



---

**TECHNICAL UNIVERSITY**

OF CLUJ-NAPOCA, ROMANIA

**FACULTY OF AUTOMATION AND COMPUTER SCIENCE  
COMPUTER SCIENCE DEPARTMENT**

# DISTRIBUTED SYSTEMS

# Lab Resources

Prof. Tudor Cioara  
As. Liana Todorean  
As. Gabriel Antonesi

S.I. Marcel Antal  
As. Alexandru Rancea

Conf. Cristina Pop  
As. Dan Mitrea

2024-2025




---

**TECHNICAL UNIVERSITY**  
OF CLUJ-NAPOCA, ROMANIA  
**FACULTY OF AUTOMATION AND COMPUTER SCIENCE**  
**COMPUTER SCIENCE DEPARTMENT**

---

## Contents

Contents .....	2
1 Java .....	3
1.1 Java JDK and JRE .....	3
1.2 Set JAVA_HOME and JAVA_JRE variables .....	6
2. Database Server .....	7
2.1. MySQL .....	7
2.2. PostgresSQL .....	9
3 Web Server .....	11
3.1 Apache Tomcat .....	11
3.2 Set the CATALINA_HOME variable .....	11
4 Git .....	13
4.1 Git installation .....	13
4.2 Create Account on GitLab .....	13
4.3 Basic Instructions .....	15
4.3.1 Create a project from scratch .....	15
4.3.2 Update the project .....	17
4.3.3. Create Groups for projects .....	18
4.3.4. Create and work with a new branch .....	18
4.3.5. Getting git to work with a proxy server .....	21
4.3.6. Getting MAVEN to work with a proxy server .....	21




---

**TECHNICAL UNIVERSITY**  
OF CLUJ-NAPOCA, ROMANIA  
**FACULTY OF AUTOMATION AND COMPUTER SCIENCE**  
**COMPUTER SCIENCE DEPARTMENT**

---

## 1 Java

### 1.1 Java JDK and JRE

1) Access the next link:

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

#### Java 17 available now

Java 17 LTS is the latest long-term support release for the Java SE platform. JDK 17 binaries are free to use in production and free to redistribute, at no cost, under the [Oracle No-Fee Terms and Conditions License](#).

[Learn about Java SE Subscription](#)

JDK 17 will receive updates under these terms, until at least September 2024.

#### Java SE Development Kit 17 downloads

Thank you for downloading this release of the Java™ Platform, Standard Edition Development Kit (JDK™). The JDK is a development environment for building applications and components using the Java programming language.

The JDK includes tools for developing and testing programs written in the Java programming language and running on the Java platform.

Documentation Download

Linux   macOS   Windows

Product/file description	File size	Download
Arm 64 Compressed Archive	170.95 MB	<a href="https://download.oracle.com/java/17/latest/jdk-17_linux-aarch64_bin.tar.gz">https://download.oracle.com/java/17/latest/jdk-17_linux-aarch64_bin.tar.gz</a> (sha256 <a href="#">🔗</a> )
Arm 64 RPM Package	153.12 MB	<a href="https://download.oracle.com/java/17/latest/jdk-17_linux-aarch64_bin.rpm">https://download.oracle.com/java/17/latest/jdk-17_linux-aarch64_bin.rpm</a> (sha256 <a href="#">🔗</a> )
x64 Compressed Archive	172.19 MB	<a href="https://download.oracle.com/java/17/latest/jdk-17_linux-x64_bin.tar.gz">https://download.oracle.com/java/17/latest/jdk-17_linux-x64_bin.tar.gz</a> (sha256 <a href="#">🔗</a> )
x64 Debian Package	147.98 MB	<a href="https://download.oracle.com/java/17/latest/jdk-17_linux-x64_bin.deb">https://download.oracle.com/java/17/latest/jdk-17_linux-x64_bin.deb</a> (sha256 <a href="#">🔗</a> )
x64 RPM Package	154.73 MB	<a href="https://download.oracle.com/java/17/latest/jdk-17_linux-x64_bin.rpm">https://download.oracle.com/java/17/latest/jdk-17_linux-x64_bin.rpm</a> (sha256 <a href="#">🔗</a> )

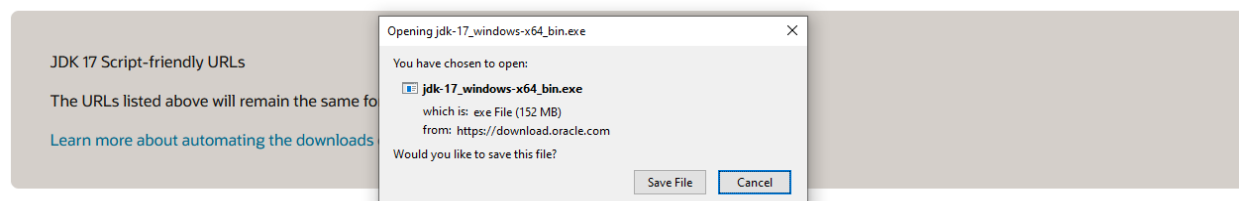
2) Click on the icon which is above Java Platform (JDK). You will be redirected to Java downloads.



