

# Runner Movement System

## Оглавление

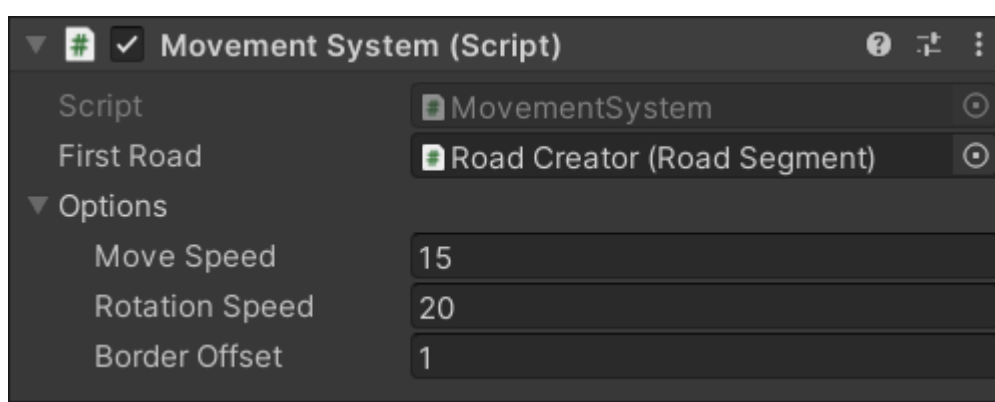
|   |   |
|---|---|
| Описание .....  | 2 |
| Добавление перемещения игроку .....                             | 2 |
| Управление игроком (добавление ввода) .....                     | 3 |
| Стыки дорог (развилки) .....                                    | 3 |
| Переходы между дорогами (прыжки) .....                          | 4 |
| Автоматическое соединение перехода со стыкующими дорогами ..... | 5 |

## Описание

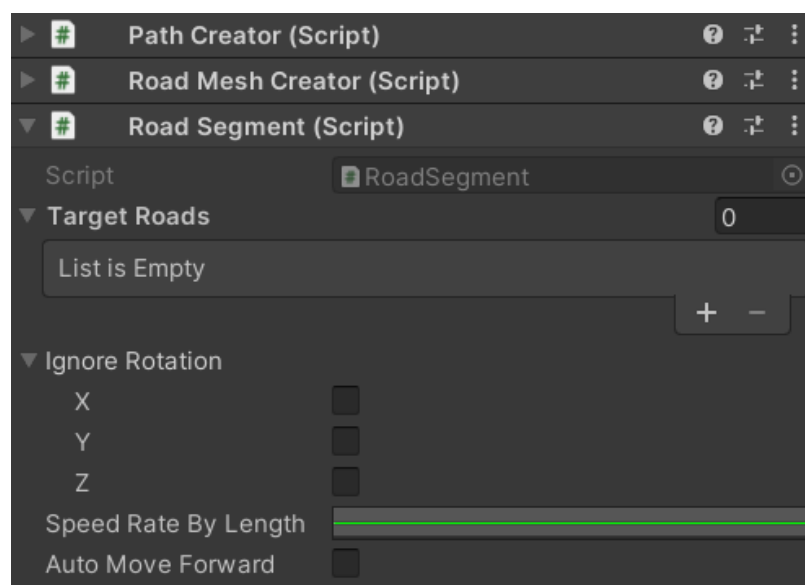
Данный пакет является дополнением к ассету [Bézier Path Creator](#). Пакет содержит готовую систему перемещения по заданному пути и возможность настраивать переходы между путями.

## Добавление перемещения игроку

Главный скрипт для перемещения – MovementSystem. Этот скрипт должен находиться на объекте, который будет управляться и перемещаться по дороге. После добавления этого скрипта автоматически добавятся несколько других скриптов, необходимых для корректной работы перемещения.



