

Gerstner Laboratory
for Intelligent Decision Making



SyRoTek: V012.2 -
User's manual for SyRoTek
e-learning system

Karel Košnar, Jan Faigl, Martin Saska, Miroslav Kulich, Jan
Chudoba, Petr Štěpán a Libor Přeučil

{kosnar,xfaijl,saska,
kulich,chudoba,stepan,preucil}@labe.felk.cvut.cz

Report No.	<i>GLR 222/10</i>	Date	10/02/2011
------------	-------------------	------	------------

Gerstner Laboratory
Katedra Kybernetiky (K13133)
Faculty of Electrical Engineering
Technická 2, 166 27, Prague 6,
Czech Republic

<http://gerstner.felk.cvut.cz>

System pro robotickou tele-výuku

SyRoTek

V012.2- User's manual for SyRoTek e-learning system

ver. 1.0

10/02/2011



ČVUT v Praze, FEL, Gerstner Laboratory, ProTyS a.s.

Contents

1	Introduction	9
2	Installation of the required software	10
2.1	The Player Robot Device Interface adapted for SyRoTek requirements . . .	10
2.2	NetBeans with the SyRoTek Plugin	10
2.3	Subversion client software	15
2.4	Recapitulation	16
3	SyRoTek Plugin NetBeans description	17
3.1	Run in Simulator	17
3.2	Show Simulator/Arena in NetBeans	17
3.3	Run in Arena	18
3.4	Distant run in Arena	19
3.5	Log in the SyRoTek System	22
3.6	My SyRoTek page	23
3.7	My reservations	23
3.8	Make a reservation	24
3.9	Evaluate task	24
4	Subversion	26
5	Reservation of the system	28
5.1	How to create a new reservation in SyRoTek	28
5.1.1	Initial setting of your reservation	28
5.2	How to view and delete your reservations	31

List of Figures

2.1	SyRoTek Plugins installation.	12
2.2	SyRoTek login form.	12
2.3	Form 1 with details specifying new project to open in NetBeans.	13
2.4	Form 2 with details specifying new project to open in NetBeans.	14
2.5	How to set project as the main project.	15
3.1	NetBeans GUI with denoted buttons added by the SyRoTek plugin.	18
3.2	Two possibilities for the Run in Simulator action.	18
3.3	External window with Player Stage simulator.	19
3.4	Output window of the Run in Simulator mode.	20
3.5	Simulator displayed in the NetBeans GUI.	21
3.6	An augmented scene from the real arena with highlighted menu for setting the displayed picture.	22
3.7	A real picture from a camera observing the arena, which is augmented by a model of obstacles and simulated range of robots' sensors.	23
3.8	Two options of the Evaluate task action.	24
3.9	Warning displayed if your codes are not a working copy of SVN.	25
3.10	List of files in your project folder that are not in SVN.	25
4.1	A Subversion plugin window for adding files located in your working folder into SVN.	27
5.1	Required date and duration of the reservation.	29
5.2	A reservation form enabling to reserve the system for solving a task with 6 robots in duration of 90 minutes. A task with 6 robots is already reserved at time 10:00-11:30, 2009-12-25. The total amount of the robots available in the system is 10.	30

5.3	A reservation form enabling to reserve the system for solving a task with 2 robots in duration of 90 minutes. A task with 6 robots is already reserved at time 10:00-11:30, 2009-12-25. The total amount of the robots available in the system is 10.	31
5.4	A reservation form enabling to reserve the system for solving a task with 6 robots in duration of 30 minutes. A task with 6 robots is already reserve at time 10:00-11:30, 2009-12-25. The total amount of the robots available in the system is 10.	32
5.5	Confirm the selected data.	33
5.6	Reservation was successfully saved into the database.	33

Chapter 1

Introduction

This project report is framed as a user manual, which will be utilized as a guideline for new users of the SyRoTek system. Based on this report, a manual in a pdf file will be created and placed for download at the project's web pages. This document will be updated whenever it will be necessary due to later changes of the SyRoTek system. Beyond this, a web manual will be created from this report and attached to the web pages of SyRoTek.

In this report, the process, from enrolment of the first course and the first task therein to successful submission of the task for evaluation, will be described step by step. Readers can find all necessary information on installation of proper development environments, on downloading special software tools and project templates, on reservation of the arena, on system remote control and on utilization of all SyRoTek tools.

Chapter 2

Installation of the required software

First of all, we should highlight that the SyRoTek system is actually designed for Linux users only. The second important restriction, which needs to be mentioned, lies in supported development environments. The access to the SyRoTek robotic platform can be easily realized through NetBeans 6.9 with a special SyRoTek Plugin and the robotic platform can be controlled only using the Player hardware abstraction layer supporting the robot device interface.

2.1 The Player Robot Device Interface adapted for SyRoTek requirements

As highlighted above, the only tool for controlling the remote multi-robot platform SyRoTek is the Player interface. Here, we strongly recommend you to use the Player version available at the project web-sites <http://syrotek.felk.cvut.cz/download/>, which includes all functionalities necessary for utilizing of the SyRoTek system. Installation of this version is the same as the installation of standard Player environment and you can follow instructions at <http://playerstage.sourceforge.net/doc/Player-svn/player/install.html> for it.

2.2 NetBeans with the SyRoTek Plugin

As mentioned above the only development environment supported by developers of the SyRoTek system is the NetBeans 6.9 application platform complemented by a special SyRoTek Plugin. Let us now describe how to optimally download the required software and to set up your system to maximize effect of functionalities provided by SyRoTek.

Step 1.

In the first step you can download the required version 6.9 of NetBeans from the project web-page <http://netbeans.org/>. The installation is relatively straightforward and you can simply follow instructions mentioned there.

Step 2.

Once the NetBeans are installed, you will need to download the SyRoTek Plugins from the SyRoTek web-pages <http://syrotek.felk.cvut.cz/download/>.

Step 3.

Downloaded Plugins can be added to NetBeans using the option **Tools**→**Plugins** from the NetBeans menu. In the opened window, select **Downloaded** and press the **Add Plugins** button. Select all .nbm files from the folder, where the downloaded SyRoTek Plugins are saved, as depicted in Fig. 2.1. Confirm the selection by OK button and the necessary Plugins should be added. Once the Plugins are added and NetBeans restarted, a set of new icons in the NetBeans tool bar should be displayed. You can find a description of functionalities of these buttons in Section 3 of this guideline.

Step 4.

Beside the operating buttons, you can find a login dialog appeared in the menu. You should be logged in the SyRoTek system whenever you need to start solving a new task, to test your algorithms using the SyRoTek arena or to submit a final solution for corrections. Before your first login, you need to be registered using the form at the SyRoTek main page. If you are attending a regular course at CTU, you may be registered automatically based on CTU databases of students. Your teacher will inform you about details.

Once, you are registered, simply press the login button and fulfill your Username and Password in the displayed form (see Fig 2.2). If your login was processed successfully, your username will be displayed at the NetBeans status bar. You are strongly recommended to login always when you are implementing a SyRoTek task. Nevertheless, if you are working off-line, you will be asked for login, whenever you call for services of the SyRoTek server.

Step 5.

The last step that needs to be done, ahead of implementation of your task, is to open a new NetBeans project. Here, we should distinguish between two types of users. The most commonly, you are going to solve a task as a part of a course prepared by a teacher. In such a case, it is recommended to use a template of the project, which is attached to the

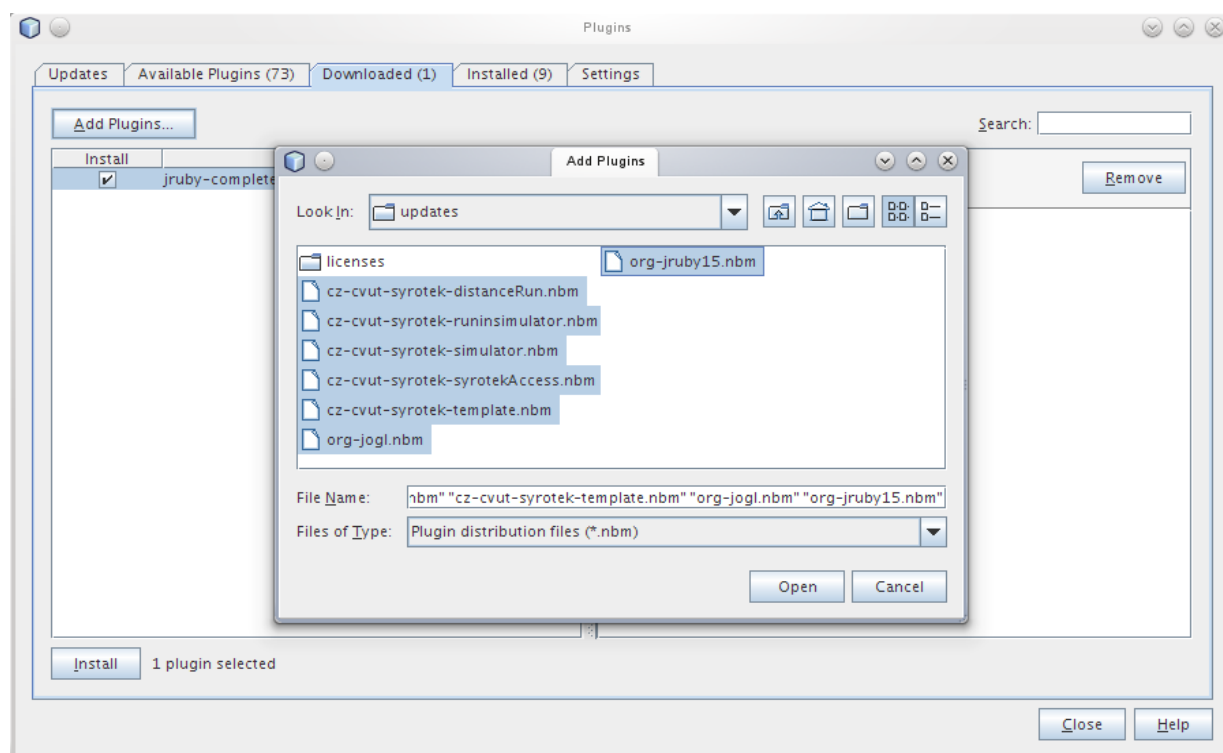


Figure 2.1: SyRoTek Plugins installation.

