MKS

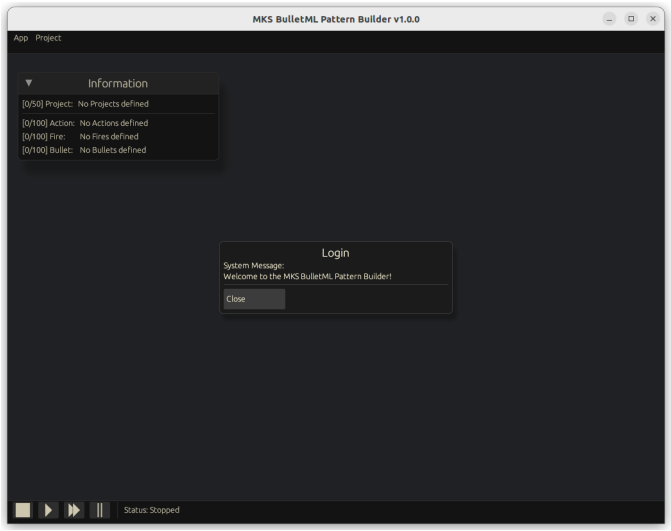
BulletML Pattern Builder

Player Aimed Tutorial

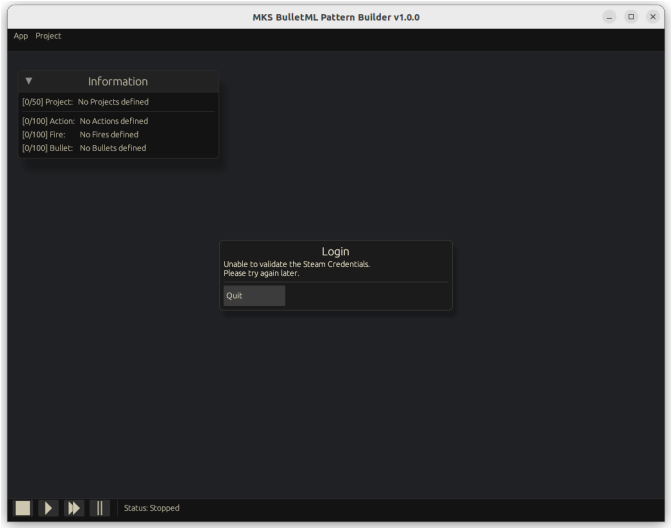
by Micah Koleoso Software, 2025
https://micahkoleososoftware.com

Getting Started

As an introduction, the goal here is to show the quick workflow setting up bullet patterns and previewing the results playing in the Playfield of the editor. On starting the application, the main user interface shows the application after a successful login and providing any system information that may be important for working with the tool. In the case shown below, there is no specific message available and the login has been successful.

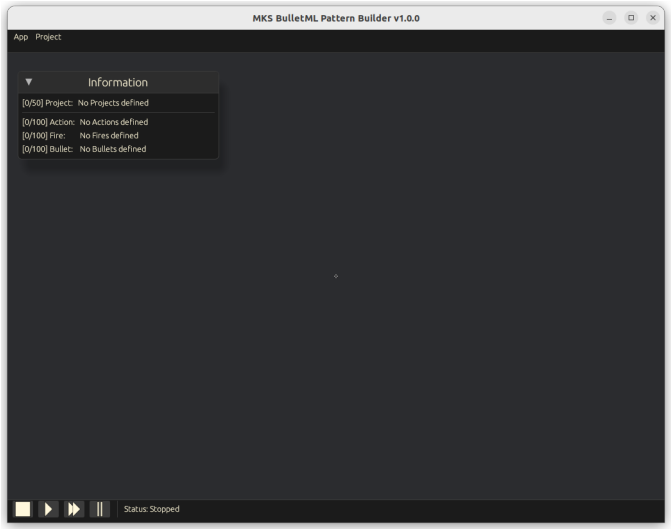


The application is launched through the Steam Launcher and requires a valid purchase on the user's Steam account as well as the execution of the Steam Launcher at all times for validation of the access rights. If the Steam Launcher is not able to verify the correct credentials and purchase or the application account is not in good standing, the Pattern Builder tool will indicate the following on start-up:



Please contact our support per email or come into our Discord Server if you encounter any issues here that we can help with.

