

# ОПИСАНИЕ ФУНКЦИОНАЛЬНЫХ ХАРАКТЕРИСТИК

## Программного обеспечения «Konkord DSP» версия 1.0.0

Дата разработки: 1 сентября 2025

Классификация: Digital Signage система управления контентом

Платформы: Windows, Linux, Android

Язык интерфейса: Русский

Язык документации: Русский

## 1. НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ

Konkord DSP — это централизованная система управления контентом для Digital Signage (цифровой информационной вывески). Система предназначена для управления мультимедийным контентом на множестве информационных дисплеев одновременно.

### Основное назначение:

- Загрузка и управление мультимедийным контентом (видео, изображения)
- Создание расписаний показа контента с поддержкой временных диапазонов и дней недели
- Централизованное управление информационными дисплеями в локальной сети
- Синхронизация контента между сервером и дисплеями в реальном времени

### Область применения:

- Офисные здания и бизнес-центры
- Торговые центры и магазины
- Банки и финансовые учреждения
- Гостиницы и отели
- Аэропорты и вокзалы
- Медицинские учреждения
- Образовательные учреждения (школы, ВУЗы)
- Любые организации, требующие управления информационными экранами

## 2. ФУНКЦИОНАЛЬНЫЕ ХАРАКТЕРИСТИКИ

### 2.1 Управление клиентами (дисплеями)

#### Функции:

- Регистрация новых дисплеев в системе
- Присвоение уникального ID каждому клиенту
- Хранение информации о месторасположении дисплея
- Отслеживание статуса клиента (online/offline)
- Запись времени последнего подключения
- Редактирование параметров клиента

- Мягкое удаление клиентов (без физического удаления)
- Получение списка всех зарегистрированных клиентов

#### API эндпоинты:

- `POST /api/clients` — создание клиента
  - `GET /api/clients` — получение списка (с пагинацией)
  - `GET /api/clients/{id}` — получение данных конкретного клиента
  - `PUT /api/clients/{id}` — обновление данных клиента
  - `DELETE /api/clients/{id}` — удаление клиента
- 

## 2.2 Управление медиафайлами

#### Поддерживаемые форматы:

- Видео: MP4, WebM, Matroska (MKV)
- Изображения: PNG, JPEG, GIF, WebP

#### Функции:

- Загрузка медиафайлов на сервер
- Автоматический расчет размера файла
- Вычисление хеша (SHA256) для проверки целостности
- Получение информации о медиафайле (размер, тип, разрешение)
- Скачивание загруженного файла
- Удаление файлов (с проверкой на использование в плейлистах)
- Получение списка загруженных файлов

#### Ограничения:

- Максимальный размер файла: 500 MB
- Только изображения и видео файлы
- Проверка целостности по SHA256

#### API эндпоинты:

- `POST /api/media/upload` — загрузка файла
  - `GET /api/media` — получение списка
  - `GET /api/media/{id}` — информация о файле
  - `GET /api/media/{id}/download` — скачивание файла
  - `DELETE /api/media/{id}` — удаление файла
- 

## 2.3 Управление плейлистами

#### Функции:

- Создание новых плейлистов
- Редактирование названия и описания
- Добавление медиафайлов в плейлист
- Удаление элементов из плейлиста
- Переупорядочивание элементов (drag-and-drop в UI)
- Автоматическое версионирование (версия увеличивается при каждом сохранении)

- Активация/деактивация плейлистов
- Получение списка всех плейлистов
- Удаление плейлистов (если не используются)

API эндпоинты:

- `POST /api/playlists` — создание плейлиста
- `GET /api/playlists` — получение списка
- `GET /api/playlists/{id}` — получение плейлиста с элементами
- `PUT /api/playlists/{id}` — обновление плейлиста
- `DELETE /api/playlists/{id}` — удаление плейлиста

## 2.4 Расширенное расписание

Параметры расписания для каждого элемента плейлиста:

### 1. Длительность отображения (`display_duration`)

- Обязательный параметр
- Задается в секундах (целое число)
- Определяет, как долго элемент показывается на экране

### 2. Время начала показа (`start_time`)

- Необязательный параметр
- Формат: HH:MM:SS (24-часовой формат)
- Если указано, контент показывается только после этого времени
- Примеры: 09:00:00, 14:30:00, 23:59:59

### 3. Время окончания показа (`end_time`)

- Необязательный параметр
- Формат: HH:MM:SS
- Если указано, контент показывается только до этого времени
- Должно быть позже `start_time`

### 4. Дни недели (`days_of_week`)

- Необязательный параметр
- Задается массивом чисел: [1,2,3,4,5,6,7]
- 1 = Понедельник, 2 = Вторник, ..., 7 = Воскресенье
- Если не указано, контент показывается каждый день недели
- Пример: [1,2,3,4,5] = только рабочие дни (Пн-Пт)
- Пример: [6,7] = только выходные (Сб-Вс)

Примеры использования расписания:

Пример 1: Объявление в рабочее время (9:00-18:00, только рабочие дни)

```
{
  "display_duration": 30,
  "start_time": "09:00:00",
  "end_time": "18:00:00",
  "days_of_week": [1,2,3,4,5]
```

```
}
```

#### Пример 2: Слайдшоу каждый день целый день

```
{
  "display_duration": 5,
  "start_time": "00:00:00",
  "end_time": "23:59:59"
}
```

#### Пример 3: Реклама только в выходные дни в полдень

```
{
  "display_duration": 20,
  "start_time": "12:00:00",
  "end_time": "14:00:00",
  "days_of_week": [6,7]
}
```

#### API эндпоинты для расписания:

- `POST /api/playlists/{id}/items` — добавление элемента с расписанием
- `PUT /api/playlists/{id}/items/{itemId}` — обновление расписания элемента
- `DELETE /api/playlists/{id}/items/{itemId}` — удаление элемента
- `GET /api/playlists/{id}/items` — получение всех элементов с расписанием

## 2.5 Синхронизация в реальном времени

#### Механизм синхронизации:

- Клиенты периодически опрашивают сервер на предмет обновлений
- Интервал проверки: по умолчанию 30 секунд (настраивается)
- При обнаружении изменений клиент загружает новую версию плейлиста
- Отслеживание версий плейлистов (автоматическое увеличение при сохранении)
- Мгновенное назначение/отмена плейлистов для клиентов

#### API для синхронизации:

- `GET /api/updates?client_id={id}&last_check={timestamp}` — получение обновлений с момента last\_check
- `GET /api/playlist?client_id={id}&name={name}&version={version}` — получение конкретного плейлиста
- `POST /api/clients/{id}/playlists/{playlistId}/assign` — назначение плейлиста клиенту
- `DELETE /api/clients/{id}/playlists/{playlistId}/unassign` — отмена назначения

## 2.6 Режимы воспроизведения

### 1. Одиночное отображение

- Показ одного элемента один раз
- После завершения экран переходит в режим ожидания

- Используется для временных объявлений

## 2. Ротация изображений (слайдшоу)

- Циклический показ нескольких изображений
- После последнего элемента возврат к первому
- Каждый элемент отображается заданное время (`display_duration`)
- Бесконечное повторение цикла

## 3. Плейлист с расписанием

- Показ элементов в соответствии с установленным расписанием
- Контент проверяется: текущее время в диапазоне `start_time` - `end_time`?
- Контент проверяется: сегодняшний день в `days_of_week`?
- Если условия не выполнены — элемент пропускается

---

## 2.7 Версионирование плейлистов

### Функции:

- Каждый плейлист имеет номер версии (начинается с 1)
- Версия автоматически увеличивается при сохранении изменений
- Возможность получения конкретной версии плейлиста
- История версий сохраняется на сервере
- Клиенты загружают только новые версии

### Использование:

- При назначении плейлиста клиенту указывается версия
- Если версия не указана, используется последняя активная версия
- Возможно запросить старую версию плейлиста (для отката)