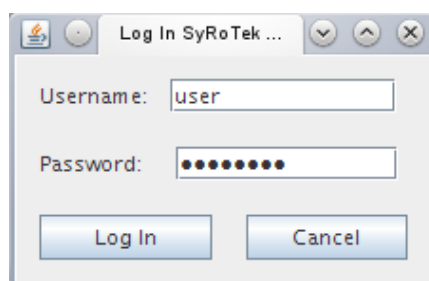


Figure 2.2: SyRoTek login form.

particular task by the teacher. If you are interesting to solve an individual task, you must ask administrators of the SyRoTek system for upgrading your status in a level enabling to create new tasks. In this case, you have to attach an appropriate template to your task from a list of available templates. In both schemes, the process of opening the new NetBeans project is identical.

In the NetBeans menu, press **File**→**New Project**. In the displayed window, select category C/C++, project SyRoTek template, as depicted in Fig. 2.3, and press next. In the second form (Fig. 2.4), specify project name, location and folder in which your source

codes have to be stored. In the **Course** and **Task** drop down lists, select a course you are attending and a task you are interesting to solve and press **Download Template**. Afterwards press the button **Finish** to finish the action.

If the required course is not available in the drop down list, you should consult project web-sites for general rules on attending SyRoTek courses or contact your teacher if you are sure that you are subscribed to the course correctly. If the required task is not available in the drop down list, you should check all dependencies between the tasks of the particular course. If you are sure, that you have accomplished all tasks demanded for solving the required task and that the time schedule of the course enables the solving of the required task, contact your teacher or system administrators.

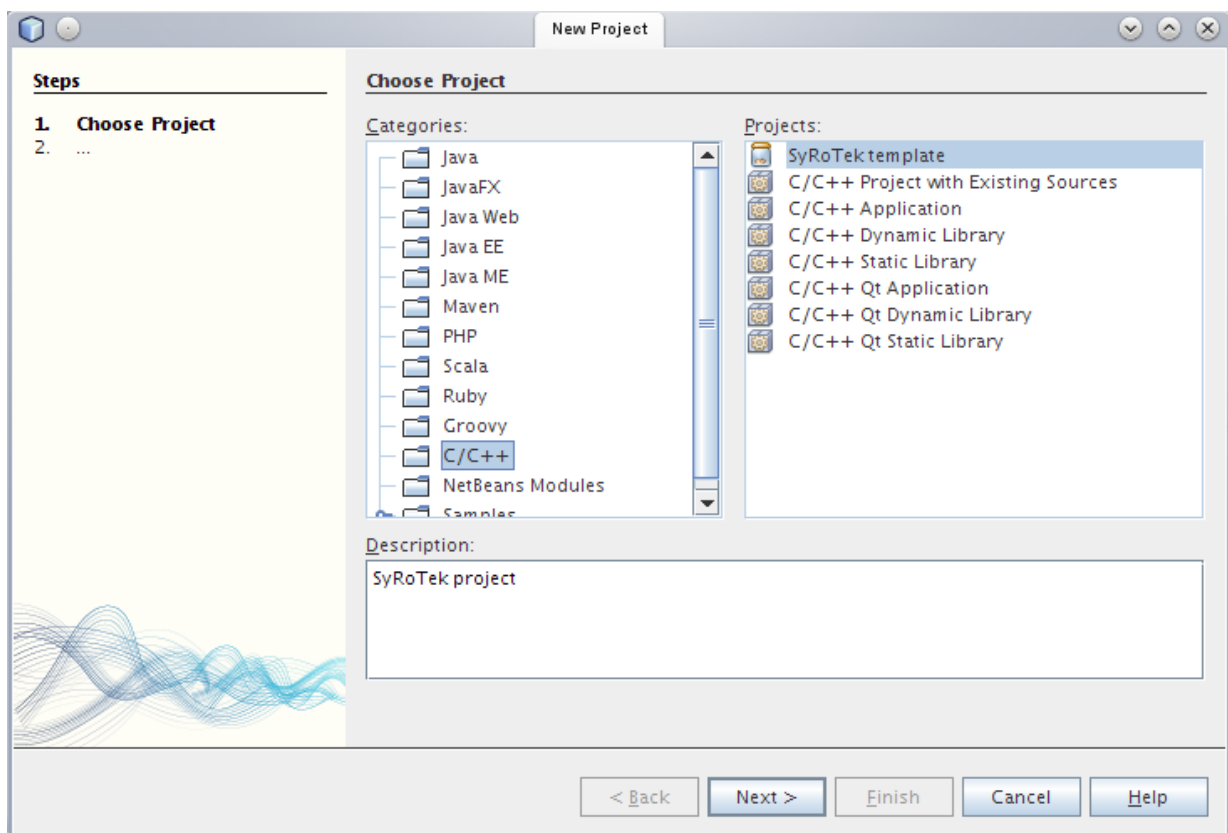


Figure 2.3: Form 1 with details specifying new project to open in NetBeans.

We strongly suggest to use only the above described approach for creating a new NetBeans project for solving SyRoTek tasks. Using this procedure, the creating of new project of NetBeans at your local desktop is not the only action performed. Confirming the form depicted in Fig. 2.4, a repository with copy of template files is automatically created in the SyRoTek control PC. In a consequence step, these files are automatically checked out to a folder in your PC. Name and path of this folder was specified in the form depicted in

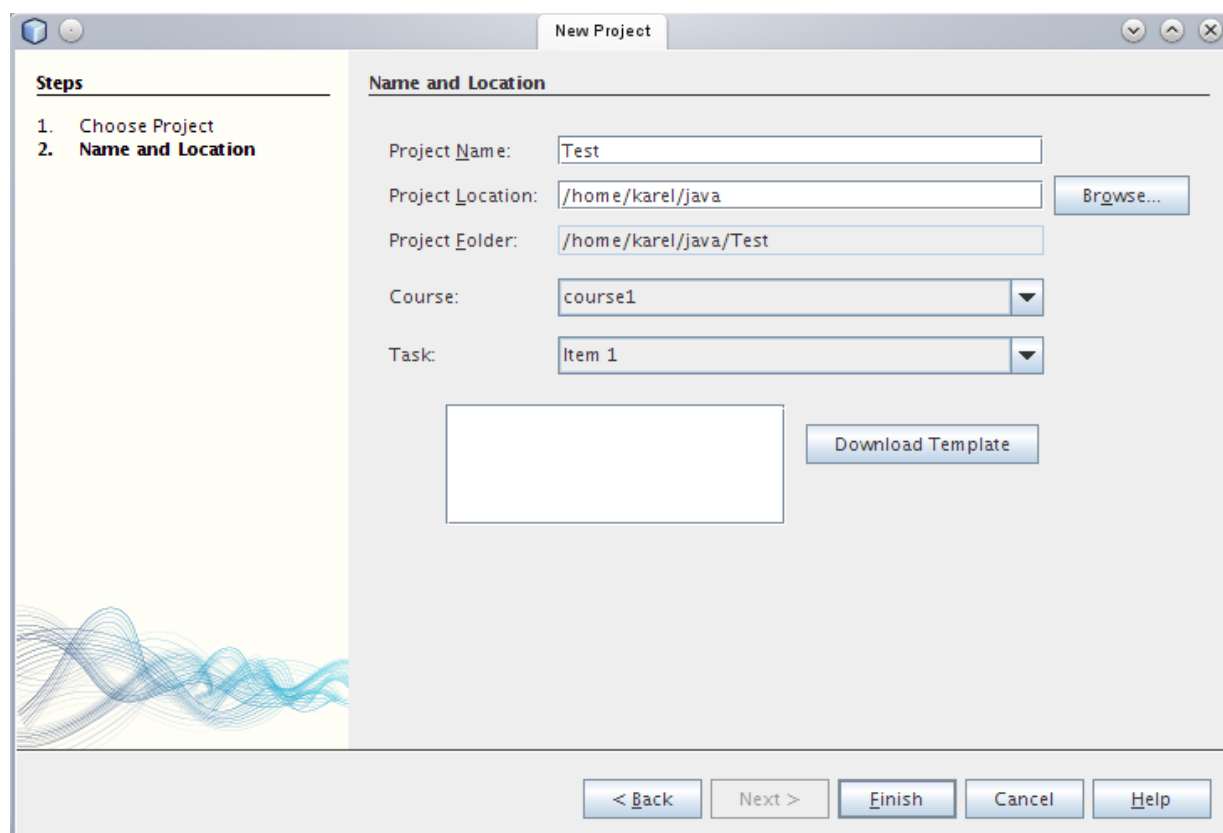


Figure 2.4: Form 2 with details specifying new project to open in NetBeans.