**TECHNICAL UNIVERSITY**  
OF CLUJ-NAPOCA, ROMANIA  
**FACULTY OF AUTOMATION AND COMPUTER SCIENCE**  
**COMPUTER SCIENCE DEPARTMENT**

Linux   macOS   **Windows**

Product/file description	File size	Download
x64 Compressed Archive	170.64 MB	<a href="https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.zip">https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.zip</a> (sha256 <a href="#">🔗</a> )
x64 Installer	151.99 MB	<a href="https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.exe">https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.exe</a> (sha256 <a href="#">🔗</a> )
x64 MSI Installer	150.88 MB	<a href="https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.msi">https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.msi</a> (sha256 <a href="#">🔗</a> )



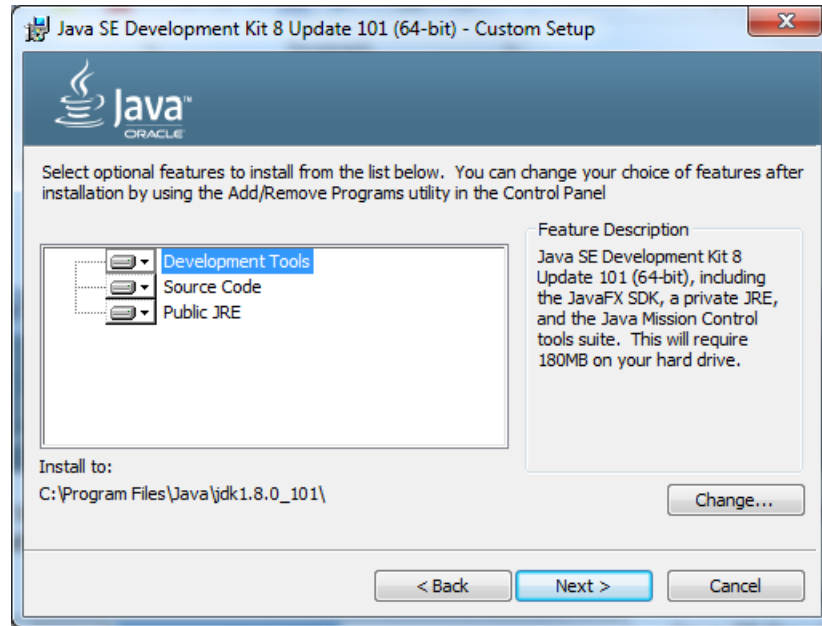
- 3) Click on the link Accept License Agreement.
- 4) Click on the link which corresponds to your version of the Operating System. In the example the version which is used corresponds to Windows x64 and the file is named jdk-17\_windows-x64\_bin.exe.
- 5) After *java-version.exe* is pressed, a file with the same name will be downloaded.
- 6) Click on *java-version.exe*.
- 7) You will be asked the next question: Do you want to allow the following program to make changes to this computer? Click Yes.
- 8) Click Next.
- 9) You will be asked where you want to install Java. Use the default location and click Next.



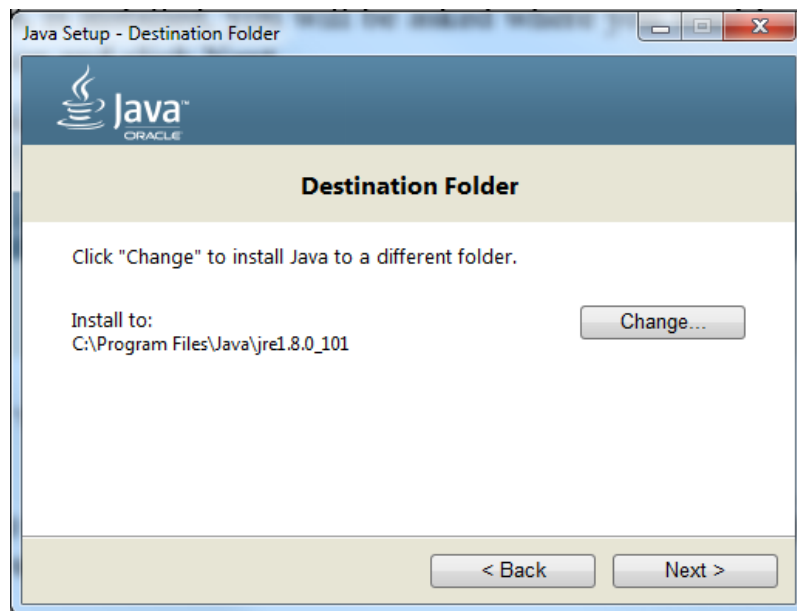
---

**TECHNICAL UNIVERSITY**  
OF CLUJ-NAPOCA, ROMANIA  
**FACULTY OF AUTOMATION AND COMPUTER SCIENCE**  
**COMPUTER SCIENCE DEPARTMENT**

---



10) After the JDK is installed, you will be asked where you want to install the JRE. Use the default location and click Next.