After confirmation of the start-up message dialog, the following screen is presented. As we haven't created any projects yet, the menu shows only relevant next steps.

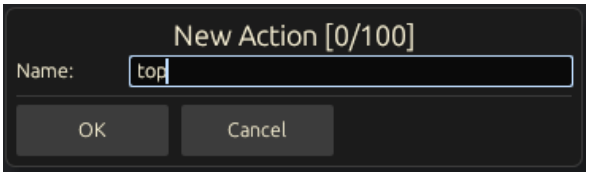


At this stage, we can go through the introductions as described below. Note that the Information Window can be moved per mouse drag in the window title or toggled visible using the F2 key or the sub-menu option “Show/Hide UI Overlays” under the App menu.

Aimed Shot

In this introduction, we want to use the player position as a directional goal for the bullets. The player position is demonstrated here as a small cross which appears in the virtual playback area. When the mouse pointer is within the limits of the virtual Playfield, the cross follows the mouse pointer simulating the movement of a player within the field. This can be used to see the effects of logic built with regards to aimed shots at the player.

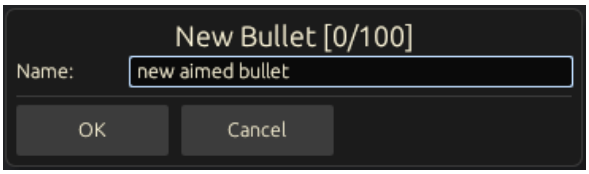
First, create a new project named “aimed” and enter the description text “use the player position for aimed shots”. Press “OK” to leave the dialog and save the settings. From now on, I will assume the user clicks the “OK” button to close the respective dialog as mentioned below. This saves the data entered and continues with the next step.



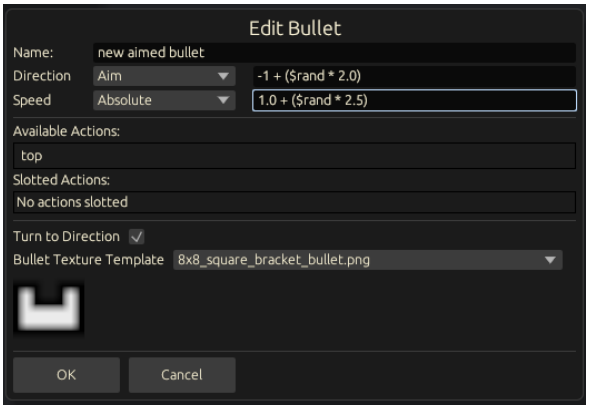
Create a new Action called “top”, which will be our entry point into the playback.

Create a second Action called “spray” but don’t open it for editing yet.

Create a Fire named “aimed fire” and OK the dialog for saving.



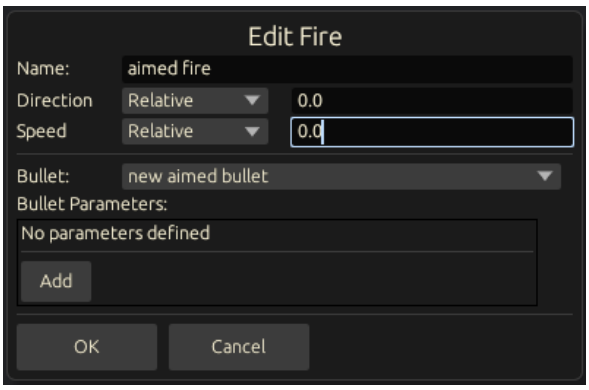
As the last step of the creation process, create a new Bullet named “new aimed bullet”. Save that and go on to edit the Bullet we just created.



Open the Bullet “new aimed bullet” for editing, it should still be selected as current bullet since it’s the only one and it’s the last one created.

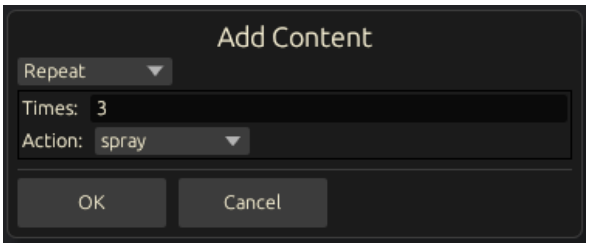
As we want the Bullet to aim at the player position, select the Direction attribute as “Aim”, enter the Expressions as shown above and make sure the Speed attribute is “Absolute”. The Direction and Speed settings enable the Bullet to be emitted having the direction to the player with up to 1 degree random variance angle to each side. The Speed Expression enables the bullet to have a movement velocity of between 1.0 and 2.5 pixels total per frame. At a playback rate of 60 frames per second, bullets will really be flying!

