

# Руководство по использованию public API

API используется для интеграции Test IT с внешними системами запуска автоматизированных тестов. API позволяет:

- связывать автоматизированные тесты с тестами в Test IT
- получать задания на запуск автоматизированных тестов
- управлять запусками автоматизированных тестов
- добавлять результаты автоматизированных тестов

## Содержание

- [Содержание](#)
- [Описание](#)
- [Авторизация](#)
- [Основной URL](#)
- [Связывание автоматизированного теста с тестом в Test IT](#)
- [Получение всех автотестов в системе](#)
- [Получение всех автотестов в конкретном проекте](#)
- [Получение всех автотестов, связанных с конкретным тестом](#)
- [Удаление связи автотеста с тестом Test IT](#)
- [Получение запусков для проекта](#)
- [Перевод запуска в статус InProgress \(Выполняется\)](#)
- [Перевод запуска в статус Stopped \(Остановлен\)](#)
- [Перевод запуска в статус Completed \(Завершен\)](#)
- [Добавление результата автотеста](#)

## Описание

API работает по протоколу HTTP и представляет собой набор методов, с помощью которых совершаются запросы и возвращаются ответы для каждой операции. Все ответы приходят в виде JSON структур.

## Авторизация

Авторизация возможна при наличии логина и секретного ключа пользователя, которые должны быть добавлены в заголовки каждого запроса к API.

Секретный ключ может сгенерировать каждый пользователь в настройках своего профиля.

Заголовки, которые необходимы для авторизации:

Ключ	Значение
secretKeyBase64	{secret_key}
userName	{login}

## Основной URL

Все ссылки на запросы к API в текущей документации включают базовый URL системы Test IT, например <http://testit.meistersoft.ru>

## Связывание автоматизированного теста с тестом в Test IT

Для того, чтобы связать автотест с тестом в Test IT, необходимо отправить запрос:

**POST** /api/Public/AddAutoTestToTestCase/{testCaseId}

Где:

{testCaseId} - это идентификатор теста в Test IT

Тело запроса:

```
{ "autoTestExternalId": "string" }
```

Пример cURL запроса в общем виде:

```
curl -X POST "http://192.168.88.140/api/Public/AddAutoTestToTestCase/{testCaseGlobalId}" -H "accept: application/json" -H "SecretKeyBase64: {SecretKeyBase64}" -H "UserName: {UserName}" -H "Content-Type: application/json-patch+json" -d "{ \"autoTestExternalId\": \"{autoTestExternalId}\"}"
```

Пример cURL запроса:

```
curl -X POST "http://192.168.88.140/api/Public/AddAutoTestToTestCase/3834" -H "accept: application/json" -H "SecretKeyBase64: ZndlUXkyNXFCbkJRUmJvUw==" -H "UserName: admin" -H "Content-Type: application/json-patch+json" -d "{ \"autoTestExternalId\": \"autoTestExternalId\"}"
```

## Получение всех автотестов в системе

Для того, чтобы получить список всех автотестов, зарегистрированных в системе, необходимо отправить запрос:

**GET** /api/Public/GetAllAutoTests

Пример cURL запроса в общем виде:

```
curl -X POST "http://192.168.88.140/api/Public/GetAllAutoTests" -H "accept: application/json" -H "SecretKeyBase64: {SecretKeyBase64}" -H "UserName: {UserName}"
```

Пример cURL запроса:

```
curl -X POST "http://192.168.88.140/api/Public/GetAllAutoTests" -H "accept: application/json" -H "SecretKeyBase64: Znd1UXkyNXFCbkJRUmJvUw==" -H "UserName: admin"
```

## Получение всех автотестов в конкретном проекте

Для того, чтобы получить список всех автотестов, зарегистрированных в проекте, необходимо отправить запрос:

**GET** /api/Public/GetAutoTestsForProject/{projectId}

Где:

{projectId} - идентификатор проекта

Пример cURL запроса в общем виде:

```
curl -X POST "http://192.168.88.140/api/Public/GetAutoTestsForProject/{projectGlobalId}" -H "accept: application/json" -H "SecretKeyBase64: {SecretKeyBase64}" -H "UserName: {UserName}"
```

Пример cURL запроса:

```
curl -X POST "http://192.168.88.140/api/Public/GetAutoTestsForProject/3504" -H "accept: application/json" -H "SecretKeyBase64: Znd1UXkyNXFCbkJRUmJvUw==" -H "UserName: admin"
```

## Получение всех автотестов, связанных с конкретным тестом

Для того, чтобы получить список всех автотестов, связанных с тестом, необходимо отправить запрос:

**GET** /api/Public/GetAutoTestsForTestCase/{testCaseId}

Где:

{testCaseId} - это идентификатор теста в Test IT

Пример cURL запроса в общем виде:

```
curl -X POST "http://192.168.88.140/api/Public/GetAutoTestsForTestCase/{testCaseGlobalId}" -H "accept: application/json" -H "SecretKeyBase64: {SecretKeyBase64}" -H "UserName: {UserName}"
```

Пример cURL запроса:

```
curl -X POST "http://192.168.88.140/api/Public/GetAutoTestsForTestCase/3507" -H "accept: application/json" -H "SecretKeyBase64: Znd1UXkyNXFCbkJRUmJvUw==" -H "UserName: admin"
```

## Удаление связи автотеста с тестом Test IT

Для того, чтобы удалить связь автотеста с тестом, необходимо отправить запрос:

**DELETE** /api/Public/DeleteAutoTestFromTestCase/{testCaseId}

Где:

{testCaseId} - это идентификатор теста в Test IT

Тело запроса:

```
{
  "autoTestExternalId": "string"
}
```

Где:

autoTestExternalId - строка, идентифицирующая внешний автоматизированный тест

Пример cURL запроса в общем виде:

```
curl -X POST
"http://192.168.88.140/api/Public/DeleteAutoTestFromTestCase/{testCaseGlobalId}" -H "accept:
application/json" -H "SecretKeyBase64: {SecretKeyBase64}" -H "UserName: {UserName}" -H
"Content-Type: application/json-patch+json" -d "{ \"autoTestExternalId\":
\"{autoTestExternalId}\"}"
```

Пример cURL запроса:

```
curl -X POST "http://192.168.88.140/api/Public/DeleteAutoTestFromTestCase/3507" -H "accept:
application/json" -H "SecretKeyBase64: Znd1UXkyNXFCbkJRUmJvUw==" -H "UserName: admin" -H
"Content-Type: application/json-patch+json" -d "{ \"autoTestExternalId\": \"autoTest#1\"}"
```

## Получение запусков для проекта

Для того, чтобы получить список автоматизированных запусков для проекта, необходимо отправить запрос:

**GET** /api/Public/GetTestRuns/{projectId}

Где:

{projectId} - идентификатор проекта

Пример cURL запроса в общем виде:

```
curl -X POST "http://192.168.88.140/api/Public/GetTestRuns/{projectId}" -H "accept:
application/json" -H "SecretKeyBase64: {SecretKeyBase64}" -H "UserName: {UserName}"
```

Пример cURL запроса:

```
curl -X POST "http://192.168.88.140/api/Public/GetTestRuns/3504" -H "accept: application/json" -H "SecretKeyBase64: Znd1UXkyNXFCbkJRUmJvUw==" -H "UserName: admin"
```

## Перевод запуска в статус InProgress (Выполняется)

Для того, чтобы перевести статус запуска в InProgress, необходимо отправить запрос:

**POST** /api/Public/StartTestRun/{testRunId}

Где:

{testRunId} - это валидный uuid запуска

При успешном выполнении запроса запуск переходит в состояние InProgress.

Пример cURL запроса в общем виде:

```
curl -X POST "http://192.168.88.140/api/Public/StartTestRun/{ID}" -H "accept: application/json" -H "SecretKeyBase64: {SecretKeyBase64}" -H "UserName: {UserName}"
```

Пример cURL запроса:

```
curl -X POST "http://192.168.88.140/api/Public/StartTestRun/cd2a79db-00cf-4606-adbe-00eb6fd0dc4a" -H "accept: application/json" -H "SecretKeyBase64: admin" -H "UserName: Znd1UXkyNXFCbkJRUmJvUw=="
```

## Перевод запуска в статус Stopped (Остановлен)

Для того, чтобы перевести статус запуска в InProgress, необходимо отправить запрос:

**POST** /api/Public/StopTestRun/{testRunId}

Где:

{testRunId} - это валидный uuid запуска

При успешном выполнении запроса запуск переходит в состояние Stopped. Всем тест-поинтам, по которым не был получен результат, проставляется результат Skipped.

Пример cURL запроса в общем виде:

```
curl -X POST "http://192.168.88.140/api/Public/StopTestRun/{ID}" -H "accept: application/json" -H "SecretKeyBase64: {SecretKeyBase64}" -H "UserName: {UserName}"
```

Пример cURL запроса:

```
curl -X POST "http://192.168.88.140/api/Public/StopTestRun/cd2a79db-00cf-4606-adbe-00eb6fd0dc4a" -H "accept: application/json" -H "SecretKeyBase64: admin" -H "UserName: Znd1UXkyNXFCbkJRUmJvUw=="
```

## Перевод запуска в статус Completed (Завершен)

Для того, чтобы перевести статус запуска в Completed, необходимо отправить запрос:

**POST** /api/Public/CompleteTestRun/{testRunId}

Где:

{testRunId} - это валидный uuid запуска

При успешном выполнении запроса Test Run переходит в состояние Completed. Всем тест-поинтам, по которым не был получен результат, проставляется результат Skipped.