11) After the installation is completed click Close.



---

**TECHNICAL UNIVERSITY**  
OF CLUJ-NAPOCA, ROMANIA  
**FACULTY OF AUTOMATION AND COMPUTER SCIENCE**  
**COMPUTER SCIENCE DEPARTMENT**

---

## 1.2 Set JAVA\_HOME and JAVA\_JRE variables

- 1) Click Start.
- 2) Right-Click on Computer.
- 3) Select Properties.
- 4) Click on Advanced System Settings.
- 5) Click on Environment Variables.
- 6) Under System Variables click New.
- 7) In the text field associated with the name of the variable insert JAVA\_HOME and in the field associated with the value of the variable insert C:\Program Files\Java\java\_version;.
- 8) Click OK.
- 9) Under System Variables click New again.
- 10) In the text field associated with the name of the variable insert JRE\_HOME and in the field associated with the value of the variable insert C:\Program Files\Java\java\_version;.
- 11) Click OK.



---

**TECHNICAL UNIVERSITY**  
OF CLUJ-NAPOCA, ROMANIA  
**FACULTY OF AUTOMATION AND COMPUTER SCIENCE**  
**COMPUTER SCIENCE DEPARTMENT**

---

## 2. Database Server

A database server must be installed on the local machine. MySQL or PostgreSQL are two examples of servers that can be used to run locally the projects.

### 2.1. MySQL

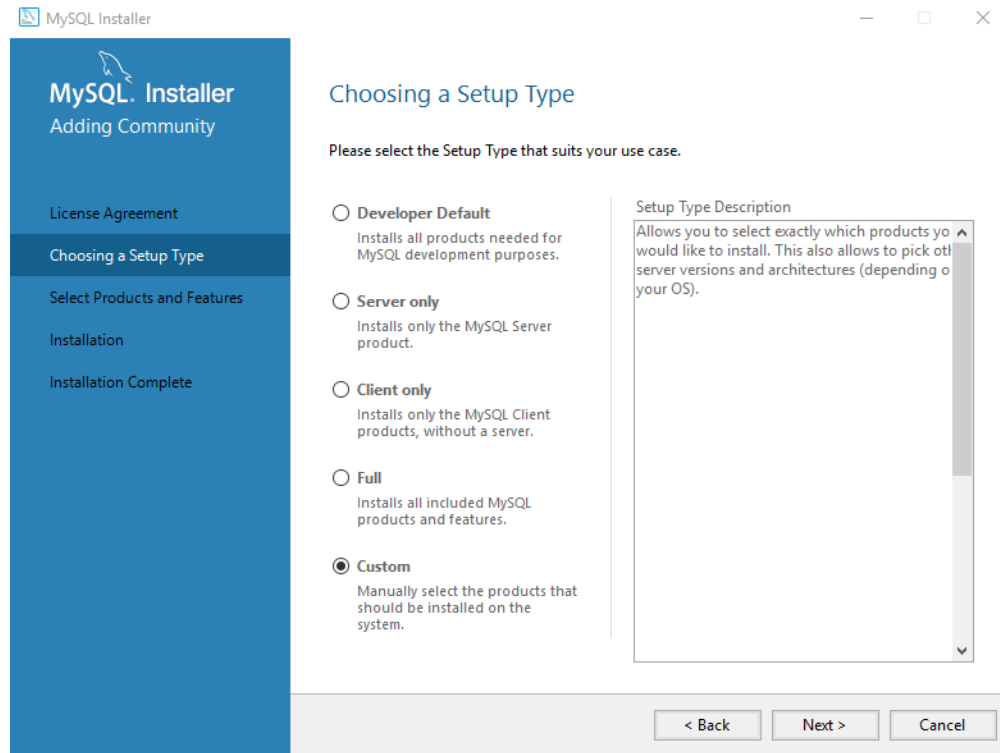
- 1) Click on the next link: <https://dev.mysql.com/downloads/windows/installer/>.

The screenshot shows the MySQL Downloads page for Windows. It features a navigation bar with 'General Availability (GA) Releases' (highlighted in orange), 'Archives', and an information icon. Below the navigation bar is the heading 'MySQL Installer 8.0.26'. A 'Select Operating System:' dropdown menu is set to 'Microsoft Windows'. To the right, there is a link 'Looking for previous GA versions?'. The main content area displays two download options for Windows (x86, 32-bit), MSI Installer. The first option is 'mysql-installer-web-community-8.0.26.0.msi' (2.4M) with a 'Download' button and an MD5 checksum of eaddc383a742775a5b33a3783a4890fb. The second option is 'mysql-installer-community-8.0.26.0.msi' (450.7M) with a 'Download' button and an MD5 checksum of b5b8e6bc39f2b163b817264ae206b815. At the bottom, a blue information icon is followed by the text: 'We suggest that you use the MD5 checksums and GnuPG signatures to verify the integrity of the packages you download.'

