

План реализации агентного портала

Этапы, задачи, зависимости и критерии готовности

Параметр	Значение
Версия / дата	0.1 / 02 октября 2026 года
Статус	Проект для обсуждения и последующего согласования
Объект	Портал, подключение моделей, агентные сценарии и эксплуатация
Граница	Развёртывание, обучение и GPU-инфраструктура моделей вне проекта
Документ	Практический план действий

Принцип: пользователь собирает помощника из разрешённых возможностей; платформа обеспечивает доступ, бюджет, выполнение и проверяемый результат.

Сроки, ресурсы и целевые показатели являются проектными оценками. Функциональность не объявляется уже реализованной. Выбор агентного runtime подтверждается сравнительными испытаниями.

Содержание

1. Цель плана и условия начала	3
2. Релизы, сроки и критический путь	4
3. Команда и ответственность	6
4. Этап 0: решения, прототип и сравнительные испытания	7
5. Этап 1: приложение, доступ и поставка	8
6. Этап 2: модели, чат, бюджеты и аналитика	9
7. Этап 3: сквозной пилот R1	10
8. Этап 4: сценарии, RAG, код и расширения	11
9. Этапы 5–6: распределение, ИБ и эксплуатация	12
10. Backlog первых двух недель	13
11. Качество, зависимости и управление изменениями	14
12. Комплект поставки и контрольные точки	15

1. Цель плана и условия начала

План превращает описание целевой системы в последовательность проверяемых поставок. Объект — корпоративный портал, подключение готовых модельных API, агентные сценарии, аналитика, управление и эксплуатация приложения. Развёртывание моделей, обучение и GPU вне проекта.

Базовая инфраструктура: VM, Docker Compose и Ansible. Задания распределяет Temporal. Nomad/Kubernetes не являются обязательным предварительным условием. Если потребуется автоматическое размещение контейнеров, решение принимается отдельным ADR после пилота.

Каждый этап заканчивается демонстрацией, проверками и списком закрытых требований F01–F30 основного документа. Для ИБ используется отдельный пакет с контролями и формами согласования. До production должны быть назначены владельцы платформы, данных, сетевых доступов и поддержки.

Входные условия

- Назначены product owner и технический ответственный; утверждена первая группа пользователей.
- Доступны хотя бы один внутренний и один разрешённый внешний/провайдерский API для контрактных тестов; модели предоставляются отдельно.
- Есть тестовый LDAP/IdP, SMTP relay, Git и S3 либо согласованы временные mocks.
- Предоставлены VM и административный путь установки; согласован offline-import.
- Определены классы данных и разрешённые пути к внешним API/search.

При задержке корпоративной интеграции разработка продолжается на mock-контракте, но соответствующая функция не объявляется принятой до реального испытания.

2. Релизы, сроки и критический путь

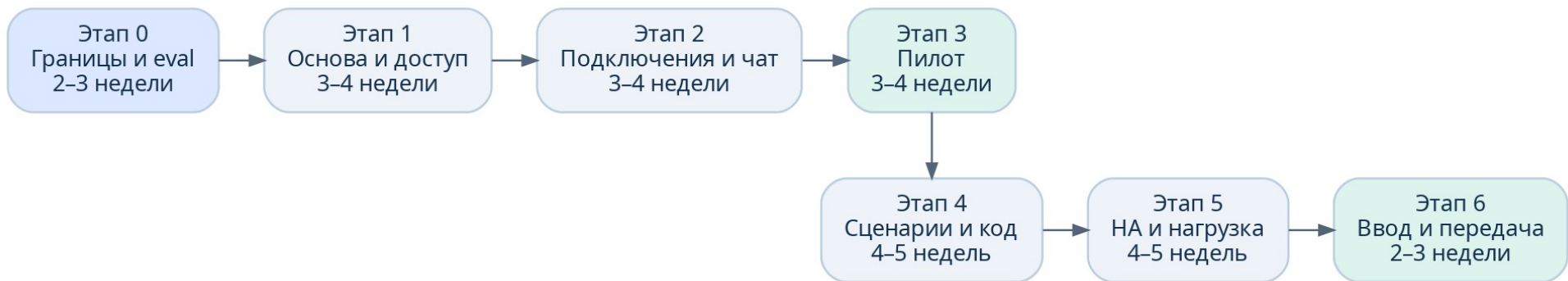


Рисунок 1. Последовательность этапов; инфраструктура моделей не включена в сроки.