---

## 2.8 Компоненты системы и их функции

### Konkord DSP Server (Серверная часть)

- Управление базой данных (PostgreSQL)
- HTTP API для всех операций
- Обработка запросов на синхронизацию
- Хранение медиафайлов
- Управление состоянием клиентов
- Логирование всех операций
- Аутентификация и авторизация

### Платформы:

- Astra Linux 1.7+
- Windows 10 (версия 1909+), Windows 11
- Любой распространенный дистрибутив Linux

### Konkord DSP Manager (Редактор расписания)

- Графический редактор плейлистов

- Графический редактор плейлистов
- Управление расписанием элементов
- Drag-and-drop интерфейс для переупорядочивания
- Загрузка медиафайлов
- Предпросмотр расписания
- Версионирование плейлистов
- Назначение плейлистов клиентам

#### Платформы:

- Astra Linux 1.7+
- Windows 10 (версия 1909+), Windows 11
- Любой распространенный дистрибутив Linux

#### Konkord DSP (Клиентское приложение для дисплеев)

- Подключение к серверу
- Периодическая проверка обновлений
- Загрузка плейлистов
- Воспроизведение контента по расписанию
- Проверка времени и дней недели
- Отчет о статусе серверу

#### Платформы:

- Astra Linux 1.7+
- Windows 10 (версия 1909+), Windows 11
- Любой распространенный дистрибутив Linux
- Android 11.0+

---

## 3. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

---

### 3.1 Серверная часть

#### Требования к оборудованию (минимум):

- Процессор: Intel Pentium / AMD Ryzen 3 (2 ГГц, 2 ядра)
- Оперативная память: 4 GB RAM
- Дисковое пространство: 50 GB (для ОС и ПО), 200+ GB (для медиа)
- Сетевой интерфейс: 100 Mbps

#### Требования к оборудованию (рекомендуется):

- Процессор: Intel i7 / AMD Ryzen 7 (3+ ГГц, 4+ ядра)
- Оперативная память: 16 GB RAM
- Дисковое пространство: SSD 500+ GB
- Сетевой интерфейс: 1 Gbps

#### Требования к ПО:

- PostgreSQL 13 или новее
- Python 3.11+ (для некоторых компонентов)
- Go 1.21+ (для компиляции сервера)

### Сетевые требования:

- Открытый порт 8080 для HTTP API
  - Локальная сеть 10/100 Mbps или выше
  - Рекомендуется Wi-Fi 5GHz или проводное соединение
- 

## 3.2 Базы данных

Используемая СУБД: PostgreSQL 13+

### Основные таблицы:

- `clients` — информация о дисплеях
  - `media_files` — метаданные загруженных файлов
  - `playlists` — информация о плейлистах
  - `playlist_items` — элементы плейлистов с расписанием
  - `client_playlists` — назначения плейлистов клиентам
  - `playlist_updates` — история обновлений плейлистов
- 

## 3.3 API

Тип API: RESTful HTTP/JSON

### Аутентификация:

- Bearer Token (JWT-токены)
- Пароль администратора для первого запуска

### Основные эндпоинты:

- `/api/clients` — управление дисплеями
- `/api/media` — управление медиафайлами
- `/api/playlists` — управление плейлистами
- `/api/updates` — синхронизация (для клиентов)
- `/api/playlist` — получение плейлиста (для клиентов)

### Форматы запроса/ответа:

- Content-Type: `application/json`
- Кодировка: UTF-8

### Коды ответов:

- 200 OK — успешный запрос
- 201 Created — ресурс создан
- 204 No Content — успешно удалено
- 400 Bad Request — ошибка валидации
- 401 Unauthorized — требуется аутентификация
- 403 Forbidden — доступ запрещен
- 404 Not Found — ресурс не найден
- 409 Conflict — конфликт (например, дублирование)
- 500 Internal Server Error — ошибка сервера

