Basic Tutorial for *ConText – A Text/Choice Adventure Game Framework*

This tutorial will guide you through the interface of the framework and in the process you will create a basic text adventure game.

Part 1 – Setup

1. Before anything else, you will need to download and install Unity Personal Edition from here: https://unity3d.com/get-unity/download?ref=personal
   Unity is the game engine this framework was developed in and is to be used in. The first time you start Unity, you will need to create/login with an account. This account is necessary to verify your license information. The account can be created and used free of charge.
2. Once you have installed Unity, you will need to either use the ConText package (ConText_UserStudy1.unitypackage) provided in your initial invitation email or download it from here: https://www.dropbox.com/sh/ad48e0u7hz3583b/AABUAKveQAnkVYoRA5sBqlAqa?dl=0. You may download the entire folder which contains the UnityPackage, a *'ConText Project.wlt'* file as well as all PDFs also attached to the email.
3. The *'ConText Project.wlt'* file needs to be copied to the Unity Editor Layout folder. In Windows, that is **%APPDATA%\Unity\Editor-5.x\Preferences\Layouts\**. In Mac OS, it is **~/Library/Preferences/Unity/Editor-5.x/Layouts**. (Simply paste these directory paths to Explorer/Finder to access the respective folder)
4. Start Unity, log in with your account and select **NEW**. On the following screen, enter a name for your project and select **2D**.

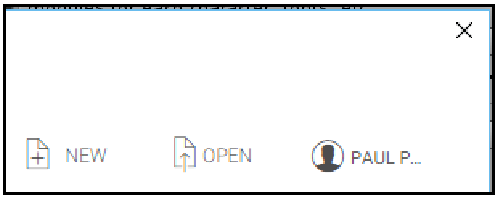
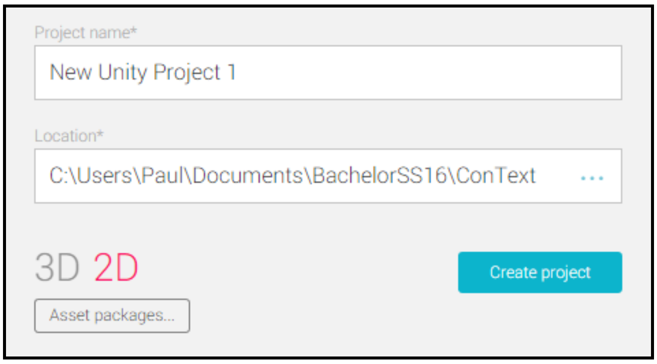


Figure 1: NEW


Project name*

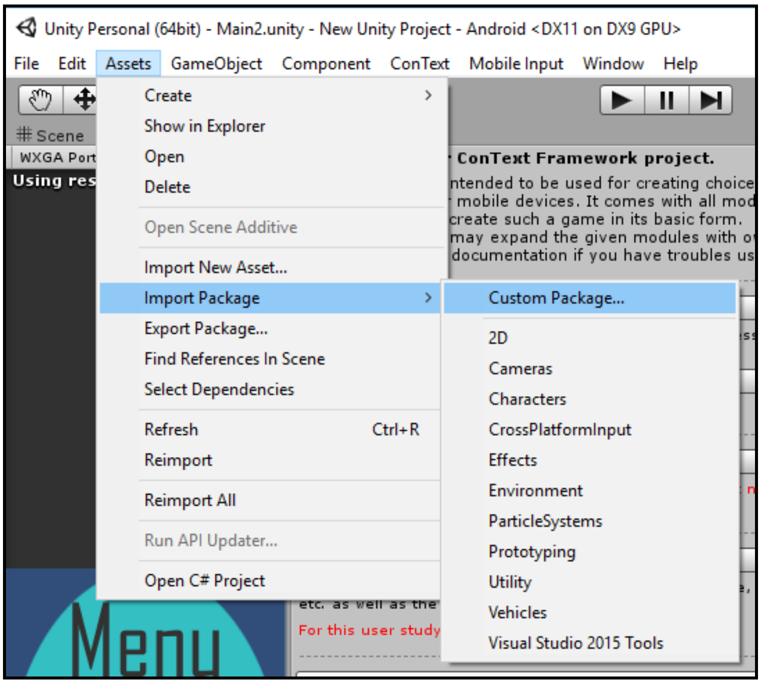
New Unity Project 1

Location*

C:\Users\Paul\Documents\BachelorSS16\ConText

Figure 2: Create

5. Import the UnityPackage into Unity through **Assets – Import Package – Custom Package...** . A window will pop up listing all items that can be imported from that package. At the bottom of that window, select **All** and then click **Import**.