For this example, I chose the square bracket sprite as the image. Leave all other settings as per default.

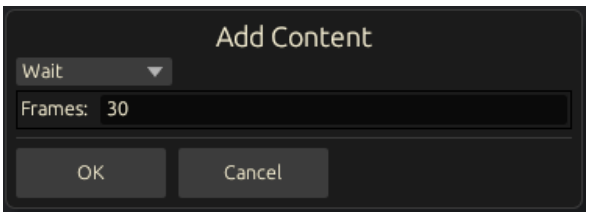


Next, open the Fire “aimed fire” for editing. We set Direction and Speed to “Relative” and the values to “0.0” to prevent this Fire having any further influence over the Bullet “new aimed bullet” it will emit. Both Fires and Bullets can have attributes set, and the Bullet is evaluated first, then the Fire. So it is possible, that the Fire can completely override the Bullet values at runtime, this depends which attributes are set for Direction and Speed in both edits.

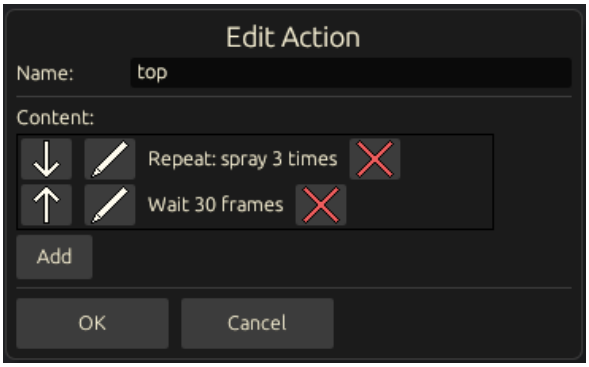
Select and open the “spray” Action to enter a FireRef content with the Fire “aimed fire” to be referenced. Save that and continue. For the next two steps, open the “top” Action for editing.



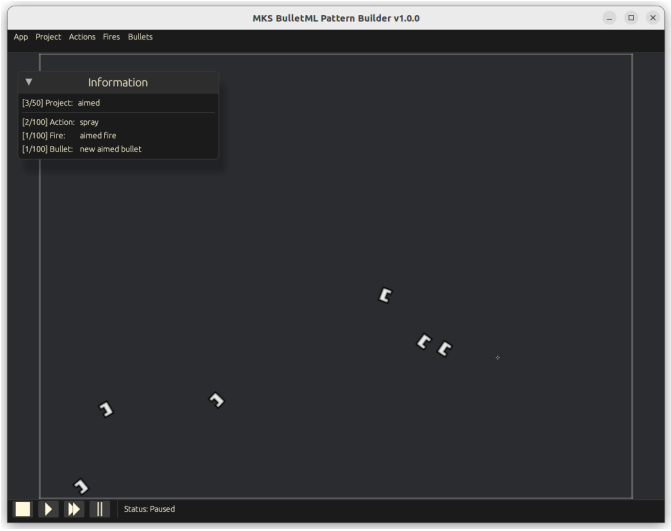
In the “top” Action, add content for a repeat with 3 times set and let it call the “spray” Action each time.



Add another content to the “top” Action, add a Wait with 30 frames, which should be half a second.



The final contents of the Action “top” are shown above. We are calling the “spray” Action 3 times and then waiting half a second before this is looped again.



After pressing the “Play” button in the status bar, you can see bullets being emitted around the position of the player position, shown as a cross, in the virtual Playfield. In this case, we are emitting 3 bullets from the center of the Playfield to an angle of -1 to +1 degrees around the player position.

When the mouse cursor leaves the Playfield, a standard player position in the center bottom half of the field is fixed as player position representative, regardless of the type of

Playfield BulletML type attribute (None, Horizontal, Vertical) set. Move the mouse in the Playfield to simulate the movements of a player craft.

15th August, 2025	v1.0 - Initial version.