

Opc Server with Milo and Spring Boot

Documentazione di progetto per la materia Informatica Industriale

- [Introduzione](#)
- [Perché Spring](#)
- [Descrizione del progetto](#)
- [Funzionameto](#)
- [Annex1: GitHub](#)
 - [Step1: Creare il progetto su github](#)
 - [Step2: Readme and .gitignore](#)
 - [Step3: Repo Download](#)
- [Annex 2: Spring Boot](#)
 - [Step1: Spring Init](#)
 - [Step2: Importare il progetto Maven dentro Eclipse](#)
 - [Step3: Apache Milo](#)

Introduzione

Il presnete documento presenta la documentazione di progetto per la materia informatica industriale





Perché Spring

Descrizione del progetto

Funzionameto

Annex1: GitHub

Step1: Creare il progetto su github



 Search or jump to... / [Pull requests](#) [Issues](#) [Marketplace](#) [Explore](#)   

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).


Owner


Repository name *

 cristinalombardo / 

Great repository names are short and memorable. Need inspiration? How about [fantastic-train](#)?

Description (optional)


☒  **Public**
Anyone can see this repository. You choose who can commit.

☐  **Private**
You choose who can see and commit to this repository.

Skip this step if you're importing an existing repository.


☒ **Initialize this repository with a README**
This will let you immediately clone the repository to your computer.

Add .gitignore: **None**

Add a license: **MIT License** 

[Create repository](#)

© 2020 GitHub, Inc. [Terms](#) [Privacy](#) [Security](#) [Status](#) [Help](#)



[Contact GitHub](#) [Pricing](#) [API](#) [Training](#) [Blog](#) [About](#)

Step2: Readme and .gitignore


Aggiungere Readme e .gitignore

Come template del .gitignore è stato scelto Maven.

Skip this step if you're importing an existing repository.

☒ **Initialize this repository with a README**
This will let you immediately clone the repository to your computer.


Add .gitignore: **None**

Add a license: **MIT License** 

.gitignore

Maven

2020 GitHub, Inc. [Terms](#) [Privacy](#) [Security](#) [Status](#) [Help](#)



[Cont](#)

Step3: Repo Download

Creare il repository e clonarlo in locale con il programma preferito di gestione del git.

Annex 2: Spring Boot

Prerequisiti:

- Java 9+
- Eclipse per j2ee con Spring Tools Suite Plugin installato

Step1: Spring Init

Per inizializzare il progetto utilizzeremo Spring Initializr

[Go to Spring initializr](#)

Inizializzare il progetto in accordo alla seguente immagine

The screenshot shows the Spring Initializr web interface. On the left, there is a hamburger menu icon and a Twitter icon. The main content area is divided into several sections:

- Project:** Includes radio buttons for **Maven Project** (selected) and **Gradle Project**.
- Language:** Includes radio buttons for **Java** (selected), **Kotlin**, and **Groovy**.
- Spring Boot:** Includes radio buttons for versions: 2.3.1 (SNAPSHOT), **2.3.0** (selected), 2.2.8 (SNAPSHOT), 2.2.7, 2.1.15 (SNAPSHOT), and 2.1.14.
- Project Metadata:** Includes text input fields for:
 - Group:** com.github.cristinalombardo
 - Artifact:** opcsrver
 - Name:** Spring Boot OPC Server
 - Description:** OPC Server using Apache Camel Milo into Spring Boot
 - Package name:** com.github.cristinalombardo.opcsrver
- Packaging:** Includes radio buttons for **Jar** (selected) and **War**.
- Java:** Includes radio buttons for versions: 14, **11** (selected), and 8.
- Dependencies:** Includes a button **ADD DEPENDENCIES...** with a plus icon and a list of dependencies. The first dependency is **Spring Boot DevTools** (marked as a **DEVELOPER TOOL**) with the description: "Provides fast application restarts, LiveReload, and configurations for enhanced development experience."

