

# User Manual

Ricardo Aguirre  
Andrea Graziano  
Marco Teoli  
Alberto Zuccato

May 2, 2012

### About this document

<b>Subject</b>	User Manual
<b>Version</b>	0.2
<b>Date</b>	May 2, 2012
<b>Authors</b>	Aguirre Ricardo Graziano Andrea Teoli Marco Zuccato Alberto

# Contents

<b>1</b>	<b>System requirements</b>	<b>3</b>
<b>2</b>	<b>Compilation of the sources</b>	<b>4</b>
2.1	Installation procedure . . . . .	4
2.1.1	Dependencies . . . . .	4
2.1.2	GNAT . . . . .	4
2.1.3	XMLAda . . . . .	5
2.1.4	PolyORB . . . . .	5
2.1.5	YAMI4 . . . . .	5
2.1.6	Soccer Simulator . . . . .	5
<b>3</b>	<b>Starting the system</b>	<b>6</b>
3.1	Local execution . . . . .	6
3.2	Distributed execution . . . . .	7
<b>4</b>	<b>Using the Spectator GUI</b>	<b>8</b>
4.1	Overview . . . . .	8
4.2	Connecting . . . . .	9
4.3	Match and associated information . . . . .	10
4.4	Render quality . . . . .	10
4.5	Players magnification . . . . .	10
4.6	Players numbers . . . . .	10
4.7	Errors and exceptional situations . . . . .	10
<b>5</b>	<b>Simulator GUI</b>	<b>11</b>
5.1	Overview . . . . .	11
5.2	Connecting . . . . .	12
5.3	Connected instances of the others GUIs . . . . .	12
5.4	Match configuration . . . . .	12
5.5	Start match . . . . .	13
5.6	Start second half . . . . .	13
5.7	Abort match . . . . .	14
5.8	Errors and exceptional situations . . . . .	14
<b>6</b>	<b>Manager GUI</b>	<b>15</b>
6.1	Overview . . . . .	15
6.2	Connecting . . . . .	16
6.3	Line-up configuration . . . . .	17

6.3.1	Team . . . . .	17
6.3.2	Players . . . . .	18
6.3.3	Tactic . . . . .	19
6.4	Save line-up configuration . . . . .	20
6.5	Tactic change . . . . .	21
6.6	Substitute player . . . . .	22
6.7	Errors and exceptional situations . . . . .	22

# Chapter 1

## System requirements

The Soccer Simulator requires GNAT and the Java Virtual Machine.

It has been tested on GNU/Linux platforms. It could be deployed on other platforms, provided that Makefiles and other scripts are adapted to the underlying platform.

The minimum hardware requirements are:

- cpu speed: 1.2 GHz
- main memory: 1 GB
- disk space: 700 MB are required by GNAT, XMLAda and PolyORB; 60 MB are required by the simulator.

These requirements are intended for the execution of the whole system on a single node. The system can be deployed on a distributed environment requiring less resources on each node. In that case the same CPU frequency is required for the “core” and “Billboard out” subsystems; due to memory leak caused by a known bug found on DSA implementation, “Billboard out” requires a big amount of memory, so it is recommended to provide the same memory otherwise provided to the full system when deployed on a single node. Each Java graphical user interfaces requires 60 MB RAM.

A reliable network connection is recommended between “Core” and “Billboard Out” subsystems because of the high amount of data transmitted by the core subsystem. A reliable connection is also recommended for the Spectator GUI when the match is watched at its highest resolution.

## Chapter 2

# Compilation of the sources

### 2.1 Installation procedure

The following procedure has been tested under 32-bit installations of both Ubuntu 12.04 Desktop and Server.

#### 2.1.1 Dependencies

In order to install GNAT, XMLAda, PolyORB and YAMI4 core libraries you need to ensure that you have some packages already installed. Execute:

```
sudo apt-get install libjpeg62 make gcc
```

In order to build the YAMI4 Java library and the user interfaces provided with our project, you need to have “ant” and a Java Development Toolkit (such as the OpenJDK) installed on your system:

```
sudo apt-get install g++ ant openjdk-6-jdk
```

#### 2.1.2 GNAT

Download GNAT from [libre.adacore.com](http://libre.adacore.com), unpack and enter the GNAT set-up directory. Execute:

```
sudo ./doinstall
```

as explained in the INSTALL file provided with the package.