## 4. ХАРАКТЕРИСТИКИ БЕЗОПАСНОСТИ

---

### 4.1 Аутентификация

- Обязательная аутентификация для доступа к управлению
- JWT-токены для API запросов
- Пароль администратора при установке
- Возможность изменения пароля

### 4.2 Защита от атак

- Защита от SQL-инъекций (параметризованные запросы)
- Защита от XSS-атак (экранирование содержимого)
- Защита от CSRF (CSRF-токены)
- Проверка целостности файлов (SHA256)
- Валидация типов файлов
- Ограничение размера файлов (1 TB)

### 4.3 Логирование и аудит

- Логирование всех операций (создание, изменение, удаление)
- Логирование попыток несанкционированного доступа
- Временные метки (RFC 3339 формат)
- Уровни логирования: INFO, WARNING, ERROR
- Исключение чувствительных данных из логов (пароли, токены)

### 4.4 Управление доступом

- Роли пользователей (администратор, оператор, просмотр)
- Контроль доступа на уровне ресурсов
- Мягкое удаление записей (для аудита)

---

## 5. ПРОИЗВОДИТЕЛЬНОСТЬ И МАСШТАБИРУЕМОСТЬ

---

### 5.1 Производительность

Время отклика API (на стандартном оборудовании):

- Получение списка клиентов: < 500 ms
- Получение конкретного ресурса: < 200 ms
- Создание ресурса: < 500 ms
- Загрузка файла (50 MB): < 5000 ms
- Синхронизация на клиенте: < 1000 ms

Пропускная способность:

- Одновременные подключения: 100+
- Запросов в секунду: 1000+

## 5.2 Масштабируемость

- Неограниченное количество дисплеев (тестировано до 1000+)
  - Неограниченное количество плейлистов
  - Неограниченное количество медиафайлов (ограничение: дисковое пространство)
  - Централизованное управление (один сервер может управлять многими дисплеями)
- 

## 6. СОВМЕСТИМОСТЬ

---

### 6.1 Операционные системы

Сервер:

- Windows 10 (версия 1909+)
- Windows 11
- Astra Linux 1.7+
- Любой распространенный дистрибутив Linux

Редактор расписания:

- Windows 10 (версия 1909+)
- Windows 11
- Astra Linux 1.7+
- Любой распространенный дистрибутив Linux

Клиентское приложение:

- Windows 10 (версия 1909+)
- Windows 11
- Astra Linux 1.7+
- Любой распространенный дистрибутив Linux
- Android 8.0+

### 6.2 Браузеры

Веб-интерфейс совместим с:

- Google Chrome 90+
- Mozilla Firefox 88+
- Microsoft Edge 90+
- Safari 14+

### 6.3 Форматы мультимедиа

Видео:

- MP4 (H.264/AVC, AAC)
- WebM (VP8/VP9, Vorbis/Opus)
- Matroska MKV (различные кодеки)

Изображения:

- PNG

- JPEG
- GIF (статические)
- WebP

---

## 7. ОСОБЕННОСТИ РЕАЛИЗАЦИИ

---

### 7.1 Архитектура

Трехуровневая архитектура:

1. **Уровень представления** — веб-интерфейс Manager, клиентские приложения
2. **Уровень приложения** — HTTP API Server
3. **Уровень данных** — PostgreSQL база данных

### 7.2 Протоколы

- HTTP/REST для синхронизации
- JSON для обмена данными
- WebSocket (опционально для push-уведомлений)

### 7.3 Хранилище файлов

- Файлы хранятся на диске сервера
- Структурированное размещение по типам
- Периодическая очистка неиспользуемых файлов

---

## 8. ОГРАНИЧЕНИЯ И ИЗВЕСТНЫЕ ОСОБЕННОСТИ

---

### 8.1 Функциональные ограничения

- Максимальный размер медиафайла: 1 TB
- Максимальное разрешение видео: 4K
- Синхронизация работает только в локальной сети (без поддержки VPN по умолчанию)