Fig. 2.4. This action, initialized by the SVN checkout command, creates a working copy there. You can edit files of this copy and commit them via SVN whenever you need to backup your work. If you create new files, which are going to be an inseparable part of your application, you must add them to SVN, otherwise they are not backed up and your codes sent for a final evaluation will be incomplete. Details on adding new files to SVN can be found in Section 4 or at <http://netbeans.org/kb/docs/ide/subversion.html>.

Step 6.

The new buttons, included by the SyRoTek plugins, put their effect only on the Main project opened in the NetBeans development environment. Therefore you should check if the project, you are going to submit to the SyRoTek control PC, is the main project. If not, you can set the project as the main project with right mouse button applied on the name of the project in the Projects window. See Fig. 2.5.

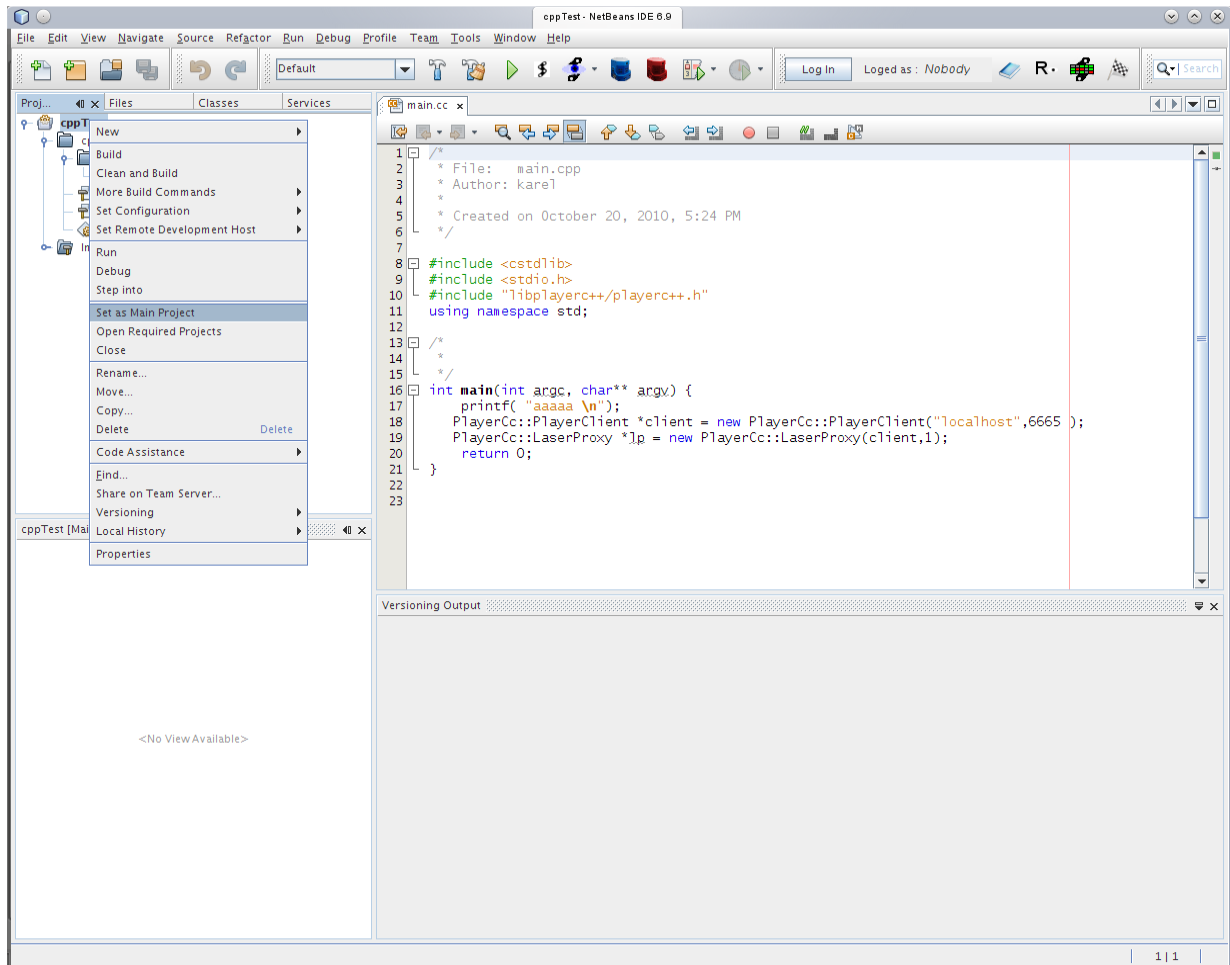


Figure 2.5: How to set project as the main project.

2.3 Subversion client software

Different versions of users' codes as well as descriptions of tasks and courses are maintained by Subversion (SVN) in the SyRoTek project. We recommend to use SVN for storing subsequently arising versions of your code throughout development of your application. This approach enables an easy backtracking if a modification of your code would decrease performance of your application. If you prefer another type of backup, you will still need the SVN for submitting your code to the SyRoTek system for the evaluation. This enables to resubmit your work several times, while the teacher can keep track of all versions and can easily identify, which version has been already tested etc.

For your convenience, we suggest to employ a NetBeans IDE's Subversion plugin mainly if you are not familiar with SVN. NetBeans IDE's Subversion support works by interacting with the Subversion client to carry out versioning commands. Before you can take

advantage of the IDE's Subversion support, you need to have Subversion client software installed on your computer. You can download the software from <http://www.collab.net/downloads/netbeans/> and for the installation simply follow instructions presented there.

2.4 Recapitulation

Make sure you have the following software installed on your computer, before you start solving SyRoTek tasks:

- NetBeans IDE 6.9 (download: <http://netbeans.org>)
- Java SE Development Kit (download: <http://www.oracle.com/technetwork/java/javase/downloads/index.html>)
- Subversion client software (download: <http://www.collab.net/downloads/netbeans>)
- SyRoTek plugins for NetBeans (download: <http://syrotek.felk.cvut.cz/download/>)
- Player for SyRoTek (download: <http://syrotek.felk.cvut.cz/download/>)

Chapter 3

SyRoTek Plugin NetBeans description

Let us now describe the new buttons in the menu of NetBeans GUI, which were added by the SyRoTek Plugin. All these buttons serve for using SyRoTek special functionalities with actions applied on the actual main project.

3.1 Run in Simulator

The **Run in Simulator** button (button denoted as 1 in Fig. 3.1) runs your application in the Player with plugin Stage, adapted for requirements of SyRoTek project. Once you press the button, you will be asked to choose between two options: restart the server and connect to running server (Fig. 3.2). Select **Restart the server**, if you would like to restart your simulation from the initial conditions. Select **Connect to the running server**, if you would like to continue your simulation from the last state. After the selection, the simulator will be opened in a separate window (see Fig. 3.3 as an example). You can watch the outputs from the simulator directly in the NetBeans output window as shown in Fig. 3.4.

3.2 Show Simulator/Arena in NetBeans

If you prefer to display the screen of the simulator directly in a NetBeans window, you can press the button 2 in Fig. 3.1. A snapshot with such a view is depicted in Fig 3.5. Using this button, you can select layers of the image that have to be displayed, similarly as in the Player Stage. The list of different possibilities is shown in Fig. 3.6, where the same button can be used for displaying the augmented scene from the real arena.