- 2) Click on the second Download button.
- 3) Click on No thanks, just start my download.
- 4) Click on the file downloaded file.
- 5) Click Yes.
- 6) Click Yes.
- 7) Click I accept the license terms and then Next.
- 8) You will be asked to select the Setup Type that suits your use case. Select Custom and click Next.



**TECHNICAL UNIVERSITY**  
OF CLUJ-NAPOCA, ROMANIA  
**FACULTY OF AUTOMATION AND COMPUTER SCIENCE**  
**COMPUTER SCIENCE DEPARTMENT**

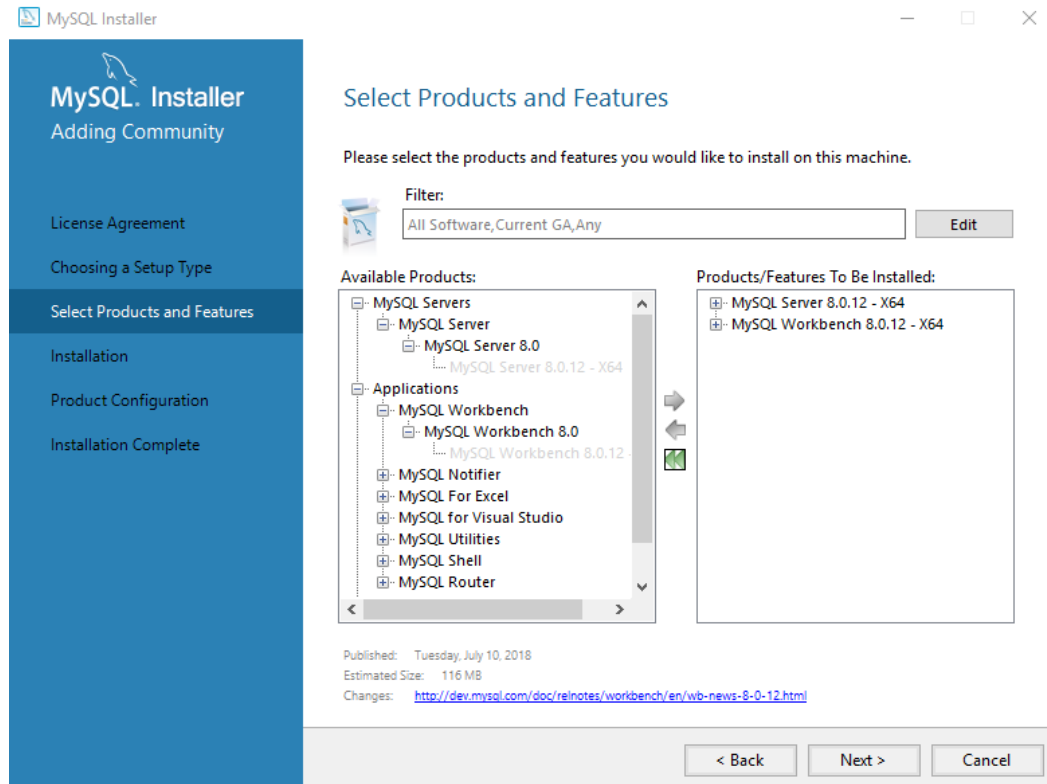


- 9) You will be redirected to “Select Products and Features”. Select “MySQL Server 8.0.26–X64” and “MySQL Workbench 8.0.21–X64” and click Next.





**TECHNICAL UNIVERSITY**  
OF CLUJ-NAPOCA, ROMANIA  
**FACULTY OF AUTOMATION AND COMPUTER SCIENCE**  
**COMPUTER SCIENCE DEPARTMENT**



10) Click Next.

11) Click Execute.

12) Click Next.

13) Click Next and follow the steps for the configuration of the MySQL Server.

## 2.2. PostgreSQL

- 1) Click on the following link: <https://www.postgresql.org/download/>
- 2) Choose the appropriate installer for your operating system
- 3) Click on *Download the installer* on the first row of the page you are redirected to.
- 4) Choose the appropriate version, we recommend the latest one.