Этап	Оценка	Результат	Зависимость
0	2–3 недели	Scope, UX, eval и ADR	Владельцы и тестовые входные данные
1	3–4 недели	Основа приложения и доступа	Решения этапа 0
2	3–4 недели	Модельные подключения, чат, бюджеты, аналитика	API/ACL и секреты этапа 1
3	3–4 недели	Документы, search и сквозной пилот R1	Подключения и учёт этапа 2
4	4–5 недель	Функциональный релиз R2	Пилот и одобренные расширения
5	4–5 недель	Multi-VM, HA и проверка эксплуатации R3	Стабильная функциональность и ресурсы
6	2–3 недели	Ввод, обучение и передача поддержки	Приёмка R3 и пакет ИБ

Базовая сумма — 21–28 недель. Плановый диапазон с резервом — 24–32 недели. Пилот R1 — 11–15 недель при готовых зависимостях. Это предварительная оценка для команды, указанной ниже; этап 0 должен заменить её оценкой backlog. Даты старта/финиша пока не устанавливаются.

Критический путь: авторизация и модельный контракт → резерв бюджета и поток событий → сквозное задание → изоляция tools/documents → fault/restore → production. UX, offline-поставка, мониторинг и security review идут с первого этапа; их нельзя целиком отложить до конца.

3. Команда и ответственность

Роль	Ответственность	Участие
Product owner	Приоритеты, задачи пользователей, критерии пользы и приёмка	Постоянные решения и review
Technical lead / backend	Архитектура, контракты, policy/ledger и code review	Полное
Backend-разработчик	API, данные, интеграции и администрирование	Полное
Frontend-разработчик	Портал, аналитика, редакторы и design tokens	Полное
Runtime / integration	Готовый harness, сценарии, MCP, документы и sandbox	Полное
QA	Автоматизация, отрицательные тесты, нагрузка и приёмка	От частичного на старте к полному
DevOps	Offline-установка, VM, backup, monitoring и HA	Постоянное; интенсивное на этапах 1 и 5
UX	Макеты, пользовательские сценарии и доступность	Этапы 0–2, review далее
ИБ	Требования, модель угроз, review и допуск	Все контрольные точки
Владельцы сервисов	LDAP, сеть, SMTP, Git, API моделей, S3	По соответствующим интеграциям

Совмещение backend/runtime и DevOps допускается для небольшой команды с увеличением сроков. Отдельная команда эксплуатации моделей не включается в оценку разработки портала.

Еженедельный ритм: обновление backlog и зависимостей; демонстрация завершённого функционального среза; review рисков; фиксация ADR и тестового evidence. Изменение score отражается в версиях описания, плана и security package.

4. Этап 0: решения, прототип и сравнительные испытания

Цель — не закрепить библиотеку по популярности, а доказать пригодность подхода на коротком наборе задач. Исходные требования превращаются в согласованный каталог сценариев и ограничений.

Работа	Результат	Ответственный
P0.1 Зафиксировать score	R1/R2/R3, F01–F30, внешние зависимости и исключённые модели	Product owner + lead
P0.2 Карта данных/сети	Trust zones, классы данных, обязательные endpoints и owners	Lead + ИБ + сеть
P0.3 UX-макет	Портал, управление, аналитика, sharing и настройка темы	Frontend + UX
P0.4 Контракт API модели	Capabilities, usage, streaming, auth и errors	Runtime + backend
P0.5 Eval-корпус	40–60 задач, рубрика и измерение cost/success	QA + пользователи
P0.6 Сравнение runtime	Direct API, управляемый agent/workflow, один сложный harness	Runtime
P0.7 Deployment ADR	Compose/Ansible base; условия перехода к Nomad/K8s	DevOps + lead
P0.8 Начальный пакет ИБ	Реестр угроз, открытые вопросы и gate пилота	ИБ + lead

Результат eval включает все модельные вызовы, retries, суммаризацию и неудачные задания. Модель и tools одинаковы между harness-кандидатами. Кандидат, не прошедший обязательные ACL/egress проверки, не выигрывает выбор только за счёт меньшей цены.