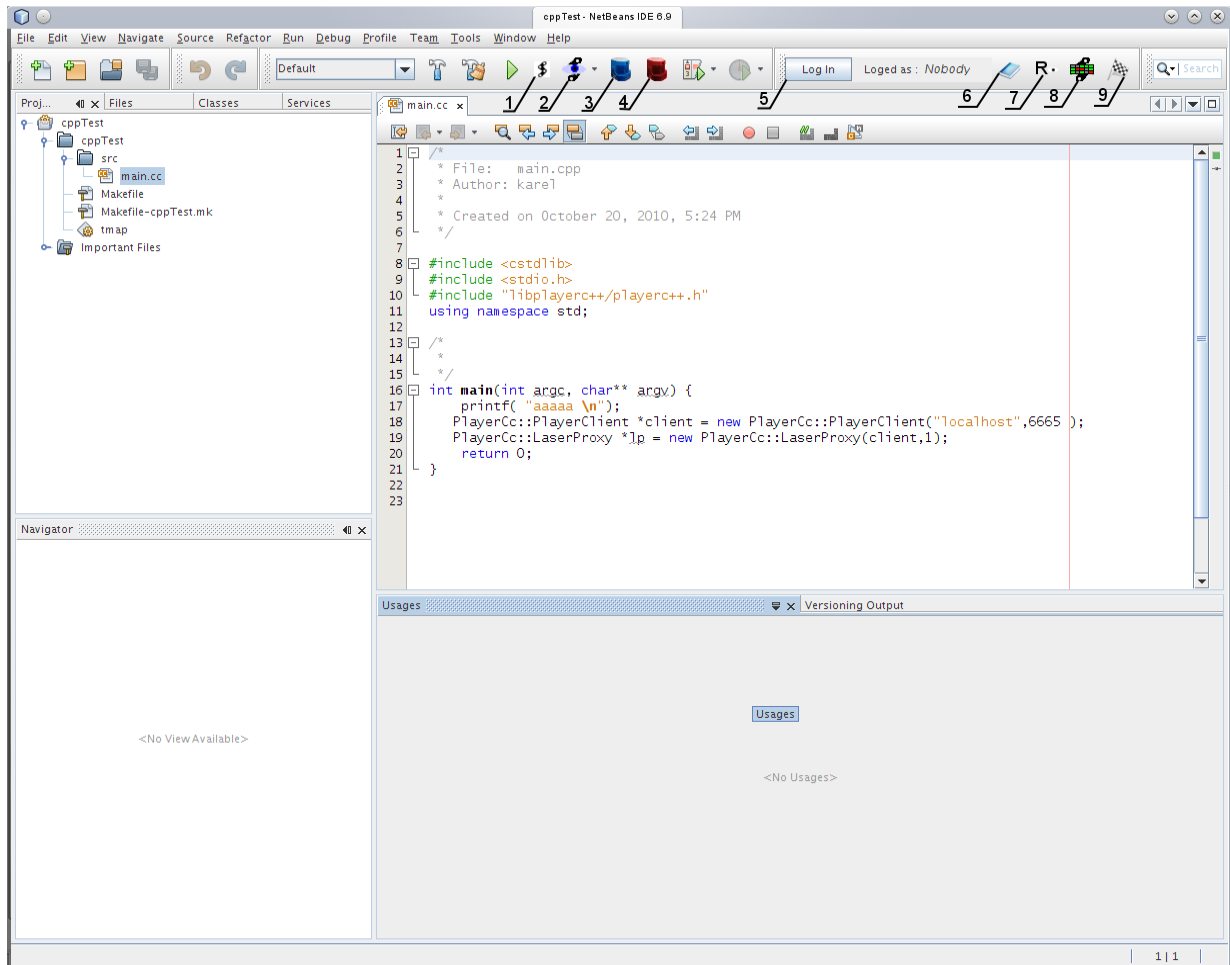


Figure 3.1: NetBeans GUI with denoted buttons added by the SyRoTek plugin.

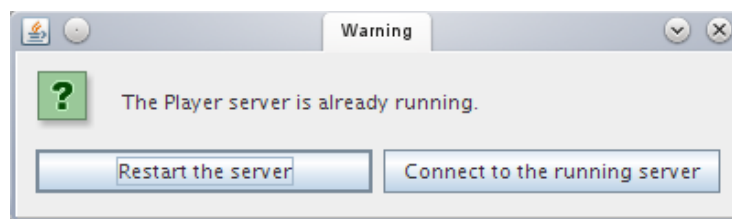


Figure 3.2: Two possibilities for the **Run in Simulator** action.

3.3 Run in Arena

The button with symbol of the blue SyRoTek robot (button denoted as 3 in Fig. 3.1) runs a remote server on the control PC of the SyRoTek arena. Your application is run locally

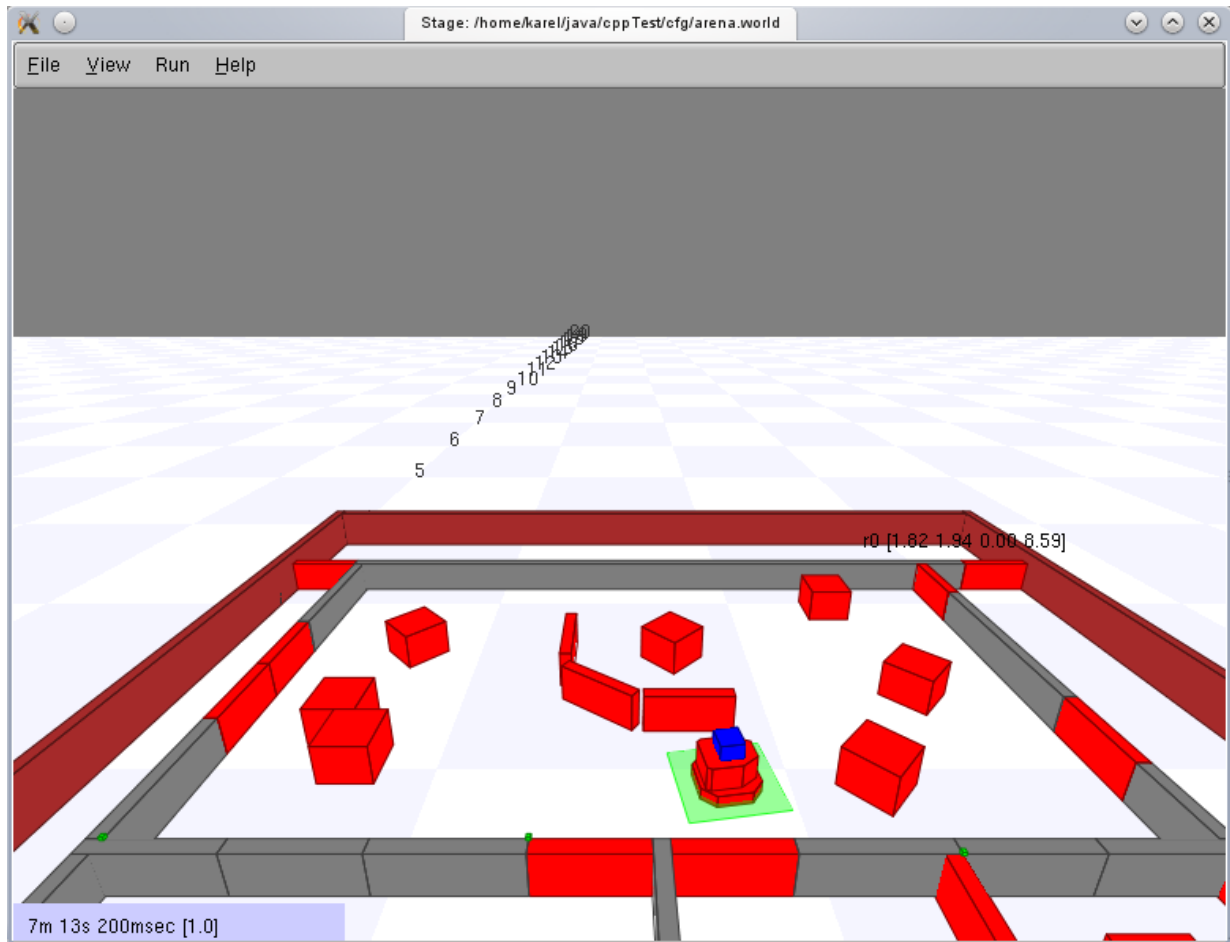
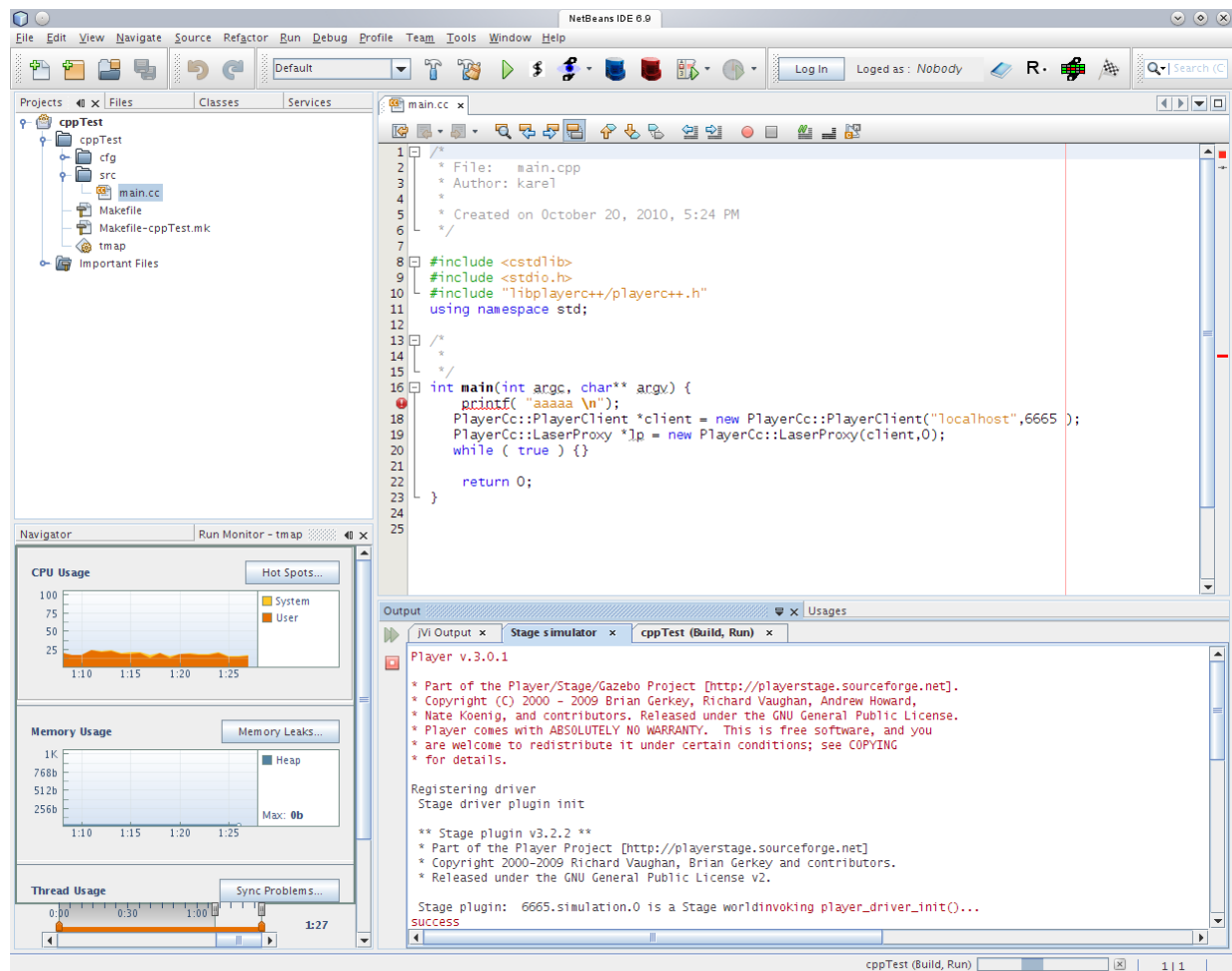


Figure 3.3: External window with Player Stage simulator.

on your desktop computer, which is connected to the server. You can read the outputs of your program as well as of the server in particular output NetBeans windows. Similarly as with the simulator, you can watch the actual state of the arena in another output window using the button 2. An example combining a real view of a camera observing the arena with a model of obstacles and simulated range of robots' sensors is depicted in Fig 3.7.

