

План конфигурационного управления

по проекту «Система учёта IT-сервисов, обеспечивающая хранение досье и инструкций по их использованию и сопровождению» (ITSIMS)

Дата: 01.11.2025

Версия: 1.2 **Статус:** Утверждено

Автор: Носов А.А. Гречко И.В. **Ответственный:** Носов А.А. Гречко И.В

Лист изменений

Версия	Дата	Автор	Описание изменений	Основание
1.0	17.10.2025	Носов А.А.	Первая версия документа	https://www.notion.so/282cf70d42c980d1a58ce054f3b49f7b
1.1	17.10.2025	Gregory K	Обновление структуры и ролей	Приведение документа к общему формату
1.2	01.11.2025	Гречко И. В	Обновление структуры репозетория	Приведение названий документов к общему формату

Подпись таблицы: Таблица 1. Лист изменений.

Оглавление

1. Цели и область применения
2. Организация КУ - роли и ответственность
3. Конфигурационные единицы (КЕ) - классификация и CIL
 - 3.1 Классификация КЕ
 - 3.2 Реестр конфигурационных единиц (CIL)
4. Политика идентификации и именования
5. Процесс управления изменениями
 - 5.1 Стадии процесса

1. Цели и область применения

Цель: обеспечить контроль над идентификацией, выпуском, изменением и хранением всех артефактов проекта ITSIMS, гарантию отслеживаемости и возможности восстановления конкретных конфигураций системы.

Область применения: документ распространяется на все артефакты проекта: ТЗ, проектные документы, исходные коды, инструкции, эксплуатационные досье, базы данных, скрипты миграции, конфигурационные файлы, скрипты развертывания, сборки и т.д.

2. Организация КУ - роли и ответственность

Роль	ФИО	Обязанности
Менеджер конфигурации	Носов А.А.	Составление и актуализация Плана конфигурационного управления.
Руководитель проекта / Тимлид	Баранов А.Т.	Утверждение графиков, принятие ключевых решений по проекту, распределение задач и ответственности среди членов команды.
Представитель заказчика	Суставов С.А.	Утверждает изменения, контролирует соответствие требованиям.
Специалист по документированию (Технический писатель)	Когановский Г.И.	Разработка и поддержка шаблонов, стандартов и стилей для проектной документации. Контроль за соблюдением правил оформления документов.
Руководитель обеспечения качества (QA Lead)	Зацепилин А.В.	Планирование и контроль процессов тестирования, проведение инспекций.
Администратор репозитория	Гречко И.В.	Управление структурой и доступом к хранилищу проекта (репозиторию), контроль версионности и наименования артефактов, выдача прав доступа.
Архитектор / Аналитик требований	Вальковец Д.И.	Формирование и описание функциональных требований к системе, разработка содержательной части технического задания и другой технической документации.
Специалист по поддержке проектных процессов	Ловков К.И.	Оказание помощи членам команды в выполнении вспомогательных и рутинных задач, связанных с подготовкой документации и сопровождением проектных процессов.

2.1 Правила структуры репозитория (полный текст)

Правила структуры репозитория и правила размещения создаваемых документов в нем

1. Общие положения

Настоящий документ описывает: - структуру каталогов репозитория проекта; - правила размещения создаваемых документов и исходных файлов; - требования к поддержанию порядка в структуре и именованиях. Все участники команды обязаны соблюдать данную структуру при добавлении новых материалов в репозиторий. ## 2. Основная структура репозитория

ROOT/
|

└─ planning/	# Планирование, стандарты, шаблоны и процессы
└─ standards/	# Регламенты и правила
└─ repo/	# Правила по репозиторию
└─ docs/	# Правила оформления документов
└─ templates/	# Шаблоны документов (например, ТЗ)
└─ processes/	# Методики, гайды, формальные процедуры
└─ diagrams/	# Диаграммы, схемы, жизненные циклы
└─ deliverables/	# Готовые артефакты проекта
└─ specs/	
└─ tech_spec/	# ТЗ/технические спецификации
└─ requirements/	# Системные требования (MD/PDF)
└─ code/	# Исходный код системы
└─ tools/	# Скрипты и ресурсы конвертации/сборки
└─ README.md	

3. Правила размещения файлов

3.1 planning/

- Содержит материалы, управляющие проектом: стандарты, шаблоны, процессы и диаграммы.
- Рекомендуемая структура уже закреплена: standards/, templates/, processes/, diagrams/.
- Внутри standards/ разделяйте правила по тематике (например, repo/, docs/).
- Не создавайте чрезмерную глубину: если в каталоге 2–3 файла, держите их на одном уровне.

3.2 deliverables/

- Содержит готовые, согласованные артефакты:
 - specs/tech_spec/ — ТЗ/технические спецификации;
 - requirements/ — системные требования (в т.ч. экспорт в PDF);
 - code/ — исходный код системы.
- Для рабочих версий документов используйте ветки/PR; в deliverables/ попадают утверждённые версии.
- Именование файлов должно быть осмысленным и стабильным (например, spec_technical.md, spec_system_requirements.pdf).

3.3 tools/

- Содержит скрипты и ресурсы для сборки/конвертации (например, Pandoc/LaTeX ресурсы).
- Скрипты должны работать из любой текущей директории, использовать относительные пути от расположения скрипта.
- Не являются частью проекта, используются как вспомогательный инструмент

3. Конфигурационные единицы (KE) - классификация и CIL

Конфигурационные единицы (Configuration Items, CI) - это объекты, на которые распространяется управление конфигурацией в проекте ITSIMS.

Все КЕ регистрируются в *реестре конфигурационных единиц (CIL)* и отслеживаются на всём жизненном цикле.