At the bottom, there are three buttons: **GENERATE** (with a plus icon and a download icon), **EXPLORE** (with **CTRL + SPACE** shortcut), and **SHARE...**

Per inizializzare il progetto abbiamo utilizzato le seguenti configurazioni:

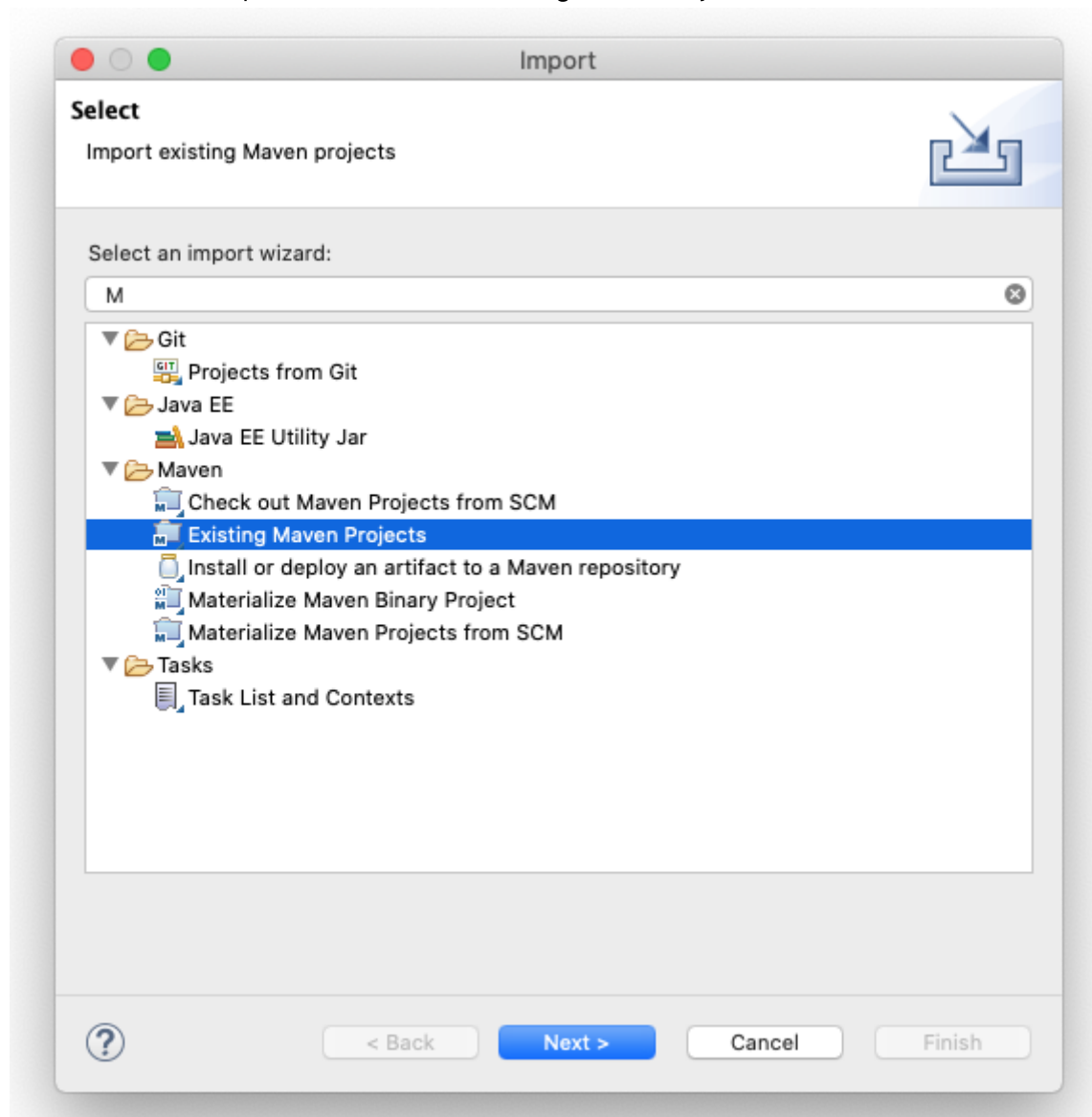
- Maven: Utilizzeremo Maven per compilare il progetto
- Spring Boot 2.3.0: Versione Stabile di Spring Boot al momento di questo tutorial
- Packaging Jar: Si vuole realizzare un'applicazione standalone
- Java 11: Apache Camel Milo richiede Java 9+
- Dependencies Spring Boot DevTools: dipendenza utile durante lo sviluppo

A questo punto è possibile scaricare il progetto cliccando sul tasto GENERATE. Una volta scaricato il progetto è stato copiato nella root del repository e decompattato.