3.4 Distant run in Arena

The button with symbol of the red SyRoTek robot (button denoted as 4 in Fig. 3.1) also runs a remote server on the control PC of the SyRoTek arena. The difference is that your application is run on the control PC of the SyRoTek arena too. Similarly as with the blue button, you can read the outputs of your program as well as of the control PC in particular output NetBeans windows and you can watch the actual state of the arena in

Figure 3.4: Output window of the **Run in Simulator** mode.

another output window using the button 2.

Let us now highlight differences in utilization of the local and distant run of your application.

- Probably the most significant advantage of running your application locally on your desktop, is the easier access to your log files. If your application is running on the SyRoTek control PC, also your log files are stored there and you must copy the files always when you need to use them for debugging.
- The local approach is not recommended whenever you need to process big amount of sensory data in your application. The sensory data are transferred through internet, which may cause unacceptable delays.
- A delay caused by sending commands from your application to the robots and sensory

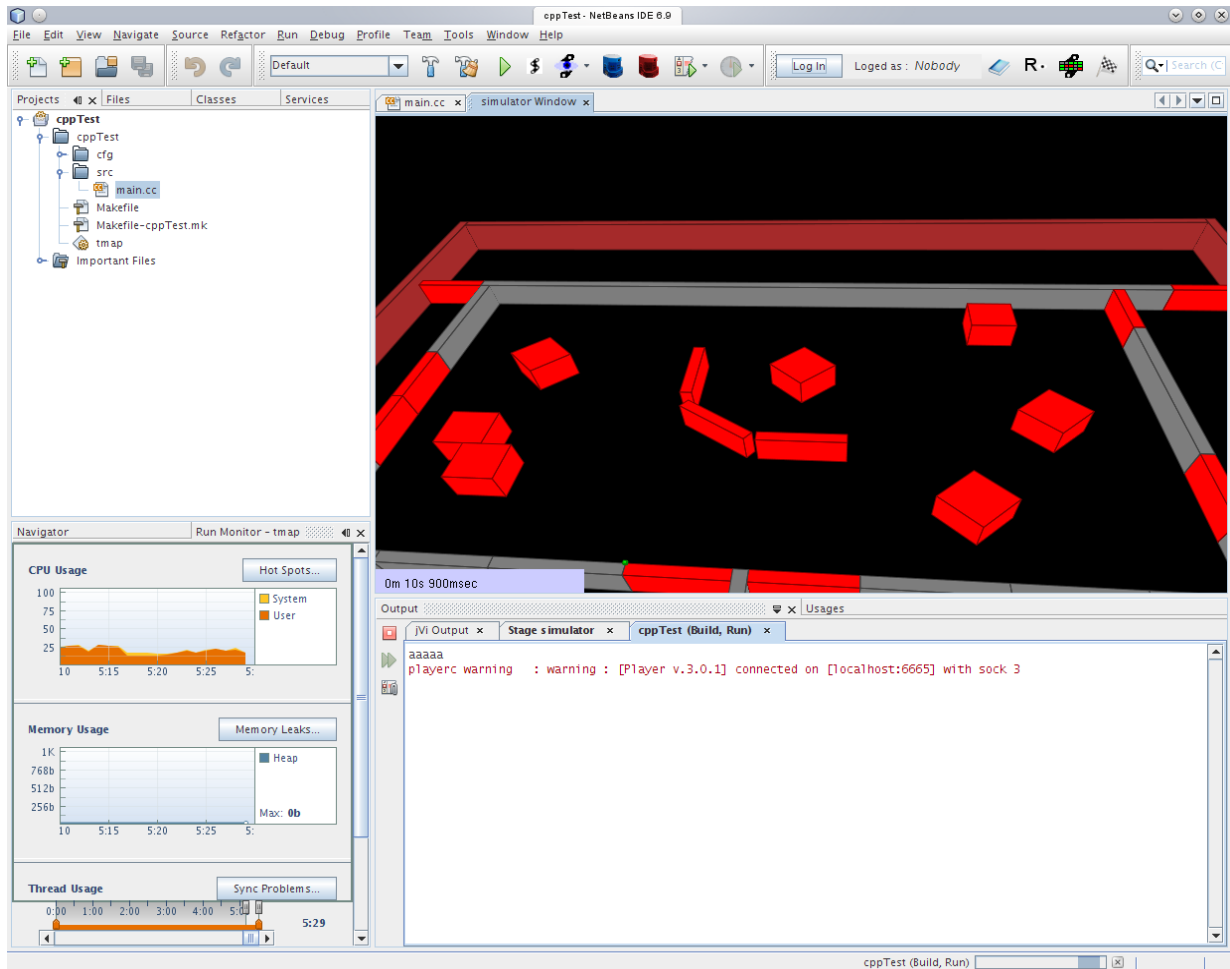


Figure 3.5: Simulator displayed in the NetBeans GUI.

output from the robotic system back to your computer is unavoidable, if you are using the locally running application. In some application this transfer delay can cause unwanted behavior. Always take this into account during implementation, if you plan to control the system remotely from your locally running application.

- If you choose the distant run in the arena, your code needs to be sent to the control PC whenever it is changed. This can be annoying mainly during the debugging of your application and if size of your code is big.
- Finally, if you have access to big computational resources and/or the SyRoTek control PC is overloaded, you should run your code locally. But usually, the computationally resources available on the SyRoTek control PC markedly surpasses computational resources of common desktop PCs.

To summarize these points, we recommend to test your code in simulator first (the

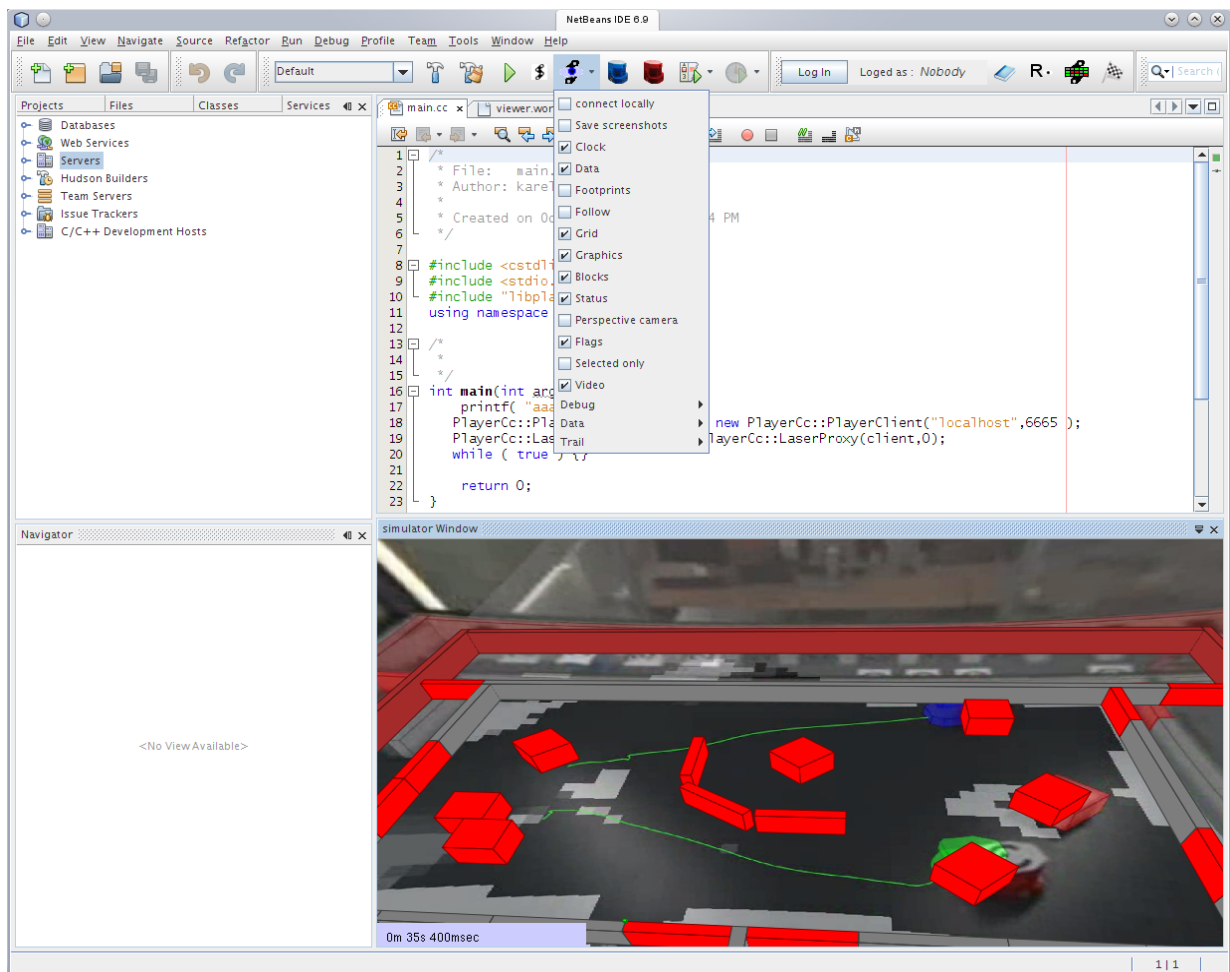


Figure 3.6: An augmented scene from the real arena with highlighted menu for setting the displayed picture.

