляционной БД (PostgreSQL) и по запросу могут извлекаться инструментами агентов (SQL-агент через LangChain). Также хранится массив метаданных по обращениям (темы, сроки, исполнители, исходы) – это поможет обучать модели на

собственных данных и выявлять узкие места (например, часто ли определённый отдел просрочивает ответы).

- **Разделяемая память между агентами.** Если несколько агентов последовательно работают над одной задачей, им необходим единый взгляд на прогресс. Для этого внедрён механизм *общего контекста*: результат каждого шага (например, черновик ответа, список выделенных фактов) сохраняется либо в кэше Redis, либо сразу в векторном хранилище как новый «документ» с меткой текущего обращения. Другой агент, приступая к работе, делает запрос к памяти по ID обращения и получает все уже известные факты. Такая координация предотвращает ситуации, когда агенты повторяют работу друг друга или запрашивают уже найденную информацию повторно. Практика показывает, что на real-life примерах, когда агентам явно указывают, какие инструменты/данные уже применены, они работают эффективнее [18].

Организация знаний. Наполнение базы знаний Agent Smith – это непрерывный процесс. На старте туда загрузят массив нормативных документов (Конституция, кодексы, типовые регламенты услуг и т.д.), а также исторические данные обращений за несколько последних лет (обезличенные). Далее, по мере работы системы, каждое новое обращение и сформированный по нему ответ будут добавляться в базу: таким образом, агент как бы учится на реальных кейсах. Предусмотрены задачи по *очистке и актуализации* знаний: устаревшие записи помечаются, противоречивые – выявляются и передаются специалисту для разрешения (человек может откорректировать или удалить). Для эффективного поиска по базе используются современные методы векторного семантического поиска, обладающие контекстностью (например, **Hybrid Search** – комбинация классического ключевого поиска и sematic search). Это значит, что агент найдёт нужный документ даже если запрос задан не буквально теми же словами, а синонимично.

Модель памяти AI-агентов спроектирована с учётом масштабируемости: Milvus способен хранить миллиард+ эмбеддингов, а его шардирование позволяет распределить нагрузку между серверами. Таким образом, по мере подключения новых источников данных (например, обращений из других систем, результатов мониторинга соцсетей и т.д.) архитектура хранения выдержит рост. В то же время реализована концепция *data locality* – чувствительные данные остаются внутри периметра организации. Если для ответа нужно обратиться к внешнему знанию (например, новость на сайте), агент делает это через прокси-сервис, фильтрующий результаты, чтобы не допустить утечки.

В итоге, сочетание краткосрочной и долгосрочной памяти позволяет Agent Smith выступать действительно интеллектуальным помощником, **помнящим контекст** и обладающим корпоративной памятью организации. Это соответствует лучшим практикам построения AI-ассистентов: добавление внешнего хранилища знаний устраняет «склероз» LLM и даёт системе долгосрочную устойчивость в эксплуатации.

6. Принципы безопасности, приватности и суверенитета

При внедрении AI-системы в государственном секторе критически важно соблюсти требования информационной безопасности, защиты персональных данных и технологического суверенитета. Архитектура Agent Smith изначально спроектирована с учётом этих принципов:

Безопасность данных и инфраструктуры. Базовый уровень безопасности (Layer I по классификации ENISA [19]) обеспечивает защиту ИТ-инфраструктуры, на которой работает система. Все компоненты Agent Smith развёрнуты в сертифицированном государственном ЦОД с необходимыми сетевыми экранирующими средствами. Внутренние коммуникации защищены шифрованием, конфиденциальные данные хранятся в зашифрованном виде. Проводится

регулярный анализ уязвимостей и управление патчами. Согласно рекомендациям, внедрён **процесс управления рисками безопасности ИИ** – на этапе проектирования выполнен анализ угроз (например, перехват токенов API LLM, атаки внедрения подсказок prompt injection, утечки через ответы модели и др.). Для каждой категории рисков определены меры: ограничение прав сервисных аккаунтов, фильтрация пользовательского ввода, логирование и мониторинг аномалий. ENISA рекомендует двухэтапный процесс security management (анализ и управление рисками) как динамичный цикл на всём протяжении эксплуатации AI [8] – в проекте предусмотрено, что политика безопасности будет регулярно пересматриваться по мере появления новых угроз и по мере эволюции самой модели (например, при обновлении LLM до новой версии проводится повторное тестирование на уязвимости).