Confirm or change the installation directory as you wish. From now on, the installation directory will be called GNAT\_HOME; to follow these instructions, replace the string <GNAT\_HOME> with the one you are using (by default, it is /usr/gnat). As requested at the end of the installation procedure, add <GNAT\_HOME> to your PATH environment variable:

```
export PATH=$PATH:<GNAT_HOME>/bin
```

To make this change permanent, edit the /etc/environment file; at the end, it should look like:

```
PATH="<GNAT_HOME>:[...]"
```

Please follow this workaround before executing the next steps on new versions of Ubuntu, because otherwise the compiler will be unable to find and link some files, such as the one called “crti.o”. By executing:

```
locate crt1.o
```

you should find its containing directory (it should be /usr/lib/i386-linux-gnu); export it in your `LIBRARY_PATH` environment variable:

```
export LIBRARY_PATH=$LIBRARY_PATH:/usr/lib/i386-linux-gnu
```

We recommend to add the `LIBRARY_PATH` variable to the `/etc/environment` file, because the same problem will otherwise happen during the project build process.

### 2.1.3 XMLAda

Download XMLAda from [libre.adacore.com](http://libre.adacore.com). Unpack and enter the set-up directory. Execute:

```
./configure --prefix=<GNAT_HOME>
make all install
```

as explained in the README file provided with the package.

### 2.1.4 PolyORB

Download PolyORB from [libre.adacore.com](http://libre.adacore.com). Unpack and enter the PolyORB set-up directory. Because the Ada GCC compiler has substituted the default one, the installation procedure will fail on guessing the right C compiler. Override the `CC` environment variable and install PolyORB by executing:

```
CC=/usr/bin/gcc ./configure --prefix=<GNAT_HOME> --with-appli-perso
="dsa" --with-PROTO-Perso="giop"
make
make install
```

Add this line to your `/etc/environment` file:

```
POLYORB_CONFIG='polyorb-config'
```

For more information about the procedure above, see the `INSTALL` file provided with the package

### 2.1.5 YAMI4

YAMI4 download ([www.inspirel.com/yami4](http://www.inspirel.com/yami4)) is not required because all the sources used by the project are already included.

### 2.1.6 Soccer Simulator

Enter the AdaWay project source directory and then build by executing:

```
make
```

## Chapter 3

# Starting the system

### 3.1 Local execution

Enter the simulator source directory and execute all partitions using one commands (it needs xterm):

```
make start-all
```

Or execute each partition in this order:

1. execute name\_server and Billboard\_Output

```
make start-billboard-out
```

2. execute Core

```
make start-core
```

3. execute billboard\_input

```
make start-billboard-in
```

4. execute Simulator-gui

```
make start-simulator-gui
```

5. execute spectator-gui

```
make start-spectator-gui
```

6. execute manager-gui (local)

```
make start-manager-gui
```

7. execute manager-gui (visitor)

```
make start-manager-gui
```



## 3.2 Distributed execution

Edit the file DISTRIBUTION.txt contained inside the “config” directory of the source project folder. Change the IP and port values of “Billboard in” and “Billboard out” to reflect the current network configuration. Copy and build (as described in the previous section) the project source directory on each node. It is possible to copy only the executables (along with the “config” folder) if each node executes on the same architecture.

Make sure that firewall is turned off on each node or open the ports contained on DISTRIBUTION.txt (8080, 8081, 8082).

1. execute name\_server (this should keep running until simulation ends)

```
make start-name-server
```

Once executed, the nameserver creates a file called “IOR.txt” inside the “config” directory of the project sources. That file must be copied inside the same directory on each distributed node.

2. Execute billboard\_output:

```
make start-billboard-out
```

3. Execute Core:

```
make start-core
```

4. Execute billboard\_input:

```
make start-billboard-in
```

5. Execute Simulator GUI:

```
make start-simulator-gui
```

6. Execute Spectator GUI:

```
make start-spectator-gui
```

7. Execute Manager GUI (local):

```
make start-manager-gui
```

8. Execute Manager GUI (visitor):

```
make start-manager-gui
```

## Chapter 4

# Using the Spectator GUI

### 4.1 Overview

The Spectator GUI is the front-end of the simulation. It displays the simulation and it offers basic features aimed to control various aspects of the visualization. Once started by the user, the GUI connects automatically to the system, live data are gathered as soon as the simulation is started.



Figure 4.1: The Spectator GUI

The main window is divided into two resizable areas. The soccer field widget, on the left side, shows the soccer field, the players and the ball. The bottom-left area contains the functions related to the visualization of the match. The area on the right side is divided into three sub-areas: the top shows the status of the match and the score of the challenging teams, the central shows a text area that displays the commentary of the match, the bottom area displays the statistics dialog button. The non-modal statistics dialog is opened on button's pressure. The statistics dialog displays various statistic parameters associated with each

team. The colors are fixed: the red color is always associated with the local team, the blue color with visitor team.



