

Описание жизненного цикла, поддержки и обслуживания программного обеспечения МБорд

1. Термины и определения

Термин	Определение
ПО	Программное обеспечение
МБорд	Программное обеспечение «МБорд» — веб-платформа мониторинга промышленного оборудования
Правообладатель	ООО «Метрикор» — исключительный правообладатель ПО «МБорд»
Пользователь	Физическое или юридическое лицо, использующее ПО на основании лицензионного соглашения
Инцидент	Незапланированное прерывание или снижение качества работы ПО
SLA	Service Level Agreement — соглашение об уровне обслуживания
SemVer	Semantic Versioning — семантическое версионирование (MAJOR.MINOR.PATCH)
Релиз	Версия ПО, подготовленная для передачи пользователям
Патч	Обновление ПО, направленное на устранение дефектов без изменения функциональности
Техническая поддержка	Комплекс мероприятий по обеспечению работоспособности ПО
Жизненный цикл	Совокупность этапов существования ПО от создания до вывода из эксплуатации
Модернизация	Процесс доработки ПО с целью улучшения функциональных характеристик
Эксплуатация	Этап жизненного цикла, на котором ПО используется по назначению
Конфигурация	Совокупность настроек и параметров, определяющих поведение ПО
CI/CD	Continuous Integration / Continuous Deployment — непрерывная интеграция и доставка
TimescaleDB	Расширение PostgreSQL для работы с временными рядами
Docker	Платформа контейнеризации для развёртывания ПО

2. Назначение документа

Настоящий документ описывает:

- Принципы и порядок поддержания жизненного цикла программного обеспечения «МБорд»;
- Процессы технической поддержки пользователей;
- Порядок устранения неисправностей;
- Процедуры модернизации и развития ПО;
- Требования к персоналу, осуществляющему сопровождение;
- Уровни обслуживания и показатели качества;
- Процессы управления изменениями и конфигурацией;
- Условия завершения жизненного цикла ПО.

Документ предназначен для:

- Сотрудников ООО «Метрикор», осуществляющих разработку и сопровождение ПО;
- Администраторов и инженеров, эксплуатирующих ПО;
- Пользователей ПО, обращающихся в службу технической поддержки;
- Контролирующих органов при проведении экспертизы ПО.

3. Общие положения

3.1. Программное обеспечение «МБорд» разработано ООО «Метрикор» (ОГРН: зарегистрировано на территории Российской Федерации) и является результатом интеллектуальной деятельности, охраняемым в соответствии с Гражданским кодексом Российской Федерации (часть четвёртая).

3.2. ПО «МБорд» предназначено для непрерывного мониторинга параметров промышленного оборудования, включая:

- Сбор телеметрических данных с контроллеров;
- Визуализацию показаний датчиков в реальном времени;
- Оповещение персонала о выходе параметров за пределы нормативных значений;
- Хранение и анализ исторических данных.

3.3. Жизненный цикл ПО «МБорд» включает следующие этапы:

- Разработка и тестирование;
- Ввод в эксплуатацию;
- Промышленная эксплуатация;
- Техническая поддержка и сопровождение;
- Модернизация и развитие;
- Вывод из эксплуатации.

3.4. Правообладатель гарантирует поддержку каждой мажорной версии ПО в течение не менее 24 месяцев с момента выпуска.

3.5. Все процессы жизненного цикла осуществляются в соответствии с ГОСТ Р ИСО/МЭК 12207-2010 «Информационная технология. Системная и программная инженерия. Процессы жизненного цикла программных средств».

4. Общие принципы поддержания жизненного цикла ПО

Поддержание жизненного цикла ПО «МБорд» основывается на следующих принципах:

4.1 Системность

Все процессы сопровождения ПО рассматриваются как единая система взаимосвязанных мероприятий. Изменение одного компонента анализируется с точки зрения влияния на всю систему в целом. Архитектура ПО (backend, frontend, file-watcher, база данных, кэш) поддерживается как единое целое с синхронизацией версий.