---

**TECHNICAL UNIVERSITY**  
OF CLUJ-NAPOCA, ROMANIA  
**FACULTY OF AUTOMATION AND COMPUTER SCIENCE**  
**COMPUTER SCIENCE DEPARTMENT**

### PostgreSQL Database Download

Version	Linux x86-64	Linux x86-32	Mac OS X	Windows x86-64	Windows x86-32
13	N/A	N/A	<a href="#">Download</a>	<a href="#">Download</a>	N/A
12.4	N/A	N/A	<a href="#">Download</a>	<a href="#">Download</a>	N/A
11.9	N/A	N/A	<a href="#">Download</a>	<a href="#">Download</a>	N/A
10.14	<a href="#">Download</a>	<a href="#">Download</a>	<a href="#">Download</a>	<a href="#">Download</a>	<a href="#">Download</a>
9.6.19	<a href="#">Download</a>	<a href="#">Download</a>	<a href="#">Download</a>	<a href="#">Download</a>	<a href="#">Download</a>
9.5.23	<a href="#">Download</a>	<a href="#">Download</a>	<a href="#">Download</a>	<a href="#">Download</a>	<a href="#">Download</a>
9.4.26 (Not Supported)	<a href="#">Download</a>	<a href="#">Download</a>	<a href="#">Download</a>	<a href="#">Download</a>	<a href="#">Download</a>
9.3.25 (Not Supported)	<a href="#">Download</a>	<a href="#">Download</a>	<a href="#">Download</a>	<a href="#">Download</a>	<a href="#">Download</a>

- 5) Next, just follow the basic steps of the installer. You should choose to install all the suggested programs listed there, especially pgAdmin, which is the equivalent of MySQL Workbench, but for PostgreSQL. Moreover, pay attention to select the System Local Time and Date for Romania, or else the timestamps will be different than the actual system time of your computer.
- 6) When asked about other applications you would like to install, we recommend you choose *Database Drivers - pgJDBC* and *Database Server - PostgreSQL (64 bit) v13.0-1*.
- 7) Finish the installation process then enter pgAdmin program to see your fresh database.



---

**TECHNICAL UNIVERSITY**  
OF CLUJ-NAPOCA, ROMANIA  
**FACULTY OF AUTOMATION AND COMPUTER SCIENCE**  
**COMPUTER SCIENCE DEPARTMENT**

---

## 3 Web Server

### 3.1 Apache Tomcat

- 1) Click on the next link: <https://tomcat.apache.org/download-10.cgi>
- 2) Under Binary Distributions look for Core and click on zip.

#### Binary Distributions

- Core:
  - [zip \(pgp, md5, sha1\)](#)
  - [tar.gz \(pgp, md5, sha1\)](#)
  - [32-bit Windows zip \(pgp, md5, sha1\)](#)
  - [64-bit Windows zip \(pgp, md5, sha1\)](#)
  - [32-bit/64-bit Windows Service Installer \(pgp, md5, sha1\)](#)
- Full documentation:
  - [tar.gz \(pgp, md5, sha1\)](#)
- Deployer:
  - [zip \(pgp, md5, sha1\)](#)
  - [tar.gz \(pgp, md5, sha1\)](#)
- Extras:
  - [JMX Remote jar \(pgp, md5, sha1\)](#)
  - [Web services jar \(pgp, md5, sha1\)](#)
- Embedded:
  - [tar.gz \(pgp, md5, sha1\)](#)
  - [zip \(pgp, md5, sha1\)](#)

- 3) A file called `apache-tomcat-version.zip` is downloaded.
- 4) Extract the content of this file on C:\. The file `startup.bat` should be at the location `C:\apache-tomcat-version\bin`.

### 3.2 Set the CATALINA\_HOME variable

- 1) Click Start.
- 2) Right-Click on Computer.
- 3) Select Properties.
- 4) Click on Advanced System Settings.
- 5) Click on Environment Variables.
- 6) Under System Variables click New.
- 7) In the text field associated with the name of the variable insert `CATALINA_HOME` and in the field associated with the value of the variable insert `C:\apache-tomcat-version;`.



---

**TECHNICAL UNIVERSITY**  
OF CLUJ-NAPOCA, ROMANIA  
**FACULTY OF AUTOMATION AND COMPUTER SCIENCE**  
**COMPUTER SCIENCE DEPARTMENT**

---



8) Click OK.



---

**TECHNICAL UNIVERSITY**  
OF CLUJ-NAPOCA, ROMANIA  
**FACULTY OF AUTOMATION AND COMPUTER SCIENCE**  
**COMPUTER SCIENCE DEPARTMENT**

---

## 4 Git

### 4.1 Git installation

- 1) Click on <https://git-scm.com/downloads>.
- 2) Select your operating system.

## Downloads



Older releases are available and the Git source repository is on GitHub.



- 3) If you select Windows, a file called *Git-2.33.0-64-bit.exe* should be downloaded. In the case you select another operating system or if your system is on 32 bits then a file with a similar name should be downloaded.
- 4) Click on this file and follow the default installation guidelines, except for the step where you are asked which terminal emulator you want to use. Select the second option.

### 4.2 Create Account on GitLab

- 1) Click on <https://about.gitlab.com/>.
- 2) In the right corner, click on Register. You will be asked to introduce your personal information. Or, if you already have an account, just *Sign In*.



