**1. Цели и результат проекта «PTE-QR»**

* **Цель:** обеспечить на каждом листе PDF-документа машиночитаемую ссылку (QR) на онлайн-проверку актуальности конкретной **ревизии** документа и **страницы**, с понятным ответом:
  1. «**Принята Заказчиком**»,
  2. «**Утверждена в производство работ**»,
  3. «**Внесены изменения — получите новый документ**»,
  4. тех. детали (номер ревизии, номер изменения, дата утверждения/изменения, ссылка на актуальную версию — согласно правам доступа).
* **Результат:** сервер с API + фронтенд-страница проверки, библиотека/утилита встраивания QR-кодов в PDF, регламент интеграции с PLM (ENOVIA), корпоративная политика безопасности, журналирование и отчётность.

**2. Глоссарий и основные сущности**

* **DocumentID** — постоянный идентификатор документа в ENOVIA (например, 3DSpace/ENOVIA PLMExternalID).
* **Revision** — ревизия/издание в ENOVIA (A, B, C… или 1, 2, 3).
* **Change** – номер изменения.
* **Page** — номер страницы PDF.
* **Maturity/State** — статус жизненного цикла в ENOVIA (In Work / Frozen / Released и т. п.).
* **Status (бизнес-отображение):**
  + Accepted → «Принята Заказчиком»;
  + AFC/Released → «Утверждена в производство работ»;
  + Outdated/Obsolete/ superseded → «Внесены изменения — получите новый документ».

**3. Роли пользователей**

* **Гость по QR:** видит факт актуальности и минимальные метаданные (без скачивания файлов).
* **Сотрудник Заказчика и ПТИ (аутентифицированный):** видит расширенные сведения, ссылки на актуальную ревизию в ЛК Заказчика.
* **Админ/ОНК:** доступ к журналам, настройкам маппинга статусов, webhook-нотификациям, отчётам.

**4. Архитектура (высокоуровнево)**

* **Сервис QR/Status (Backend, REST API)**
  + Модули: авторизация (SSO), коннектор ENOVIA, кеш (Redis), БД (PostgreSQL), генерация QR, PDF-stamper (на стороне ENOVIA при сохранении документа в системе).
* **Frontend (Single Page / SSR)**
  + Страница резолвинга QR: показывает статус, карточку документа, ссылки (с учётом прав).
* **Интеграция с ENOVIA**
  + Service Account (3DPassport/OAuth2) для чтения метаданных документа и текущей ревизии/статуса.
  + Вариант: webhooks/подписки для инвалидации кеша при изменениях.
* **Хранилище**
  + Таблица кэша статусов, журнал обращений, карта соответствия DocumentID↔Revision (при необходимости).

**5. Формат QR и URL-схема**

* **Читаемое содержимое QR (URL):**

https://qr.pti.ru/r/{docUid}/{rev}/{page}?t={sig}&ts={unix}

где:

* + docUid — ENOVIA DocumentID (или стабильный UUID-прокси),
  + rev — ревизия, **обязательна** для однозначности,
  + page — номер страницы,
  + ts — метка времени выпуска QR,
  + sig — HMAC-подпись (SHA-256) по строке {docUid}|{rev}|{page}|{ts}, ключ хранится в KMS.
* **Почему подпись:** исключить подмену URL/параметров на распечатках.
* **Короткий домен:** допустимо применять pti.is/… для компактности QR.

**6. User Stories (ключевые сценарии)**

1. **Как рабочий на площадке,** я сканирую QR на листе и вижу «Утверждена в производство работ / ревизия B / дата 2025-09-10».
2. **Как Технический Заказчик,** я сканирую старый лист и получаю «Внесены изменения — получите новый документ», с кнопкой «Открыть актуальную ревизию» (по авторизации).
3. **Как Авторский Надзор,** я выгружаю PDF c вшитыми QR на каждой странице и уверен, что URL подписан и не изменится.
4. **Как админ,** я настраиваю соответствие ENOVIA-статусов бизнес-статусам (mapping), вижу отчёты по сканированиям.

