

ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ  
«ЛАСП ТЕХНОЛОГИИ»

Программное обеспечение (ПО)  
«ЦЕНТР 2.0»

---

(наименование продукта)

Инструкция технического писателя

---

(наименование документа)

Москва 2024

## Оглавление

Инструкция технического писателя	3
Верстка	4
Включение документов на стартовую страницу (index.md)	5
Включение разделов в документ	6
Ссылка на внешний документ внутри проекта	7
Ссылка на раздел документа	8
Картинки	9
Файл pdf	10
Титул на pdf	11
Редактирование титула pdf	12
Включение в документ диаграммы	13
Переименование комментариев к объектам в базе данных	14
Расширенное описание таблиц базы данных	15
Тестовый заголовок четвертого уровня	15



## Верстка

Текст документа набирается в синтаксисе markdown в любом редакторе.

Используем это из стандартного:

```
# Заголовок 1
## Заголовок 2
### Заголовок 3
#### Заголовок 4
##### Заголовок 5
##### Заголовок 6

*Курсив*
**Жирный**
***Жирный курсив***

>цитата

1. <!--1--> Нумерованный список 1
2. <!--2--> Нумерованный список 2
3. <!--3--> Нумерованный список 3

* Маркированный список уровень 1
* Маркированный список уровень 1
  * Маркированный список уровень 2
  * Маркированный список уровень 2
```

Блоки кода и значения в строке текста выделяются стандартно апострофами (').

Внешние ссылки тоже стандартные - пример [GitLab](#).

Для удобства лучше использовать полноценные версии Typora, JetBrains IntelliJ IDEA, Microsoft Visual Studio Code и т.д.

## Включение документов на стартовую страницу (index.md)

Путь прописываем со слэшем ( / ) впереди. Перед ссылкой на pdf два пробела от основной ссылки.

**Пример:**

```
[Описание программного обеспечения](/DOCS/Описание программного обеспечения)  [(pdf)](/DOCS/Описание программного обеспечения.pdf)
```

## Включение разделов в документ

Неоднократно используемые разделы включаются в основной документ тегами `{{` и `}}` с указанием пути к подключаемому разделу.

Расширение `md` не указываем.

**Пример:**

```
{{LIBRARY/CLASSES}}
```

При неправильной ссылке будет ошибка на сборке.

## Ссылка на внешний документ внутри проекта

Ссылку на внешний документ внутри проекта делаем подобно ссылкам на картинки. Путь прописываем со слэшем ( / ) впереди.

**Пример:**

```
[ACTION](/LIBRARY/CLASSES/ACTION)
```

| В pdf-файлах такая конструкция не работает.

## Ссылка на раздел документа

Ссылку на раздел внутри текущего документа делаем указывая заголовок раздела и перед ним # без пробела.

**Пример:**

```
[Переход на тестовый заголовок четвертого уровня](#Тестовый заголовок четвертого уровня)
```

**Как работает:**

[Переход на тестовый заголовок четвертого уровня](#)



## Картинки

Картинки помещаются в соответствующий раздел каталога **images** каждого проекта и подключаются ссылками с относительными путями (удобнее пользоваться JetBrains IntelliJ IDEA).

### **Пример:**

```

```

Во избежание проблем при сборке - **обязательно**:

- наименования картинок с разделением слов дефисами;
- наименование каталогов внутри **src/images** латинскими буквами.

## Файл pdf

Создаем рядом с файлом md, файл с тем же именем и расширением pdf.md.

**Пример:**

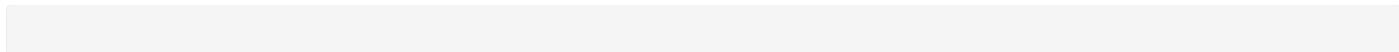
```
MANUAL.md  
MANUAL.pdf.md
```

В файле pdf.md вставляем тот же заголовок, что и в родительском и вставляем ссылку на родительский файл тегами `{{` и `}}`.

В результате содержание родительского документа подтянется в pdf.

## Титул на pdf

Для помещения наименования документа на титул, внутри pdf.md вставляем после заголовка следующую конструкцию:



### заменить на заголовок документа, который будет на титуле.

## Редактирование титула pdf

Содержание и внешний вид титула редактируется в файле frontpage.html расположенном в корневом каталоге проекта  
...\src

## Включение в документ диаграммы

Диаграмма, созданная в Camunda, подключается в документ следующей конструкцией:

```

```

Файл с диаграммой должен быть расположен в каталоге проекта ...\\src\\diagrams

## Переименование комментариев к объектам в базе данных

Переименовывать описание можно применять к схемам, таблицам и полям базы данных.

В проекте [gitlab.com/coreweb/objects/имя\\_проекта](https://gitlab.com/coreweb/objects/имя_проекта) создаем ветку hotfix-ГОДМЕСЯЦДЕНЬ.set-comments.

В директории [gitlab.com/coreweb/objects/имя\\_проекта/liquibase/changesets/](https://gitlab.com/coreweb/objects/имя_проекта/liquibase/changesets/) создаем файл ГОД-МЕСЯЦ-ДЕНЬ-add-comments.sql следующего содержания:

```
--liquibase formatted sql
--changeset ИМЯ_ПОЛЬЗОВАТЕЛЯ:ГОД-МЕСЯЦ-ДЕНЬ-add-comments.sql
COMMENT ON НАИМЕНОВАНИЕ_ОБЪЕКТА_В_СХЕМЕ IS 'Текст_который_будет_внесен_в_базу';
--rollback;
```

### Пример:

```
--liquibase formatted sql
--changeset user:2021-05-24--01-add-comments.sql
COMMENT ON SCHEMA admin IS 'Администрирование';
COMMENT ON TABLE admin.storage_user IS 'Пользователи хранилища';
COMMENT ON COLUMN admin."group".start_date IS 'Начало действия';
--rollback;
```

После редактирования *push* и *pull request* в master.

## Расширенное описание таблиц базы данных

Для вставки расширенного описания таблиц в документ, не включая его в базу данных, в каталоге проекта DB.details создаем файлы:

- НАИМЕНОВАНИЕ МОДУЛЯ.НАИМЕНОВАНИЕ СХЕМЫ.НАИМЕНОВАНИЕ ТАБЛИЦЫ.md - описание над таблицей;
- НАИМЕНОВАНИЕ МОДУЛЯ.НАИМЕНОВАНИЕ СХЕМЫ.НАИМЕНОВАНИЕ ТАБЛИЦЫ.footer.md - описание под таблицей.

Пример пути:

```
...\src\DB.details\ADMIN.stat.track.md  
...\src\DB.details\ADMIN.stat.track.footer.md
```

После чего наполняем их содержимым.

## Тестовый заголовок четвертого уровня