3.1 Классификация КЕ

Класс КЕ	Обозначение	Состав и описание
Общие правила	rule	Документы, определяющие общие регламенты проекта: структуру репозитория, правила наименования файлов и другие стандарты организации и оформления материалов.
Технические спецификации	spec	Документы, описывающие требования и характеристики реализуемых решений. Включают шаблон технического задания и конкретные спецификации, определяющие цели, функционал, ограничения и критерии приемки.
Диаграммы	diagram	Графические материалы, иллюстрирующие процессы и структуры. Включают диаграммы жизненного цикла, архитектуры, взаимодействий и других аспектов системы. На текущем этапе содержит диаграмму жизненного цикла технического задания.
Принятые решения	decision	Документы, фиксирующие ключевые проектные решения и их обоснование. На текущем этапе содержит решение о формате файлов для документов с требованиями, направленное на упрощение индексирования и отслеживания покрытия требований.

3.2 Реестр конфигурационных единиц (CIL)

Все документы хранятся в репозитории GitHub по ссылке:

<https://github.com/ITSIMS/ITSIMS-IT-Service-Inventory-and-Management-System>

Все дальнейшие указания пути хранения начинаются относительно корня ветки “main” указанного репозитория.

CI ID	Название КЕ	Тип	Владелец (ФИО)	Хранилище / путь
spec technical	Техническое задание ITSIMS	spec	Вальковец Д.И.	deliverables/specs/tech_spec
rule configuration management	План конфигурационного управления	rule	Носов А.А.	planning/standards/docs

CI ID	Название KE	Тип	Владелец (ФИО)	Хранилище / путь
spec technical template	Шаблон технического задания ITSIMS	spec	Когановский Г.И.	planning/templates
decision document format	Решение о формате документов с требованиями	decision	Когановский Г.И.	planning/processes
rule file naming	Правила наименования документов	rule	Гречко И. В.	planning/standards/repo
rule repository structure	Правила структурирования репозитория	rule	Гречко И. В.	planning/standards/repo
diagram spec lifetime	Диаграмма жизненного цикла ТЗ	diagram	Ловков К. И.	planning/diagrams
rules spec formatting	Правила оформления ТЗ	rule		planning/standards/docs

4. Политика идентификации и именования

Базовые правила (согласно `planning/standards/repo/guideline_naming_rules.md`):

- Имена файлов — на английском, в формате `snake_case`.
- Не допускаются пробелы, кириллица и специальные символы.
- Имя отражает содержание, а не дату или версию.
- Версии и даты не включаются в имя файла — их отслеживает Git.
- Расширение стандартное: `.md`, `.pdf`, `.docx`, `.png`, `.tsx` и т.п.

Структура имени файла:

`<content_type>_<short_description>.<extension>`

Примеры: `spec_requirements.md`, `diagram_er_model.png`, `report_verification_results.`

4.1 Полный текст правил наименования

Правила наименования файлов

1. Общие принципы

1. Все имена файлов пишутся **на английском языке**.
2. Используется **snake_case** — слова разделяются нижним подчёркиванием.
3. В именах **не используются**:
 - пробелы;
 - кириллица;
 - специальные символы (`!@#$%^&*()+=[]{};'"<>/?`);
4. Название файла должно **отражать его содержание**, а не формат или дату.
5. **Версии и даты не указываются** в названии — их отслеживает Git (история коммитов).
6. Расширение файла — стандартное: `.md`, `.docx`, `.pdf`, `.tsx`, `.png`, `.xlsx` и т.п. **## 2. Структура имени файла**

`<content_type>_<short_description>.<extension>`

где: `-content_type` — тип документа (например, `report`, `diagram`, `spec`); `-short_description` — краткое описание содержимого; `-<extension>` — формат файла. **### Примеры**

spec_requirements.docx diagram_er_model.png report_team_feedback.pdf component_question_editor.tsx

3. Примеры типов документов

Тип	Назначение / содержание	Примеры файлов
spec	Спецификация, техническое задание, требования	spec_requirements.docx, spec_functional.md
report	Отчёты, аналитические документы, результаты проверки	report_verification_results.pdf, report_analysis.docx
diagram	Диаграммы, схемы, графические материалы	diagram_er_model.png, diagram_architecture.drawio
model	Формальные модели (данных, классов, процессов)	model_restaurant_domain.md, model_user_roles.png
plan	Планы, дорожные карты, расписания	plan_development.docx, plan_verification.xlsx
note	Небольшие пояснительные документы, внутренние заметки	note_discussion_summary.md
component	Элементы исходного кода (фронтенд, backend, утилиты)	component_question_editor.tsx, component_drag_handler.ts
data	Исходные данные, тестовые примеры, CSV- или JSON-файлы	data_test_cases.json, data_menu_items.csv
form	Шаблоны и формы для заполнения (инспекции, отчёты)	form_inspection_checklist.docx
guideline	Руководства, стандарты, правила оформления	guideline_coding_style.md, guideline_document_structure.md

5. Процесс управления изменениями

Процесс изменения конфигурации ITSIMS регламентируется следующей последовательностью:

5.1 Стадии процесса

№	Этап	Описание
1	Инициирование RFC	Подача заявки на изменение (форма RFC)
2	Регистрация RFC	Присвоение номера и занесение в журнал
3	Анализ влияния	Оценка технических и рисковых последствий
4	Рассмотрение CCB	Принятие решения (Approve / Reject / Defer)
5	Реализация	Внесение изменений
6	Тестирование	Проведение тестов
7	Верификация и релиз	Обновление CIL, выпуск версии
8	Архивация и отчёт	Подготовка отчёта о внедрении