Step2: Importare il progetto Maven dentro Eclipse

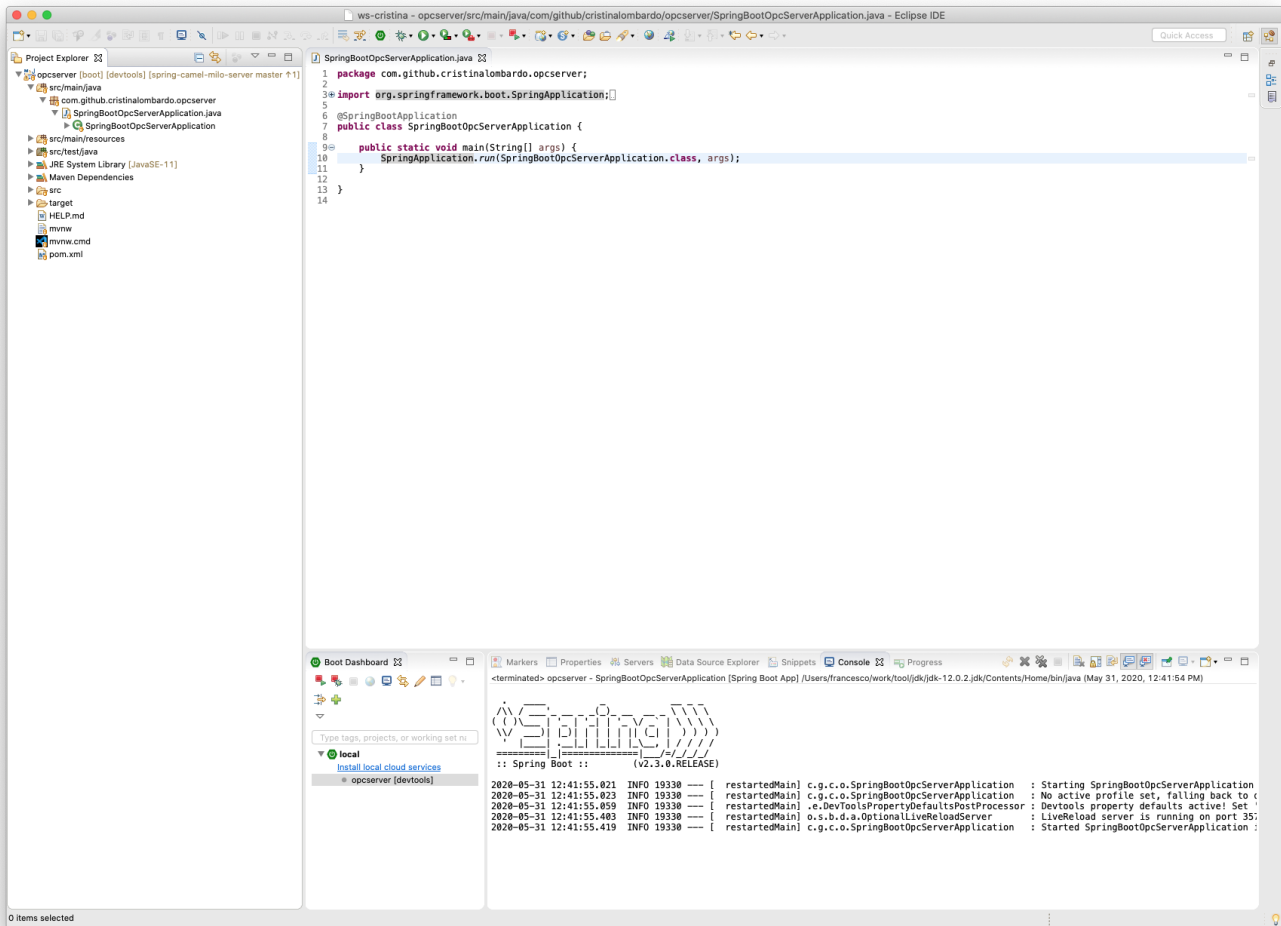
Per importare il progetto i passi sono:

1. Aprire Eclipse
2. Cliccare su File->Import... e selezionare Existing Maven Project



3. Selezionare la cartella dove risiede il progetto e cliccare su Finish

A questo punto il progetto dovrebbe presentarsi nel seguente modo:



Step3: Apache Milo

In questo step aggiungeremo le librerie Apache Camel Milo.

1. Apriamo il file `pom.xml`
2. Aggiungiamo la seguente dipendenza al file

```

<dependency>
  <groupId>org.apache.camel.springboot</groupId>
  <artifactId>camel-milo-starter</artifactId>
  <version>${camel.milo.version}</version>
  <!-- use the same version as your Camel core version -->
</dependency>

```

La versione di Camel viene inserita nella sezione `<properties>` del `pom.xml`

```

<camel.milo.version>3.0.0</camel.milo.version>

```