Особое внимание уделяется **API-вызовам внешних LLM**. Поскольку часть запросов может направляться во внешние облака (OpenAI, Anthropic и т.д.), действует строгая политика: персональные данные граждан **никогда не отправляются во внешние сервисы в открытом виде**. Перед передачей текста обращения во внешнюю модель все поля с персональными идентификаторами (ФИО, ИИН, адрес и т.д.) обезличиваются или заменяются токенами. Обратно полученный ответ проходит пост-обработку, где реальные данные подставляются на места токенов. Таким образом, внешние LLM «видят» только обезличенные обращения. Кроме того, с каждым облачным провайдером заключается соглашение (DPA) об обработке данных, гарантирующее неиспользование предоставленных данных в чужих целях. Для критичных же сценариев (например, обращение содержит гостайну или конфиденциальную информацию) система маршрутизирует запрос исключительно на локальную модель.

Приватность и соответствие законодательству. Agent Smith полностью соответствует закону РК «О персональных данных и их защите». Граждане информируются, что их обращение будет обработано автоматизированно; при необходимости реализованы механизмы отзыва согласия на автоматизированную обработку. Все персональные данные хранятся внутри страны. Реализован принцип минимизации данных: агент оперирует только той информацией, которая необходима для выполнения задачи. Например, модуль классификации обращений не видит личных данных заявителя – только текст обращения. Это снижает риск компрометации приватности при потенциальной утечке отдельных компонентов. При разработке также учтён международный опыт: ориентируемся на GDPR в части прав субъектов данных (право быть забытым – в системе предусмотрено полное удаление обращения по законному требованию, что включает удаление из векторной базы и блокчейн-записи с помощью механизма криптографического стирания). Логи доступа и действий агентов ведутся и могут быть аудитированы уполномоченными лицами, что создаёт дополнительный слой ответственности.

Суверенитет и независимость. Под суверенностью понимается способность государства автономно управлять системой без критической зависимости от иностранных компаний. В архитектуре Agent Smith этот принцип реализуется через максимальное использование **отечественных и открытых технологий**. Ключевые компоненты (база данных, векторное хранилище, blockchain) – с открытым исходным кодом, что позволит при необходимости аудит и модификацию под национальные стандарты. Модели LLM, обученные на языках РК, интегрированы наряду с зарубежными: например, KazLM, собранная на национальном датацентре, может обрабатывать обращения на государственном языке без отправки их за рубеж. Это также решает задачу локализации – иностранные модели не всегда хорошо знают контекст Казахстана (законы, географию и др.), тогда как местные модели, обученные на отечественных данных, восполняют этот пробел.

Суверенность включает и аппаратную составляющую: рассматривается возможность развёртывания части инфраструктуры на отечественном серверном оборудовании

(произведённом в РК). Это не только стимулирует локальную индустрию, но и снижает риски закладок на уровне аппаратного обеспечения. Конечно, полностью избежать импорта технологий невозможно (те же GPU для обучения моделей – зарубежные), но принцип диверсификации поставщиков соблюдается.

Мониторинг и контроль качества AI. Поскольку Agent Smith будет участвовать в принятии решений, важна **надежность и беспристрастность** его работы. Встроенные метрики и мониторинг (система Prometheus+Grafana) отслеживают ключевые показатели: долю обращений, обработанных автоматически, среднее время ответа, процент ошибок или отклонённых агентом обращений (когда AI не уверен и передаёт человеку). Настраиваются алерты на аномалии – например, если вдруг агент начинает чаще обычного давать сбои или ответы получают низкую оценку граждан, это будет сигналом для команды поддержки.

