



**Università degli Studi di Bologna  
Facoltà di Ingegneria**

# **Principles, Models, and Applications for Distributed Systems M**

***Lab 7 (assignment)  
Java RMI***

**Luca Foschini**

# Specification

---

Use java RMI to develop a C/S application that provides remote methods to:

- **count the occurrences of a character** in a **file** stored at the server node;
- **return the list of file names** (in a **remote directory**) **that contain a character** chosen by the client **in their filename**.

Clients continues to interact with the user until the end of file (EOF).

# Remote methods

---

The RMI project uses a **remote interface** (**RemOp**, defined in file `RemOp.java`) which defines the methods **to be invoked by the client**:

- **count\_occurrences** method receives as input parameters **the file name** and **the character to search** and returns the number of occurrences of the character as an **integer** value; if there is an error, it throws a *remote exception* (for example, if the file does not exist);
- **list\_files\_char** method takes as input parameters **the directory name** and **a character**; then, if the specified directory exists opens it and returns to the client the **list of file names** (as an array of `String`) that contain the requested character in their filename, or error, for example, if the directory does not exist.

# Classes

---

Design and develop:

- **ServerImpl** (file ServerImpl.java), that implements the methods that can be invoked by a client and can be started specifying optionally the registry port:

**ServerImpl [registryPort]**

- **Client** (file Client.java), that implements user interaction, until EOF, asking the user which service she wants to access and printing on console the results:

**Client HostName [registryPort]**

# Remark:

## how to obtain a reference to Server

---

The **rmiregistry** running on a known host and listening on a known port.

The **Server** (instance of `ServerImpl`) registers its remote reference to the Registry.

A **Client** (instance of `Client`) obtains the reference to the remote object `ServerImpl` from the registry and invokes the provided methods.