

# «PM.integrator» (Система проектной интеграции и координации). Инструкция по установке ПО

---

Редакция 1.0

Всего листов: 13

---

---

**Содержание**

<b>1. Общие положения .....</b>	<b>3</b>
1.1. Термины .....	3
1.2. Аббревиатуры .....	3
1.3. Используемые форматы данных .....	3
<b>2. Общие положения .....</b>	<b>4</b>
<b>3. Подготовка к работе .....</b>	<b>5</b>
3.1. Подготовка сервера .....	5
3.2. Подготовка исходного кода .....	5
3.3. Подготовка «образов» .....	5
<b>4. Настройка .....</b>	<b>7</b>
4.1. Настройка пользователей хоста и права доступа .....	7
4.2. Конфигурация переменных окружения.....	7
4.3. Настройка https .....	8
4.4. Настройка директории данных .....	9
4.5. Запуск PM.integrator .....	10
4.6. Настройка Active Directory.....	10
4.7. Настройка Elasticsearch .....	11

## 1. Общие положения

### 1.1. Термины

Система, ПО, PM.integrator      «PM.integrator» (Система проектной интеграции и координации)

### 1.2. Аббревиатуры

### 1.3. Используемые форматы данных

## 2. Общие положения

«PM.integrator» (Система проектной интеграции и координации) предназначена для интеграции и координации участников проекта, требований, обязательств и ограничений, информации по проекту, а также групп процессов управления проектами в едином информационном пространстве, в том числе:

- Создания комплексной и интегрированной системы проектного контроля.
- Объединения и улучшения коммуникаций между всеми участниками проектной деятельности в единой среде управления проектами.
- Снижения затрат на исполнение и поддержку групп процессов управления проектами и повышения эффективности команд.
- Интеграционного взаимодействия с профессиональными решениями для календарно-сетевое планирования, управления стоимостью и другими специализированными инструментами для повышения степени проектного контроля.
- Обеспечения выполнения процессов управления техническими и проектными документами, рисками, изменениями, контроля исполнения поручений, администрирования проектов.
- Обеспечения мобильного доступа к проектным данным и выполнению задач по оперативному сбору данных о выполнении работ проекта непосредственно в месте их выполнения.
- Поддержки международных стандартов и лучших практик проектного управления, таких как PMI, IPMA, ISO, AACE, AWP, WFP.

«PM.integrator» решает следующие задачи:

- Обеспечивает регламентированный доступ к актуальной информации и поддерживает совместную работу, включая подключение подрядчиков к процессам управления.
- Автоматизирует и интегрирует ключевые группы процессов управления проектом.
- Формирует единый и достоверный источник информации о статусе выполнения проекта.
- Поддерживает контроль над основными параметрами реализации, позволяя оперативно реагировать на изменения и обеспечивая управление качеством на всех этапах.

Более подробную информацию можно найти на странице продукта: <https://www.pmssoft.ru/pmintegrator/>.

### 3. Подготовка к работе

#### 3.1. Подготовка сервера

Виртуальный выделенный сервер на базе операционной системы Astra Linux, наличие ПО Docker, docker-compose (в новых версиях Docker предоставляется «из коробки»).

#### 3.2. Подготовка исходного кода

На выделенную машину (далее – «хост», «сервер», «стенд») клонируется или загружается посредством удалённого доступа необходимая «*git ветка*» (по умолчанию «*main*») или необходимый определённый «*commit*» из репозитория или переданного исходного кода.

---

Для использования заранее собранных Docker образов клонировать ветку исходного кода необязательно.

---

#### 3.3. Подготовка «образов»

Для того, чтобы получить необходимый набор образов, необходимо иметь доступ к реестру Docker образов, в котором хранятся актуальные сборки. Имея Linux-подобную ОС, выполнить команду:

```
docker save {REPOSITORY} /{IMAGE_NAME}:{IMAGE_VERSION} | gzip > {IMAGE_ARCHIVE_NAME}.tar.gz
```

- REPOSITORY – URL адрес репозитория
- IMAGE\_NAME – название образа
- IMAGE\_VERSION – версия образа
- IMAGE\_ARCHIVE\_NAME – название архива на диске

