

Super Powered LEGO - Colorable Bricks

MAXIMILIAN DAVID

s1madavi 1349951

maximilian.david@uni-bayreuth.de

December 19, 2022

When assembling LEGO, searching for the correctly colored brick can take a long time and can annoy. By introducing an interaction that makes it possible to make any brick the desired color, these issues can be alleviated.

To help users understand the process and help them build a mental model, the needed actions to color in bricks need to be as natural as possible. Therefore, a brush that can be loaded with paint by dipping it inside paint pots and then coloring the bricks seemed the most natural for this purpose. For the realization of this new feature, the correct models were needed. These were found for free after scouring the internet.

The model of the bucket was well suited for its intended use since the bucket and the paint inside it were two separate meshes. This makes it possible to give both objects different materials. The brush's model, on the other hand, needed more preparation. Bristles and handle were one solid object, which made it impossible to give them different materials. Therefore, the bristles and the handle were manually separated in blender and reimported into unity. Now the bristles were separately colorable from the handle. This represents a very important feature since the bristles can be used to inform the user about which paint is currently on the brush.



Figure 1: Brush with uncolored bristles



Figure 2: Loading brush with paint

The "PaintBucket" and "PaintBrush" scripts were created to provide the needed functionality for the new objects. These scripts were then added to their respective game objects, as well as trigger colliders. Additionally, the brush received the "Throwable" script, and the tags "Bucket", "Brush", and "Brick" were created and added to the corresponding objects. These tags make it possible to check if a colliding object has the correct type. Now it was possible to pick up the brush and receive collisions between it, the buckets, and the bricks.

To make the bricks colorable, the brush stores a material and changes the current material of a brick, if it touches one. If the brush collides with one of the paint pots, the brush's material will change to the one that is stored inside the bucket. This material change is visualized by the brush's bristles. Therefore, the desired functionality is now implemented.

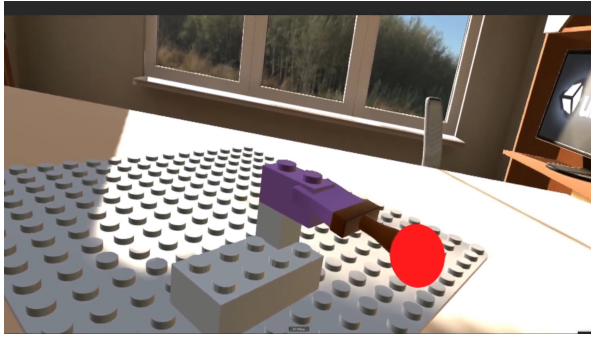


Figure 3: Coloring a brick

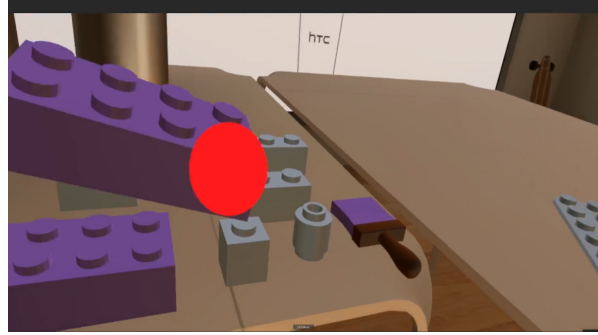


Figure 4: Colored spawner duplicating a colored brick

To fully complete this development step, however, another small detail needed to be added. Since it is possible to pick up the brush and throw it around the scene, the user can also lose it. Therefore, some mechanic to retrieve the brush was needed as well. Whenever the user presses the down button on the controller's directional pad, all loose bricks will be deleted and the scene will be cleaned. Thus, it would be in sync with the user's mental model to also return the brush to its original position. Adding a function to do so, therefore, concludes this step of the development process.