Критерий готовности: подписаны границы R1; макет просмотрен будущими пользователями; утверждён тестовый набор; выбран один основной runtime-подход либо явно зафиксировано, какое испытание ещё требуется; нет неизвестного пути передачи закрытых данных наружу.

5. Этап 1: приложение, доступ и поставка

Цель — рабочий skeleton с реальной авторизацией и повторяемой установкой. Техническую основу проверяем до добавления сложных агентных сценариев.

Работа	Требования	Приёмочный результат
P1.1 Repository/CI и миграции	F26	Build воспроизводим, версии/digests фиксируются
P1.2 React shell и design tokens	F05, F23	Навигация портала/управления, темы, иконки, focus и empty/error states
P1.3 Go API и OIDC	F01	LDAP/local вход, logout, роли и blocking
P1.4 Spaces и policy engine	F02	Серверная авторизация всех объектов и отрицательные тесты
P1.5 Secrets и connector registry	F03, F19	Секреты не возвращаются UI и не попадают в logs/export
P1.6 Audit/event contract	F24, F10	Уникальные events, correlation IDs и schema
P1.7 Compose/Ansible skeleton	F26	Preflight и install на чистых VM без общего интернета
P1.8 Monitoring baseline	F25	Health/readiness и внутренние alerts

Уже на этом этапе проверяется запрет интернет-доступа worker и песочницы. Если для разработки используется разрешённая connected-среда, offline-test всё равно выполняется на изолированной среде.

Критерий готовности: два пользователя из разных пространств не видят объекты друг друга; пользователь не может добавить произвольный endpoint; repeat install не дублирует начальные настройки; настройки оформления действуют во всех разделах.

6. Этап 2: модели, чат, бюджеты и аналитика

Модели подключаются как сервисы. Этап не включает provisioning модельных серверов. Ledger и аналитика вводятся одновременно, чтобы фактический расход не восстанавливался позже по неполным логам.

Работа	Требования	Приёмочный результат
P2.1 Model gateway/adapters	F03–F04	Внутренний и внешний/провайдерский API проходят один контракт
P2.2 Streaming/history	F05	SSE reconnect, сохранённая история и читаемые ошибки
P2.3 Assistant editor/versioning	F06–F07	Черновик, версия, публикация и закрепление в run
P2.4 Budget accounts/ledger	F08–F09	Индивидуальный предел + общий счёт; race/fault tests
P2.5 Price/usage normalization	F10, F29	Нет double count cache; неизвестный usage отмечен
P2.6 Analytics base	F10	Расход, calls, users, tools/docs/share events и filters
P2.7 Admission/cancel	F11	Лимит блокирует следующий шаг; отмена сверяет расход
P2.8 Global config revisions	F23	Валидация, запись версии и audit изменения

Analytics сначала показывает обязательные финансовые и пользовательские события, даже если сложные инструменты появятся позже. Sharing grant и фактическое использование — разные показатели. Export проверяется под теми же ACL, что dashboard.

Критерий готовности: A01–A07, A16 и базовый A17; два параллельных задания не используют один остаток повторно; внешнее направление запрещается по policy; сумма аналитики сверяется с ledger за тестовый период.

7. Этап 3: сквозной пилот R1

Цель — законченные пользовательские задачи с документами и разрешённым поиском, а не набор изолированных экранов. На пилоте измеряем фактическую нагрузку и полезность.

