

Programiranje v okolju Unity 3D

Rok Kos

10. januar 2016

Kazalo

1	SET UP	3
1.1	Editor	3
1.2	Plugins	5

Povzetek

1 SET UP

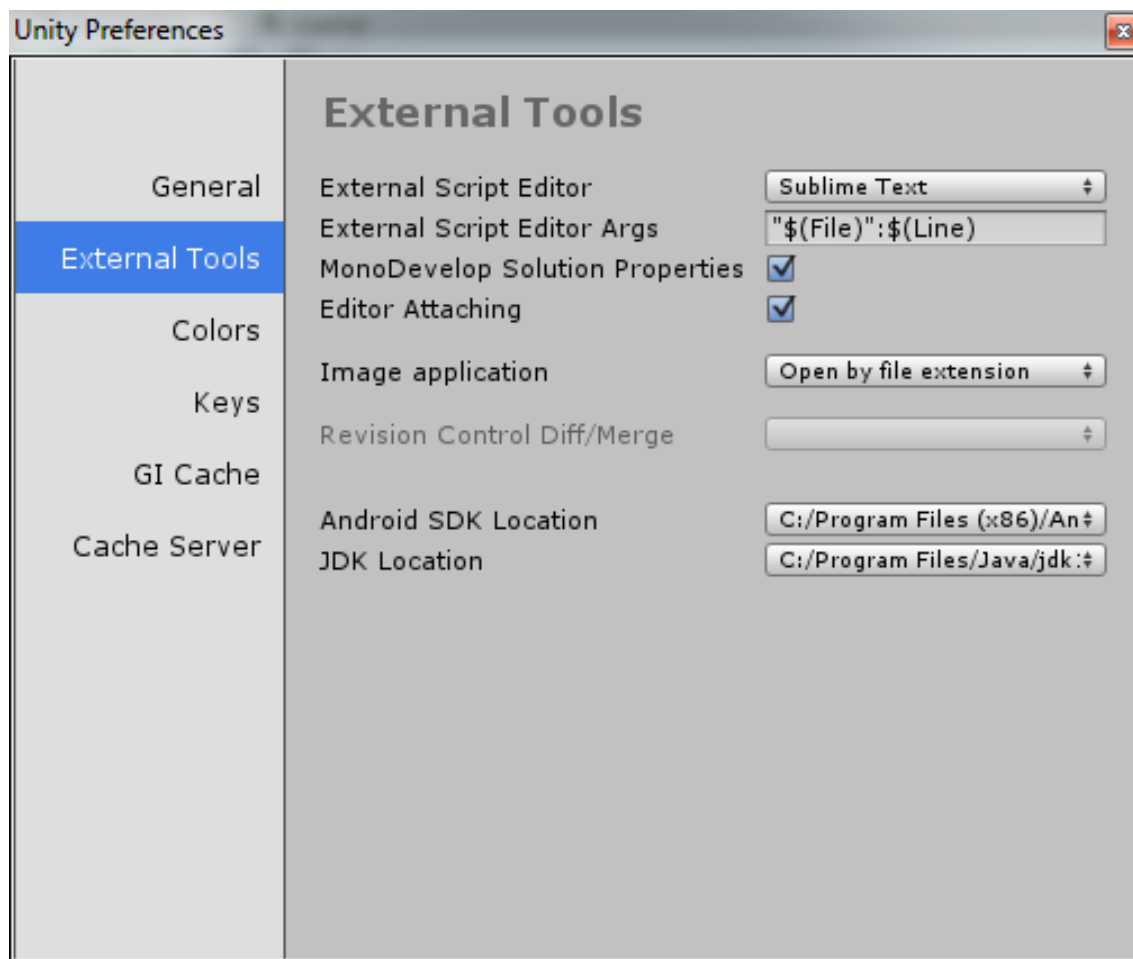
1.1 Editor

Kot vsako razvojno okolje ima tudi Unity 3D svoj privzeti editor. To je MonoDevelop, ki podpira večplatformski razvoj(to pomeni, da lahko isto kodo pišemo za različne operacijske sisteme Linux, Windows, Mac itd.). Podpira naslednje jezika:

- C#
- F#
- VisualBasic
- ...

