Общество с ограниченной ответственностью «КС-ИТ»

УТВЕРЖДАЮ

Директор

ООО «КС-ИТ»

_____О.В. Иванова

«____» _____202__г.

М. П.

СИСТЕМА РАЗРАБОТКИ БЕЗОПАСНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Регламент управления конфигурацией в рамках жизненного цикла ПО

ИНФОРМАЦИОННАЯ ПЛАТФОРМА ДЕЛЬТА-К

Версия 1

ИСТОРИЯ ПЕРЕСМОТРА ДОКУМЕНТА

Версия	Дата	Описание изменений
1	23.09.2025	Первоначальный выпуск

Содержание

Определения и сокращения

API (Application Programming Interface) - набор правил, методов и протоколов для взаимодействия между программами.

CI/CD – аббревиатура от *Continuous Integration / Continuous Delivery (или Deployment)*:

- **Continuous Integration (CI)** – практика регулярного интегрирования изменений кода в общую ветку, с автоматическим сборкой и тестированием.
- **Continuous Delivery (CD)** – процесс автоматической подготовки и тестирования релизов, чтобы они могли быть развернуты в любой момент.
- **Continuous Deployment (часть CD)** – автоматическое развертывание изменений на рабочие среды после успешного прохождения всех тестов.

Changelog (журнал изменений) – файл, в котором фиксируются все изменения продукта между версиями: новые функции, исправления ошибок, улучшения.

Commit - фиксирование изменений в системе контроля версий (Git)

Git – это распределенная система контроля версий, используемая для отслеживания изменений в файлах проекта.

GitLab CI – система непрерывной интеграции и доставки, встроенная в GitLab. Позволяет автоматически запускать сборку, тестирование и деплой кода при каждом изменении в репозитории.

MAJOR.MINOR.YEAR.MONHTDAY – схема версионирования программного обеспечения:

- MAJOR — крупные изменения, несовместимые с предыдущими версиями;

- MINOR — новые функции, сохраняющие совместимость;
- YEAR.MONTHDAY — дата сборки или выпуска.

Redmine – система управления проектами и задачами с открытым исходным кодом. Используется для ведения баг-трекера, управления проектами, документооборота и совместной работы.

SDK (Software Development Kit) – комплект инструментов, библиотек и документации, предоставляемый разработчикам для создания приложений под конкретную платформу или сервис.

Ticket – запись в системе управления задачами, описывающая конкретную проблему, запрос или работу, которую нужно выполнить. Содержит идентификатор, описание, приоритет, статус и исполнителя.

1. Общие положения

1.1 Назначение документа

Регламент устанавливает единые правила управления конфигурацией программного обеспечения (ПО) в компании, обеспечивает контроль над составом, состоянием и изменениями конфигурационных единиц на всех стадиях жизненного цикла ПО.

Цель управления конфигурацией – сохранить целостность продукта, предотвратить неконтролируемые изменения, повысить управляемость разработки и сопровождения.

1.2 Область применения

Регламент применяется ко всем проектам компании, связанным с:

- разработкой нового ПО;
- модернизацией существующего ПО;
- сопровождением и эксплуатацией ПО у заказчиков;
- интеграцией с внешними системами.

2. Перечень элементов конфигурации ПО

2.1.Общие положения

Перечень элементов конфигурации формируется для того, чтобы определить, какие компоненты и документы подлежат обязательному контролю, версионированию и отслеживанию в рамках жизненного цикла собственного программного продукта.

К элементам конфигурации (далее — КЕ) относятся как артефакты ПО (исходный код, библиотеки, сборки), так и сопроводительная документация.