1	Milan - Inter	1
1	Shootings	0
3	Catches	5
0	Corner Kicks	0
0	Fouls Done	0
0	Fouls Suffered	0
0	Yellow Cards	0
0	Red Cards	0
0	Free Kicks	0
0	Penalty Kicks	0
0	Autogoals	0
3	Passings	6
2	Tackles Done	2
2	Tackles Suffered	2
0	Throw Ins	0
0	Headings	0
0	Goal Kicks	0
1	Kick-offs	1

Figure 4.2: The statistics dialog

## 4.2 Connecting

A modal dialog blocks the main interface during the initial connection step. If the Billboard Input partition is found, the dialog will be closed automatically. The dialog is only a practical way to improve usability of the GUI, it makes explicit the fact that the GUI requires the connection.

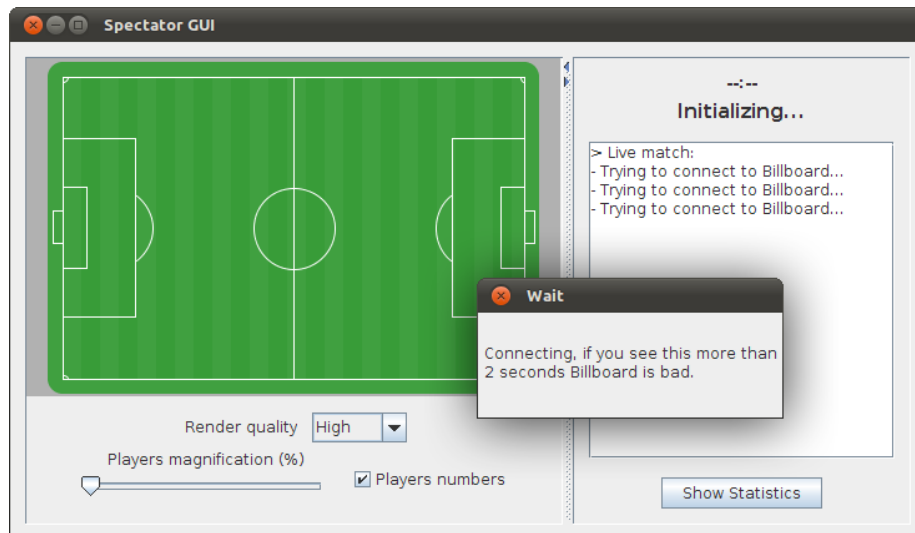


Figure 4.3: The Spectator GUI during connection

### 4.3 Match and associated information

The match and its associated information are displayed live on the main widget exclusively when the simulation is running. The match status and the score of the teams are displayed on the top-right side of the interface, the commentary is displayed on the text area and the statistics are displayed on the separate dialog that opens through the button. The latter information updates every second, the frequency of the match rendering, on the contrary, can be changed by the user.

### 4.4 Render quality

The user can change the rendering quality of the simulation, intuitively more quality is better leading to a smoother rendering. On the contrary a less qualitative render can help in the case of clients with limited resources, or slow network transmission. There are three possible choices: high quality rendering (data are received and repainted every 30 ms), medium quality rendering (60 ms) and low quality (90 ms) rendering.

### 4.5 Players magnification

The slider is used to change the players and ball dimensions. The minimum size is 100% that is the real size, the maximum scaling is 300% that means three times larger.

### 4.6 Players numbers

The check-box enable or disable the visualization of the player's uniform numbers from the graphical representation of the players on the soccer field widget.

### 4.7 Errors and exceptional situations

- The GUI connects automatically to Billboard at the beginning, if the Billboard Input partition is not available, the GUI keeps reconnecting every five seconds until: the Billboard Input becomes available or the user exits from the GUI.
- The GUI needs to register its subscribers to Billboard channels after the connection has succeeded but the registration won't happen if the Billboard Output is not available. The user must start the missing Billboard partitions and restarts the GUI.
- It is possible that the soccer field widget does not display the match, even after successful connection and registration. This is due to the fact that the GUI works live, so it starts to display something as soon as the match is started.

## Chapter 5

# Simulator GUI

### 5.1 Overview

The Simulator GUI is the front-end to control the flow of the simulation. It offers basic features: the configuration of the simulation using custom settings, the start match signal, the abort match signal and the second half start signal. In addition the user can see the presence of the other GUI instances connected to the system. Once started by the user, the Simulator GUI connects automatically to the system and initializes on the appropriate state of the simulation. The possible states are: warm up, first half, break time, second half, match end. The configuration of the match is allowed only during the warm up state. Anyhow, the notification of the new state will trigger the GUI to pass to the appropriate state.

