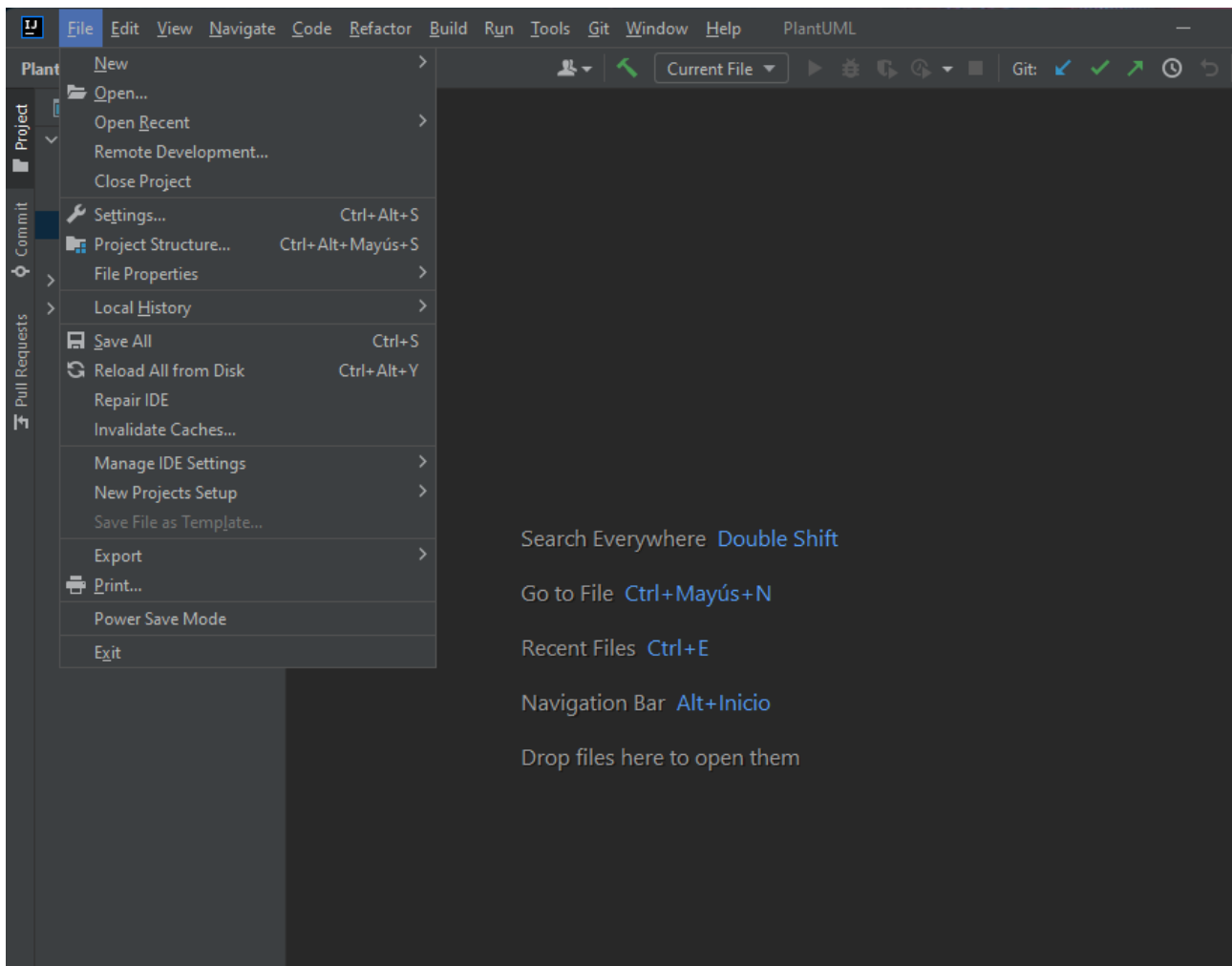


Para instalar PlantUML en IntelliJ, nos vamos a la pestañita que pone file y apretamos a settings.



Dentro de settings vamos a plugins en el panel de la izquierda y veremos arriba en el centro que pone: Marketplace y Installed, le apretamos a marketplace y buscamos el plugin PlantUML. Lo instalamos, y nos pedirá reiniciar el IntelliJ.