Также реализован механизм *«человек в цикле»*: на пилотном этапе все ответы, сгенерированные AI, будут проверяться ответственными сотрудниками перед отправкой заявителю. Автономность системы будет наращиваться постепенно, и на каждом этапе будут введены **ограничители**: например, агент не сможет сам окончательно ответить на обращение, относящееся к категории высокой важности (обращения к Президенту, массовые петиции и т.п.) – он лишь подготовит проект ответа для должностного лица. Такой подход соответствует мировым принципам осторожного внедрения AI в управление: автоматизируем рутинное, но оставляем человеку последнее слово в спорных или важных вопросах.

Надёжность и защита от ошибок. Будучи сложной распределённой системой, Agent Smith проектируется с запасом по отказоустойчивости. Кластеры Kubernetes распределены по двум зонам доступности, данные в PostgreSQL реплицируются, снапшоты важнейших баз (MinIO, Milvus) создаются ежедневно. Смарт-контракты в блокчейне прошли аудит на отсутствие уязвимостей (например, переполнения, возможности неавторизованных голосований). Если какой-то из внешних сервисов LLM недоступен, маршрутизатор автоматически переключится на альтернативный (пусть менее точный, но доступный) движок, чтобы система продолжала работу. Таким образом достигается **устойчивость к сбоям**.

В итоге, принципы безопасности, приватности и суверенитета в Agent Smith не являются второстепенным аспектом, а пронизывают всю архитектуру. Соблюдение этих принципов подкреплено как техническими решениями (шифрование, изоляция, блокчейн-аудит), так и организационными мерами (политики, регламенты взаимодействия человека и AI). Такая комбинация обеспечивает доверие к системе со стороны и госорганов, что критически важно для успешного внедрения.

7. Roadmap: этапы, вехи и контрольные точки

Для успешной реализации проекта Agent Smith разработана поэтапная дорожная карта. Каждый этап соответствует определённым целям и результатам, позволяющим поэтапно нарастить функциональность системы и охват пользователей. Ниже представлена краткая дорожная карта на ближайшие ~6 месяцев с указанием основных Milestones:

- **Этап 0: Проектирование и подготовка (март – апрель 2025).** Уточнение требований, сбор команды, разработка архитектуры (см. выше), необходимые исследования.
  Результаты: архитектурный документ, согласованный с ИТ-службами госорганов; прототипы основных модулей (маршрутизатор, базовая AI-модель, каркас веб-

интерфейса); план пилотного внедрения, одобренный руководством eOtinish. *Контрольные точки:* утверждение ТЗ, готовность тестового контура для пилота.

- **Этап 1: Пилотная интеграция с eOtinish (май – июнь 2025).** Фокус на базовой функциональности для обработки обращений. **План спринтов:**
- *Недели 1–2:* Настройка авторизации и доступа (SSO через Keycloak для пользователей пилота), запуск коннектора к API eOtinish для импорта обращений. К концу 2-й недели система должна автоматически подтягивать новые обращения во внутреннюю БД и отображать их на канбан-доске для инспекторов. *KPI:* время импорта обращения < 5 с; все участники пилота имеют доступ к интерфейсу.
- *Недели 3–4:* Внедрение базового AI-агента для анализа обращений. Модель определяет тему обращения, критичность, рекомендует исполнителя. Подключение векторной базы знаний с начальными данными и реализация RAG при классификации. Также – прототип автозаполнения профиля заявителя (*MyData-lite*) на тестовом примере. *KPI:* точность тематической классификации > 80%; авто-подстановка ≥3 полей в форме обращения; время автозаполнения < 1 с【12†】.
- *Недели 5–6:* Реализация мониторинга сроков (SLA-трекер). Настройка фонового процесса (cron-задача) для проверки времени нахождения обращения в статусе. Механизм эскалации: при просрочке триггерится уведомление в eOtinish или email ответственному. Параллельно – интеграция блока GovChain: развёртывание приватного блокчейна, запись первых транзакций (например, хэша полученного обращения). *KPI:* 0 незамеченных просрочек (каждая просрочка порождает уведомление); задержка между просрочкой и нотификацией < 5 мин.
- *Недели 7–8:* Модуль обратной связи и дашборды. В интерфейс eOtinish внедряется виджет оценки ответа (★1–5 и комментарий). Agent Smith собирает эти оценки и публикует метрики для пилотных органов: средняя удовлетворённость, доля обращений с оценкой и т.д. Разворачивается Grafana-дэшборд с показателями в реальном времени (доступен участникам пилота). *KPI:* ≥70% обработанных пилотом обращений имеют выставленную оценку; дашборд обновляется онлайн при каждом новом ответе.
- *Недели 9–10:* Автоматизация отчётности по AI (AI Impact Assessment). Запуск агента, который по шаблону генерирует PDF-отчёт о влиянии использования AI на основе данных пилота: фиксирует инциденты (если были), оценивает корректность решений AI, выполняет пункты этического чек-листа. Хэш отчёта сохраняется в блокчейне (Besu). *KPI:* генерация отчёта < 60 с; отчёт проверен и подписан ответственным лицом; смарт-контракт зафиксировал хэш документа.