**7. Интеграция с ENOVIA**

* **Аутентификация:** 3DPassport OAuth2 Client Credentials (service account) → access token.
* **Методы коннектора (внутренние):**
  + getDocumentMeta(docUid) → { title, number, currentRevision, maturityState, lastModified, … }
  + getRevisionMeta(docUid, rev) → { maturityState, releasedDate, supersededBy, … }
  + (опц.) getLatestReleased(docUid) → последняя актуальная ревизия.
* **Маппинг статусов:** конфигурационная таблица:

| **ENOVIA.maturity** | **BusinessStatus** | **Color** | **Action** |
| --- | --- | --- | --- |
| Released/AFC | Утверждена в производство работ | Green | «Открыть документ» |
| Accepted/Approved by Customer | Принята Заказчиком | Blue | «Открыть документ» |
| Obsolete/Superseded | Внесены изменения | Red | «Скачать актуальную» |
| In Work | На доработке | Gray | «Обратиться к ответственному» |

**8. Backend API (внешний контракт)**

**Аутентификация API:** JWT (RS256), роли: guest, user, admin. Rate limiting.

1. **Проверка статуса (использует фронтенд):**

GET /api/v1/documents/{docUid}/revisions/{rev}/status?page={page}

**200:**

{

"docUid":"…",

"revision":"B",

"page": 3,

"businessStatus":"APPROVED\_FOR\_CONSTRUCTION",

"enoviaState":"Released",

"isActual": true,

"releasedAt":"2025-09-10T10:32:00Z",

"supersededBy": null,

"links": {

"openDocument":"https://…/enovia/3dspace/doc/…",

"openLatest": null

}

}

1. **404:** документ/ревизия не найдены.

**410:** ревизия устарела (вернуть isActual=false и links.openLatest).

1. **Генерация QR (для ПД или CI/CD):**

POST /api/v1/qrcodes

**Body:**

{ "docUid":"…", "revision":"B", "pages":[1,2,3], "style":"black", "dpi":300 }

1. **200:** массив PNG/SVG (base64) с URL и подписью sig.

Опция: mode:"pdf-stamp" → вернуть готовый PDF с вшитыми QR (штамп/угол листа/поля — по профилю).

1. **Валидатор подписи (опц.):**

GET /api/v1/qr/verify?docUid=…&rev=…&page=…&ts=…&sig=… → 200/400.

1. **Админ-настройки маппинга:**

GET/PUT /api/v1/admin/status-mapping

1. **Служебные:**

GET /api/v1/health | GET /api/v1/metrics | GET /api/v1/audit?from=&to=&docUid=…

**9. Frontend (страница резолвинга QR)**

* **Маршрут:** GET /r/{docUid}/{rev}/{page}?t={sig}&ts={unix}

Шаги:

* 1. Проверить sig локально (HMAC), при невалидности — «QR повреждён/подделан».
  2. Вызвать /api/v1/documents/{docUid}/revisions/{rev}/status?page={page}.
  3. Отобразить карточку:
     + Крупный индикатор: **Актуален / Не актуален** (цвет/иконка).
     + Поля: Наименование, Номер, Ревизия, Дата релиза/утв., Страница.
     + Если устарел: показать кнопку «Перейти к актуальной ревизии» (требует входа/прав).
     + QR-история (опц.): дата генерации, кем выпущен.
  4. Авторизация (SAML/OIDC) → доступ к ссылкам на файлы.
* **Адаптация под стройплощадку:** ультра-контрастная тема, офлайн-кэш под загрузку статуса (TTL 5–15 мин), крупные кнопки.

**10. Встраивание QR в PDF/чертёж**

* **Профили размещения:**
  + «Правый нижний угол страницы документа» (35×35 мм)
  + «Правый верхний угол над основной надписью» (35×35 мм),
  + «Верхнее поле» (для схем).
* **Размер QR:** 35–35 мм на стороне; ECC = M или Q; обязательная подпись URL или короткая подпись PTI QR.
* **Технически:**
  + Серверная простановка (pdf-stamp) — при выпуске ревизии и ее созранении в ENOVIA.
  + Плагины/скрипты для CAD/PDM экспорта — опционально.
  + Проверка контраста, белой подложки, контроль зоны тишины (quiet zone ≥ 4 modules).

