Grading Ecosystem – Development & Testing Environments



Contents

[Introduction 3](#_Toc368055542)

[Development Environment - Setup Guide 3](#_Toc368055543)

[Setting up the VM 3](#_Toc368055544)

[GIT 8](#_Toc368055545)

[JDK 8](#_Toc368055546)

[Maven 8](#_Toc368055547)

[Eclipse 8](#_Toc368055548)

[MySQL 9](#_Toc368055549)

[MySQL Workbench 9](#_Toc368055550)

[MongoDB 9](#_Toc368055551)

[Curl 10](#_Toc368055552)

[Ruby on Rails 10](#_Toc368055553)

[RestClient 11](#_Toc368055554)

[Setting up the Development Environment 11](#_Toc368055555)

[Testing the Setup 14](#_Toc368055556)

[Testing Environment – Setup Guide 14](#_Toc368055557)

[References 14](#_Toc368055558)

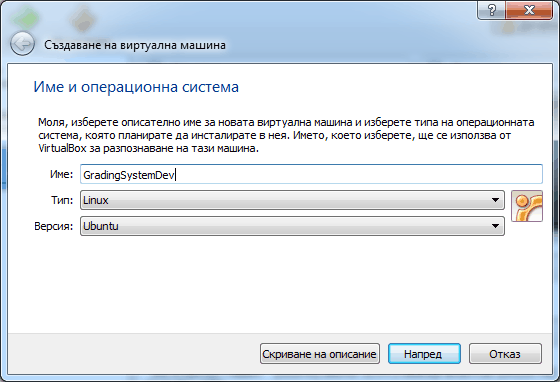
# Introduction

The following document provides a detailed overview of the steps required to setup a development environment for the grading ecosystem. Since we will be developing on top of different systems it is best to have a separate virtual machine for each separate component (ecosystem server, graders and third-party clients such as Eclipse) for testing the integration between them. However for the purpose of development all of the components can reside on single Linux VM (since spoj0, for example, is coupled strongly to a Linux environment in order to be able to run and test all of the components simultaneously the VM must be running Linux).

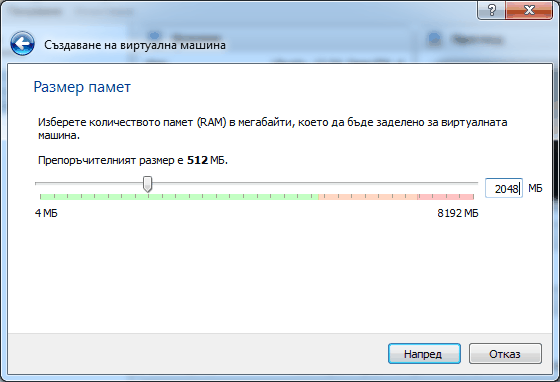
# Development Environment - Setup Guide

## Setting up the VM

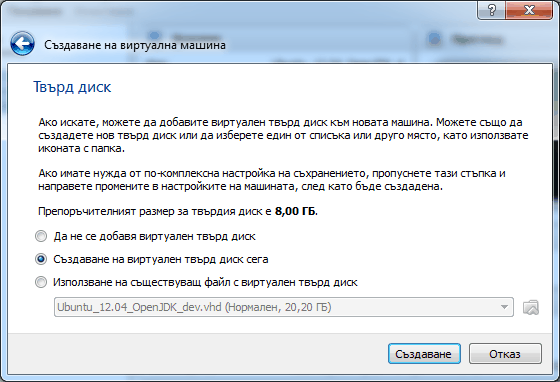
Download and install VirtualBox from [1] – this guide uses VirtualBox 4.2.18. This guide uses Ubuntu 12.04 64bit as the OS for the VirtualBox VM - download Ubuntu from [2]. Power up Virtual Box and click the **New** button. Specify **GradingSystemDev** as the name of the VM, for the OS type select **Linux** and for version – **Ubuntu**.