---

**TECHNICAL UNIVERSITY**  
OF CLUJ-NAPOCA, ROMANIA  
**FACULTY OF AUTOMATION AND COMPUTER SCIENCE**  
**COMPUTER SCIENCE DEPARTMENT**

---

### Register for GitLab

First name	Last name
<input type="text"/>	<input type="text"/>


Username

Email

Password  
  
Minimum length is 8 characters

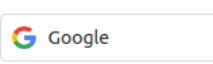

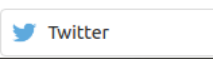
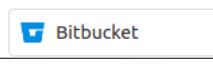
I accept the [Terms of Service and Privacy Policy](#)

I'd like to receive updates via email about GitLab.

I'm not a robot   
reCAPTCHA  
Privacy - Terms

or

Create an account using:

 Google	 GitHub
 Twitter	 Bitbucket

- 3) In the next window, you can choose the role of “Software Developer”. In the next checkbox, you can choose the “Just me” option.
- 4) Following up, you will have to create a group. The name of the **PRIVATE** group must be of the format: *DS2024\_GroupNumber\_LastName\_FirstName*  
(e.g. *DS2024\_30441\_Popescu\_Ioan*)
- 5) For now, do not invite any other teammates to have access to the group, as we will do this later.




---

**TECHNICAL UNIVERSITY**  
OF CLUJ-NAPOCA, ROMANIA  
**FACULTY OF AUTOMATION AND COMPUTER SCIENCE**  
**COMPUTER SCIENCE DEPARTMENT**

---

- 6) Click *Create group*.
- 7) For the next section, *Create/import your first project*, just write a random name in the *Project name* text field, like *test project*. We will create a project for each assignment, but in a few minutes.
- 8) Click *Create project*
- 9) You can choose any experience level you consider having, because this document will detail all the steps necessary to complete your assignments.
- 10) After selecting the experience level, you will see the *home screen* of the group you have just created.
  - a. If you already had an account and just signed in, you must create the group we have just talked about. Go to: *Groups* → *Your Groups* → *New Group* then create a new **PRIVATE** group with the format: *DS2024\_GroupNumber\_LastName\_FirstName*  
(e.g. *DS2024\_30441\_Popescu\_Ioan*)
- 11) Now you must give access to your group, to the DS lab assistants. On your Group page, go to: *Members* → *Invite Member* → and offer **Maintainer** rights for the user: [utcn.dsrl@gmail.com](mailto:utcn.dsrl@gmail.com).
- 12) Inside the group, you can create your own projects for different applications of the DS lab. Remember to keep the same naming conventions for the projects, considering the following formats:
  - a. *DS2024\_GroupNumber\_LastName\_FirstName\_AssigNumber*
  - b. *DS2024\_GroupNumber\_LastName\_FirstName\_AssigNumber\_Backend*
  - c. *DS2024\_GroupNumber\_LastName\_FirstName\_AssigNumber\_Frontend*

## 4.3 Basic Instructions

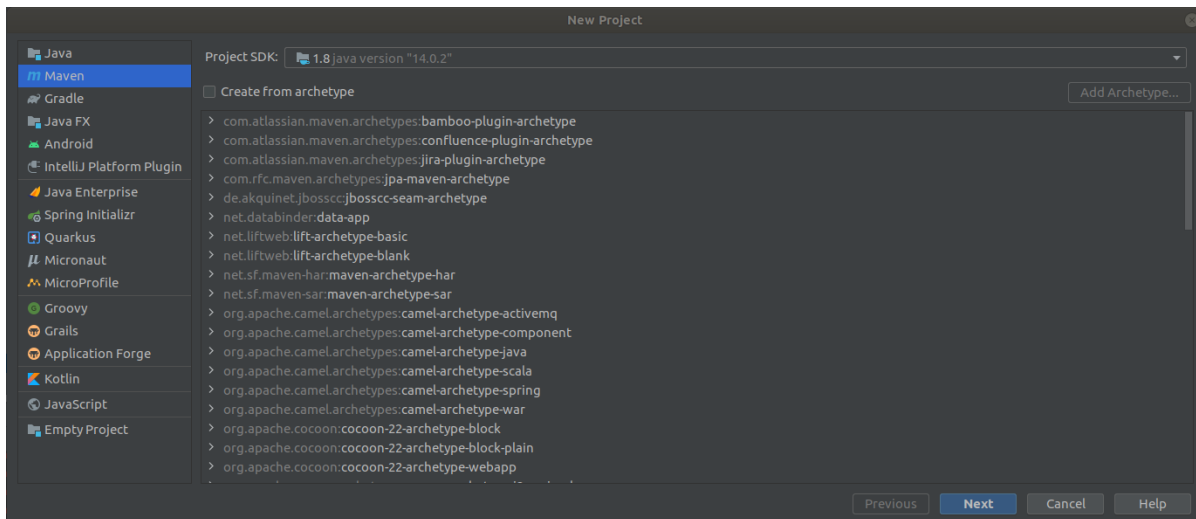
### 4.3.1 Create a project from scratch

- 1) Create the folder *DS2024\_GroupNumber\_LastName\_FirstName\_AssigNumber* on *D:\*.
- 2) Right click on this folder and click *Git Bash Here*.
- 3) Write the next commands:
  - a) `git init`
  - b) `git remote add origin`  
[https://gitlab.com/group\\_name/project\\_name.git](https://gitlab.com/group_name/project_name.git)