Например

- `docker save gitea.pmssoft.ru/packages/pmportal-frontend:9fef535.3418 | gzip > pmportal-frontend.tar.gz`
- `docker save gitea.pmssoft.ru/packages/pmportal-api:9fef535.3418 | gzip > pmportal-api.tar.gz`
- `docker save gitea.pmssoft.ru/packages/pmssoft_docstorage:1.0.0 | gzip > pmssoft_docstorage.tar.gz`
- `docker save gitea.pmssoft.ru/packages/nginx:1.25.1-bookworm | gzip > nginx.tar.gz`
- `docker save gitea.pmssoft.ru/packages/pmssoft-repmgr-pgbackrest:14.7.1 | gzip > pmssoft-repmgr-pgbackrest.tar.gz`
- `docker save gitea.pmssoft.ru/packages/nginx_load_balancer:1.0.7 | gzip > nginx_load_balancer.tar.gz`
- `docker save gitea.pmssoft.ru/packages/pmssoft_elastic:7.14.3 | gzip > pmssoft_elastic.tar.gz`
- `docker save gitea.pmssoft.ru/packages/kibana:7.14.2 | gzip > kibana.tar.gz`
- `docker save rabbitmq:3.10.7-management | gzip > rabbitmq.tar.gz`

- `docker save gitea.pmssoft.ru/packages/pmportal-webdav-server:f9d5626.5 | gzip > pmportal-webdav-server.tar.gz`
- `docker save gitea.pmssoft.ru/packages/gantt-converter:0.6.0 | gzip > gantt-converter.tar.gz`
- `docker save gitea.pmssoft.ru/packages/msp_gantt_converter_module:1.0.0 | gzip > msp_gantt_converter_module.tar.gz`

Ниже представлен список образов, необходимых для успешного запуска ПО:

- pmportal-frontend
- pmportal-api
- pmssoft\_docstorage
- nginx
- pmssoft-repmgr-pgbackrest
- nginx\_load\_balancer
- pmssoft\_elastic
- kibana
- rabbitmq
- pmportal-webdav-server
- gantt-converter
- msp\_gantt\_converter\_module

---

Если системному администратору, который осуществляет установку PM.integrator, уже переданы эти архивы, этап получения образов пропустить.

---

Для запуска Системы необходимо несколько файлов, хранящихся вместе с исходным кодом в системе управления версиями:

- `docker-compose.yml`
- `.env` (конфигурируется из лежащего в ветке/переданного `.env-example` файла)

## 4. Настройка

### 4.1. Настройка пользователей хоста и права доступа

Для обеспечения безопасности рекомендуется создать пользователя (например, pm.int) без «*sudo*» доступа, через который будет производиться запуск и обновление Системы:

- `sudo useradd -m pm.int`
- `sudo passwd pm.int`
- (установить пароль)
- `sudo usermod -aG docker pm.int`

PM.integrator использует дополнительную «директорию данных», например, `/opt/pmssoft` для хранения данных и конфигураций Системы. Пользователь, через которого производится запуск Системы должен иметь доступ к директории данных, «*docker-compose.yml*» и «*.env* файлам».

### 4.2. Конфигурация переменных окружения

В директории «*Docker*» исходного кода расположен файл «*.env-example*», содержащий названия переменных окружения. Необходимо установить значения этих переменных.

- `PMPORTAL_VERSION` – переменная отвечает за версию «*Docker*» образов «*Backend*» и «*Frontend*» сервисов.
- `APPLICATION_URL` – URL-адрес, по которому будет доступна Система на данном хосте.
- `WEBDAV_VERSION` – версия образа «*WebDav*», реализующего доступ к документам через проводник.
- `NGINX_LOAD_BALANCER_VERSION` – версия «*revers-proxy*».
- `POSTGRES_PASSWORD`, `POSTGRES_POSTGRES_PASSWORD`, `REPMGR_PASSWORD`, `RABBITMQ_DEFAULT_PASS`, `ELASTIC_PASSWORD` – пароли, которые необходимо установить
- `BACKEND_MAIL_*` – набор переменных, которые необходимо установить для обеспечения работы почтовых уведомлений.
- `BACKEND_SECURITY_SECRET` – случайный GUID.
- `DATA_DIRECTORY` – путь к директории данных (см. «4.4 Настройка директории данных»).
- `APP_DIRECTORY` – путь к директории конфигурации данных (см. «4.4 Настройка директории данных»)
- `BACKUP_DIRECTORY` – путь к директории резервного копирования (см. «4.4 Настройка директории данных»)
- `SSL_CERT_HOST` – название хоста в файлах SSL сертификата (см. «4.3 Настройка https»)