**11. Безопасность и доступ**

* Подпись QR (HMAC SHA-256, секрет — в KMS/HSM).
* JWT (RS256), ролевой доступ.
* Минимизация данных гостю (только факт актуальности + минимум метаданных).
* CORS: deny by default, allow фронтенд-домен.
* Rate-limit + WAF.
* Аудит-лог: кто/когда сканировал (анонимный/авторизованный), какие статусы показаны.
* GDPR/152-ФЗ: не собирать лишних персональных данных, IP — в агрегированную аналитику.

**12. Производительность и надёжность**

* **SLA чтения:** p95 < 500 мс (при попадании в кеш), p95 < 1500 мс (промах в кеш → ENOVIA).
* **Кеширование:** Redis на {docUid,rev} с TTL 5–15 мин; инвалидация по webhook из ENOVIA.
* **Нагрузочное тестирование:** 100 RPS на status с долей промахов ≤ 20%.
* **Доступность:** 99.9%/месяц; health-checks, авто-рестарт, горизонтальное масштабирование.

**13. Логи и метрики**

* **Метрики:** RPS, p50/p95/p99, error rate, доля устаревших ревизий, средний «возраст» отсканированных листов, гео/время суток.
* **Логи:** request/response IDs, docUid, rev, результат, пользователь/роляметка, ENOVIA-latency.
* **Дэшборд:** Grafana/Prometheus; алерты по SLA.

**14. Набор тестов и приёмка**

* **Юнит-тесты:** подпись/валидация QR, парсинг URL, маппинг статусов.
* **Интеграционные:** mock ENOVIA → статусы «Released», «In Work», «Obsolete»; кеш-хит/мис; сбой ENOVIA.
* **E2E:** сканирование реального QR (эмулятор), проверка трёх основных ответов, авторизация и переход к актуальной ревизии.
* **Регрессы:** после смены маппинга статусов и ключей подписи.
* **Приёмочные критерии:**
  1. Для «Released/AFC» отображается «Утверждена…», isActual=true.
  2. Для устаревшей ревизии — «Внесены изменения…», isActual=false, ссылка на актуальную ревизию (для авторизованных).
  3. Подделка URL/параметров даёт отказ (невалидная sig).
  4. Пиковая нагрузка и SLA подтверждены отчётом.

**15. Нефункциональные требования**

* **Стек (рекомендация):**
  + Backend: Python FastAPI / Node.js (NestJS);
  + Frontend: React/Next.js;
  + DB: PostgreSQL (метаданные, аудит), Redis (кеш);
  + QR: qrcode/segno (Python) или qrcode (Node); PDF-stamp: pikepdf/PyPDF2 или pdf-lib (Node).
* **Контейнеризация:** Docker; окружения DEV/TEST/PROD; IaC (Terraform/Ansible).
* **CI/CD:** линт, тесты, SAST, контейнер-скан, деплой с миграциями.

**16. Миграция/внедрение**

* Пилот на одном направлении работ (1–2 подразделения), 2–4 недели.
* Обучение ОНК и ИТР (инструкция по чтению статуса по QR).
* Постепенная ретро-простановка QR на ключевые комплекты (опц., массовый pdf-stamp пачками).

**17. Риски и меры**

* **Несовместимость статусов ENOVIA:** сделать настраиваемый mapping + справочник.
* **Плохая сканируемость QR:** контроль контраста, размер, тестовые вычитки на типовых плоттерах.
* **Отсутствие связи на площадке:** фронтенд с вежливым офлайн-сообщением и повторной попыткой; кеш на стороне сервера короткий.
* **Секреты/ключи:** KMS, ротация, dual-control.
* **Юзабилити:** крупный UI, светлая/тёмная темы, локализация RU/EN.

**18. Артефакты, которые нужно изготовить**

1. Спецификация API (OpenAPI/Swagger).
2. Техническая схема интеграции с ENOVIA (диаграмма последовательностей).
3. Руководство по генерации и простановке QR (PDF-stamp и плагины).
4. Политика безопасности и регламент ротации ключей.
5. План тестирования и акт приёмки.
6. Инструкция для пользователей стройплощадки (1 страница, с картинками).