**TECHNICAL UNIVERSITY**  
OF CLUJ-NAPOCA, ROMANIA  
**FACULTY OF AUTOMATION AND COMPUTER SCIENCE**  
**COMPUTER SCIENCE DEPARTMENT**

- 4) Open IntelliJ, select *File -> New -> Project*. Then choose *Maven* from the list from the left, choose the appropriate Java SDK, then click on *Next*.



- 5) Instead of using the default location suggested, use this one:

*D:\DS\_Group\_LastName\_FirstName*

*Do not forget to also choose the appropriate name for your new Project, most likely using the format we requested (unless it is a test project, not for one of your assignments).*

- 6) Click *Finish*.
- 7) Now, before pushing anything to the remote project, you must create a `.gitignore` file, which tells git which files to ignore when committing and pushing to your remote projects. You don't want unnecessary files, such as IDE configuration files, to be pushed, because they are strictly relevant for your local system. Just create a file named `".gitignore"` and write the following lines:






---

**TECHNICAL UNIVERSITY**  
OF CLUJ-NAPOCA, ROMANIA  
**FACULTY OF AUTOMATION AND COMPUTER SCIENCE**  
**COMPUTER SCIENCE DEPARTMENT**

---

```

.gitignore 268 Bytes
1 /target/
2 !.mvn/wrapper/maven-wrapper.jar
3
4 ### STS ###
5 .apt_generated
6 .classpath
7 .factorypath
8 .project
9 .settings
10 .springBeans
11 .sts4-cache
12
13 ### IntelliJ IDEA ###
14 .idea
15 *.iws
16 *.iml
17 *.ipr
18

```

\*\* You can go to our demo project on GitLab and just copy the .gitignore file from there.

- 8) Right click on the folder *DS\_Group\_LastName\_FirstName* and introduce the next commands:
  - a) `git add .`
  - b) `git commit -a -m "initial commit"`
  - c) `git push -u origin master`

#### 4.3.2 Update the project

- 1) Create a new class named *Main* in the same package as the class *App*.
- 2) Navigate to folder *DS\_Group\_LastName\_FirstName*, right click and select *Git Bash*
- 3) Insert the next commands:
  - a) `git add .`
  - b) `git commit -a -m "add new class"`




---

**TECHNICAL UNIVERSITY**  
OF CLUJ-NAPOCA, ROMANIA  
**FACULTY OF AUTOMATION AND COMPUTER SCIENCE**  
**COMPUTER SCIENCE DEPARTMENT**

---

- c) git pull origin master
  - d) git push -u origin master
- 4) You can always see the modification that were not committed yet by using:
- a) git status

#### 4.3.3. Create Groups for projects

Although you have probably created your first group and project on GitLab (if you have followed the instructions from section 4.2), we will repeat the basic steps for working with groups on GitLab.

- 1) You will have a group corresponding to all your DS projects for this semester. To create a new group, go to:

*Groups* → *Your Groups* → *New Group* then create a new **PRIVATE** group with the format:

*DS2024\_GroupNumber\_LastName\_FirstName*

*(e.g. DS2024\_30441\_Popescu\_Ioan)*

- 2) Now you must give access to your group, to the DS lab assistants. On your Group page, go to: *Members* → *Invite Member* → and offer **Maintainer** rights for the user: [utcن.dsrl@gmail.com](mailto:utcن.dsrl@gmail.com) .
- 3) Inside the group, you can create your own projects for different applications of the DS lab. Remember to keep the same naming conventions for the projects, considering the following formats:
  - a. *DS2024\_GroupNumber\_LastName\_FirstName\_AssigNumber*
  - b. *DS2024\_GroupNumber\_LastName\_FirstName\_AssigNumber\_Backend*
  - c. *DS2024\_GroupNumber\_LastName\_FirstName\_AssigNumber\_Frontend*

#### 4.3.4. Create and work with a new branch

The real value of working with git, is the power of *branches*. They allow multiple developers to work simultaneously on the same project, on different features, and then to merge all the new changes in the master branch. In the DS laboratory, you will have to create a production branch, which will then be configured to be automatically tested and deployed, on each new commit and push.



---

**TECHNICAL UNIVERSITY**  
OF CLUJ-NAPOCA, ROMANIA  
**FACULTY OF AUTOMATION AND COMPUTER SCIENCE**  
**COMPUTER SCIENCE DEPARTMENT**

---

### 1) Create a branch production

The first step towards working with branches, is to create a new branch. When you create a new branch, it will automatically be initialized with the currently existent code. Then, while inside that branch, all changes will be added only on that branch.

