Before testing the service task feature, you'll have to use some components.
First, please choose a directory where you'll clone all the components repositories (in the following, this directory will be ~/service-task/).

# I. Testing

## A. All-in-one blockchain

```
Unset
# first install dependencies
npm i

#for testing
npm run autotest fast

#for creating docker images for testing other components
./createTestChains.sh

#for creating docker image for deployment
./createChain.sh

#in order to use PoCo in other components, you first have to minify the
contracts
make minify
```

## B. R_MARKET-SDK

```
Unset
# first install dependencies
npm ci

# SDK needs PoCo as a dependency, in order to use our custom PoCo, we'll
remove it before createing a link to our new PoCo
rm -rf node_modules/@iexec/poco
ln -s ~/service-task/All-in-One_Blockchain/ ./node_modules/@iexec/poco
```

```
#for testing, first run some testing services (./createTestChains.sh in
All-in-one_blockchain must be run before)
cd test
docker compose up -d

# run tests
npm test

#stop testing services
cd test
docker compose down
```

## C. Iexec-common

```
Unset
#build project (it will execute tests and generate jar)
DOCKER_IO_USER=$DOCKER_IO_USER DOCKER_IO_PASSWORD=$DOCKER_IO_PASSWORD
./gradlew build
```

## D. R_MARKET-Scheduler

```
Unset
#link custom iexec-common
ln -s ~/service-task/R-MARKET-common/build/libs
~/service-task/R-MARKET_Scheduler/

#build project (it will execute tests)
./gradlew build
```

## E. Market-API

### 1. api

```
Unset
cd api

# first install dependencies
npm ci

# Market-API needs PoCo as a dependency, in order to use our custom
PoCo, we'll remove it before createing a link to our new PoCo
rm -rf node_modules/@iexec/poco
ln -s ~/service-task/All-in-One_Blockchain/ ./node_modules/@iexec/poco

# Market-API needs the SDK as a dependency, in order to use our custom
SDK, we'll remove it before createing a link to our new SDK

rm -rf node_modules/iexec
ln -s ~/service-task/R-MARKET_SDK ./node_modules/iexec

#for testing, first run some testing services (./createTestChains.sh in
All-in-one_blockchain must be run before)
cd test
docker compose up -d

# run tests
npm test

#stop testing services
cd test
docker compose down
```

### 2. watcher

```
Unset
cd watcher

# first install dependencies
npm ci
```

```
# Market-API needs PoCo as a dependency, in order to use our custom
PoCo, we'll remove it before createing a link to our new PoCo
rm -rf node_modules/@iexec/poco
ln -s ~/service-task/All-in-One_Blockchain/ ./node_modules/@iexec/poco

# Market-API needs the SDK as a dependency, in order to use our custom
SDK, we'll remove it before createing a link to our new SDK

rm -rf node_modules/iexec
ln -s ~/service-task/R-MARKET_SDK ./node_modules/iexec

#for testing, first run some testing services (./createTestChains.sh in
All-in-one_blockchain must be run before)
cd test
docker compose up -d

# run tests
npm test

#stop testing services
cd test
docker compose down
```

# II. Image Generation

## A. All-in-one blockchain

```
Unset
#for creating docker image for deployment
./createChain.sh
```

## B. R_MARKET-SDK

```
Unset
#generate binary
```

```
npm run build

#install custom SDK (optional) otherwise you'll have to use relative or
absolute path
npm i -g ~/service-task/R-MARKET_SDK/
```

## C. Iexec-common

```
Unset
#build project (it will execute tests and generate jar)
DOCKER_IO_USER=$DOCKER_IO_USER DOCKER_IO_PASSWORD=$DOCKER_IO_PASSWORD
./gradlew build
```

## D. R_MARKET-Scheduler

```
Unset
#link custom iexec-common
ln -s ~/service-task/R-MARKET-common/build/libs
~/service-task/R-MARKET_Scheduler/

#build project (it will execute tests)
./gradlew build

#create docker image
./gradlew buildImage
```

### E. Market-API

1. api

```
Unset
docker build -f Dockerfile.api -t iexec-market-api:5.3.1 .
```

2. watcher

```
Unset
docker build -f Dockerfile.watcher -t iexec-market-watcher:5.3.1 .
```

## III. Deployment

```
Unset
cd deployment/

./restart_services.sh
./publish_wp_order.sh
```