- *Недели 11–12:* Итоговое улучшение и документация. По результатам пилота дорабатываются модели и правила. Готовится **White Paper проекта** (рабочее название «KZ-GovFrame») – публичный документ, описывающий архитектуру, достигнутые показатели, сравнение с международными аналогами. Планируется публикация White Paper на GitHub и рассылка заинтересованным сторонам. *KPI:* ≥100 звёзд на GitHub-репозитории (интерес сообщества); упоминание проекта в отчётах NIA или профильных международных обзорах.

- **Этап 2: Масштабирование и расширение (III–IV кв. 2025).** После успешного пилота – расширение охвата Agent Smith на большее число госорганов и типов обращений. **Основные вехи:** интеграция с другими системами (например, *e-Нотария* для нотариальных запросов, *eLicense* для лицензий) с использованием наработок пилота; обучение дополнительных доменных моделей (например, казахоязычной модели для соцблока); повышенные требования к отказоустойчивости (развёртывание резервных экземпляров, бэкапы в другом ЦОД). *Контрольные точки:* принято решение о промышленной эксплуатации (уровень Минцифры); включение Agent Smith в реестр

госуслуг как подсистемы; выпущены методические рекомендации для ведомств по подключению к платформе.

- **Этап 3: Эксплуатация и автономизация (2026 г.).** Agent Smith становится штатным инструментом обработки обращений по всей стране. **Приоритеты:** оптимизация и поддержка, постепенное увеличение степени автономности агентов (доля обращений, обрабатываемых без участия человека); расширение функциональности – внедрение **прогнозной аналитики** (анализ трендов обращений, предупреждение социальных рисков), интеграция с проактивными сервисами (например, проактивное информирование граждан при выявлении массовой проблемы). На этом этапе платформа может выйти за рамки системы обращений и стать основой для других AI-инициатив в госуправлении. *Контрольные точки:* целевые показатели (среднее время ответа, уровень удовлетворённости) сравнялись или улучшились относительно доктрины «цифрового правительства»; решение о масштабировании опыта на другие процессы принято на государственном уровне.

Данный Roadmap остаётся гибким: по мере прогресса пилота и появления новых технологий (например, выход более продвинутых моделей) план может корректироваться. Важным механизмом контроля выступает **ретроспектива по вехам**: после каждого этапа команда и стейкхолдеры оценивают достижение KPI, выявляют уроки и только затем переходят к следующему этапу. Такой адаптивный подход обеспечит поступательное развитие системы, минимизируя риски срыва сроков и несоответствия ожиданиям.

