Описание процессов жизненного цикла

1. Общие положения

Жизненный цикл ПО включает планирование изменений, разработку, проверку, выпуск, установку, сопровождение, устранение неисправностей и прекращение поддержки версии.

Организация процесса определяется правообладателем. В исходном коде подтверждено использование Git и автоматических тестов; применение Scrum, Kanban или иной методологии не заявляется без отдельного организационного документа.

2. Хранение исходного кода

История разработки ведется в Git. Проверенный рабочий репозиторий расположен в /srv/ac700/own-chat. Для промышленного процесса правообладатель должен назначить основной закрытый Git-сервер, резервное хранение и правила доступа.

Для каждого выпуска фиксируются:

- идентификатор версии;
- хеш коммита;
- состав файлов;
- контрольные суммы дистрибутива;
- перечень изменений;
- результаты тестов;
- совместимые конфигурации;
- план обновления и отката.

3. Разработка изменений

Требование или дефект регистрируется в корпоративной системе учета обращений (далее — Система учета). Ответственный определяет затронутые функции, данные и интерфейсы.

Изменение выполняется в отдельной ветви или рабочей копии. Добавляются или обновляются тесты. Проводится проверка изменений другим разработчиком либо формализованная самопроверка для малой команды.

После успешных тестов изменение включается в выпуск. Документация обновляется одновременно с кодом.

4. Проверка

Минимальная проверка выпуска:

```
python3 -m compileall -q app gateway tools
python3 -m unittest discover -v
node --check static/app.js
node --check static/admin.js
node --check static/common.js
git diff --check
```

Сквозная проверка выполняется с имитатором OpenAI API и включает вход, получение списка моделей, создание чата, потоковый ответ и сохранение двух сообщений. Для проверенного аппаратного профиля дополнительно проверяются запуск модели, телеметрия, генерация, переключение, очередь и восстановление после остановки контейнера.

Результат аппаратного испытания относится только к конкретной версии драйвера, среды вывода, модели, ОС и оборудования.

5. Выпуск

Выпуск формирует назначенный ответственный. Дистрибутив создается из утвержденного коммита. Python-пакеты устанавливаются в отдельное виртуальное окружение по зафиксированному requirements.txt; клиентская часть не требует сборки.

Перед выпуском проверяются:

- отсутствие секретов и пользовательских данных в дистрибутиве;
- соответствие версии в исходном коде и документации;
- актуальность THIRD_PARTY_NOTICES.md;
- наличие лицензионных текстов при поставке сторонних пакетов;
- воспроизводимость установки;
- целостность резервной копии;
- работоспособность отката.

6. Установка обновлений

Обновления устанавливает системный администратор вручную из доверенного внутреннего источника. Собственный код не выполняет принудительное обновление и не предоставляет внешнему лицу механизм удаленного обновления.

Порядок обновления описан в руководстве администратора. Изменение среды модели или драйвера выполняется отдельно от приложения и требует повторной проверки совместимости.

7. Прием и обработка обращений

Канал поддержки: support@stc.com.ru, +7 (499) 380-83-80.

Режим работы: рабочие дни с 09:00 до 18:00 по московскому времени.

Обращение должно содержать версию ПО, время ошибки, действия пользователя, текст сообщения и допустимый фрагмент журнала без секретов и конфиденциальных данных.

Классификация обращений и целевое время первичной реакции:

Приоритет	Пример	Целевое время первичной реакции
Критический (P1)	Сервис полностью недоступен, повреждение данных	1 рабочий час
Высокий (P2)	Недоступна ключевая функция без обхода	4 рабочих часа
Обычный (P3)	Частичная ошибка с обходным решением	1 рабочий день
Низкий (P4)	Консультация или предложение	3 рабочих дня

Значения SLA утверждены правообладателем. До заключения отдельных договоров с пользователями указанные значения являются целевыми ориентирами.

8. Устранение неисправностей

Обращение регистрируется и получает приоритет.

Специалист поддержки воспроизводит проблему на тестовой среде или собирает диагностические данные.

Определяется компонент: браузер, веб-приложение, база, шлюз, сервер модели, драйвер или инфраструктура.

Для эксплуатационной проблемы применяется утвержденная инструкция восстановления. Для дефекта кода готовится изменение и регрессионный тест.

Выпуск проверяется и устанавливается по процедуре обновления. В обращении фиксируются причина, решение и затронутые версии.

Если отказ связан со сторонней моделью, драйвером или средой вывода, правообладатель ПО предоставляет диагностику в пределах собственного продукта и взаимодействует с поставщиком внешнего компонента по отдельным условиям.

9. Резервное копирование и восстановление

Для SQLite предусмотрена ежедневная служба резервного копирования. Архив проверяется PRAGMA integrity_check. Рекомендуется не реже одного раза в месяц переносить копии на отдельный носитель и не реже одного раза в квартал выполнять контрольное восстановление.

Исходный код, документация, конфигурация без секретов и материалы выпуска также должны резервироваться.

10. Совершенствование

Предложения пользователей и результаты эксплуатации рассматриваются ответственным за продукт. При изменении функций оцениваются обратная совместимость API, схема данных, безопасность, документация и необходимость повторной проверки аппаратных профилей.

11. Персонал

Минимально необходимые роли:

Роль	Задачи	Требуемая квалификация
Владелец продукта	Требования, приоритеты, приемка выпуска	Знание предметной области и продукта
Разработчик Python	Сервер, шлюз, база и тесты	Python 3.10, HTTP, asyncio, SQLite, безопасность веб-приложений
Разработчик интерфейса	Браузерный интерфейс	HTML, CSS, JavaScript, доступность и безопасность DOM
Системный администратор	Установка, Linux, службы, TLS, резервные копии	Linux, systemd, сети, контейнеры
Специалист среды вывода	Модели и ускорители	Выбранный движок вывода, модели и аппаратная платформа
Специалист поддержки	Обращения и первичная диагностика	Эксплуатационная документация и журналы

Один сотрудник может совмещать роли при наличии квалификации и разграничении доступа.

12. Прекращение поддержки

Правообладатель уведомляет пользователей о прекращении поддержки версии за 6 месяцев, предоставляет порядок выгрузки или сохранения истории и рекомендуемый путь обновления. Удаление данных выполняется по правилам организации и применимым требованиям к персональным данным.