4.2 Непрерывность

Техническая поддержка и сопровождение осуществляются непрерывно на протяжении всего жизненного цикла ПО. Правообладатель обеспечивает доступность каналов связи и реагирование на обращения в соответствии с установленными SLA.

4.3 Предсказуемость

Плановые обновления и изменения выполняются по заранее определённому расписанию с уведомлением пользователей. Внеплановые работы проводятся только для устранения критических дефектов и сопровождаются информированием заинтересованных сторон.

4.4 Документируемость

Все изменения, обращения, инциденты и их разрешения фиксируются документально. Ведётся журнал изменений (CHANGELOG), история версий, база знаний по типичным проблемам и их решениям.

4.5 Трассируемость

Каждое изменение в ПО может быть отслежено от момента постановки задачи до внедрения в продуктивную среду. Используется система контроля версий (Git) с обязательным указанием связи коммитов с задачами.

4.6 Разделение ответственности

Чётко определены зоны ответственности между:

- Правообладателем (разработка, обновления, устранение дефектов ПО);
- Администратором (развёртывание, конфигурирование, резервное копирование);
- Пользователем (эксплуатация в соответствии с документацией).

4.7 Прозрачность

Пользователи информируются о:

- Планируемых обновлениях и их содержании;
- Известных ограничениях и дефектах;
- Сроках устранения проблем;
- Изменениях в условиях обслуживания.

5. Техническая поддержка пользователей

5.1 Назначение

Техническая поддержка обеспечивает:

- Консультирование пользователей по вопросам эксплуатации ПО;
- Приём и обработку обращений о неисправностях;
- Помощь в настройке и конфигурировании;
- Предоставление обновлений и исправлений;
- Информирование о планируемых изменениях.

5.2 Каналы обращения

Канал	Контакт	Время работы
Электронная почта	info@metrikor.ru	Круглосуточно (приём), обработка — рабочие дни
Телефон	По договору	Рабочие дни, 09:00–18:00 МСК
Система заявок	По договору	Круглосуточно

Основным каналом обращения является электронная почта. При наличии договора расширенной поддержки доступны дополнительные каналы.

5.3 Основные функции технической поддержки

- Регистрация обращений и присвоение приоритетов;
- Первичная диагностика и классификация проблем;
- Предоставление рекомендаций по решению типовых вопросов;
- Эскалация сложных проблем разработчикам;
- Контроль сроков устранения проблем;
- Информирование пользователя о статусе обращения;
- Закрытие обращений после подтверждения решения.

5.4 Требования к обращению

Для эффективной обработки обращения пользователь должен предоставить:

1. **Описание проблемы** — что именно не работает или работает некорректно.
 2. **Шаги воспроизведения** — последовательность действий, приводящая к проблеме.
 3. **Ожидаемый результат** — что должно происходить по мнению пользователя.
 4. **Фактический результат** — что происходит на самом деле.
 5. **Окружение** — браузер, версия ПО, URL системы.
 6. **Скриншоты** — визуальное подтверждение проблемы (при наличии).
 7. **Контактная информация** — email и/или телефон для обратной связи.
-

6. Устранение неисправностей

6.1 Этапы устранения неисправностей

Процесс устранения неисправностей включает следующие этапы:

Этап 1. Регистрация и классификация

- Приём обращения через канал связи;
- Регистрация в системе учёта;
- Определение категории и приоритета;
- Назначение ответственного специалиста.

Этап 2. Диагностика

- Воспроизведение проблемы в тестовой среде;
- Анализ логов и состояния системы;
- Определение корневой причины (root cause);
- Оценка влияния на работу системы.

Этап 3. Разработка решения

- Проектирование исправления;
- Реализация и модульное тестирование;
- Code review;
- Интеграционное тестирование.