Ima tudi integrirano dopolnjevanje kode(code auto-completion) in svoj debugger. Sam sem kar nekaj časa uporabljal MonoDevelop in v njemu razvijal, ampak mi pri je pri njem vedno nekaj manjkalo. Zdelo se mi je, da je njihov code auto-completion vsiljeval svoje stvari in da mi highlighter ni vedno obarval kakšnih klasov. Zato sem se odločil, da bom začel uporabljati meni ljubši urejevalnik besedila to je Sublime Text 3. Pri tem urejevalniku sem dobil veliko svobodo pri urejanju, iskanju, popravljanju ter pri samem urejanju urejevalnika. To sublime omogoča s svojim Package Controlom, ki se ga da inštalirati preko te strani: <https://packagecontrol.io/installation>. Preko Package Controla lahko inštaliramo dodatne plugine in snippete za sublime, ki izboljšajo samo delovanje tega. O tej temi bi se dalo še veliko govoriti zato jo bom pustil za drugič. Seveda moramo Unityju povedati, da naj svoje datoteke odpira z sublimom. To naredimo tako:

1. Inštaliramo Sublime Text 3/2 preko te [strani](#)
2. Gremo v **Edit** → **Preferences** → **External Tools**(kot lahko vidimo v spodnji sliki) in spremenimo na[urejevalnik
3. Spremenimo **External Script Editor Args** v "\$(File)":\$(Line) zato, da bo sublime skočil v vrstico, kjer je error
4. Sedaj bi nam moral Unity odpreti sublime, ko dvokliknemo na datoteko, ki je skripta



Slika 1: Sublime Text Set Up

5. V sublimu bomo še spremenili, katere datoteke hočemo videti v projeknem drevesu. To naredimo tako, da gremo v **Project** → **Save Project As** in notri vtipkamo tole kodo:

```
1 {
2     "folders":
3     [
4         {
5             "path": "Assets/Scripts",
6             "file_exclude_patterns":
7             [
8                 "*.dll",
9                 "*.meta"
10            ]
11        }
12    ]
13 }
```

in potem shranimo datoteko. To nam srije vse nepotrebne .meta in

.dll datoteke

6. S Package Controlom na koncu še inštaliramo dodatne snippete, ki nam pomagajo pri auto-completion in barvanju kode. Vse potrebne package najdemo na tej strani http://wiki.unity3d.com/index.php/Using_Sublime_Text_as_a_script_editor, kjer so tudi bolj podrobna navodila za celoten setup Sublime Text-a.

Kot zanimivost pa bi vam rad pokazal, kako enostavno se da narediti snippete v Sublime Text-u, ki se potem sprožijo ob določenem zaporedju tipk in nakoncu tabulatorja.

```
1
2 "folders":
3 [
4     {
5         "path": "Assets/Scripts",
6         "file_exclude_patterns":
7         [
8             "*.dll",
9             "*.meta"
10        ]
11    }
12 ]
13
```

1.2 Plugini

Slike

1	Sublime Text Set Up	4
	[Tec] [Ala14] [Tho15]	

Viri in literatura

- [Ala14] THORN Alan. *Pro Unity Game Development with C#*. Vol. 1. izdaja. New York: Apress, 2014. ISBN: 9781430267461.
- [Tho15] FINNEGAN Thomas. *Learning Unity Android Game Development*. Vol. 1. izdaja. Birmingham: Packt Publishing, 2015. ISBN: 9781784394691.
- [Tec] Unity Technologies. *Unity Manual*. URL: <http://docs.unity3d.com/Manual/index.html> (visited on 2015).