2.2.Состав элементов конфигурации

В состав КЕ включаются следующие группы:

1. **Исходный код и зависимости**
   - собственные модули, пакеты, сервисы;
   - сторонние библиотеки и SDK, используемые в продукте;
   - конфигурационные файлы, влияющие на работу продукта.
2. **Сборки и артефакты**
   - бинарные файлы (исполняемые модули, библиотеки);
   - установочные пакеты, дистрибутивы;
3. **Инфраструктурные файлы**
   - скрипты сборки и развертывания;
   - CI/CD-конфигурации (GitLab CI и др.);
   - схемы баз данных и миграции.
4. **Документация**
   - техническая документация (архитектура, описание API);
   - эксплуатационная документация (руководства пользователей, администраторов);
   - тестовая документация (тест-кейсы, чек-листы).
5. **Тестовые и эталонные данные**
   - демо-базы данных, тестовые наборы, сценарии нагрузочного тестирования.

2.3.Порядок формирования перечня

- Перечень КЕ формируется исходя из состава продукта и архитектуры, принятой в команде разработки.
- В реестр в обязательном порядке включаются:

  - все модули и сервисы, находящиеся в исходных кодах;

  - артефакты, используемые при релизе и развертывании;

  - документация, необходимая для эксплуатации и сопровождения продукта.

- Формирование перечня КЕ осуществляется архитектором продукта.

2.4.Обновление перечня

- Перечень пересматривается при появлении новых модулей или сервисов, изменении архитектуры, добавлении новой документации.
- Обновление осуществляется по инициативе архитектора продукта.

3. Порядок идентификации ПО

3.1.Идентификация ПО

3.1.1. Идентификация конфигурационных единиц выполняется для обеспечения уникальности и прослеживаемости. Каждый элемент получает версию продукта.

3.2. Идентификация ПО

- Для исходного кода используется Git-репозиторий с присвоением тегов релизам.
- Система версионирования:
  - MAJOR.MINOR.YEAR.MONHTDAY (например, 3.5.25.909).
  - Присвоение версий выполняет архитектор системы после успешного тестирования.

4. Управление версиями и изменениями

4.1. Управление версиями

- Все изменения исходного кода фиксируются в Git с комментариями в commit.
- Релизы маркируются тегами и сопровождаются файлами changelog.
- Базовые линии фиксируются и не подлежат изменению.

4.2. Управление изменениями

Любое изменение конфигурационной единицы инициируется через ticket в системе управления задачами (Redmine).

Ticket должен содержать: описание изменения, обоснование, влияние на систему, ответственного исполнителя.

Решение о внесении изменений принимает архитектор системы.

5. Хранение и доступ

5.1. Системы хранения

- Исходный код – GitLab корпоративный.
- Документация – Redmine или файловое хранилище.
- Сборки – артефактный репозиторий.

5.2. Контроль доступа

- Доступ предоставляется по принципу **минимально необходимого уровня**.
- Администрирование прав доступа выполняет системный администратор по заявке Project Manager или руководителя отдела.

6. Ответственность

- Менеджер конфигурации – ведёт реестр КЕ, контролирует идентификацию и версионность.

- Tech Arch – определяет состав КЕ, обеспечивает хранение и развертывание утверждённых сборок, принимаем решения об изменениях.
- Разработчики – реализуют изменения в соответствии с утверждёнными процессами.
- QA - инженеры – подтверждают корректность релизов.

7. Контроль и аудит

7.1. Аудит конфигурации

Плановый аудит проводится не реже 1 раза в квартал. Внеплановый аудит возможен при инцидентах, связанных с ошибочными версиями.

7.2. Критерии аудита

- наличие всех КЕ в реестре;
- корректность версий;
- соответствие фактического состава ПО утверждённой базовой линии.

7.3. Отчетность

По итогам аудита формируется отчёт, содержащий:

- выявленные несоответствия;
- предложения по устранению;
- сроки и ответственных.

8. Заключительные положения

Регламент вступает в силу с момента утверждения. Все сотрудники компании, вовлечённые в процессы разработки и сопровождения ПО, обязаны соблюдать настоящий регламент. Контроль соблюдения настоящего регламента возлагается на Project Manager и Technical Architect.