Settings

Appearance & Behavior

Keymap

Editor

Plugins1

Version Control

Build, Execution, Deployment

Languages & Frameworks

Tools

Advanced Settings

Plugins


Marketplace

Installed1


Q- Type / to see options

Featured


Show all

 **Test Management**
± 164K ☆ 4.55


Install

 **Kubernetes**
± 3,4M ☆ 4.06


Install

 **AWS Toolkit**
± 4,9M ☆ 2.99


Install

 **Solarized Theme**
± 286,2K ☆ 4.42


Install

 **Grazie Professional**
± 93,2K ☆ 4.86


Install

 **Big Data Tools**
± 1,7M ☆ 4.04


Install

 **Key Promoter X**
± 4,6M ☆ 4.95

Install

 **JetBrains Academy**

Install

 **Test Management**

Install

± 164K ☆ 4.55 JetBrains s.r.o.

Testing 222.4345.14 jun 07, 2023

Plugin homepage ↗

Test Management is a powerful tool for TMS integration. It supports Java, Kotlin, and Python, and provides many smart features for QA specialists.

Some features currently are available for Java/Kotlin & Python only.

Integrate with TMS

- TestRail v6.7 and higher is supported;
- Browse test cases and test runs;
- Search via search everywhere;
- Add links to TMS items in the code with an in-place preview popup.

Java/Kotlin & Python

- Generate tests from TMS test cases.
- Use custom annotations / decorators to reference automated tests with test cases.
- Easily filter in/out test cases referenced from code to find automated/non-automated tests.
- Find potentially outdated automated tests referencing

OK Cancel Apply

Settings

Appearance & Behavior

Keymap

Editor

Plugins1

Version Control

Build, Execution, Deployment

Languages & Frameworks

Tools

Advanced Settings

Plugins


Marketplace

Installed1

Q- Plan


Search Results (45)

Sort By: Relevance

 **PlantUML Integration**


Installed

± 2,8M ☆ 4.63

 **PlantUML Parser**


± 65,2K ☆ 4.20

Install

 **PlantUML Syntax Check**


± 36,7K

Install

 **PlantUML Diagram Generator**


± 62,3K ☆ 3.20

Install

 **PlantUML2DDL**


± 16,1K

Install

 **SpiraPlan Integration**


± 1K

Install


 **Context Free Grammar**

± 216,1K

Install

 **NASM Assembly Language**

Install

 **PlantUML Integration**

Installed

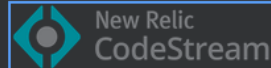
± 2,8M ☆ 4.63 Eugene Steinberg, Vojtech Krasa

Modeling and CASE tools 6.1.0-U2022.2 jun 04, 2023

Plugin homepage ↗

PlantUML diagramming tool integration. Now better and faster, with code navigation and highlighting.

Sponsored by:



Manage pull requests and conduct code reviews in your IDE with full source-tree context. Comment on any line, not just the diffs. Use jump-to-definition, your favorite keybindings, and code intelligence with more of your workflow.