button 2). If it works sufficiently, you should debug it using the **Run in Arena** approach with your algorithm launched locally (the button 3). And the final testing, before the submitting for evaluation, should be done using the **Distant run in Arena** (the button 4). If your internet connection is too slow or you require a big amount of sensory data to be processed in your application, we recommend to skip the second point and to use the **Distant run in Arena**, once your application has been verified on the simulator.

3.5 Log in the SyRoTek System

How to use the button 5 in Fig. 3.1 for log into the system has been already described above.

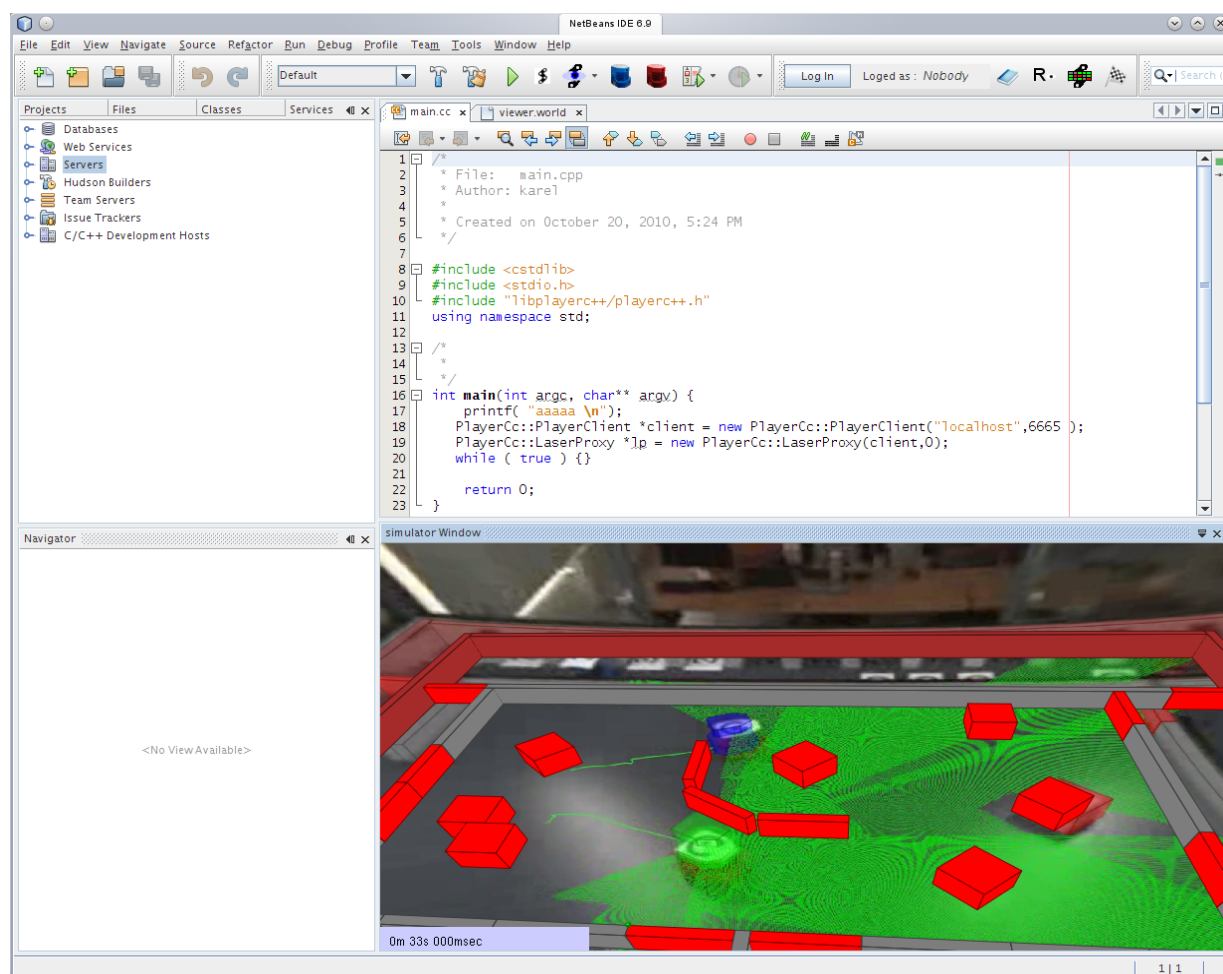


Figure 3.7: A real picture from a camera observing the arena, which is augmented by a model of obstacles and simulated range of robots' sensors.

3.6 My SyRoTek page

Using the button 6 in Fig. 3.1, you can open a new window in your internet browser to subscribe new courses and tasks as also mentioned above.

3.7 My reservations

The button 7 in Fig. 3.1 displays a set of all your reservations of the SyRoTek arena in a separate window of your internet browser. You can also delete your reservations with this functionality as described in Section 5.2.

3.8 Make a reservation

You can make your new reservation of the SyRoTek arena with button 8 in Fig. 3.1. A guideline of the reservation process is presented in Section 5.1.

3.9 Evaluate task

Once you are satisfied with your application and SyRoTek robots behave in the arena as required by the task specification, you can submit your codes for automatic evaluation with button 8 in Fig. 3.1. You have two options of the **Evaluate task** action, **Test evaluation** and **Final evaluation**. If you choose the **Test evaluation** (in the form depicted in Fig. 3.8), your codes will be submitted to the system's repository and automatically evaluated as soon as the arena is available. We strongly recommend to try the test evaluation before the final one, since it is hard to objectively decide if the performance of your application meets the task requirements. Once, you pass through the **Test evaluation**, you can select the **Final evaluation** and your teacher will be informed that your work is ready for evaluation. Usually, you can submit as many versions of your application as you want before the task deadline and the most actual one will be selected for the evaluation. Such an approach makes possible to deliver your homework in advance and to enable corrections if your application is not acceptable by the teacher or you would like to get a better mark. On the other hand, you can replace your codes with an improved version if you a mistake is found later on. Nevertheless, these rules can be modified individually by each teacher to optimize the evolution process.

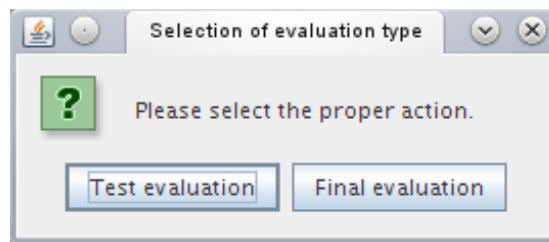


Figure 3.8: Two options of the **Evaluate task** action.

Two kinds of warning messages can be displayed if you select one of the evaluations depicted in Fig. 3.8. Both warnings are related to the Subversion system (SVN), which is used as a tool for submitting of your codes to the SyRoTek system. The advantage of the SVN lies in the possibility to digestedly maintain different versions of your work consequently submitted for evaluation. The first warning, shown in Fig. 3.9, is displayed if your project folder is not maintained with SVN and therefore your codes cannot be sent to the SyRoTek repository for evaluation. This may happen, for example, if you moved your source files to another folder. To enable submitting, you need to assign the

status of working copy to your source codes using the command checkout or with the NetBeans IDE's Subversion plugin (see the guidelines at <http://netbeans.org/kb/docs/ide/subversion.html> for details).

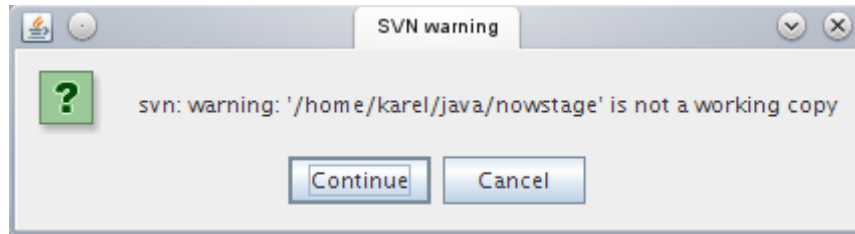


Figure 3.9: Warning displayed if your codes are not a working copy of SVN.