На дороге, по которой игрок может перемещаться, должен висеть скрипт RoadSegment. Этот скрипт также требует, чтобы на объекте присутствовали компоненты RoadMeshCreator и PathCreator, с помощью которых можно настроить внешний вид дороги и траекторию.



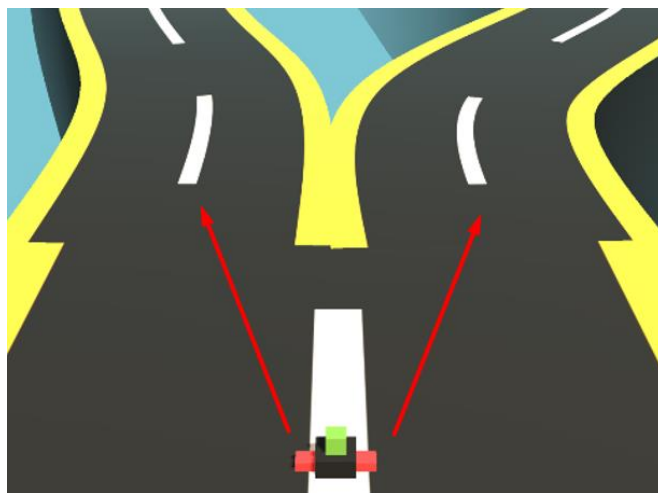
В компонент `MovementSystem` нужно передать дорогу, с которой игрок начнёт движение. После этого остаётся добавить ввод и перемещение по дороге будет готово.

Обратите внимание, что начальная позиция игрока задаётся динамически: берется ближайшая точка к игроку, находящаяся на заданной первой дороге, и игрок начинает движение с этой точки. Таким образом, если Вы хотите, чтобы игрок начинал движение с начала дороги, то позиция игрока на сцене должна быть приближена к началу дороги.

### Управление игроком (добавление ввода)

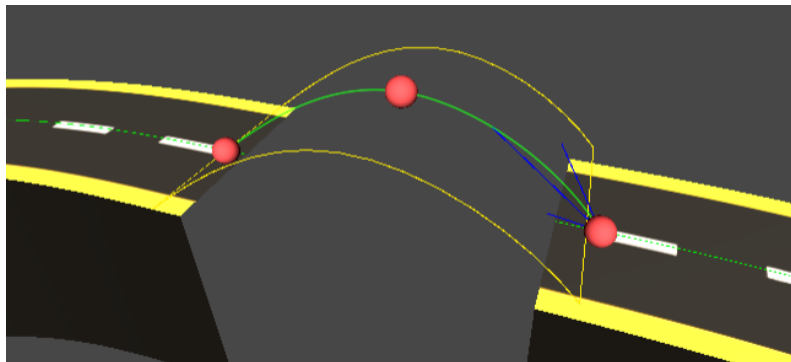
Управление происходит через скрипт `MovementSystem`, в котором есть два публичных метода: `MoveForward` – движение вперед по сплайну, и `SetOffset` – установка горизонтального смещения. Это смещение ограничено шириной дороги. В пакете присутствует скрипт `MouseInput`, демонстрирующий пример ввода для перемещения.

### Стыки дорог (развилки)

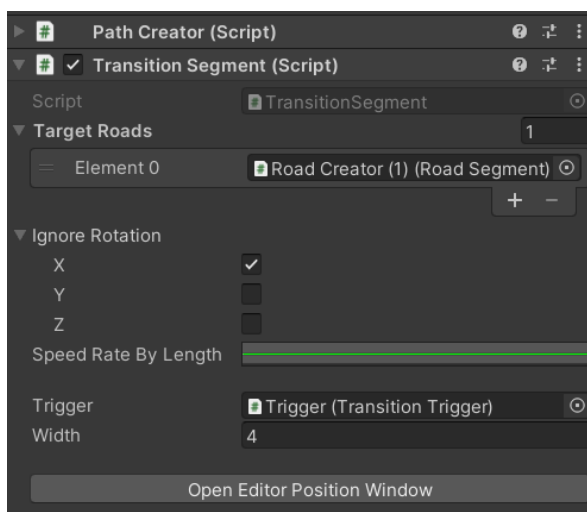


Бывают случаи, когда одна дорога вытекает в несколько (2 или более), и нужно осуществить связь между ними. В скрипте `RoadSegment` в инспекторе можно добавлять в список примыкающие дороги (`TargetRoads`). Когда игрок доходит до конца дороги, по которой он перемещается в данный момент, то он автоматически перемещается на ближайшую дорогу из списка `TargetRoads`. Если список пустой – игрок останавливается.