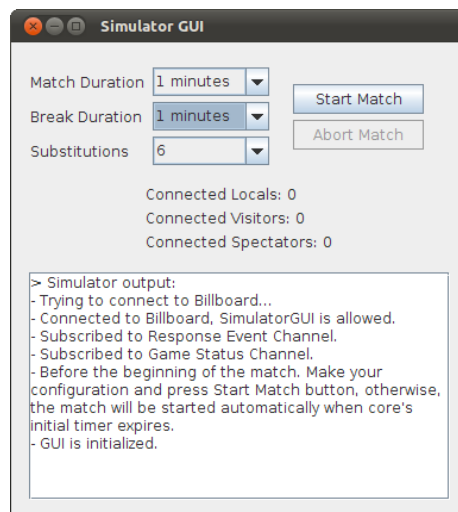


Figure 5.1: The Simulator GUI

The main window is divided into three parts: the top area displays the tools needed to make the configuration and to control the flow of the simulation. The central area shows the information regarding the number of connected GUI

instances within the system. The bottom part displays relevant information on the GUI state changes.

## 5.2 Connecting

A modal dialog blocks the main interface during the initial connection step. If the Billboard Input partition is available, the dialog will be closed automatically. The dialog is only a practical way to improve usability of the GUI, it makes explicit the fact that the GUI requires the connection. After the connection there is the initialization step that brings the information needed for the population of the combo boxes (match settings) and for the triggering of the appropriate initial state.

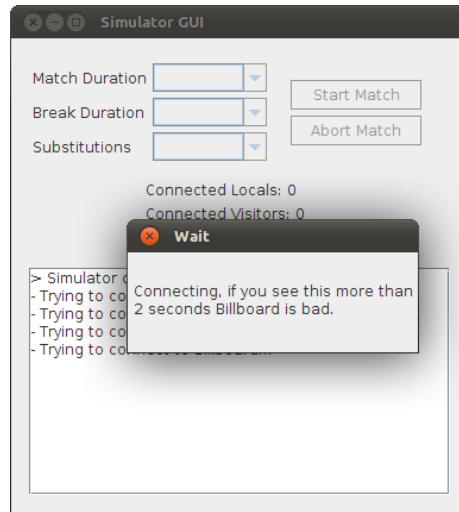


Figure 5.2: The Simulator GUI during connection

## 5.3 Connected instances of the others GUIs

The labels on the central area of the GUI display the number of instances of each GUI's category currently connected to the system. The maximum allowed connected instance for locals and visitors is one. On the contrary, the connected instances of Spectator GUI are not bounded in multiplicity. The correspondent GUI counter is updated whenever a new GUI initializes or leaves the system.

## 5.4 Match configuration

The match settings are available for use only during warm up state. The possible values for the duration of the period are: 1, 2, 5, 10, 15, 25 and 45 minutes. The values for the duration of the break time are: 1, 2, 5, 10 and 20 minutes. The maximum allowed substitutions can be 3 or 6. The setting are transmitted on start match button's pressure.

## 5.5 Start match

The start match button is used to control the beginning of the simulation. Once pressed, the GUI requests the start of the simulation on the Core. It may happen on the contrary that the simulation starts in the meanwhile (the GUI gets notified because the initial countdown had expired on the Core). Then the GUI passes to first half state.