# PTE-QR: OpenAPI‑спецификация, диаграмма последовательностей и регламент QR‑размещения

Ниже представлены три артефакта:

1. OpenAPI (YAML),
2. диаграмма последовательностей (Mermaid),
3. шаблон регламента QR‑размещения для документов ПТИ.

## 1) OpenAPI 3.0.3 (YAML)

openapi: 3.0.3

info:

title: PTE QR/Status API

version: 1.0.0

description: |

API для генерации QR-кодов и проверки актуальности листов документации (ENOVIA/3DEXPERIENCE).

Поддерживает проверку статуса конкретной ревизии документа и страницу PDF.

servers:

- url: https://api.qr.pte.ru

description: Прод

- url: https://sandbox.api.qr.pte.ru

description: Тест

security:

- bearerAuth: []

paths:

/api/v1/health:

get:

summary: Health check

responses:

'200':

description: OK

/api/v1/metrics:

get:

summary: Метрики Prometheus (только для внутренней сети)

responses:

'200':

description: Prometheus metrics text

/api/v1/documents/{docUid}/revisions/{rev}/status:

get:

summary: Проверка статуса конкретной ревизии документа и страницы

parameters:

- in: path

name: docUid

required: true

schema: { type: string }

description: ENOVIA DocumentID (или стабильный UUID-прокси)

- in: path

name: rev

required: true

schema: { type: string }

description: Ревизия документа (A/B/C или 1/2/3)

- in: query

name: page

required: true

schema: { type: integer, minimum: 1 }

description: Номер страницы PDF

responses:

'200':

description: Успешно

content:

application/json:

schema:

$ref: '#/components/schemas/StatusResponse'

examples:

released:

value:

docUid: 3D-00001234

revision: 2

page: 3

businessStatus: APPROVED\_FOR\_CONSTRUCTION

enoviaState: Released

isActual: true

releasedAt: '2025-09-10T10:32:00Z'

supersededBy: null

links:

openDocument: 'https://enovia.pti.ru/3dspace/document/3D-00001234?rev=B'

openLatest: null

'404': { description: Документ или ревизия не найдены }

'410':

description: Ревизия устарела

content:

application/json:

schema:

$ref: '#/components/schemas/StatusResponse'

examples:

obsolete:

value:

docUid: 3D-00001234

revision: A

page: 3

businessStatus: CHANGES\_INTRODUCED\_GET\_NEW

enoviaState: Obsolete

isActual: false

releasedAt: '2024-06-01T08:00:00Z'

supersededBy: 'B'

links:

openDocument: 'https://enovia.pti.ru/3dspace/document/3D-00001234?rev=A'

openLatest: 'https://enovia.pti.ru/3dspace/document/3D-00001234?rev=B'

'401': { description: Неавторизован }

'429': { description: Слишком много запросов }

/api/v1/qrcodes:

post:

summary: Генерация QR-кодов для набора страниц

requestBody:

required: true

content:

application/json:

schema:

$ref: '#/components/schemas/QRCodeRequest'

examples:

example1:

value:

docUid: 3D-00001234

revision: B

pages: [1,2,3]

style: black

dpi: 300

mode: images

responses:

'200':

description: Сгенерированные QR (PNG/SVG как base64)

content:

application/json:

schema:

$ref: '#/components/schemas/QRCodeResponse'

'400': { description: Ошибка запроса }

'401': { description: Неавторизован }

'403': { description: Недостаточно прав }

/api/v1/qr/verify:

get:

summary: Валидация подписи параметров QR (опционально для отладки)

parameters:

- in: query

name: docUid

required: true

schema: { type: string }

- in: query

name: rev

required: true

schema: { type: string }

- in: query

name: page

required: true

schema: { type: integer }

- in: query

name: ts

required: true

schema: { type: integer }

- in: query

name: sig

required: true

schema: { type: string }

responses:

'200':

description: Подпись валидна

'400':

description: Невалидные параметры или подпись

/api/v1/admin/status-mapping:

get:

summary: Получить текущий маппинг статусов ENOVIA → BusinessStatus

responses:

'200':

description: Текущая конфигурация

content:

application/json:

schema:

$ref: '#/components/schemas/StatusMapping'

'401': { description: Неавторизован }

'403': { description: Недостаточно прав }

put:

summary: Обновить маппинг статусов (только admin)

requestBody:

required: true

content:

application/json:

schema:

$ref: '#/components/schemas/StatusMapping'

responses:

'204': { description: Обновлено }

'400': { description: Ошибка валидации }

'401': { description: Неавторизован }

'403': { description: Недостаточно прав }

/r/{docUid}/{rev}/{page}:

get:

summary: Frontend‑маршрут резолвинга QR (SSR или редирект на веб‑клиент)

parameters:

- in: path

name: docUid

required: true

schema: { type: string }

- in: path

name: rev

required: true

schema: { type: string }

- in: path

name: page

required: true

schema: { type: integer }

- in: query

name: ts

required: true

schema: { type: integer }

- in: query

name: t

required: true

schema: { type: string }

responses:

'200': { description: HTML }

'400': { description: Неверная подпись }

'302': { description: Редирект на SPA }

components:

securitySchemes:

bearerAuth:

type: http

scheme: bearer

bearerFormat: JWT

schemas:

StatusResponse:

type: object

required: [docUid, revision, page, businessStatus, enoviaState, isActual]

properties:

docUid: { type: string }

revision: { type: string }

page: { type: integer }

businessStatus:

type: string

enum: [APPROVED\_FOR\_CONSTRUCTION, ACCEPTED\_BY\_CUSTOMER, CHANGES\_INTRODUCED\_GET\_NEW, IN\_WORK]

enoviaState: { type: string }

isActual: { type: boolean }

releasedAt: { type: string, format: date-time, nullable: true }

supersededBy: { type: string, nullable: true }

links:

type: object

properties:

openDocument: { type: string, nullable: true }

openLatest: { type: string, nullable: true }

QRCodeRequest:

type: object

required: [docUid, revision, pages]

properties:

docUid: { type: string }

revision: { type: string }

pages:

type: array

items: { type: integer, minimum: 1 }

minItems: 1

style:

type: string

enum: [black, inverted, with\_label]

default: black

dpi:

type: integer

minimum: 96

maximum: 1200

default: 300

mode:

type: string

enum: [images, pdf-stamp]

default: images

QRCodeItem:

type: object

properties:

page: { type: integer }

format:

type: string

enum: [png, svg, pdf]

dataBase64: { type: string }

url: { type: string, description: 'URL, закодированный в QR' }

QRCodeResponse:

type: object

properties:

docUid: { type: string }

revision: { type: string }

items:

type: array

items:

$ref: '#/components/schemas/QRCodeItem'

StatusMapping:

type: object

additionalProperties:

type: object

properties:

businessStatus:

type: string

enum: [APPROVED\_FOR\_CONSTRUCTION, ACCEPTED\_BY\_CUSTOMER, CHANGES\_INTRODUCED\_GET\_NEW, IN\_WORK]

color: { type: string }

actionLabel: { type: string }

## 2) Диаграмма последовательностей (Mermaid)

sequenceDiagram

actor U as Пользователь (сканер/смартфон)

participant FE as Frontend (SSR/SPA)

participant API as QR/Status API

participant Cache as Redis Cache

participant ENO as ENOVIA Connector

U->>FE: GET /r/{docUid}/{rev}/{page}?ts=...&t=sig

FE->>FE: Проверка HMAC-подписи (t)

alt Невалидная подпись

FE-->>U: 400 QR повреждён/подделан

else Валидная подпись

FE->>API: GET /api/v1/documents/{docUid}/revisions/{rev}/status?page=page

API->>Cache: GET status {docUid,rev}

alt Кеш-хит

Cache-->>API: Status OK

else Кеш‑промах

API->>ENO: getRevisionMeta(docUid, rev)