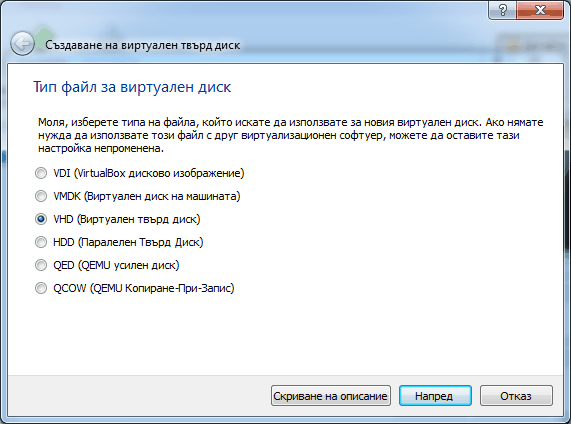
Specify a memory size of 2048 MB.



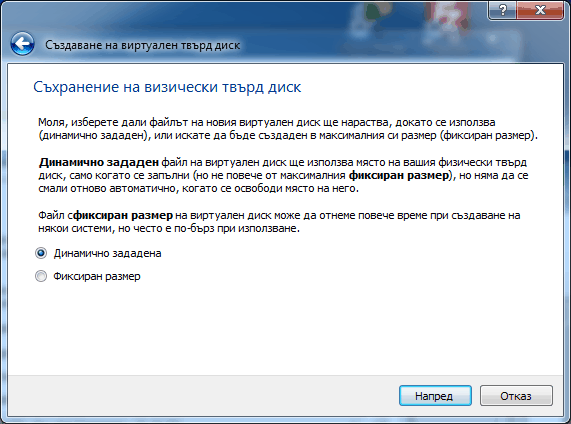
Specify the option for creating a new virtual disk.



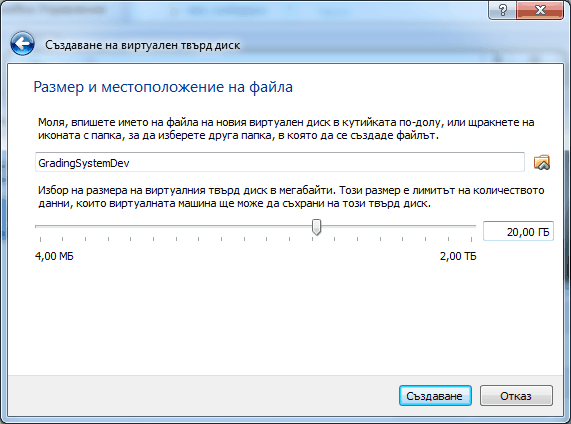
Specify **VHD** as the type of the virtual disk.



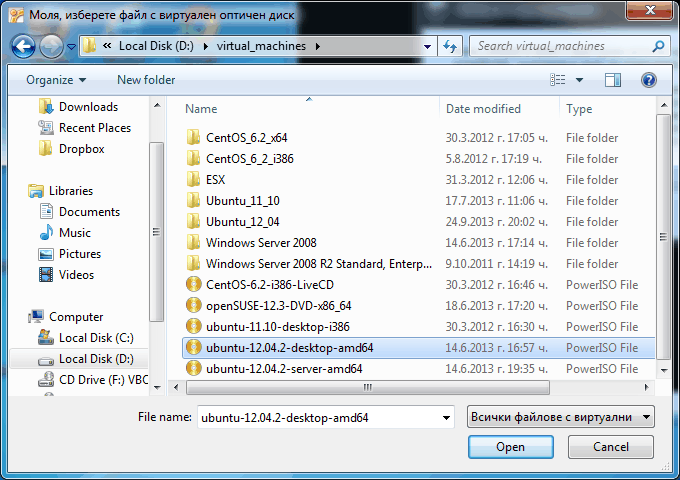
Specify dynamic size allocation for the new disk.

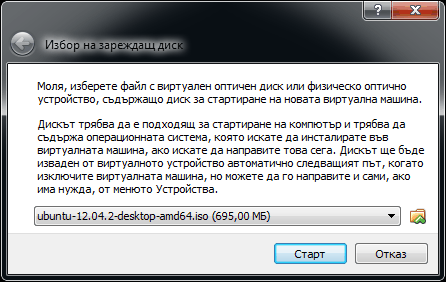


Specify a size of 20 GB for the virtual disk.