- WEBDAV\_PKCS12\_PASSWORD – пароль от SSL сертификата (см. «4.3 Настройка https»)
- DOCKER\_REGISTRY – репозиторий «**Docker**» образов (при установке портала с заранее собранными образами значение переменной должно соответствовать тому репозиторию, откуда переданы образы)

### 4.3. Настройка https

Для настройки TLS компонента потребуется преобразовать готовый сертификат – извлечь сам сертификат и приватный ключ в отдельные файлы. Для этого выполните следующие команды:

1. Для извлечения сертификата сервера:

```
openssl pkcs12 -in ${путь до файла pfx} -clcerts -nokeys -out ${путь до файла сертификата в формате crt}
```

При выполнении команды будет запрошен пароль для сертификата pfx.

2. Для извлечения приватного ключа в зашифрованном виде:

```
openssl pkcs12 -in ${путь до файла pfx} -nocerts -out ${путь до файла приватного ключа (.pkey)}
```

При выполнении команды сначала будет запрошен пароль для сертификата pfx, затем новый пароль для защиты созданного файла с приватным ключом.

3. Для расшифровки приватного ключа:

```
openssl rsa -in ${путь до файла приватного ключа} -out ${путь до файла с расшифрованным ключем (.key)}
```

4. Для генерации файла набора параметров для шифров с использованием протокола Диффи-Хеллмана:

```
openssl dhparam -out ${путь до файла параметров} 4096
```

---

Генерация файла требовательна к ресурсам ЦПУ и может занять более 10 минут.

---

Например, для хоста pmportal.test.ru:

- `openssl pkcs12 -in pmportal.test.ru.pfx -clcerts -nokeys -out pmportal.test.ru.crt -legacy`
- `openssl pkcs12 -in pmportal.test.ru.pfx -nocerts -out pmportal.test.ru.pkey -legacy`
- `openssl rsa -in pmportal.test.ru.pkey -out pmportal.test.ru.key`
- `openssl dhparam -out dhparam.pem 4096`

Значение переменной «**SSL\_CERT\_HOST**» устанавливается как название файлов сертификата, то есть в данном примере «**SSL\_CERT\_HOST= pmportal.test.ru**»



#### 4.4. Настройка директории данных

Для хранения данных, конфигурации и «**backup**» используется отдельная директория (допустимо указать путь к внешнему диску). В данном примере данная директория будет «**/opt/pmssoft**». Таким образом:

- DATA\_DIRECTORY=/opt/pmssoft/data
- APP\_DIRECTORY=/opt/pmssoft/deploy
- BACKUP\_DIRECTORY=/opt/pmssoft/backup

В данных директориях должна быть создана определенная иерархия директорий и загружены определенные файлы:

- deploy/Assests/logo.png
- deploy/certs/
- deploy/conf/repmgr/pg\_hba.conf
- deploy/conf/pgbackrest.conf
- deploy/data/portal-files/nginx.conf
- deploy/data/DisplayNames/display-names.json
- deploy/Export/assignmentTemplate.ods
- deploy/Export/dynamicAppSettings.json
- deploy/Export/EntityTemplate.ods
- deploy/Export/Passport.ods
- deploy/Export/presentationLogo.png
- deploy/Export/Risk.ods
- deploy/Export/TechnicalDocumentation.ods
- deploy/Export/template.odp
- deploy/Export/template.ods
- deploy/logs/
- deploy/SSO/
- data/elastic/
- data/files/
- data/postgres/

В директорию «**deploy/certs/**» необходимо загрузить полученные в результате выполнения команд, указанных в разделе «**4.3 Настройка https**» файлы:

- pmportal.test.ru.crt
- pmportal.test.ru.key
- pmportal.test.ru.pfx
- dhparam.pem

После загрузки файлов необходимо установить права на определенные директории (через «**sudo**» пользователя):

- sudo chown 1001:1001 /opt/pmssoft/data/postgres -R

- `sudo chown 1000:0 /opt/pmsoft/data/elastic -R`

#### 4.5. Запуск PM.integrator

Запуск Системы осуществляется командой, находясь в директории «**Docker**»:

**`docker compose up --profile backend --profile frontend -d`**

Для получения логов в режиме реального времени использовать ту же команду, но без флага «**-d**».