8. Рекомендации по оргструктуре и пилотированию

Успешное внедрение Agent Smith зависит не только от технической реализации, но и от правильно выстроенной организационной структуры управления проектом и процесса пилотирования. Ниже представлены рекомендации, основанные на опыте аналогичных проектов и специфике данной инициативы:

1. Межведомственная рабочая группа. Поскольку система затрагивает несколько государственных органов (разработчики, владельцы данных eOtinish, пользователи-исполнители из разных министерств, надзорные органы), целесообразно создать единую рабочую группу или координационный совет. В её состав должны войти: представитель Минцифры (владелец продукта от государства), представители ключевых ведомств-участников пилота (для обратной связи и учёта требований), технический лидер от команды разработки (Bitmagic/QOSI), эксперт по безопасности, а также представитель гражданского общества (например, из офиса Омбудсмена, чтобы учитывать интересы заявителей). Такая группа будет регулярно собираться для обзора хода проекта, решения спорных вопросов и определения приоритетов. Практика показывает, что высокий уровень вовлечённости заказчика и конечных пользователей на этапе разработки существенно повышает шансы на успех.

2. Команда разработки и эксплуатации. Рекомендуется чётко разделить роли внутри технической команды. Необходимы **ML-инженеры** (обучение и настройка моделей, мониторинг качества ответов), **backend-разработчики** (интеграция с eOtinish, разработка микросервисов агентов), **DevOps-инженеры** (поддержка инфраструктуры Kubernetes, CI/CD, масштабирование), **специалисты по безопасности** (пентесты, настройка IAM, анализ соответствия требованиям по защите данных). Помимо них, нужен **аналитик предметной области**, знакомый с процессами обработки обращений – он поможет правильно интерпретировать результаты, настроить бизнес-правила (например, маршрутизацию по ведомствам) и сформулировать данные для обучения модели. На этапе пилота важно также иметь **службу поддержки** (хотя бы 1–2 человека), которая

будет оперативно помогать пользователям пилота, собирать от них проблемы и пожелания. В перспективе, по мере масштабирования, поддержкой займутся штатные ИТ-отделы самих госорганов, но на старте необходима централизованная команда реагирования.

3. Обучение и onboarding пользователей. Даже лучшая AI-система бесполезна, если пользователи ей не доверяют или не умеют пользоваться. Поэтому параллельно с техническим внедрением следует организовать **обучающие сессии** для госслужащих, участвующих в пилоте. Необходимо объяснить, какие задачи решает Agent Smith, каковы его ограничения (например, модель может ошибаться, поэтому нужен контроль), как интерпретировать подсказки и что делать при нестандартных ситуациях. Полезно подготовить *FAQ для пользователей* и краткую инструкцию. Кроме того, стоит внедрить механизм обратной связи: в интерфейсе инспектора добавить кнопку «Сообщить о неточности AI» – чтобы сотрудники могли быстро пометить случаи, где модель дала некорректный или неполезный результат. Такие сигналы будут собираться и анализироваться ML-командой для улучшения системы.

4. Постепенное наращивание автономности. В организационном плане важно определить *границы ответственности* AI и человека на каждом этапе внедрения. На первом пилотном этапе AI выполняет **вспомогательную роль**: он готовит аналитику и черновики ответов, но финальные решения по обращению принимают должностные лица. Это должно быть закреплено регламентом пилота: какие категории обращений обрабатываются с участием AI, кто проверяет и утверждает ответы, как фиксировать ошибки AI и кому о них докладывать. По мере роста доверия к системе эти границы можно расширять – обновлять регламент, делегируя AI больше полномочий. Однако всегда должно оставаться ответственное лицо, которое несёт юридическую ответственность за конечный ответ (даже если AI подготовил его без участия человека). Такой подход обеспечивает баланс между инновациями и ответственностью.

5. Правовое обеспечение пилота. Желательно заранее проработать нормативную базу. На период эксперимента можно издать приказ (распоряжение) профильного органа, разрешающий использование AI при работе с обращениями в рамках пилота. В документе оговорить, что ответы, подготовленные системой Agent Smith, считаются официальными только после верификации уполномоченным сотрудником. Также можно предусмотреть информирование заявителей: хотя запрашивать отдельное согласие граждан на «обработку их обращений AI» необязательно (они обращаются через eOtinish, а внутренние инструменты – прерогатива ведомства), но хорошей практикой будет уведомить их. Например, добавить на портал eOtinish заметку: «Ваше обращение может быть обработано автоматически с использованием ИИ; персональные данные защищены». Это повысит прозрачность и доверие.