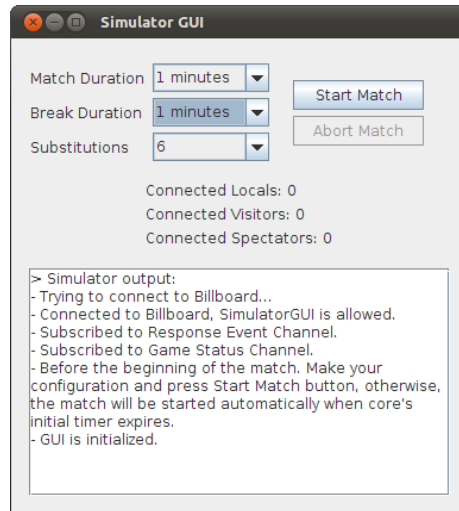


Figure 5.3: The Simulator GUI during warm up

## 5.6 Start second half

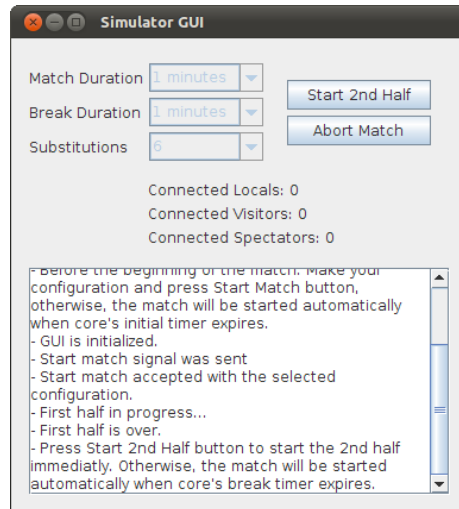


Figure 5.4: The Simulator GUI during break time

The start second half button is used to the resume of the simulation from the break, it is available only during break time. Once pressed, the GUI requests the start of the second half on the Core. It may happen on the contrary that

the second half starts because the GUI gets notified from the Core (the break time countdown had expired on the Core). Then the GUI passes to second half state.

## 5.7 Abort match

The abort match button is used to force the termination the simulation, it is available during the course of the match, also during the break. Once pressed, the GUI requests the termination of the simulation on the Core. Then the GUI becomes no longer usable and needs to be restarted in order to control another match.

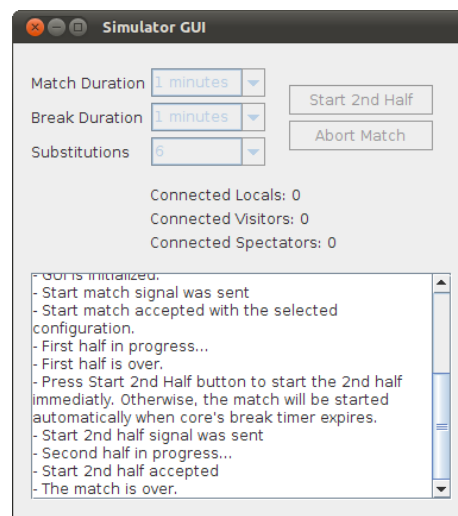


Figure 5.5: The Simulator GUI after the end of the second half

## 5.8 Errors and exceptional situations

- The GUI connects automatically to Billboard at the beginning, if the Billboard Input partition is not available, the GUI keeps reconnecting every five seconds until: the Billboard Input becomes available or the user exits from the GUI.
- The GUI needs to register its subscribers to Billboard channels after the connection has succeeded but the registration won't happen if the Billboard Output is not available. The user must start the missing Billboard partitions and restarts the GUI.
- The GUI needs to receive the initialization from Billboard output, if for some reasons the initialization fails, the GUI is no longer usable and the GUI must be restarted.



## Chapter 6

# Manager GUI

### 6.1 Overview

The Manager GUI is the front-end to control the team within the simulation. The user can access both roles (local team or visitor team) using the same interface. The GUI does not connect automatically, the role selection needs to be made before connecting. If the connection succeeds then the selection will be applied permanently until the GUI's closure.

The Manager GUI offers the following features: the configuration of the team using custom settings, the save team configuration signal, the change of the tactic and the player substitution, the visualization of the dynamic player's parameters (during the match). Once started by the user, the Manager GUI becomes visible, the user chooses the role and connects to the system pressing the connect button. The GUI connects, then receives the initialization.

Before the beginning of the match the user can set the line-up and save the configuration on the Core. The configuration is structured on three step: selection of the team, selection of the team players and selection of the team tactic.

Once the configuration has been accepted on the Core, the GUI passes to the in-game state where it waits for the start of the match. Once the match has started, the main table updates the player's parameters. The user can watch the status of the players as the match evolves, change the tactic of the team or substitute a player.

The main windows takes diverse graphic configurations. If the GUI is instanced at the beginning, during warm up, it will show the first page of the team configuration. Here the user chooses the team and progress to the next page. The second page displays the players of the selected team along with their static characteristics. The third page shows the team formation selector and the pressing level slider. Once the user has done with the configuration can proceed to save the settings on the core, pressing the save button on the third page.

If the GUI is instanced during the game, it displays the table view of the players statistics. Here the user can see the dynamic status of each player of the team. The bottom area of the interface is occupied by the buttons related to the change team and the substitute player functionality, and by the text area displaying the text messages of the players notifications. The change tactic dialog is structures like the third page of the configuration layout, the substitute

player dialog is structured as the second page of the configuration layout except for the not-convened players.

The screenshot shows a window titled "Manager GUI" with a sub-header "(LOCAL / Milan) CT Panel". It contains a table with 10 columns: Name, Shirt Num..., Status, Catches, Goals, Autogoals, Shootings, Passings, and Headin. Below the table are two buttons: "Change Tactic" and "Substitute Player". At the bottom is a log area with a scroll bar.

