games_vision

The games_vision package is made of different games for HRI based on vision.

games_vision uses loose dependency strategies to the following packages: - etts - gesture_player - screen_msgs - touch_skill

For more info, please check the documentation of these packages.

touch_skill

There are several ways to use touch_skill: * Using TouchListener (TouchListener.h):

```
#include "TouchListener.h"
class FooTouchListener : public TouchListener {
  public:
  //! the function that will be called when a touch event is recevied
  void touch_cb() {
    if (is_touched())
      etts.sayTextNL("|en:Touched.|es:Tocado.");
    else
      etts.sayTextNL("|en:Released.|es:Soltado.");

    if (is_touched_left_shoulder)
      ...
  } // end touch_cb()
}; // end class FooTouchListener
```

- Loose dependency: Using the small footprint "capacitive*touch" ROS node, on topic "capacitive* touch"

```
#include <std_msgs/String.h>
ros::NodeHandle nh_public;
//! fake touch publisher
ros::Publisher touch_pub;
touch_pub = nh_public.advertise<std_msgs::String>("capacitive_touch");
std_msgs::String msg;
msg.data = "left_shoulder";
pub.publish(msg);
//! touch subscriber
ros::Subscriber touch_sub;
touch_sub = nh_public.subscribe<std_msgs::String>("capacitive_touch", 1, touch_
cb);

void touch_cb(const std_msgs::StringConstPtr & msg) {
  if (msg->data == "left_shoulder")
    ...
} // end touch_cb()
```

Licence

LGPL v3 (GNU Lesser General Public License version 3).