ENO-->>API: { enoviaState, releasedDate, supersededBy, ... }

API->>Cache: SET status TTL=5..15m

end

API-->>FE: 200 {isActual, businessStatus, links, ...}

alt isActual=true

FE-->>U: Показать «Актуален (Утверждён/Принят)», кнопка «Открыть документ»

else isActual=false

FE-->>U: «Внесены изменения — получите новый документ», кнопка «Перейти к актуальной ревизии»

end

end

## 3) Шаблон регламента QR‑размещения (ПТИ)

**Документ:** Регламент простановки QR‑кодов на листах проектной и рабочей документации ПТИ

**Код документа:** PTE‑QR‑REG‑XXX

**Версия:** 1.0

**Дата вступления:** XX.XX.XXXX

**Ответственный владелец:** Разработчик ответственный за выпуск комплекта и его запуск по маршруту до подписания УКЭП.

### 1. Назначение и область применения

1.1. Регламент устанавливает единые правила генерации, простановки и проверки QR‑кодов на листах PDF‑документов, загружаемых в ENOVIA/3DEXPERIENCE.

1.2. Обязателен для всех подразделений ПТИ, выпускающих техническую документацию, а также для подрядных организаций, работающих по стандартам ПТИ.

### 2. Термины и сокращения

QR, DocumentID, Revision, Page, ENOVIA Maturity/State, BusinessStatus (Принята Заказчиком / Утверждена в производство работ / Внесены изменения…).

### 3. Ответственность и роли

* **ОНК:** методология, контроль соответствия, утверждение мест размещения, выбор профиля штампа.
* **Проектные отделы:** выпуск PDF с QR, проверка читабельности и корректности статуса при выпуске.
* **Администратор ENOVIA/IT:** доступы, ключи подписи, интеграция, кеш/вебхуки, мониторинг.
* **Технический заказчик/подрядчики:** использование QR на площадке для проверки актуальности.

### 4. Требования к QR‑коду и URL

4.1. Содержимое QR — URL вида: https://qr.pte.ru/r/{docUid}/{rev}/{page}?ts={unix}&t={sig}.

4.2. Подпись t — HMAC‑SHA256 по строке {docUid}|{rev}|{page}|{ts}; ключ хранится в KMS/HSM; ротация не реже 1 раза в 6 месяцев.

4.3. Корректность URL проверяется фронтендом; при невалидной подписи выводится предупреждение «QR повреждён/подделан».

### 5. Размещение на листе

5.1. Базовые профили:

* Профиль А: правый нижний угол, 35×35 мм.
* Профиль B: правый верх над основной надписью, 35×35 мм.
* Профиль C: верхнее поле (для схем/планов), 35×35 мм.

5.2. **Зона тишины:** ≥ 4 модуля; **ECC:** не ниже M; **Контраст:** коэффициент ≥ 4:1; подложка белая.

5.3. Текстовая подпись под QR: PTI QR или короткий URL‑идентификатор.

### 6. Процесс генерации и простановки

6.1. Генерация выполняется сервисом PTE QR/Status API (endpoint /api/v1/qrcodes).

6.2. Режимы: **images** (PNG/SVG на страницу) и **pdf‑stamp** (серверная простановка).

6.3. Штамповка производится автоматически при переводе документа в состояние «Готов к выпуску» в ENOVIA.

6.4. При ручном выпуске ответственное подразделение использует утверждённый профиль размещения и проверяет превью.

### 7. Проверка и приёмка

7.1. Перед публикацией обязательны автоматические проверки: валидность подписи УКЭП, корректность параметров URL, читаемость QR (модуль‑тест AQL).

7.2. Контрольная выборка: не менее 5 листов из комплекта, включая схемы с плотной графикой.

7.3. Результат E2E‑теста: скан с мобильного устройства (iOS/Android) → отображение статуса и корректных ссылок.

### 8. Отображение статуса

8.1. **Актуален:** «Утверждена в производство работ» / «Принята Заказчиком» (зелёный/синий индикатор).

8.2. **Не актуален:** «Внесены изменения — получите новый документ» (красный индикатор) + кнопка перехода к актуальной ревизии (по правам).

8.3. Для статуса «In Work» — серый индикатор, без выдачи материалов.