In order to create a new branch:

- pull all the changes from the remote project, in order to be up to date:
  - *git pull*
- create the branch on your local machine and switch directly to that branch:
  - *git checkout -b <branch-name>*
- push the newly created branch to the remote repository:
  - *git push origin <branch\_name>*

An example to create the branch production is the following:

```
#git pull
#git checkout -b production
#git push origin production
```

### 2) switch between branches

When working with multiple branches, it is important to keep track of all the available branches, and to always know on which branch you currently are.

- To bring locally meta-data information about existing branches:
  - *git fetch --all*
- In order to see all existent branches:
  - *git branch -a*
- In order to see on which branch you currently are:
  - *git status*
- *in order to switch from a branch to another, use the above-mentioned command:*



---

**TECHNICAL UNIVERSITY**  
OF CLUJ-NAPOCA, ROMANIA  
**FACULTY OF AUTOMATION AND COMPUTER SCIENCE**  
**COMPUTER SCIENCE DEPARTMENT**

---

- *git checkout -b <branch-name>*
  - *if the branch with that name is already existent, it will just switch to that one, instead of creating a new one*

*You can try to create a new branch, make some small change, then switch back to the master branch, and see that that change is not present in the master branch.*

### 3) commit changes to new branch

When making changes on a new branch, you must always commit and push them to the remote branch, just like working on master.

- First, make sure you are on the right branch:
  - *git status*
- Repeat the same process as if you were working with master. However, pay attention to the names:
  - *git add .*
  - *git commit -m "commit message"*
  - *git push -u origin <branch-name>*
- And now, your remote branch <branch-name> contains all the changes you have pushed.

### 4) merge branch with master

An important step when working with branches, is to always keep the master branch up to date with the latest **working and functional** code from your other branches. Merging two branches, as the name suggests, is the process of merging the code from two branches. If the branches contain changes on different parts of the code, the merge process will work instantly. If both branches contain changes on the same parts of code, git will require you to solve the conflicts: from the two modifications, you must choose the one which you want to remain in the final version.

**DO NOT FORGET:** do not merge code which is not working properly, or which is not tested, into master. Master must always contain the latest functional version of your project.

In order to merge two branches:

- *git merge <branch-with-new-changes> <branch-to-be-updated>*

For a more in-depth explanation of branches and how they can be manipulated to serve your needs, we suggest checking the following tutorial:

<https://git-scm.com/book/en/v2/Git-Branching-Basic-Branching-and-Merging>



---

**TECHNICAL UNIVERSITY**  
OF CLUJ-NAPOCA, ROMANIA  
**FACULTY OF AUTOMATION AND COMPUTER SCIENCE**  
**COMPUTER SCIENCE DEPARTMENT**

---

#### 4.3.5. Getting git to work with a proxy server

- 1) In the UTCN laboratories you need to set the proxy server in order to use GIT bash
- 2) Open Git Bash
- 3) Insert the following commands:
  - a) `git config --global http.proxy http://proxy.utcluj.ro:3128`
  - b) `git config --global --get http.proxy`
- 4) In order to unset the proxy, use the following command:
  - a) `git config --global --unset http.proxy`

#### 4.3.6. Getting MAVEN to work with a proxy server

- 1) In the UTCN laboratories you need to set the proxy server in order to use MAVEN projects
- 2) Go to Windows Explorer-> Drive C-> Users -> *Your User* -> .m2
- 3) Create the folder **conf**
- 4) Go to conf folder and create the file **settings.xml** with the following content:

```
<settings xmlns="http://maven.apache.org/SETTINGS/1.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/SETTINGS/1.0.0
    http://maven.apache.org/xsd/settings-1.0.0.xsd">
  <localRepository/>
  <interactiveMode/>
  <usePluginRegistry/>
  <offline/>
  <pluginGroups/>
  <servers/>
  <mirrors/>
  <proxies>
    <proxy>
      <id>myproxy</id>
      <active>true</active>
      <protocol>http</protocol>
      <host>proxy.utcluj.ro</host>
```



---

**TECHNICAL UNIVERSITY**  
OF CLUJ-NAPOCA, ROMANIA  
**FACULTY OF AUTOMATION AND COMPUTER SCIENCE**  
**COMPUTER SCIENCE DEPARTMENT**

---

```
<port>3128</port>
<username></username>
<password></password>
<nonProxyHosts>localhost,127.0.0.1</nonProxyHosts>
</proxy>
</proxies>
<profiles/>
<activeProfiles/>
</settings>
```

- 5) Go back to folder **.m2**
- 6) Delete the folder **repository**
- 7) Open **Eclipse**
- 8) Go to **Window->|Preferences->|Maven->|User Settings**
- 9) At the **User Settings** tab browse for the **settings.xml** file created at step 4
- 10) Click **Apply** and **OK**
- 11) Go on the project, right click and go to **Maven->Update Project**