Name	Shirt Num...	Status	Catches	Goals	Autogoals	Shootings	Passings	Headin
Abbiati	32	ACTIVE	1	0	0	0	1	1
Antonini	77	ACTIVE	0	0	0	0	0	0
Nesta	13	ACTIVE	1	0	0	0	0	0
Thiago Sil...	33	ACTIVE	0	0	0	0	0	0
Abate	20	ACTIVE	0	0	0	0	0	0
Van Bom...	4	ACTIVE	0	0	0	0	0	0
Nocerino	22	ACTIVE	2	0	0	0	2	0
Seedorf	10	ACTIVE	2	0	0	0	2	0
Boateng	27	ACTIVE	0	0	0	0	0	0
Ibrahimovic	11	ACTIVE	2	1	0	2	0	0
Robinho	70	ACTIVE	0	0	0	0	3	0
Amelia	1	INACTIVE	0	0	0	0	0	0
Mexes	5	INACTIVE	0	0	0	0	0	0
Zambrotta	19	INACTIVE	0	0	0	0	0	0
Aquilani	18	INACTIVE	0	0	0	0	0	0
Gattuso	8	INACTIVE	0	0	0	0	0	0
Cassano	99	INACTIVE	0	0	0	0	0	0

Log messages:

- Trying to connect to billboard as local team Manager...
- Connected to Billboard, ManagerGUI is allowed.
- Subscribed to Response Event Channel.
- Subscribed to Players Statistics Channel.
- Subscribed to Players Messages Channel.
- Subscribed to Game Status Channel.
- GUI is initialized.
- Save config signal was sent
- Configuration not allowed. Redirecting to ct panel.
- Referee ends the first half.

Figure 6.1: The Manager GUI

## 6.2 Connecting

A modal dialog provides the way to the user to connect with specified role after the GUI becomes visible. If the Billboard Input partition is available, the dialog will be closed automatically. The dialog is only a practical way to improve usability of the GUI, it makes explicit the fact that the GUI requires the connection. After the connection there is the initialization step that brings the information needed for the population of the graphical structures of the GUIs (team settings) and for the triggering of the appropriate initial state.

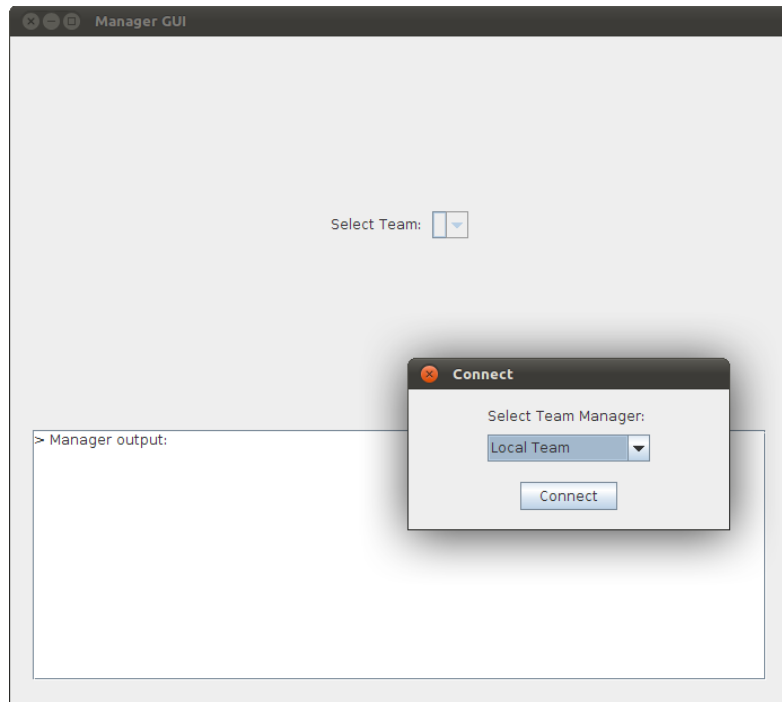


Figure 6.2: The Manager GUI during connection

## 6.3 Line-up configuration

The configuration of the line up is divided into three sequential step. The page layout holds the three pages of the configuration. The user navigate through the pages with the next and previous buttons and make the preferred choice.

### 6.3.1 Team

The user chooses the favourite team on the combo-box, this action will load the appropriate data on the following pages.