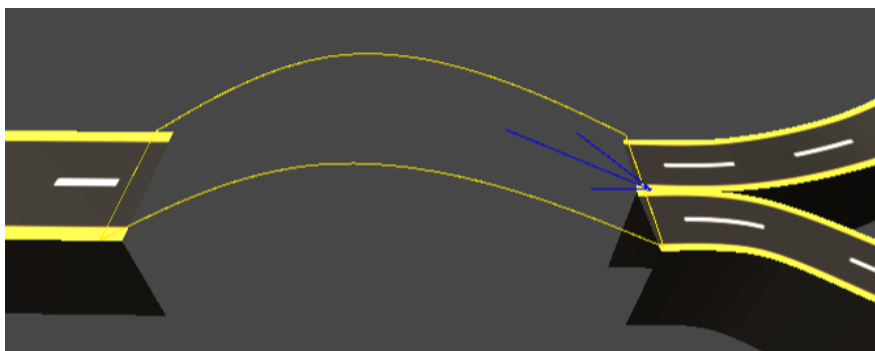
## Переходы между дорогами (прыжки)



Когда игрок попадает в специальный триггер, он начинает переход по заранее заданной траектории. Движение вперед по переходу осуществляется автоматически, а управление горизонтальным смещением как на обычной дороге. Переходы между дорогами задаются отдельными объектами, на которых находится скрипт `TransitionSegment`. Траектория перехода задаётся с помощью `PathCreator`.

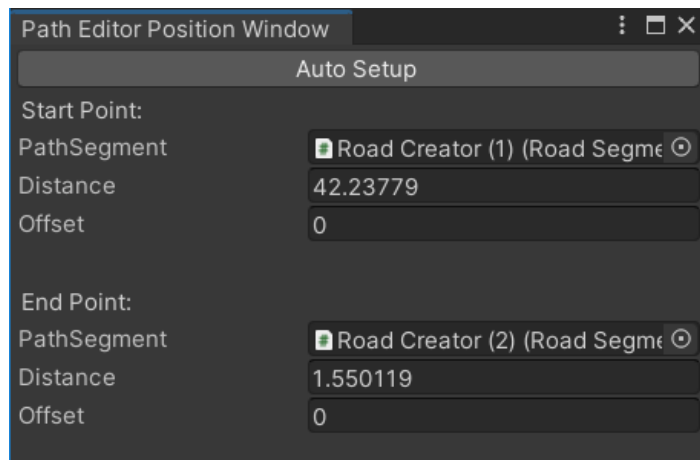


Дорога, на которую игрок должен перейти, задаётся в окне инспектора путём добавления нужной дороги в список `TargetRoads`. Причём возможна ситуация, когда прыжок может привести на несколько дорог. Тогда в списке `TargetRoads` должно быть все дороги, на которые игрок может попасть после окончания перехода.



## Автоматическое соединение перехода со стыкующими дорогами

В окне инспектора на компоненте `TransitionSegment` есть кнопка `OpenEditorPositionWindow`. По нажатии на эту кнопку открывается следующее окно.



Изначально все поля будут пустыми, но после нажатия `AutoSetup` произойдет автоматическая синхронизация начальной и конечной позиции перехода с ближайшими дорогами. В этом окне также можно менять положение и горизонтальное смещение стартовой и конечной точки перехода так, что эти точки будут четко лежать на дороге.