Installation Guide

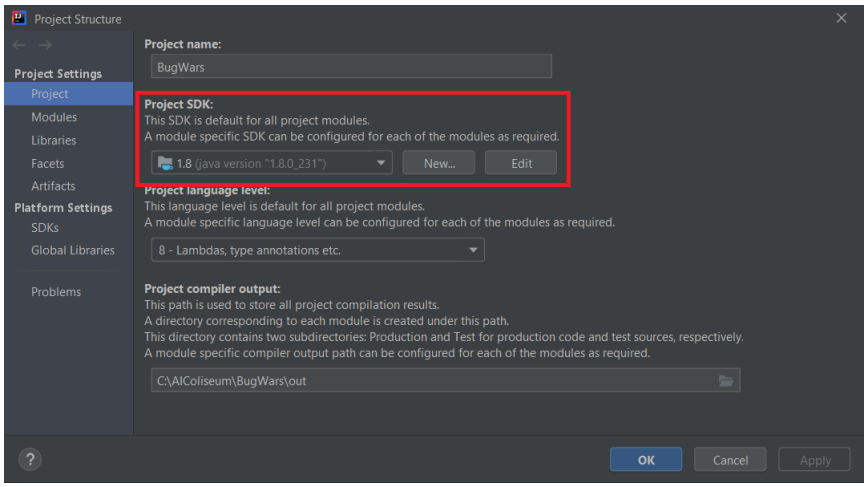
1 Project Configuration

1. Download JetBrains's IntelliJ Community Edition, choose the OS (Windows, Linux or MacOS), and follow the suggested installation steps. For more information about the installation process, you can check this link.

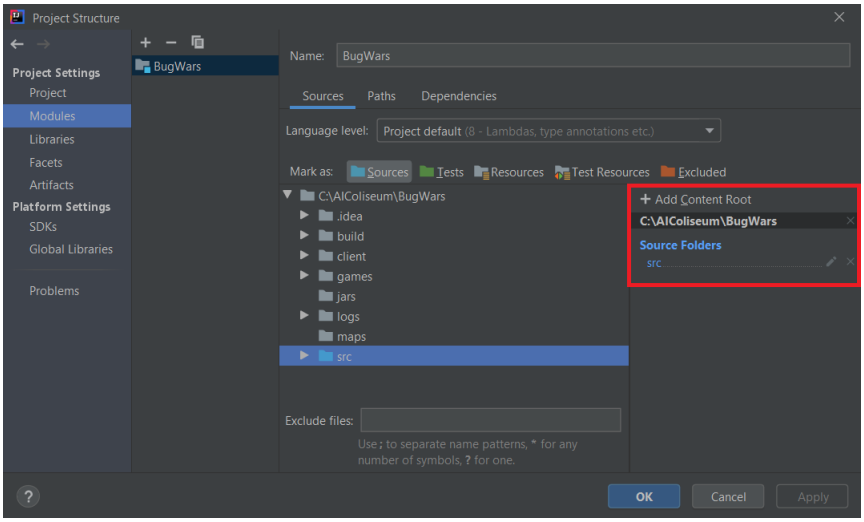
2. Download the corresponding game project at AI Coliseum's download tab.

3. Open IntelliJ IDEA and select *Import Project*. Then, select the project folder (downloaded at step 2). A new window will open, select *Create project from existing sources* and leave all default settings (name, location, libraries and module structure). If there is no 1.8 option for *Project SDK* leave that empty. If this is done correctly, it will save most of the work over the next steps.

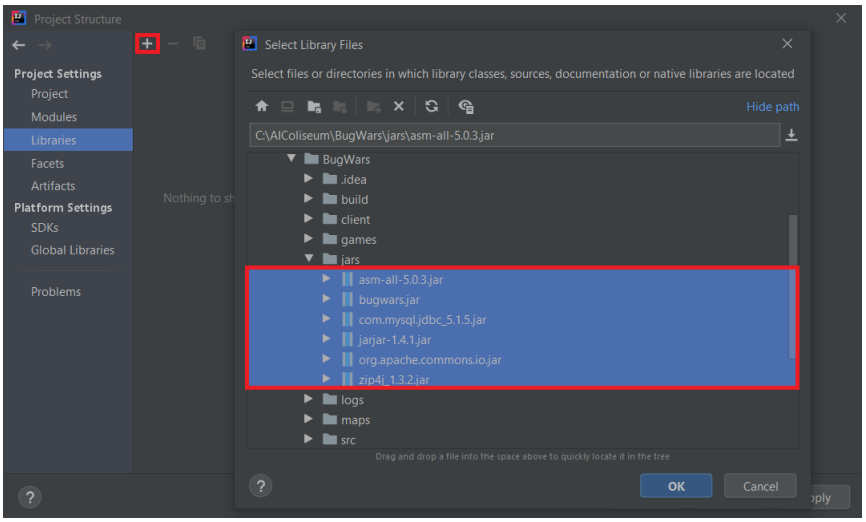
4. Press Alt+Ctrl+Shift+S. On Project Settings > Project, check that there's the JDK 1.8 version at *Project SDK*. Otherwise, follow the instructions at Section 2 or, alternatively, follow the instructions here (It seems that sometimes this may bring unexpected errors! We suggest following the instructions at Section 2).