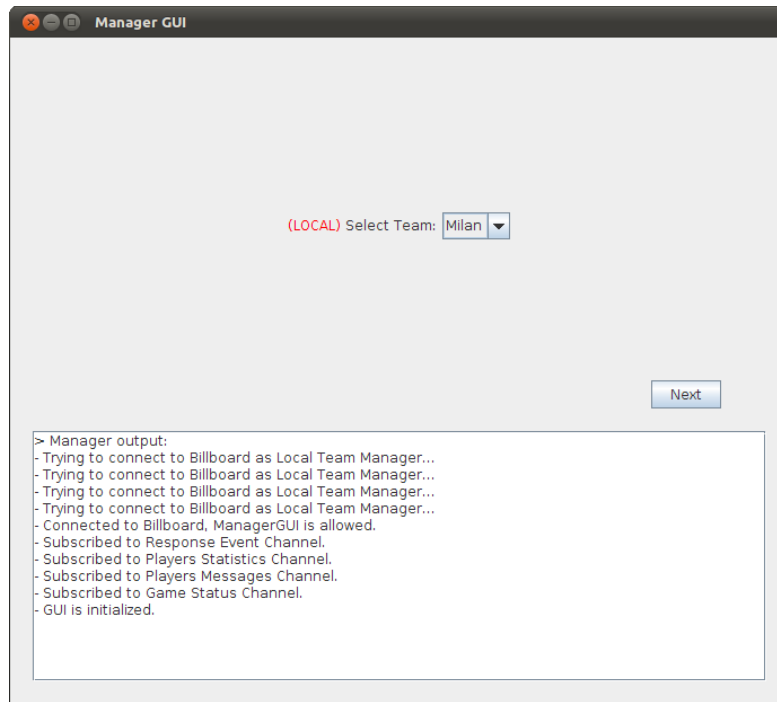


Figure 6.3: The first page of the team configuration

### 6.3.2 Players

The user chooses the favourite combination of players through the classic swap players mechanism. The first player is selected for the swap, then the pick player button is pressed, after that the other player is selected, and the button is pressed again. This action will swap the players. The user may consider players skills during the constitution of the preferred combination. The players are divided into three groups: the first-strings that are the first 11 players, the substitutes that are the following 6 players and the not-convened players, that will not take part to the match.

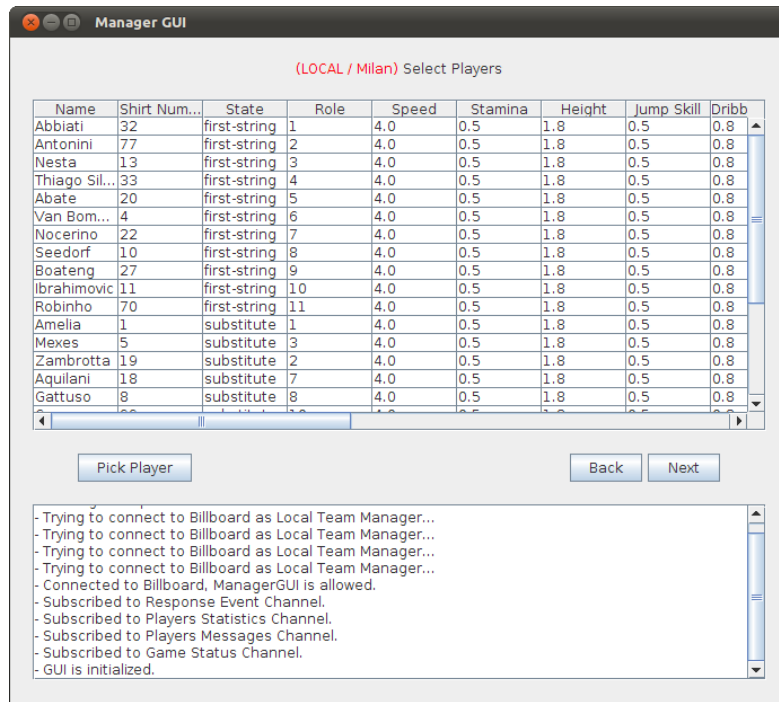


Figure 6.4: The second page of the team configuration

### 6.3.3 Tactic

The user chooses the favourite combination of formation and pressing level. The soccer field widget helps on visualizing the line-up. The formations are limited to two choices: 4-4-2 and 4-3-3 scheme. The pressing level can be set on ten levels, from the most defensive (left side of the slider) on the most offensive (right side of the slider).



Figure 6.5: The third page of the team configuration

## 6.4 Save line-up configuration

The save configuration button will apply the selected configuration on the Core. If the action succeeds then the GUI passes to in-game state waiting for the start of the match. There are other potential situations in which the save signal does not cause the side effect on the Core. If the match is started before the user issued the save request, the GUI passes to in-game state with the configuration received by the Core at that moment. If the user save the configurations choosing the same team of the other Manager after the latter has successfully saved his configuration, the GUI is not allowed to proceed and the user is forced to choose another team.