ConText Framework project.
ntended to be used for creating choice
mobile devices. It comes with all mod
create such a game in its basic form.
may expand the given modules with o
documentation if you have troubles us
etc. as well as the
For this user study

Figure 3: Import package

6. Almost done setting up the framework. In the upper right, select **ConText Project** as the Layout.

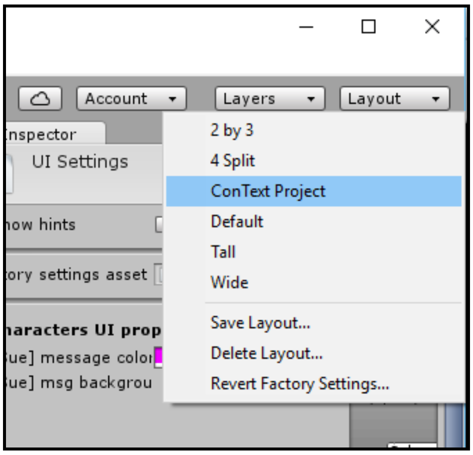


Figure 4: Layout

7. In the Project/Asset folder window, select the top most folder **Asset** and double click **Main** to open the main Unity scene.

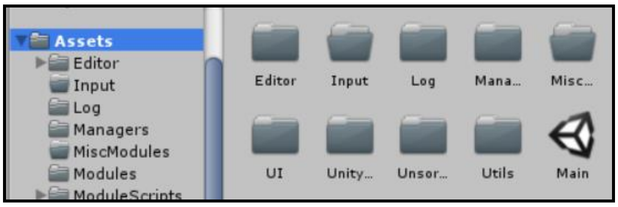
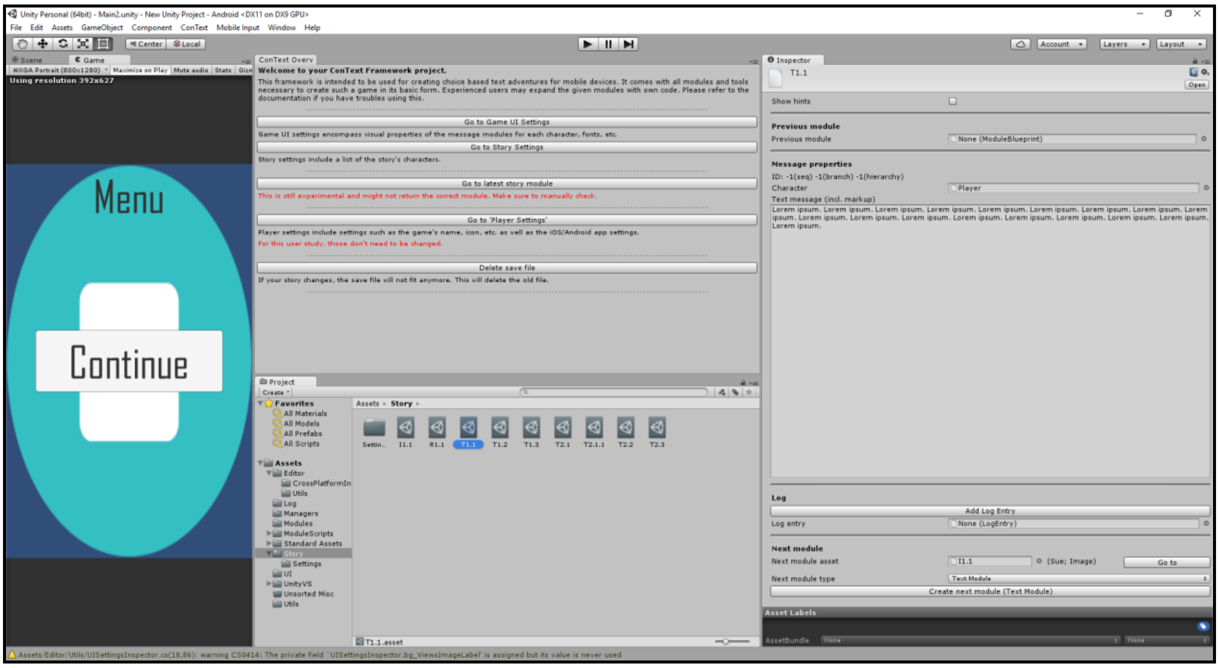


Figure 5: Open Main

Now your Unity editor should look like this:



Unity Personal (64bit) - Main2.unity - New Unity Project - Android <DX11 on DX9 GPU>

Using resolution 392x627

Welcome to your ConText Framework project.