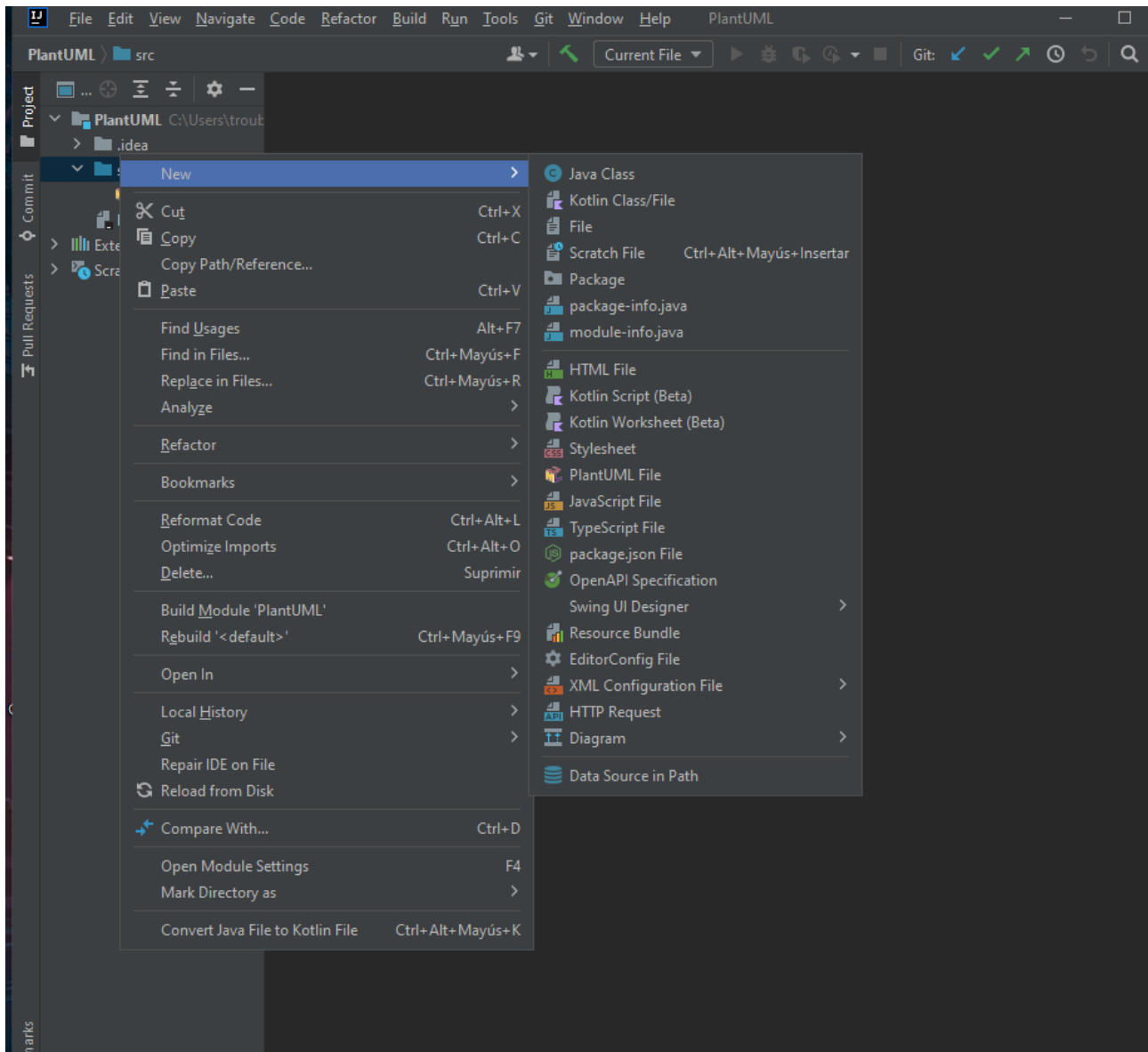
Learn More

Change Notes

Size: 11,3 MB

OK Cancel Apply

Luego de reiniciar apretamos click derecho sobre src o el nombre del proyecto, y en new veremos que pone PlantUML file, le damos y ya nos abre el fichero a editar.



Por último seguimos los pasos del pdf dado con la solución y nos ayudamos de la pagina oficial también dada en el ejercicio para saber calcar el ejercicio UML resuelto. Tiene que quedar parecido a esto.

PlantUML - Test.puml

Project: PlantUML / src / Test.puml

```
1 @startuml
2
3 class AAUOC {
4   newLocation(l : Location): void
5   newEvent(e : Event): void
6   newPerson(p : Person): void
7   informEvent(e : Event): void
8   register(m : Member, e : Event): void
9 }
10
11 AAUOC o-- "0..*" Location
12 AAUOC o-- "0..*" Event
13 AAUOC o-- "0..*" Person
14 class Location {
15   description : String
16   address : String
17 }
18
19 Location "1" -- "0..*" Event : isLocated In
20
21 class Person {
22   name : String
23 }
24 class Event {
25   date : Date
26   description : String
27   assign(l : Location): void
28 }
29
30 Event <|-- BoardMeeting
31 Event <|-- Conference
32 Event "0..*" -- "0..*" Member : attendsTo
33
34 class Member {
35   e-mail : String
36 }
37
38 Member <|-- BoardMember
39
40 class BoardMember {
```

UML Class Diagram:

```
classDiagram
    class AAUOC {
        +newLocation(l : Location): void
        +newEvent(e : Event): void
        +newPerson(p : Person): void
        +informEvent(e : Event): void
        +register(m : Member, e : Event): void
    }
    class Location {
        +description : String
        +address : String
    }
    class Event {
        +date : Date
        +description : String
        +assign(l : Location): void
    }
    class Person {
        +name : String
    }
    class Member {
        +e-mail : String
    }
    class BoardMeeting
    class Conference
    class BoardMember

    AAUOC "0..*" --> "0..*" Location
    AAUOC "0..*" --> "0..*" Event
    AAUOC "0..*" --> "0..*" Person
    Location "1" --> "0..*" Event : isLocated In
    Event <|-- BoardMeeting
    Event <|-- Conference
    Event "0..*" --> "0..*" Member : attendsTo
    Member <|-- BoardMember
```

17/2 CRLF UTF-8 4 spaces master