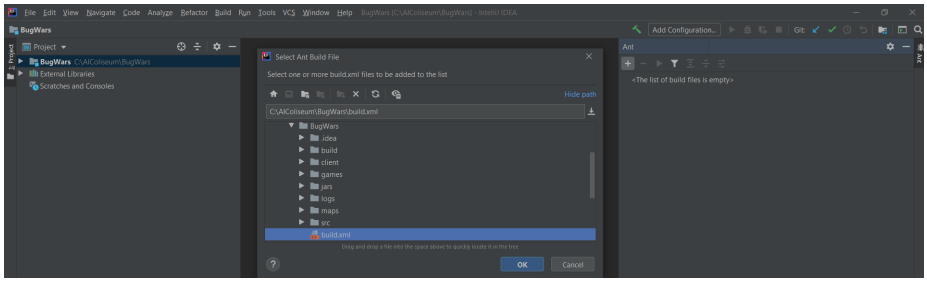
5. Without closing this window, at the *Modules* section check that there is a module with the name of the project downloaded at Step 2. Otherwise, add a new Java module (by clicking over the "+" symbol) and add the main folder as *root*. Also check that the *src* folder is marked as *Source Folder*, otherwise, click on *src* and then on the blue folder with the *Sources* label.



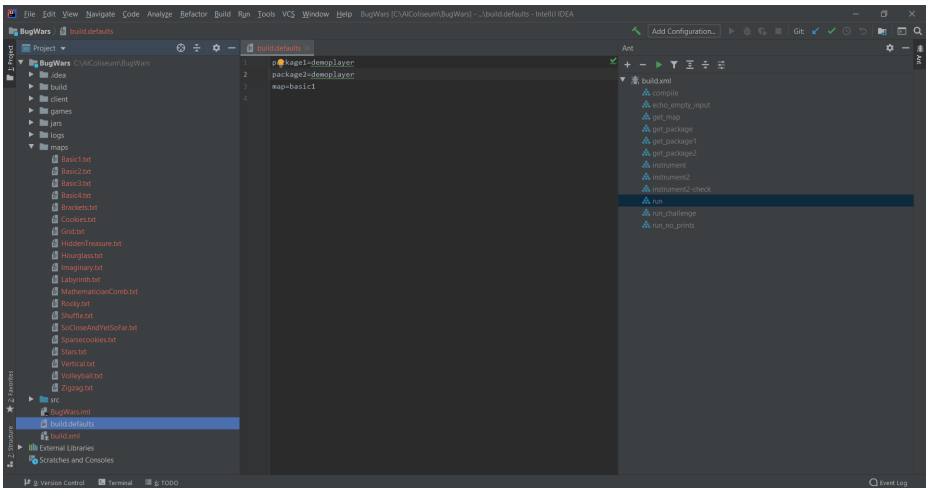
6. At the Libraries tab (same window), check that there is a *jars* library with all the .jar files inside the *jars* folder. Otherwise, click on "+" and select all .jar files inside the *jars* folder as the following image shows. Then, press *OK*.



7. Close the *Project Structure* window and, the *View* tab, select *Tool Windows* > *Ant Build*. In this new tab, click on "+" and select the *Build.xml* at the project folder (you can also drag *Build.xml* to the *Ant Build* tab. This should add a new list of options inside the *Ant Build* tab.



8. Run a test game to check that everything works smoothly. Open the *build.defaults* file at the project folder, fill the three given lines with the names of the desired players and map respectively (we suggest to use only lower letters for player names). Select the *run* option at *Ant Build* and this will run a game between the two specified bots at the given map. We suggest to try with *demoplayer* vs. *demoplayer* (we always provide a *demoplayer* for each game) at the default map (in *Bugwars* it is *basic1*). The list of all maps can be checked at the *maps* folder.