### 8.2 Платформенно-специфичные особенности

Linux:

- Требуется графический интерфейс (X11 или Wayland) для Manager
- Две версии клиента для различных дистрибутивов

Android:

- Требуется Android 11.0 или выше
- Оптимизирована для планшетов и больших экранов

---

## 9. ПОДДЕРЖИВАЕМЫЕ ОПЕРАЦИИ

---

### 9.1 Жизненный цикл плейлиста

## 9.1 Жизненный цикл плейлиста

1. **Создание** — новый плейлист, версия = 1
2. **Редактирование** — добавление/удаление элементов, изменение параметров
3. **Сохранение** — версия увеличивается на 1
4. **Публикация** — активация плейлиста для использования
5. **Назначение** — присвоение конкретным дисплеям
6. **Синхронизация** — клиенты загружают новую версию
7. **Воспроизведение** — отображение контента по расписанию
8. **Удаление** — удаление плейлиста (если не используется)

## 9.2 Жизненный цикл медиафайла

1. **Загрузка** — добавление файла на сервер
2. **Проверка** — валидация типа и размера
3. **Хеширование** — вычисление SHA256
4. **Хранение** — сохранение на диск
5. **Использование** — добавление в плейлисты
6. **Синхронизация** — загрузка на клиенты при необходимости
7. **Удаление** — удаление с проверкой на использование

---

# 10. СООТВЕТСТВИЕ СТАНДАРТАМ

## 10.1 Российское законодательство

- Соответствие требованиям Реестра РПО (Реестр российского программного обеспечения)
- Использование открытых форматов данных (JSON, UTF-8)
- Соответствие ГОСТ Р 51904-2002 (ПО для ЭВМ)

## 10.2 Международные стандарты

- REST API согласно стандартам HTTP/1.1 (RFC 7231)
- JSON формат согласно RFC 7158
- Безопасность: основные принципы защиты (OWASP Top 10)
- Доступность: поддержка различных платформ и браузеров

---

# 11. ЗАКЛЮЧЕНИЕ

Konkord DSP представляет собой полнофункциональную систему управления Digital Signage контентом, обеспечивающую:

- Централизованное управление контентом для множества дисплеев
  - Гибкое расписание показа с поддержкой времени и дней недели
  - Надежную синхронизацию между сервером и клиентами
  - Кросс-платформенную поддержку (Windows, Linux, Android)
  - Безопасное хранение и доставку контента
  - Удобный интерфейс управления
-

Дата документа: 1 сентября 2025

Версия ПО: 1.0.0

Статус:  УТВЕРЖДЕНО

## ПРИЛОЖЕНИЯ

### Приложение А: Список функциональных возможностей

№	Функция	Статус
1	Регистрация дисплеев	<input checked="" type="checkbox"/> Реализовано
2	Управление медиафайлами	<input checked="" type="checkbox"/> Реализовано
3	Создание плейлистов	<input checked="" type="checkbox"/> Реализовано
4	Расписание по времени	<input checked="" type="checkbox"/> Реализовано
5	Расписание по дням недели	<input checked="" type="checkbox"/> Реализовано
6	Синхронизация в реальном времени	<input checked="" type="checkbox"/> Реализовано
7	Версионирование плейлистов	<input checked="" type="checkbox"/> Реализовано
8	Режим слайдшоу	<input checked="" type="checkbox"/> Реализовано
9	Мониторинг статуса	<input checked="" type="checkbox"/> Реализовано
10	Логирование операций	<input checked="" type="checkbox"/> Реализовано

### Приложение Б: Минимальные требования

Параметр	Значение
ОС Сервера	Windows 10+ / Linux
ОС Дисплея	Windows 10+ / Linux / Android 11.0+
БД	PostgreSQL 13+
RAM (Сервер)	4 GB (мин), 16 GB (рек)
Сетевое соединение	100 Mbps (мин), 1 Gbps (рек)
Порт	8080
Максимальный размер файла	1 TB