Этап 4. Внедрение

- Подготовка релиза с исправлением;
- Уведомление пользователей о плановых работах;
- Развёртывание обновления;
- Проверка работоспособности в продуктивной среде.

Этап 5. Верификация и закрытие

- Подтверждение устранения проблемы пользователем;
- Документирование решения в базе знаний;
- Закрытие обращения;
- Анализ для предотвращения повторения.

6.2 Результаты

Результатами процесса устранения неисправностей являются:

- Восстановление работоспособности ПО;
 - Обновлённая версия ПО (патч);
 - Запись в журнале изменений;
 - Обновлённая документация (при необходимости);
 - Рекомендации по предотвращению аналогичных проблем;
 - Отчёт об инциденте (для критических случаев).
-

7. Модернизация и развитие ПО

7.1 Общие положения

Модернизация ПО «МБорд» осуществляется на основании:

- Запросов пользователей на новую функциональность;
- Результатов анализа тенденций рынка промышленного мониторинга;
- Технических требований к совместимости с новым оборудованием;
- Результатов аудита безопасности;
- Требований регулирующих органов.

Модернизация не должна:

- Нарушать обратную совместимость без обоснования и предварительного уведомления;
- Снижать производительность системы;
- Создавать уязвимости безопасности;
- Приводить к потере данных.

7.2 Процесс модернизации

1. **Сбор требований** — анализ потребностей пользователей и бизнес-целей.
2. **Проектирование** — разработка технического решения, оценка трудозатрат.
3. **Планирование** — включение в дорожную карту, определение сроков.
4. **Разработка** — реализация в отдельной ветке, модульное тестирование.
5. **Тестирование** — интеграционное и приёмочное тестирование.
6. **Документирование** — обновление пользовательской и технической документации.
7. **Релиз** — подготовка и выпуск новой версии.
8. **Внедрение** — развёртывание на продуктивных средах пользователей.
9. **Мониторинг** — контроль стабильности после обновления.

7.3 Управление версиями — SemVer

ПО «МБорд» использует семантическое версионирование (SemVer) в формате **MAJOR.MINOR.PATCH**:

Компонент	Изменяется при	Пример
MAJOR	Несовместимые изменения API, архитектурные изменения	1.0.0 → 2.0.0
MINOR	Новая функциональность с сохранением обратной совместимости	2.0.0 → 2.1.0
PATCH	Исправление дефектов без изменения функциональности	2.1.0 → 2.1.1

Правила версионирования:

- Номер MINOR синхронизируется во всех компонентах системы (backend, frontend, file-watcher, arch, deploy);
- Каждый релиз сопровождается записью в CHANGELOG;
- Тег версии создаётся в системе контроля версий (Git);
- Предрелизные версии обозначаются суффиксом: 2.4.0-rc.1.

Текущая версия: 1.0.0

8. Профилактическое сопровождение

Профилактическое сопровождение включает регулярные мероприятия по обеспечению стабильной работы ПО:

8.1 Ежедневные мероприятия

- Мониторинг health endpoint;
- Проверка доступности веб-интерфейса;
- Контроль свободного дискового пространства;
- Проверка статуса Docker-контейнеров.

8.2 Еженедельные мероприятия

- Анализ логов на наличие ошибок и предупреждений;
- Проверка целостности резервных копий;
- Мониторинг производительности базы данных;
- Обзор метрик использования ресурсов (CPU, RAM, диск).

8.3 Ежемесячные мероприятия

- Обновление зависимостей и библиотек (security patches);
- Проверка SSL-сертификатов;
- Аудит учётных записей (деактивация неактивных);
- Оптимизация базы данных (VACUUM, ANALYZE);
- Ротация логов.

8.4 Ежеквартальные мероприятия

- Аудит безопасности;
- Тестирование процедуры восстановления из резервной копии;
- Обновление документации;
- Анализ производительности и планирование масштабирования.