Работа	Требования	Приёмочный результат
P3.1 Document pipeline	F16	PDF/DOCX/TXT/CSV/XLSX, статусы, ошибки, источники и артефакты
P3.2 Search/fetch	F15	Отдельный egress, private-IP/redirect tests и citations
P3.3 Первый workflow	F12–F13	Ответ по файлам / сравнение документов с контролем шага
P3.4 Temporal execution	F11	Очередь, heartbeat, UI reconnect, остановка worker
P3.5 Sharing base	F02, F07, F10	Публикация помощника и grant/revoke документов
P3.6 Offline bundle R1	F26	Полный комплект и install evidence
P3.7 Pilot evaluation	F29	Принятые задачи, cost/success, feedback и measured workload

Пилот проводится на согласованной группе, например 20–50 сотрудников, и разрешённых данных. Расширение до 100 пользователей выполняется после подтверждения capacity. Пример размера пилота не является обязательным лимитом лицензии или системы.

Критерий готовности: пользователь проходит путь «войти → выбрать помощника → приложить документ → получить результат с источниками → увидеть расход → поделиться в разрешённую группу». Сбой worker и закрытие браузера обработаны. ИБ фиксирует допуск пилота и список ограничений. У каждого замечания есть приоритет и владелец.

8. Этап 4: сценарии, RAG, код и расширения

Работы идут функциональными срезами. Самыми рискованными являются ACL retrieval, внешние эффекты инструментов и запуск кода; их тесты разрабатываются вместе с функциональностью.

Работа	Требования	Приёмочный результат
P4.1 Knowledge/RAG	F17	ACL до передачи модели, versioning, deletion и retrieval eval
P4.2 Visual workflow editor	F13	Schema validation, bounded loops, typed IO и immutable version
P4.3 Human approval	F14	Согласование конкретных параметров и повторная проверка прав
P4.4 Coding executor	F18	Изолированный workspace, diff/tests, mirrors и resource limits
P4.5 MCP registry	F19	Tool scopes, server auth, SSRF и connector disable
P4.6 SMTP/outbox	F21	Разрешённые адресаты; ambiguous delivery и retry policy
P4.7 Plugin lifecycle	F20	Manifest, isolated service, contract checks и rollback
P4.8 Schedule/service principal	F22	Собственный бюджет и отзыв разрешений
P4.9 Full analytics/sharing	F10, F29	Tool/doc adoption, grants vs use, drill-down и privacy

Время coding/MCP зависит от заранее доступных библиотек и количества первых интеграций. В R2 планируется ограниченный утверждённый каталог; «поддержка плагинов» не означает разработку всех возможных корпоративных connectors.

Критерий готовности: A04, A09–A12 и regression A05–A08 проходят на полных сценариях; нет пользовательского пути к произвольному endpoint, секрету или privileged shell. Отчёт по сложному harness демонстрирует оправданность его использования либо ограничивает его применение.

9. Этапы 5–6: распределение, ИБ и эксплуатация

Этап 5

Развернуть API/IdP replicas, платформенные сервисы и разные worker pools на нескольких VM. Настроить LB, fencing/lease, HA PostgreSQL либо подключить корпоративный сервис. Зафиксировать все межсервисные порты. Подготовить capacity inventory и механизм maintenance/drain.

Выполнить нагрузочные профили и fault-тесты: потеря API, worker, search, replica БД, сбой gateway, недоступный provider, network partition и неизвестный usage. Проверить replay после restore и отсутствие повторной SMTP-операции. Собрать evidence для A08–A15, A17 и пакета ИБ.

Если подтверждена необходимость автоматического placement, выполнить отдельный ограниченный PoC Nomad. Переход не должен быть скрытым расширением scope: оценить offline-поставку, ACL/TLS, сеть, storage, лицензию и дополнительную эксплуатацию. Kubernetes допустим при готовом корпоративном контуре.

Этап 6

Провести финальную приёмку, закрыть критичные findings, зафиксировать версии и остаточные риски. Обучить пользователей и операторов. Передать backup/restore, incidents, upgrades, quota administration и onboarding runbooks. Согласовать поддержку и окно обновлений.