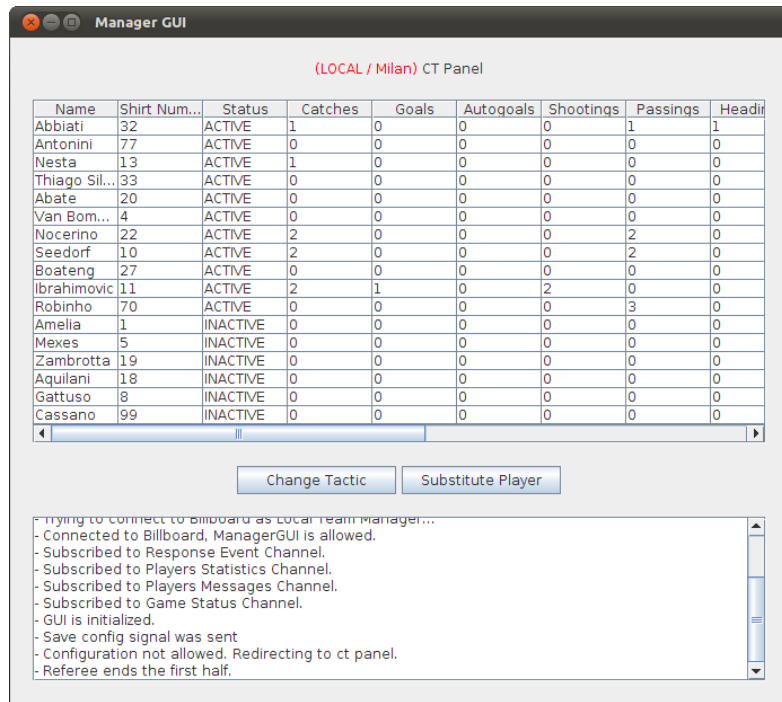


Figure 6.6: The Manager GUI after a successful save

## 6.5 Tactic change

The user opens the tactic change dialog using the button on the GUI during in-game state. The function of the dialog is identical to that of the third page of the team configuration. The user modifies the formation and the pressing level and then press the apply button to confirm the changes.



Figure 6.7: The tactic change dialog

## 6.6 Substitute player

The user opens the substitute player dialog using the button on the GUI during in-game state. The function of the dialog is identical to that of the second page of the team configuration, except for the fact that the not-convened players are not showed. The user carries out the preferred substitution swapping two players and then apply the change. Only one players a time can be substituted, moreover the swapping of two first-string players is not supported.

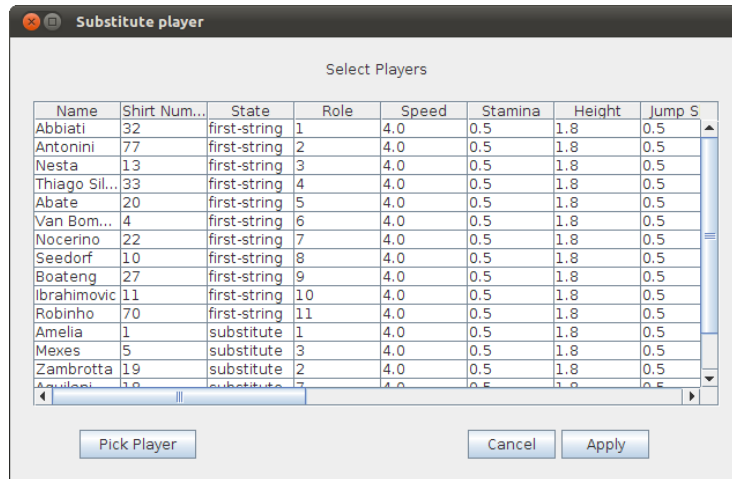


Figure 6.8: The substitution player dialog

## 6.7 Errors and exceptional situations

- When the user connects the GUI to Billboard at the beginning, if the Billboard Input partition is not available, the GUI keeps reconnecting every five seconds until: the Billboard Input becomes available or the user exits from the GUI.
- The GUI needs to register its subscribers to Billboard channels after the connection has succeeded but the registration won't happen if the Billboard Output is not available. The user must start the missing Billboard partitions and restarts the GUI.
- The GUI needs to receive the initialization from Billboard output, if for some reasons the initialization fails, the GUI is no longer usable and the GUI must be restarted.
- If additional save configuration requests is made by the same Manager, after a previous one had been accepted, those request will not cause the side effect. Once a configuration is accepted it will not be overwritten.