6. Масштабирование через последовательные пилоты. Рекомендуется после первого пилота (с ограниченным числом ведомств) спланировать следующую волну расширения. Например, на втором этапе подключить ещё несколько министерств, затем – областные акиматы. На каждом этапе важно удостовериться в готовности инфраструктуры (масштабирование кластера, добавление мощностей для новых данных) и кадров (обучение новых пользователей, наличие локальных администраторов в новых органах). Такой поэтапный рост предпочтительнее одномоментного развёртывания – он позволяет учесть специфику разных органов и постепенно адаптировать систему. Возможно, стоит назначать в каждом подключаемом ведомстве своего **координатора** – человека, отвечающего за внедрение Agent Smith в рамках данного органа, который будет связующим звеном с центральной командой.

7. Коммуникация с общественностью. Хотя Agent Smith – внутренняя система, её влияние ощутят граждане через качество ответов. Поэтому целесообразно продумать PR-стратегию: как и когда информировать общество о внедрении AI. Например, по завершении успешного пилота

можно выпустить пресс-релиз о том, что в таком-то министерстве протестирован AI-модуль, позволивший сократить средний срок ответа гражданам на X% и повысить удовлетворённость. Это продемонстрирует приверженность государства к инновациям. Также важно подготовить разъяснения на возможные вопросы или опасения: подчеркнуть, что AI работает под контролем человека, данные защищены, цель – улучшить сервис, а не заменить сотрудников и т.п.

8. Обратный анализ и обновление политики. По итогам пилотного внедрения необходимо собрать все выводы и, при необходимости, внести изменения в нормативные акты или внутренние инструкции. Если выявлено, что некоторые типы обращений плохо поддаются автоматической обработке, временно исключить их из зоны ответственности AI (до улучшения модели). Если напротив результаты отличные – инициировать закрепление этого подхода нормативно (внести изменения в стандарты оказания госуслуг, учитывающие участие AI). В идеале, результатом успешного пилота должно стать официальное решение о постоянном использовании Agent Smith и план тиражирования, поддержанный правовыми документами.

В совокупности эти рекомендации помогут встроить Agent Smith в существующую бюрократическую и ИТ-экосистему максимально эффективно. Проект подобного масштаба затрагивает не только технологии, но и людей, и процессы – поэтому необходим **комплексный подход**: технические инновации должны сопровождаться организационными мерами и прозрачной коммуникацией.

9. Выводы и следующая итерация

Реализация проекта Agent Smith открывает новую страницу в цифровой трансформации государственного управления. В этом документе представлены цели проекта, анализ текущего статуса, обзор международных практик, описание архитектуры системы и агентов, а также организационные аспекты пилотирования. Основные выводы следующие:

- **Agent Smith сочетает глобальные технологии с национальным суверенитетом.** Архитектура предусматривает использование передовых LLM-моделей и подходов (мультимодельность, RAG, микросервисы агентов) при одновременном обеспечении хранения данных и критичных вычислений внутри страны. Это позволяет получить выгоды от ИИ, не жертвуя безопасностью и независимостью.
- **Проект опирается на проверенные практики e-government.** Учитывается опыт лидеров: единое окно для заявителей, автоматическая маршрутизация и совместная обработка, жёсткие SLA с прозрачным контролем, обратная связь граждан, open-source компоненты и открытые данные. Уже сейчас Agent Smith реализует значительную часть этих элементов, а оставшиеся (MyData, публичные метрики, AI-этика) включены в дорожную карту.
- **Архитектура гибкая и масштабируемая.** Модульный принцип и разделение ролей агентов позволяют легко добавлять новые функции и модели. Интеграция слоя блокчейна (GovChain) обеспечивает доверие к результатам. Встроенная память и знаниевая база делают систему самообучающейся – с каждой итерацией Agent Smith становится умнее и полезнее.
- **Безопасность, этика и приватность встроены by-design.** Проект продемонстрировал приверженность лучшим практикам: риск-ориентированное управление безопасностью [8], соблюдение норм о данных, минимизация использования внешних сервисов, контроль решений AI человеком и публичная оценка воздействия. Это формирует фундамент доверия, без которого невозможно масштабное внедрение AI в госуправление.
- **Необходима сильная организационная поддержка.** Для успеха крайне важны правильная команда, межведомственное взаимодействие, обученность пользователей и