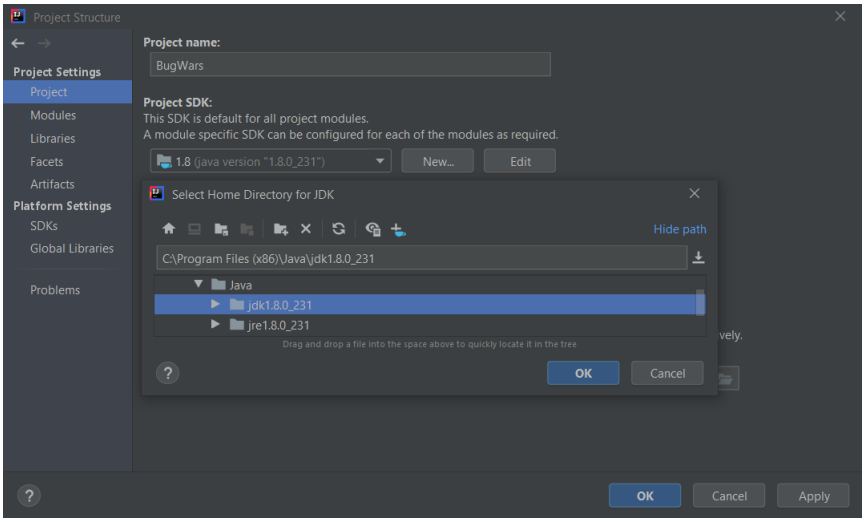


2 Update the Project SDK

1. Download, for whatever OS you are currently using, the Java SE Development Kit 8 (JDK) from Oracle's website.

2. Install the JDK. Its installation depends on the OS, for more information check here. In Windows, it is only necessary to follow the installation steps suggested by the application.

3. Open IntelliJ and press Alt+Ctrl+Shift+S. At *Project Settings* > *Project* press the *New...* button at *Project SDK* and select the folder where you installed the JDK downloaded at step 1 (in Windows, the default location is C:\Program Files\Java or C:\Program Files (x86)\Java). Then, press *OK*.

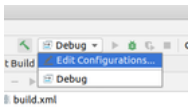


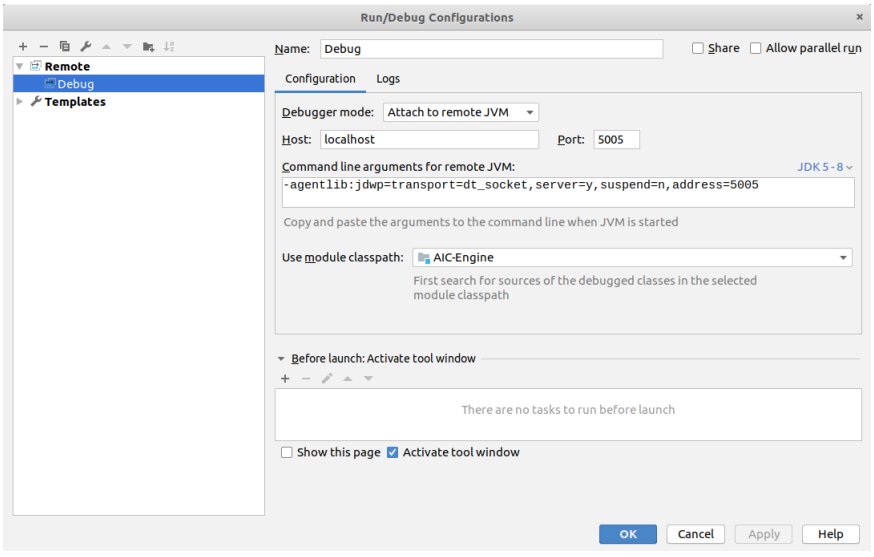
4. Optionally, at Ubuntu you can substitute steps 1 and 2 for the following instructions.

3 Add debug configuration (Optional)

To debug your bot, you must configure IntelliJ to connect to the process once a game has been launched. Debugging is very useful since it allows you to set breakpoints, inspect variables and evaluate expressions at run time.

To add the configuration go to Edit Configurations / + / Remote / and save it (you can rename it if you want).





Now you can play the run_debug target with Ant, and press the debug button to attach IntelliJ to the running game.

4 Create a new bot

To create a new bot, it is necessary to create a new package at the *src* folder with the desired name. This package must contain a *UnitPlayer* class with a public *run()* method, exactly in the same way as *nullplayer* and *demoplayer*. It is suggested to just copy the *UnitPlayer* class from *nullplayer* and fill it up.

5 Using the Client to watch replays

Tu run a game, follow the instructions of Step 8 of Section 1. All games that were run this way will be saved in the *games* folder, inside the main project folder. To watch the replay, go to the website viewer and select the corresponding game.

Note that the web viewer can be downloaded! (Click on the icon at the right side of the browser).

6 Upload a bot

To upload a new bot, you must compress its package (including the folder) in a .zip file and select it at AI Coliseum > Codes (you must also specify what game is this bot for at the *Tournament* tab).

7 Teams

During official events, all accounts of members of the same team must be inscribed as a team for the corresponding game. To create a new team, you must click over your username, select the *Teams* tab, and create a new team for the corresponding game. Once the team is created, you can invite other users to join by typing their username and clicking "+", and they may accept or reject. Teams act as single entities for the competition, which means that all members of the team have access to all previosly submitted bots and may submit new bots, however only one bot is active at a time for the whole team. Team modifications are forbidden after a certain point after the event starts to prevent abuses (for specific cases you must contact the organizers).