Пример cURL запроса в общем виде:

```
curl -X POST "http://192.168.88.140/api/Public/CompleteTestRun/{testRunId}" -H "accept: application/json" -H "SecretKeyBase64: {SecretKeyBase64}" -H "UserName: {UserName}"
```

Пример cURL запроса:

```
curl -X POST "http://192.168.88.140/api/Public/CompleteTestRun/cd2a79db-00cf-4606-adbe-00eb6fd0dc4a" -H "accept: application/json" -H "SecretKeyBase64: admin" -H "UserName: Znd1UXkyNXFCbkJRUmJvUw=="
```

## Добавление результата автотеста

Для того, чтобы добавить результат автотеста в систему, необходимо отправить запрос:

**POST** /api/Public/SetAutoTestResult

Тело запроса:

```
{
  "testRunId": "string",
  "testPlanGlobalId": 0,
  "autoTestExternalId": "string",
  "configurationGlobalId": 0,
  "status": "string",
  "outcome": "string",
  "message": "string",
  "stackTrace": "string"
}
```

Где:

testRunId - валидный uuid запуска, в котором получен результат

testPlanGlobalId - идентификатор тест-плана

autoTestExternalId - строка, идентифицирующая внешний автоматизированный тест  
configurationGlobalId - идентификатор конфигурации, на которой получен результат  
status - статус результата теста (может быть Ready или NotReady)  
outcome - исход теста (принимает значения Passed, NotPassed, Skipped)  
message - сопровождающее сообщение (\*необязательное поле)  
stackTrace - сопровождающая трассировка стека (\*необязательное поле)

Пример cURL запроса в общем виде:

```
curl -X POST "http://192.168.88.140/api/Public/SetAutoTestResult" -H "accept: application/json" -H "SecretKeyBase64: {SecretKeyBase64}" -H "UserName: {UserName}" -H "Content-Type: application/json-patch+json" -d "{ \"testRunId\": \"{testRunId}\", \"testPlanGlobalId\": 0, \"autoTestExternalId\": \"{autoTestExternalId}\", \"configurationGlobalId\": 0, \"status\": \"{Ready/NotReady}\", \"outcome\": \"{Passed/NotPassed/Skipped}\", \"message\": \"string\", \"stackTrace\": \"string\"}"
```

Пример cURL запроса:

```
curl -X POST "http://192.168.88.140/api/Public/SetAutoTestResult" -H "accept: application/json" -H "SecretKeyBase64: a3RxVjJtQlV4RFF0amF5Ng==" -H "UserName: admin" -H "Content-Type: application/json-patch+json" -d "{ \"testRunId\": \"3550656e-cfa5-44f9-812d-996230837136\", \"testPlanGlobalId\": 3036, \"autoTestExternalId\": \"test\", \"configurationGlobalId\": 49, \"status\": \"string\", \"outcome\": \"string\", \"message\": \"string\", \"stackTrace\": \"string\"}"
```

## Сценарий использования:

1. Авторизоваться
2. Генерировать секретный ключ
3. Создать проект
4. Создать test case
5. С помощью POST /api/Public/AddAutoTestToTestCase/{testCaseGlobalId} прикрепляем auto test к test case. **Результат:** test case помечен как автоматизированный
6. Создаём test plan
7. Создаём test suite
8. Добавляем конфигурацию в test suite
9. Добавляем автоматизированный test case в test suite **Результат:** в разделе Результаты 1 test point, помеченный как автоматизированный
10. Создаём Test Run, т.е. выбираем автоматизированный test point и нажимаем на кнопку создания test run
11. Получаем список Test Run по проекту с помощью GET /api/Public/GetTestRuns/{projectGlobalId}
12. Запускаем Test Run через POST /api/Public/StartTestRun/{testRunId}
13. Назначаем результат автотесту через POST /api/Public/SetAutoTestResult
14. Завершаем Test Run
  1. Если проставлены все результаты в автотестах, Test Run переходит в статус completed
  2. Для явного завершения Test Run, т.е. перевод в статус completed необходимо отправить POST /api/Public/CompleteTestRun/{testRunId}
  3. Для остановки Test Run необходимо отправить POST /api/Public/StopTestRun/{testRunId}