нормативная готовность. Предложенные рекомендации по оргструктуре и пилотированию направлены на то, чтобы инновация была воспринята и эффективно использована в существующей системе управления, а не осталась экспериментом.

Следующие шаги. После завершения текущего пилотного этапа и достижения промежуточных целей (раздел 7) проект перейдёт в фазу расширения. В 2025–2026 гг. ожидается постепенное подключение новых участников, доработка функциональности и переход от контролируемого использования AI к более **автономному режиму** в рамках установленных границ. К началу 2026 г. Agent Smith может стать неотъемлемой частью процессов работы с обращениями во многих госорганах, обеспечивая выигрыш во времени и качестве обслуживания граждан. Далее логичным шагом станет интеграция компонентов Agent Smith с другими сервисами цифрового государства (аналитические панели руководства, проактивные социальные услуги, умные чат-боты для граждан и др.), чтобы знания и возможности платформы масштабировались на всю экосистему.

В заключение, **Agent Smith** – это больше, чем ИТ-решение. Это пример того, как государство может эффективно использовать современные AI-технологии для усиления обратной связи с гражданами и повышения эффективности работы. Опираясь на международный опыт и собственные инновации, Казахстан с помощью этого проекта делает шаг к **интеллектуальному электронному правительству**, где взаимодействие «гражданин – государство» станет быстрее, прозрачнее и проактивнее. Предстоит ещё большая работа по реализации намеченного, но проведённый анализ и полученные результаты вселяют уверенность, что проект движется в верном направлении и обладает всеми предпосылками для успешной следующей итерации развития.

---

1 2 3 AI will address Kazakhstani citizens' concerns: Prosecutor General’s Office explains: 06 December 2024, 10:02 - news on Tengrinews.kz
https://en.tengrinews.kz/kazakhstan_news/ai-will-address-kazakhstani-citizens-concerns-prosecutor-265509/

4 Kazakh, Kyrgyz Officials Explore Collaboration in Space and Digital Industries - The Astana Times
https://astanatimes.com/2025/02/kazakh-kyrgyz-officials-explore-collaboration-in-space-and-digital-industries/

5 Korea's Digital Government Exhibition Hall
https://mois.go.kr/eng/sub/a03/digitalGovernmentServiceExperience/screen.do

6 7 8 9 19 ENISA Releases Comprehensive Framework for Ensuring Cybersecurity in the Lifecycle of AI Systems | Technology Law Dispatch
https://www.technologylawdispatch.com/2023/06/data-cyber-security/enisa-releases-comprehensive-framework-for-ensuring-cybersecurity-in-the-lifecycle-of-ai-systems/

10 11 12 15 GovChain-DAO-Revolutionizing-Government-Efficiency.pdf
file://file-1r3dBcQQfXMe4s7rqh5XPi

13 cdn.openai.com
https://cdn.openai.com/business-guides-and-resources/a-practical-guide-to-building-agents.pdf

14 18 AI Agent Architectures: The Ultimate Guide With n8n Examples
https://www.productcompass.pm/p/ai-agent-architectures

16 17 Building AI Agents with Long-Term Memory: A Guide to Semantic Memory Implementation | by Micheal Lanham | Apr, 2025 | Medium
https://medium.com/@Micheal-Lanham/building-ai-agents-with-long-term-memory-a-guide-to-semantic-memory-implementation-44eb48028c5e