9. Информация о персонале

9.1 Общие положения

Сопровождение и поддержка ПО «МБорд» осуществляется квалифицированным персоналом ООО «Метрикор». Правообладатель обеспечивает наличие достаточного количества специалистов для выполнения обязательств по поддержке.

9.2 Категории персонала

Категория	Функции
Разработчики	Реализация новой функциональности, устранение дефектов, код-ревью
Инженеры DevOps	Развёртывание, мониторинг инфраструктуры, CI/CD
Специалисты поддержки	Приём обращений, первичная диагностика, консультирование
Аналитики	Сбор требований, проектирование решений, документирование
Тестировщики	Функциональное и регрессионное тестирование

9.3 Требования к квалификации

Разработчики:

- Высшее техническое образование или эквивалентный опыт;
- Знание Python 3.11+, FastAPI, SQLAlchemy 2.0;
- Знание React, TypeScript;
- Опыт работы с PostgreSQL/TimescaleDB;
- Понимание принципов REST API и микросервисной архитектуры;
- Опыт работы с Docker и системами контроля версий (Git).

Инженеры DevOps:

- Опыт администрирования Linux-серверов;
- Знание Docker, Docker Compose;
- Опыт настройки Nginx;
- Знание PostgreSQL (резервное копирование, мониторинг);
- Опыт работы с CI/CD (GitLab CI).

Специалисты поддержки:

- Понимание принципов промышленного мониторинга;
- Опыт работы с веб-приложениями;
- Навыки диагностики сетевых и программных проблем;
- Коммуникативные навыки.

9.4 Ответственность

Аспект	Ответственный
Доступность системы	Инженер DevOps
Устранение дефектов	Разработчик
Приём обращений	Специалист поддержки
Качество релизов	Тестировщик
Актуальность документации	Аналитик
Безопасность данных	Инженер DevOps + Разработчик

10. Уровни обслуживания и показатели качества SLA

10.1 Общие положения

Уровни обслуживания определяют гарантированные параметры качества предоставляемых услуг по поддержке ПО «МБорд». Конкретные параметры SLA фиксируются в договоре с каждым пользователем.

Настоящий раздел описывает базовые (стандартные) уровни обслуживания.

10.2 Категории инцидентов

Категория	Описание	Пример
Критический (P1)	Система полностью неработоспособна, данные не поступают	Падение backend, недоступность БД
Высокий (P2)	Существенная деградация функциональности	Не работают алерты, не загружаются данные с части контроллеров
Средний (P3)	Частичное нарушение функциональности, есть обходной путь	Не работает экспорт Excel, ошибка отображения графика
Низкий (P4)	Косметические дефекты, пожелания	Опечатка в интерфейсе, некорректное выравнивание

10.3 Показатели SLA

Показатель	P1	P2	P3	P4
Время реакции	1 час	4 часа	8 рабочих часов	2 рабочих дня
Время решения	4 часа	1 рабочий день	5 рабочих дней	15 рабочих дней
Доступность системы	99.5% в месяц	—	—	—
RPO (допустимая потеря данных)	1 час	—	—	—
RTO (время восстановления)	4 часа	—	—	—

Примечания:

- Рабочие часы: пн-пт, 09:00–18:00 МСК (исключая государственные праздники);
- Доступность рассчитывается как: $(\text{общее время} - \text{время простоя}) / \text{общее время} \times 100\%$;
- Плановые технические работы (с предварительным уведомлением за 48 часов) не учитываются при расчёте доступности;
- Расширенные параметры SLA (24/7, сокращённые сроки) доступны по отдельному договору.