Now the virtual machine is created start it. Once started you have the option to specify a boot disk - select the Ubuntu ISO file.





During the Ubuntu installation process leave all settings to their defaults. For the name, computer’s name, username and password specify **system** and select the **Log in automatically** option (this step is optional – if you want you can change this settings at this step or later on). After Ubuntu is installed install the VirtualBox Virtual Add-ons for the VM from Devices -> Install Virtual Add-ons (if necessary mount the Virtual Add-ons iso file first – it is located in the Virtual Box directory) – it is purely optional but gives very handy tools such as copy-paste between the host OS and the VM as well as more screen modes for the VM.

All of the software packages are downloaded and installed from the VM directly. In case a custom installation process is required – the software is installed by convention in the **/opt** directory. Installation files for the applications are stored in **/home/system/Downloads**.

### GIT

GIT is the version control system of the application. To install the GIT issue:

sudo apt-get install git

To verify that GIT is installed type:

git

### JDK

JDK 7 is used to build the majority of the projects in the Grading Ecosystem. OpenJDK 7 is already installed in Ubuntu – there is no need to install Oracle JDK 7 at this time. To install OpenJDK7 (or later) if not already installed issue:

sudo apt-get install openjdk-7-jdk

To verify that the proper version of java is being used issue:

java -version

### Maven

Maven is used as the build system for most of the project in the Grading Ecosystem. To install Maven issue:

sudo apt-get install maven

To verify that Maven is installed issue:

mvn -v

### Eclipse

Eclipse is used as the IDE of choice for developing the various projects. Download and install the latest version of Eclipse Classic from [3] (this guide uses Eclipse 4.3 Kepler). Once downloaded issue the following from the download directory:

tar xzvf eclipse-standard-kepler-R-linux-gtk-x86\_64.tar.gz  
sudo mv eclipse /opt/  
sudo chown –R system:system /opt/eclipse/  
sudo chmod –R 777 /opt/eclipse

In order to be able to add an Eclipse shortcut to the desktop install gnome panel first:

sudo apt-get install --no-install-recommends gnome-panel

Add a desktop icon by issuing (/opt/eclipse/eclipse):

gnome-desktop-item-edit --create-new ~/Desktop

### MySQL

MySQL is the RDBMS used by Spoj0 and Arena Maycamp to store contest data. To install MySQL server (currently MySQL 5) issue:

sudo apt-get install mysql-server

Use the password **system** as the root password.

### MySQL Workbench

MySQL Workbench is a GUI client for use with MySQL server. Download the MySQL Workbench Ubuntu package from [4]. Add a desktop icon by issuing:

gnome-desktop-item-edit --create-new ~/Desktop

(MySQL Workbench executable /usr/bin/mysql-workbench)

### MongoDB

MongoDB is the NoSQL storage engine used by the Grading Ecosystem server instances. Download MongoDB from [5]. To install MongoDB issue:

tar xzvf mongodb-linux-x86\_64-2.4.6.tgz  
sudo mv mongodb-linux-x86\_64-2.4.6 /opt/  
sudo chown -R system:system /opt/mongodb-linux-x86\_64-2.4.6/  
sudo chmod -R 777 /opt/mongodb-linux-x86\_64-2.4.6/

In ~/.bashrc add the following line (in order to make Mongo executables visible on the path):

export PATH=$PATH:/opt/mongodb-linux-x86\_64-2.4.6/bin/

Then issue the following to make the changes visible in the shell:

source ~/.bashrc

Create the directory **mongo\_data** in **/home/system/dev**  as the directory where the test database files will be stored. To test that a MongoDB instance may be started on default port (27017) issue:

mongod --dbpath=/home/system/dev/mongo\_data/

### Curl

Curl is a command-line tool for sending HTTP requests. It can be used to test the various RESTful web services exposed by the components of the system. Install curl by typing:

sudo apt-get install curl

### Ruby on Rails

This is step is performed by the modified Maycamp Arena installer and so it is optional. Ruby on Rails is required in order to run a Maycamp Arena instance. The guide from [11] will be used to install Ruby on Rails. Using the root user (or any other user that has root privileges) install RVM (Ruby Version Manager) first:

cd ~/Downloads  
curl -L get.rvm.io | bash -s stable –auto  
source ~/.bashrc

Add the following the ~/.bashrc:

[[ -s "$HOME/.rvm/scripts/rvm" ]] && source "$HOME/.rvm/scripts/rvm" # Load RVM

And then call:

source ~/.bashrc

Then issue the following in order to retrieve and install the packages required for Ruby:

rvm requirements

Then install and configure Ruby 2.0.0:

rvm install 2.0.0  
rvm use 2.0.0

Then check the version installed:

ruby –v

Assuming this is 2.0.0p247; issue the following to add it as default:

rvm --default use 2.0.0-p247

After that install Rails 4.0 by issuing:

gem install rails -v 4.0.0

Installer the ruby bundler by issuing:

sudo apt-get install ruby-bundler

After that install the latest version of Rake by issuing:

gem install rake

After Rails is installed you should have the following commands available from the command line: ruby, gem, rails and rake.

### RestClient

The WizTools RestClient is a convenient UI client for testing RESTful web services. Download the client from [6]. To install it perform the following:

sudo mkdir /opt/restclient  
sudo mv restclient-ui-3.2.1-jar-with-dependencies.jar /opt/restclient/  
sudo chown -R system:system /opt/restclient/  
sudo chmod -R 777 /opt/restclient/  
gnome-desktop-item-edit --create-new ~/Desktop

In the **Command** field of item dialog specify:

java –jar /opt/restclient/restclient-ui-3.2.1-jar-with-dependencies.jar

## Setting up the Development Environment

The following directory structure is used:

**/home/system/dev/grading\_ecosystem** – Grading Ecosystem repository  
**/home/system/dev/workspace** – Grading Ecosystem Eclipse workspace

To clone the Grading Ecosystem repository type:

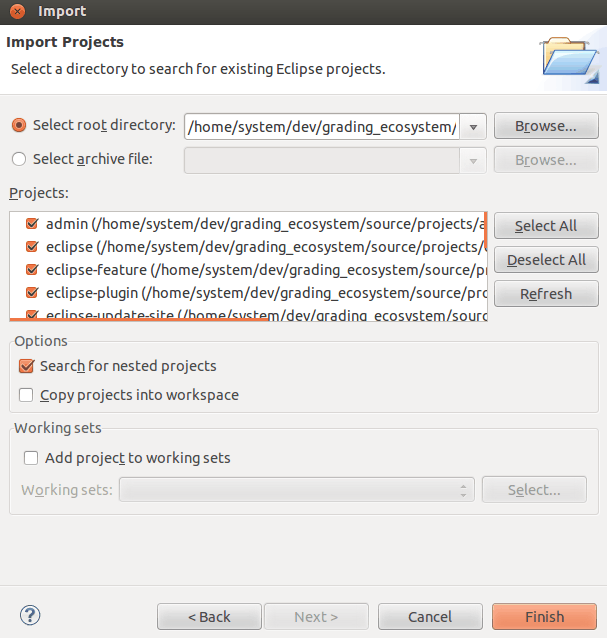
cd /home/system/dev  
git clone <https://github.com/martinfmi/grading_ecosystem>

Next start the Eclipse IDE and select the appropriate workspace (**/home/system/dev/workspace**). We will install additional features and plug-ins that will aid us throughout the development process. Install CDT (C/C++ Development Tools) using the update site specified by [7] from the **Help** -> **Install New Software**. Install the m2e (and related) - Eclipse plug-in for Maven from the the **Help** -> **Install New Software**, selecting **– All Available Sites—**and filtering on the **m2e** keyword (uncheck the **Contact All Update Sites** during Installation option). Install the EPIC Perl IDE for Eclipse (see [8]) using the update site specified by [9]). After that install the Aptana Eclipse Plug-in that comes with Ruby-on-Rails support using the update site specified by [10]. Next download RubyPeople plug-in from [12] – it provides a run configuration for Ruby applications. Unzip the contents of the plug-in archive in the Eclipse installation folder (to the corresponding **plugin** and **features** directories). Next start Eclipse and go to **Window** -> **Preferances** -> **Ruby** -> **Installed Interpreters** -> **Add** (then check that the plug-in is actually installed if the menu is not available) and add a new interpreter with the following settings:

Interpreter Name: ruby  
Location: /home/system/.rvm/rubies/ruby-2.0.0-p247/bin/ruby

*Note: The location of ruby might be different depending on the version of RVM.*

Import all of the projects in the workspace by right clicking in the **Package Explorer** and selecting **Import**. Specify **the /home/system/dev/grading\_ecosystem/source/projects** directory as parent and select the **Search for nested projects** tab. The following projects will typically be displayed:



The list of projects currently includes:

* admin – the administrative web interface client;
* eclipse – top level project for the Eclipse client;
* eclipse-feature – Eclipse feature (holds all Eclipse plug-ins and used by the update-site);
* eclipse-plugin – Eclipse plug-in for the Grading Ecosystem;
* eclipse-update-site – Eclipse update-site for the Grading Ecosystem;
* maycamp\_arena-fork – fork project of Maycamp Arena used for adding integration extensions;
* parent – parent project for all project the Grading Ecosystem;
* spoj0-fork - fork project of Spoj0 used for adding integration extensions;
* installer – custom installer for the Grading Ecosystem subprojects;
* server – Grading Ecosystem server project;
* shared - shared Java utilities (used by both the server and by Java clients);

# Setting up Spoj0

Spoj0 comes with an installer script that we will not be using since it installs the system from the original SVM repo. Instead use the modified installation script **spoj0-fork-installer.sh** from the **Installer** project – it includes additionally installation of the Dolphin web framework and Dancer::Plugin::REST plug-in used to extend spoj0.

cd ~/dev/grading\_ecosystem/source/projects/installer/  
sudo ./spoj0-fork-installer.sh

During the installation specify the following settings:

Enter new UNIX password: system  
Retype new UNIX password: system  
Full Name: spoj0  
Room Number: 1  
Work Phone: 0  
Home Phone: 0  
Other: 0

For all of the required passwords use: **system**.

Once you have made any changes to the spoj0-fork project you can update the spoj0 installation using the **spoj0-fork-update.sh** script (must be run as a root). Note that changes in the deployed spoj0 files are overridden for the files listed in the update script.

If installation is successful spoj0 should be running open typing **http://localhost/spoj0/** in the browser.

Also you should configure **cpan** (command used to download and manage dependencies for Perl from the CPAN - Comprehensive Perl Archive Network).

# Setting up Maycamp Arena

// TODO

# Testing Environment – Setup Guide

// TODO

# 

# References

[1] VirtualBox Downloads  
<https://www.virtualbox.org/wiki/Downloads>

[2] Ubuntu Desktop Edition Downloads  
<http://www.ubuntu.com/download/desktop>

[3] Eclipse Downloads  
<http://www.eclipse.org/downloads/>

[4] MySQL Workbench  
<http://dev.mysql.com/downloads/tools/workbench>

[5] MongoDB downloads  
<http://www.mongodb.org/downloads>

[6] WizTools RestClient Download  
<http://code.google.com/p/rest-client/downloads/list>

[7] Eclipse Kepler CDT update site  
<http://download.eclipse.org/tools/cdt/releases/kepler/>

[8] Epic Perl IDE for Eclipse  
<http://www.epic-ide.org/>

[9] Epic IDE update site  
<http://e-p-i-c.sf.net/updates>

[10] Aptana Eclipse Plug-in (comes with Ruby-On-Rails support)   
<http://download.aptana.com/studio3/plugin/install>

[11] Ubuntu, Ruby, RVM, Rails and You  
<http://ryanbigg.com/2010/12/ubuntu-ruby-rvm-rails-and-you/>

[12] RubyPeople Eclipse plug-in  
<http://sourceforge.net/projects/rubyeclipse/files/rdt/0.8.0/org.rubypeople.rdt-0.8.0.604272100PRD.zip/download>