The second warning, depicted in Fig. 3.10, will be displayed if a file in your working folder is not in SVN. It may happen that you have created a new source file during development of your application, which is not added to SVN. Check all file names displayed in Fig. 3.10 carefully before the final submission. If a file from the list is inseparable part of your application and it should be added to the repository on the SyRoTek server, you can use the add SVN command or the NetBeans IDE's Subversion plugin as described in Section 4 or in the guidelines at <http://netbeans.org/kb/docs/ide/subversion.html>.

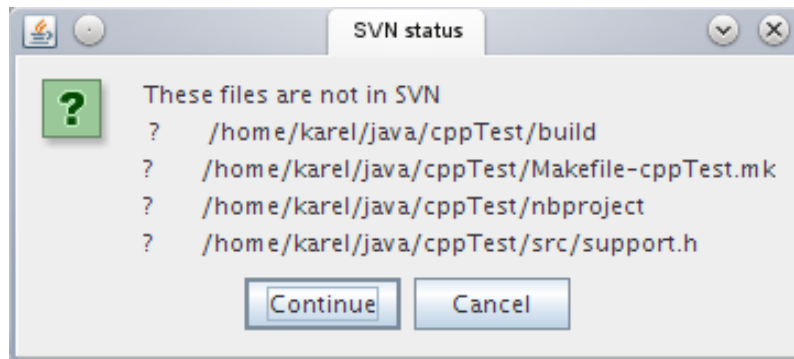


Figure 3.10: List of files in your project folder that are not in SVN.

Chapter 4

Subversion

As mentioned above, the Subversion (SVN) is utilized throughout the complete SyRoTek system. Namely, different versions of descriptions of courses and tasks are kept under the SVN to enable automatic verification even in the case that a student did not get an actual version of setting of the task. The SyRoTek system simply remembers the version of the task's description, which was given to the student, and the same version is used for the evaluation afterwards. This overall process is maintained automatically and students do not have to take care about.

The second important utilization of SVN, where an intervention from the student side is necessary, is the development of the student's code and the consequent submitting to the control PC. Together with opening a new Netbeans project for solving a particular SyRoTek task, a repository with a copy of template files of the task is automatically created in the SyRoTek control PC. In a consequence step, these files are automatically checked out to a folder in your PC. As mentioned above, you can specify the name and path of this folder in the form depicted in Fig. 2.4. The check out action, initialized by the SVN checkout command, creates a working copy in this file. You can edit files of this copy and commit them via SVN to the control PC, whenever you need to backup your work. If you create new files, which are going to be an inseparable part of your application, you must add them to SVN, otherwise they are not backed up and your codes sent for a final evaluation will be uncomplete.

To summarize this, you have two options in using SVN with SyRoTek. You can take fully the advantage of the powerful tool for versioning and backing up your work, which could be beneficial also in your future jobs. In such a case you can use the NetBeans IDE's Subversion plugin with a guideline published at <http://netbeans.org/kb/docs/ide/subversion.html> or any other available distribution of SVN you prefer. The second possibility is to use the SVN as a passive user only for the evaluation of your work. In such a case, we recommend to keep the location of your working folder and all SVN auxiliary files there unchanged throughout your work and the only intervention regarding the SVN that needs to be done is to add all new files of your code under SVN. You can do this using

the **Subversion**→**Commit** button in the Netbeans menu (of course, you have to install the NetBeans IDE's Subversion plugin, as described in Section 2, first). This action opens the form depicted in Fig. 4.1 with list of all files located in your working folder. Check all these files and select those files that have to be included with your code for final evaluation. Press the **Commit** button below the form and the files will be added to SVN. You can send your work for evaluation now.

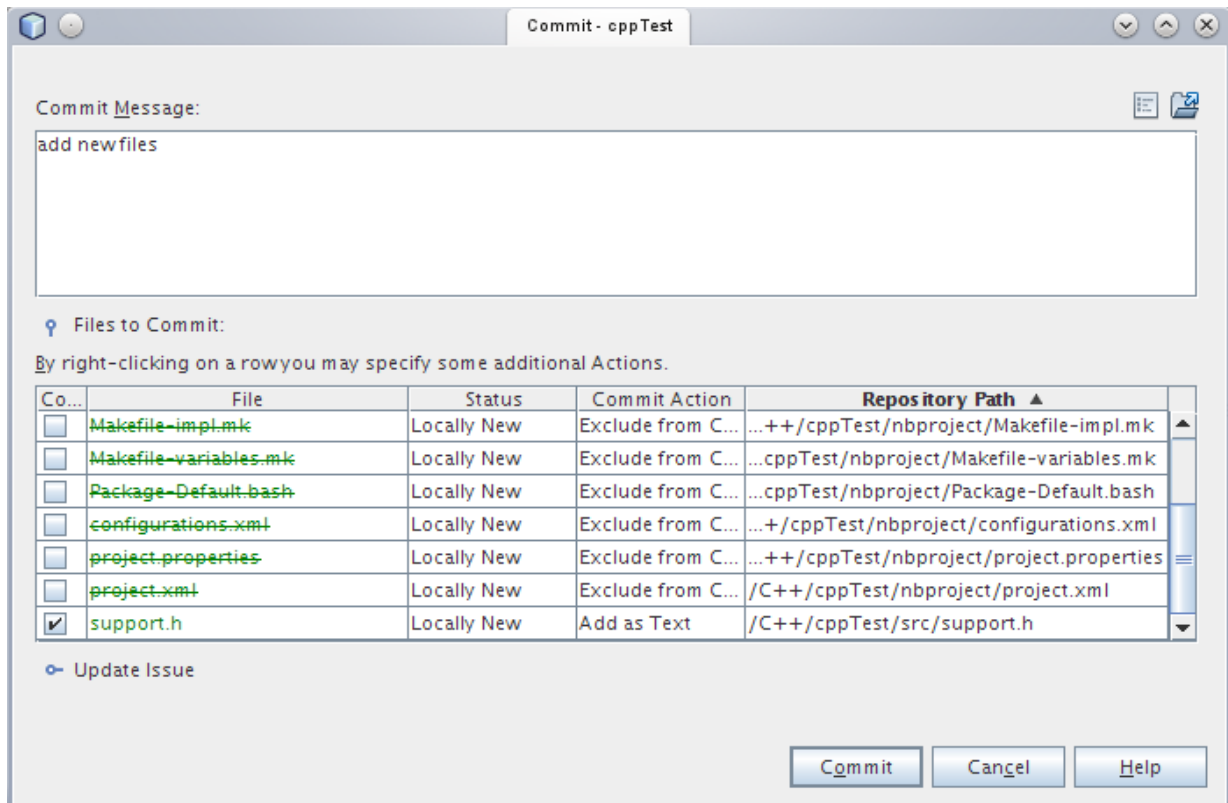


Figure 4.1: A Subversion plugin window for adding files located in your working folder into SVN.

Chapter 5

Reservation of the system for testing your codes

The following guidelines should help you with your first steps in using the SyRoTek reservation system. By reading these, you should be able to make your new reservation of robots and arena for solving various tasks offering by this autonomous multi-robot system. Besides, we will show you how to change and delete such a reservation and how to view all your actual reservations.

5.1 How to create a new reservation in SyRoTek

The reservation system takes the advantage of the architecture of SyRoTek environment based on users' tasks. As users are allowed to solve exclusively tasks defined by teachers within a course, also the reservation must be assigned to a particular task. Nevertheless, such a restriction helps users to specify their requirements on robotic platform easily from the beginning. The required platforms and equipment for each task are defined by the teacher. Using this approach, users can reserve the system for solving all unique tasks already at the beginning of the course, when the knowledge of users on the robotic tasks is limited. More experience users requiring to solve individually defined robotic experiments are required to ask administrators for a user status enabling to create new tasks.

5.1.1 Initial setting of your reservation

Let us describe how to make your first reservation of the SyRoTek system for testing your application now.

Step 1.

The first step of the reservation process is very easy. You should just click on the reservation button in the head of the NetBeans window (the button 8 in Fig. 3.1). As written above, the system itself recognize requirements given by the task: number of robots, necessary sensory equipment, required part of arena etc. This data are automatically gathered from the description of the task, which was selected during your NetBeans Project opening.

Rezervace systému SyRoTek pro řešení úlohy DPS1

Popis úlohy:

- **Celé jméno:** Detekce překážek dálkoměrnými senzory
- **Krátký popis:** Cílem úlohy je (autonomně) přemístit robot na určené místo. Toto místo je dané relativními souřadnicemi vzhledem k počáteční poloze robotu. Oproti předchozí úloze může být do trajektorie robotu umístěna překážka. Robot ji má detekovat a ukončit pohyb dostatečně včas, zamezil kolizi.
- **Motivace:** Seznámení se s použitím dálkoměrných senzorů k prevenci kolizi.

Požadovaný den rezervace: 11. ▾ 2. ▾ 2011 ▾

Požadovaná délka rezervace: 30 minut ▾

Figure 5.1: Required date and duration of the reservation.

Step 2.

Select a date when you would like to solve the task and a desired duration of the task in the form depicted in Fig. 5.1. After this, the system displays a calendar with slots of different colors.

The dark green color denotes beginnings of free time intervals (see Fig. 5.2). Besides, you can find light green slots, which denotes slots with insufficient number of robots available in the system. If you are interesting in reservation of the arena at this time, you can reserve only a task with a lower requirement on hardware. For example a task with lower amount of mobile robots needed. In Fig. 5.3, you can see the same calendar, but the slots are adjusted for a task with 2 robots and therefore you are able to choose also the time slots unavailable if reserving a task with 6 robots.