Production gate: подписанный перечень функций; evidence приёмки; допуск ИБ; проверенный restore; владельцы и on-call; мониторинг/alerting; контроль capacity; разрешённые endpoints; определённые ограничения внешних API; отсутствие нерешённых блокирующих замечаний.

10. Backlog первых двух недель

Этот список позволяет начать работу без решения всех будущих вопросов. Задачи относятся к проектированию и проверке, а не к преждевременной закупке модели/GPU.

Задача	Артефакт	Проверка готовности
B01 Уточнить 5–10 бизнес-сценариев	Короткие карточки задач и rubric	Пользователь подтверждает полезный результат
B02 Инвентаризовать доступы	Endpoint/owner/protocol table	Есть путь до тестовых API и LDAP
B03 Утвердить R1	Requirement baseline	Нет смешения инфраструктуры моделей и портала
B04 Review UI	HTML-макет и замечания	Портал, управление, аналитика и темы просмотрены
B05 Создать тестовые spaces/users	Fixture данных	Можно проверить отрицательный ACL
B06 Проверить два API	Contract report	Usage, streaming и tools имеют понятную семантику
B07 Сравнить runtime	Eval notebook/report и ADR	Выбор не основан только на бренде
B08 Проверить offline build	Release manifest prototype	Все зависимости перечислены
B09 Согласовать ИБ-пилот	Gate checklist	Данные/egress/owners определены
B10 Оценить backlog	План по итерациям	У задач есть owner, зависимости и критерии

11. Качество, зависимости и управление изменениями

Definition of Done: реализован пользовательский результат; серверная авторизация; разумные автоматические проверки; telemetry/audit; обработка errors/empty/loading; документация и схема конфигурации; offline-зависимости; миграция и совместимость; подтверждённый review. Для финансовых и внешних эффектов нужны race/fault tests.

Каждая новая функция сохраняет requirement ID. ADR нужен для runtime, deployment, ledger, identity, document ACL, sandbox и выбора platform services. Изменение API проходит версионирование. Breaking connector change не устанавливается автоматически на активные задания.

Зависимость	Владелец	Действие при задержке
LDAP/IdP	Корпоративная IAM-команда	Mock/local fixture; production gate остаётся открытым
Внешний egress	Сеть + ИБ	Интеграция внутреннего API; external feature не принята
API моделей	Владелец подключения	Contract mock; реальная цена/usage не выдумываются
SMTP/Git/MCP	Владелец системы	Test connector и отдельное расписание приёмки
HA PostgreSQL/S3	DBA/storage	Пилотное размещение и явный infrastructure backlog
Mirrors	DevOps/пакетный сервис	Ограниченный каталог coding images

Детальные нормы ИБ и сроки хранения предоставляет организация. Отсутствие названия конкретного стандарта не блокирует минимальные меры доступа, изоляции, секретов и аудита.

12. Комплект поставки и контрольные точки

Артефакт	Содержание	Передаётся
Продукт	UI/API, runtime integration, analytics и администрирование	R1 → R3
Deployment bundle	Images, packages, Ansible, manifests и checks	Каждый релиз
API/config contract	OpenAPI, schemas, events и migration notes	Каждый релиз
UX/design system	Tokens, иконки, состояния, макеты и accessibility checks	Этапы 0–2, обновления далее
Eval report	Task success, все расходы, runtime versions и выводы	Выбор и обновление runtime
Security evidence	Модель угроз, matrix, тесты и residual risks	Пилот и production
Эксплуатация	Install, backup, restore, incident и capacity runbooks	R3
Пользовательская документация	Помощники, sharing, расход, документы и сценарии	R1 → R3

Предоставленные проектные DOCX и интерактивный макет являются исходными материалами согласования. Они не заменяют реализацию, evidence испытаний и утверждение ИБ. Следующий практический шаг — review разделов R1 и макета, затем этап 0 по этому плану.