11. Управление изменениями

11.1 Классификация изменений

Тип	Описание	Процедура
Стандартное	Заранее одобренное изменение с известным риском	Выполняется по утверждённой процедуре
Нормальное	Изменение, требующее оценки и одобрения	Проходит полный цикл согласования
Экстренное	Изменение для устранения критического инцидента	Ускоренная процедура с постфактум документированием

11.2 Процесс управления изменениями

1. **Инициация** — формулирование запроса на изменение с обоснованием.
2. **Оценка** — анализ влияния, рисков, трудозатрат.
3. **Одобрение** — согласование с ответственными лицами.
4. **Планирование** — определение сроков и порядка внедрения.
5. **Реализация** — разработка и тестирование изменения.
6. **Внедрение** — развёртывание в продуктивной среде.
7. **Верификация** — проверка корректности работы.
8. **Закрытие** — документирование результатов.

11.3 Откат изменений

Для каждого изменения разрабатывается план отката (rollback plan). При обнаружении критических проблем после внедрения:

- Выполняется откат к предыдущей версии;
- Фиксируется причина отката;
- Проводится повторный анализ и доработка.

Технический откат обеспечивается:

- Система контроля версий (Git) — откат к предыдущему тегу;
- Docker-образы — откат к предыдущей версии образа;
- База данных — откат миграций (Alembic downgrade);
- Резервные копии — восстановление из бэкапа.

12. Управление конфигурацией ПО

12.1 Элементы конфигурации

Элемент	Описание	Место хранения
Исходный код	Backend, Frontend, FileWatcher	GitLab (приватный репозиторий)
Docker-образы	Собранные образы компонентов	GitLab Container Registry
Конфигурация развёртывания	docker-compose.yml, .env	Репозиторий deploy
Документация	Техническая и пользовательская	Репозиторий arch
Миграции БД	Alembic-миграции	Репозиторий backend
Зависимости	requirements.txt, package.json	Соответствующие репозитории

12.2 Принципы управления конфигурацией

- Все элементы конфигурации находятся под контролем версий;
- Изменения вносятся через merge request с обязательным code review;
- Каждый релиз маркируется тегом в системе контроля версий;
- Версии компонентов синхронизируются (MINOR-версия едина для всех);
- Конфиденциальные данные (пароли, ключи) не хранятся в репозитории;
- Среды (dev, demo, prod) используют отдельные конфигурации.

12.3 Среды

Среда	Назначение	Обновление
dev	Разработка и тестирование	При каждом коммите
demo	Демонстрация заказчиком	По готовности релиз-кандидата
prod	Промышленная эксплуатация	После успешного тестирования на demo

13. Завершение жизненного цикла ПО

13.1 Основания для завершения

Жизненный цикл ПО «МБорд» может быть завершён в следующих случаях:

- Выпуск новой мажорной версии, полностью заменяющей текущую;
- Прекращение потребности в данном классе ПО;
- Невозможность дальнейшей поддержки по техническим причинам;
- Решение правообладателя о прекращении развития продукта.

13.2 Порядок завершения

1. **Уведомление** — пользователи уведомляются не менее чем за 12 месяцев до планируемого прекращения поддержки.
2. **Переходный период** — в течение 6 месяцев после уведомления:
 - Продолжается устранение критических дефектов;
 - Выпускаются обновления безопасности;
 - Предоставляются инструменты миграции на новую версию.
3. **Ограниченная поддержка** — в течение следующих 6 месяцев:
 - Только критические обновления безопасности;
 - Консультации по миграции.
4. **Прекращение поддержки (End of Life):**
 - Прекращение выпуска обновлений;
 - Закрытие каналов технической поддержки для данной версии;
 - Архивирование исходного кода и документации.

13.3 Сохранение данных

При завершении жизненного цикла пользователям предоставляется:

- Инструмент экспорта данных из системы;
- Документация по формату экспортируемых данных;
- Рекомендации по миграции на альтернативные решения;
- Срок не менее 3 месяцев для осуществления экспорта.

13.4 Архивирование

После завершения жизненного цикла правообладатель обеспечивает:

- Архивирование исходного кода и документации;
- Сохранение информации о выпущенных версиях;
- Хранение архива в течение не менее 5 лет.

Документ подготовлен ООО «Метрикор».

Дата последнего обновления: май 2026.