The third type of time slots, which can be displayed, is colored with light red. You can assign beginning of your reservation there, but the total duration of the reservation has to

Potvrzení rezervace systému pro řešení úlohy DPS1

Popis úlohy:

- **Celé jméno:** Detekce překážek dálkoměrnými senzory
- **Krátký popis:** Cílem úlohy je (autonomně) přemístit robot na určené místo. Toto místo je dané relativními souřadnicemi vzhledem k počáteční poloze robotu. Oproti předchozí úloze může být do trajektorie robotu umístěna překážka. Robot ji má detekovat a ukončit pohyb dostatečně včas, zamezil kolizi.
- **Motivace:** Seznámení se s použitím dálkoměrných senzorů k prevenci kolizi.
- **Začátek rezervovaného bloku:** 11:00:00 13/02/2011
- **Konec rezervovaného bloku:** 11:29:59 13/02/2011

Potvrzení rezervace

Figure 5.5: Confirm the selected data.

Vybraná úloha byla úspěšně zarezervována

Popis úlohy:

- **Celé jméno:** Detekce překážek dálkoměrnými senzory
- **Krátký popis:** Cílem úlohy je (autonomně) přemístit robot na určené místo. Toto místo je dané relativními souřadnicemi vzhledem k počáteční poloze robotu. Oproti předchozí úloze může být do trajektorie robotu umístěna překážka. Robot ji má detekovat a ukončit pohyb dostatečně včas, zamezil kolizi.
- **Motivace:** Seznámení se s použitím dálkoměrných senzorů k prevenci kolizi.
- **Začátek rezervovaného bloku:** 14:30:00 11/02/2011
- **Konec rezervovaného bloku:** 14:59:59 11/02/2011

Nová rezervace

Figure 5.6: Reservation was successfully saved into the database.

Bibliography

- [1] <http://logging.apache.org/log4j/docs/index.html>.
- [2] <http://xerces.apache.org/xerces-j>.
- [3] <http://xml.apache.org/xalan-j>.
- [4] <http://ant.apache.org>.
- [5] <http://www.w3c.org>.
- [6] <http://www.w3schools.com/xml/default.asp>.
- [7] http://www.w3schools.com/xml/xml_dtd.asp.
- [8] <http://www.w3schools.com/dtd/default.asp>.
- [9] <http://www.w3schools.com/dtd/default.asp>.
- [10] <http://java.sun.com/j2se/1.4.2/docs/guide/jni/index.html>.
- [11] <http://en.wikipedia.org/wiki/JNI>.
- [12] Jan Faigl, Jan Chudoba, Miroslav Kulich, Roman Mázl, Jiří Pavlíček, Libor Přeučil, and Petr Štěpán. SyRoTek v001.1 - Stav problematiky. Technical Report GLR 78/07, ProTyS a.s. and ČVUT v Praze, FEL, Gerstner Laboratory, 2007.
- [13] Jan Faigl, Jan Chudoba, Miroslav Kulich, Roman Mázl, Jiří Pavlíček, Libor Přeučil, and Petr Štěpán. SyRoTek v001.2 - Koncepce systému. Technical Report GLR 79/07, ProTyS a.s. and ČVUT v Praze, FEL, Gerstner Laboratory, 2007.
- [14] Jan Faigl, Jan Chudoba, Miroslav Kulich, Roman Mázl, Karel Košnar, Libor Přeučil, and Petr Štěpán. SyRoTek v002.1 - Architektura robotů. Technical Report GLR 81/08, ProTyS a.s. and ČVUT v Praze, FEL, Gerstner Laboratory, 2008.
- [15] Jan Faigl, Jan Chudoba, Miroslav Kulich, Roman Mázl, Karel Košnar, Libor Přeučil, and Petr Štěpán. SyRoTek v003.1 - Detailní koncepce systému. Technical Report GLR 82/08, ProTyS a.s. and ČVUT v Praze, FEL, Gerstner Laboratory, 2008.

- [16] Jan Faigl, Jan Chudoba, Miroslav Kulich, Roman Mázl, Karel Košnar, Libor Přeučil, and Petr Štěpán. SyRoTek v003.2 - Rámcová definice úloh. Technical Report GLR 83/08, ProTyS a.s. and ČVUT v Praze, FEL, Gerstner Laboratory, 2008.
- [17] Jan Faigl, Jan Chudoba, Miroslav Kulich, Roman Mázl, Karel Košnar, Libor Přeučil, and Petr Štěpán. SyRoTek v004.1 - Popis architektury robotu a první testy. Technical Report GLR 84/08, ProTyS a.s. and ČVUT v Praze, FEL, Gerstner Laboratory, 2008.
- [18] Jan Faigl, Jan Chudoba, Miroslav Kulich, Roman Mázl, Karel Košnar, Libor Přeučil, and Petr Štěpán. SyRoTek c004.1 - Koncept hardwarové komunikace. Technical Report GLR 85/08, ProTyS a.s. and ČVUT v Praze, FEL, Gerstner Laboratory, May 2008. ver. 1.0.
- [19] Jan Faigl, Jan Chudoba, Miroslav Kulich, Karel Košnar, Libor Přeučil, and Petr Štěpán. SyRoTek s005.1 - Specifikace hardwarové komunikace. Technical Report GLR 86/09, ČVUT v Praze, FEL, Gerstner Laboratory and ProTyS a.s., 2009. ver. 2.0.
- [20] Jan Chudoba, Jan Faigl, and Miroslav Kulich. SyRoTek s005.2 - Datová vrstva hardwarové komunikace. Technical report, ČVUT v Praze, FEL, Gerstner Laboratory and ProTyS a.s., December 2009.
- [21] Jan Faigl, Jan Chudoba, Miroslav Kulich, Karel Košnar, Libor Přeučil, and Petr Štěpán. SyRoTek v005.1 - Demonstrátor základních robotických funkcí. Technical Report GLR 87/09, ČVUT v Praze, FEL, Gerstnerova Laboratoř and ProTyS a.s., February 2009.
- [22] Jan Faigl, Jan Chudoba, Miroslav Kulich, Karel Košnar, Libor Přeučil, and Petr Štěpán. SyRoTek v006.1 - Specifikace úloh mobilní robotiky. Technical Report GLR 88/09, ČVUT v Praze, FEL, Gerstnerova Laboratoř and ProTyS a.s., February 2009.
- [23] Jan Faigl, Jan Chudoba, Miroslav Kulich, Karel Košnar, Libor Přeučil, and Petr Štěpán. SyRoTek v007.1 - Návrh koncepce internetového přístupu k systému. Technical Report GLR 89/09, ČVUT v Praze, FEL, Gerstnerova Laboratoř and ProTyS a.s., February 2009.
- [24] Jan Chudoba, Karel Košnar, Miroslav Kulich, Libor Přeučil, and Petr Štěpán. SyRoTek s009.1 - Robot S1R - Hardwarová dokumentace. Technical report, ČVUT v Praze, FEL, Gerstner Laboratory and ProTyS a.s., February 2010.
- [25] Jan Chudoba, Jan Faigl, Miroslav Kulich, and Karel Košnar. SyRoTek - Řízení proudové spotřeby robotu. Technical report, ČVUT v Praze, FEL, Gerstner Laboratory, 2008.

-
- [26] Karel Košnar, Miroslav Kulich, Jan Chudoba, Jan Faigl, Libor Přeučil, and Petr Štěpán. SyRoTek v008.1 - Výsledky ověření koncepce systému SyRoTek. Technical report, ČVUT v Praze, FEL, Gerstner Laboratory and ProTyS a.s., February 2010.
- [27] Karel Košnar, Jan Chudoba, Jan Faigl, Martin Saska, Miroslav Kulich, Libor Přeučil, and Petr Štěpán. SyRoTek v009.1 - Testy a demonstrace finální verze robotu. Technical report, ČVUT v Praze, FEL, Gerstner Laboratory and ProTyS a.s., February 2010.
- [28] Karel Košnar, Jan Faigl, Miroslav Kulich, Martin Saska, Tomáš Krajník, Jan Chudoba, Libor Přeučil, and Petr Štěpán. SyRoTek v010.1 - Způsob připojení a nastavení uživatele pro systém SyRoTek. Technical report, ČVUT v Praze, FEL, Gerstner Laboratory and ProTyS a.s., February 2010.
- [29] Karel Košnar, Jan Faigl, Miroslav Kulich, Tomáš Krajník, Martin Saska, Jan Chudoba, Sol Pedre, Libor Přeučil, and Petr Štěpán. SyRoTek v011.1 - Revize způsobu připojení a nastavení uživatele pro systém SyRoTek. Technical report, ČVUT v Praze, FEL, Gerstner Laboratory and ProTyS a.s., February 2011.
- [30] Karel Košnar, Jan Faigl, Martin Saska, Miroslav Kulich, Jan Chudoba, Petr Štěpán, and Libor Přeučil. SyRoTek v012.1 - Uživatelský manuál k systému SyRoTek. Technical report, ČVUT v Praze, FEL, Gerstner Laboratory and ProTyS a.s., February 2011.
- [31] Karel Košnar, Jan Faigl, Martin Saska, Miroslav Kulich, Jan Chudoba, Petr Štěpán, and Libor Přeučil. SyRoTek v012.2 - User's manual for SyRoTek e-learning system. Technical report, ČVUT v Praze, FEL, Gerstner Laboratory and ProTyS a.s., February 2011.