Для получения списка всех образов использовать «**docker image ls -a**».

Для получения списка работающих контейнеров использовать «**docker ps**».

Для получения логов определённого контейнера (в том числе после его остановки) использовать «**docker logs**» (ID контейнера, взять из `docker ps`) «**-timestamps -since=4h**»:

- «**-timestamps**» – флаг для получения информации о времени появления каждого сообщения;
- «**-since**» – флаг для ограничения вывода логов за последние несколько часов.

Для получения подробной информации см. документацию `docker cli`.

Для административного доступа в Систему использовать «**admin**» учётную запись:

- Логин: `admin`
- Пароль: `PThGmQSiB6eGzbS6fVhK`

---

Для промышленной среды после завершения процесса установки данный пароль необходимо сменить.

---

#### 4.6. Настройка Active Directory

Данные о текущих подключенных доменах хранятся в БД в таблице «**public."ActiveDirectoryDomains"**». Запрос на получение всех записей из таблицы:

**`SELECT * FROM public."ActiveDirectoryDomains";`**

Для добавления интеграции с новым доменом нужно выполнить запрос следующего формата:

**`INSERT INTO public."ActiveDirectoryDomains"`**

**`("Id", "Name", "Hostname", "Port", "DN", "Password", "SearchBase")`**

**`VALUES ('920a4984-6a64-4833-b606-9b16a4c0b7c3', 'Основной домен', 'domain.ru', '389', 'CN=pmportal,OU=Service Users,DC=domain,DC=ru', 'Пароль', 'OU=Employees, DC=domain,DC=ru');`** где:

- `Id` - сгенерированный GUID для записи;
- `Name` - имя домена в приложении;
- `Hostname` - адрес домена;

- Port - порт подключения к домену;
- DN - CN пользователя, под которым приложение будет выполнять поиск в домене;
- Password - в открытом виде пароль пользователя, под которым приложение будет выполнять поиск в домене;
- SearchBase - ветка домена, с которого будет начинаться поиск.

Сгенерировать GUID на Linux можно с помощью команды:

- `cat /proc/sys/kernel/random/uuid`

А также на Windows помощью команды PowerShell:

- `New-Guid`

Или любым другим удобным способом.

После создания записи рекомендуется перезапустить Backend контейнер. После перезапуска контейнера все пароли, которые были записаны в открытом виде, будут зашифрованы.

#### 4.7. Настройка Elasticsearch

Для обеспечения синхронизации данных между базой данных и «*ElasticSearch*» используется «*Postgres extension – zombodb*». При установке сконфигурировать список «*Postgres*» индексов «*zombodb*»:

- подключиться к контейнеру «*pmsoft-repmgr-pgbackrest*»
- для каждой из синхронизируемых таблиц выполнить команду:
  - `CREATE INDEX tablename_zdb_idx ON public."Tablename" USING zombodb ("Tablename".*) WITH (url='http://elasticsearch-url:9200/');` где:
    - *Tablename* – название таблицы
    - *elasticsearch-url* – URL адрес сервиса Elasticsearch.
- для получения списка всех существующих *zombodb* индексов:
  - `SELECT * FROM "pg_indexes" WHERE "indexdef" LIKE "%zombodb%";`
- для удаления индекса:
  - `DROP INDEX tablename_zdb_idx;`

Полный список команд, обеспечивающих синхронизацию сущностей (пример):

- `CREATE INDEX projectinitiatives_zdb_idx ON public."ProjectInitiatives" USING zombodb ("ProjectInitiatives".*) WITH (url='http://elasticsearch:9200/');`
- `CREATE INDEX projectpassports_zdb_idx ON public."ProjectPassports" USING zombodb ("ProjectPassports".*) WITH (url='http://elasticsearch:9200/');`
- `CREATE INDEX projectquestionsolutions_zdb_idx ON public."ProjectQuestionSolutions" USING zombodb ("ProjectQuestionSolutions".*) WITH (url='http://elasticsearch:9200/');`

- CREATE INDEX projectquestions\_zdb\_idx ON public."ProjectQuestions" USING zombodb (("ProjectQuestions".\*)) WITH (url='http://elasticsearch:9200/');
- CREATE INDEX projectresources\_zdb\_idx ON public."ProjectResources" USING zombodb (("ProjectResources".\*)) WITH (url='http://elasticsearch:9200/');
- CREATE INDEX projectstatusreports\_zdb\_idx ON public."ProjectStatusReports" USING zombodb (("ProjectStatusReports".\*)) WITH (url='http://elasticsearch:9200/');
- CREATE INDEX attachmentfilesystemitems\_zdb\_idx ON public."AttachmentFileSystemItems" USING zombodb (("AttachmentFileSystemItems".\*)) WITH (url='http://elasticsearch:9200/');
- CREATE INDEX issueevents\_zdb\_idx ON public."IssueEvents" USING zombodb (("IssueEvents".\*)) WITH (url='http://elasticsearch:9200/');
- CREATE INDEX issues\_zdb\_idx ON public."Issues" USING zombodb (("Issues".\*)) WITH (url='http://elasticsearch:9200/');
- CREATE INDEX knowledgefilesystemitems\_zdb\_idx ON public."KnowledgeFileSystemItems" USING zombodb (("KnowledgeFileSystemItems".\*)) WITH (url='http://elasticsearch:9200/');
- CREATE INDEX multimediafilesystemitems\_zdb\_idx ON public."MultimediaFileSystemItems" USING zombodb (("MultimediaFileSystemItems".\*)) WITH (url='http://elasticsearch:9200/');
- CREATE INDEX notices\_zdb\_idx ON public."Notices" USING zombodb (("Notices".\*)) WITH (url='http://elasticsearch:9200/');
- CREATE INDEX pplantasks\_zdb\_idx ON public."PPlanTasks" USING zombodb (("PPlanTasks".\*)) WITH (url='http://elasticsearch:9200/');
- CREATE INDEX portfolios\_zdb\_idx ON public."Portfolios" USING zombodb (("Portfolios".\*)) WITH (url='http://elasticsearch:9200/');
- CREATE INDEX programs\_zdb\_idx ON public."Programs" USING zombodb (("Programs".\*)) WITH (url='http://elasticsearch:9200/');
- CREATE INDEX projectdetails\_zdb\_idx ON public."ProjectDetails" USING zombodb (("ProjectDetails".\*)) WITH (url='http://elasticsearch:9200/');
- CREATE INDEX projectfilesystemitemconfigurations\_zdb\_idx ON public."ProjectFileSystemItemConfigurations" USING zombodb (("ProjectFileSystemItemConfigurations".\*)) WITH (url='http://elasticsearch:9200/');
- CREATE INDEX projectfilesystemitems\_zdb\_idx ON public."ProjectFileSystemItems" USING zombodb (("ProjectFileSystemItems".\*)) WITH (url='http://elasticsearch:9200/');
- CREATE INDEX projects\_zdb\_idx ON public."Projects" USING zombodb (("Projects".\*)) WITH (url='http://elasticsearch:9200/');
- CREATE INDEX riskevents\_zdb\_idx ON public."RiskEvents" USING zombodb (("RiskEvents".\*)) WITH (url='http://elasticsearch:9200/');

- 
- CREATE INDEX riskinfluenceconfigurations\_zdb\_idx ON public."RiskInfluenceConfigurations" USING zombodb ("RiskInfluenceConfigurations".\*) WITH (url='http://elasticsearch:9200/');
  - CREATE INDEX riskmatrixconfigurations\_zdb\_idx ON public."RiskMatrixConfigurations" USING zombodb ("RiskMatrixConfigurations".\*) WITH (url='http://elasticsearch:9200/');
  - CREATE INDEX riskprobabilityconfigurations\_zdb\_idx ON public."RiskProbabilityConfigurations" USING zombodb ("RiskProbabilityConfigurations".\*) WITH (url='http://elasticsearch:9200/');
  - CREATE INDEX risks\_zdb\_idx ON public."Risks" USING zombodb ("Risks".\*) WITH (url='http://elasticsearch:9200/');
  - CREATE INDEX solutions\_zdb\_idx ON public."Solutions" USING zombodb ("Solutions".\*) WITH (url='http://elasticsearch:9200/');
  - CREATE INDEX stakeholders\_zdb\_idx ON public."Stakeholders" USING zombodb ("Stakeholders".\*) WITH (url='http://elasticsearch:9200/');
  - CREATE INDEX news\_zdb\_idx ON public."News" USING zombodb ("News".\*) WITH (url='http://elasticsearch:9200/');