This framework is intended to be used for creating choice based text adventures for mobile devices. It comes with all modules and tools necessary to create such a game in its basic form. Experienced users may expand the given modules with own code. Please refer to the documentation if you have troubles using this.

Go to Game UI Settings

Game UI settings encompass visual properties of the message modules for each character, fonts, etc.

Go to Story Settings

Story settings include a list of the story's characters.

Go to latest story module

This is still experimental and might not return the correct module. Make sure to manually check.

Go to 'Player Settings'

Player settings include settings such as the game's name, icon, etc. as well as the iOS/Android app settings.

For this user study, those don't need to be changed.

Delete save file

If your story changes, the save file will not fit anymore. This will delete the old file.

T1.1

Show hints □

Previous module

Previous module: None (ModuleBlueprint)

Message properties

ID: -1(seq) -1(branch) -1(hierarchy)

Character: Player

Text message (incl. markup)

Lorem ipsum. Lorem ipsum. Lorem ipsum. Lorem ipsum. Lorem ipsum. Lorem ipsum. Lorem ipsum. Lorem ipsum. Lorem ipsum. Lorem ipsum. Lorem ipsum. Lorem ipsum. Lorem ipsum. Lorem ipsum. Lorem ipsum. Lorem ipsum. Lorem ipsum.

Log

Add Log Entry

Log entry: None (LogEntry)

Next module

Next module asset: I1.1 (Sue; Image) Go to

Next module type: Text Module

Create next module (Text Module)

T1.1.asset

Assets/Editor/Utils/UISettingsInspector.cs(18,86): warning CS0414: The private field `UISettingsInspector.bg_ViewsImageLabel' is assigned but its value is never used

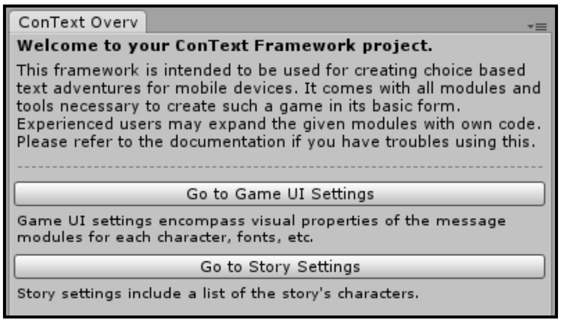
Figure 6: Overview

If so, the last thing to do is deselect **Maximize on Play** at the top of the left Game preview window, then you are done setting up the framework and can continue with Part 2 of the tutorial. If not, make sure you did all steps as described. If you are sure you did not miss a step and the overview is still wrong, feel free to contact me at paul@preissner-muc.de.

Part 2 – Sample game

In this part you will use the framework to create a simple text adventure game with a few different modules with the framework.

1. Given the layout as in Figure 5, the left part shows your *game preview window*. When you test/play your creation, it will be displayed in that window.
   The center part has the *ConText Overview* window in the top half, intended as a sort of hub from which you can go straight to the most important settings and windows. The bottom half is the *Project window*, which displays the directory structure of your asset folders, through which you may directly access story modules, settings files or ultimately script files.
   The right part is the *Inspector window*, which will always display the properties/details of the currently selected asset.
2. To get started, you will create the characters of your story. For this, click **Go to Story Settings** in the *ConText Overview* window. This will open the Story Settings asset in the Inspector window. There, click **Add a character**, which will create a new Character asset and automatically display its details in the Inspector window. You may then give it a name and a unique character ID. Add as many characters as you need.



ConText Overv

Welcome to your ConText Framework project.

This framework is intended to be used for creating choice based text adventures for mobile devices. It comes with all modules and tools necessary to create such a game in its basic form. Experienced users may expand the given modules with own code. Please refer to the documentation if you have troubles using this.

Go to Game UI Settings

Game UI settings encompass visual properties of the message modules for each character, fonts, etc.

Go to Story Settings

Story settings include a list of the story's characters.

Figure 7: Go to story settings

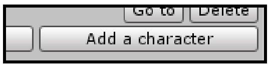
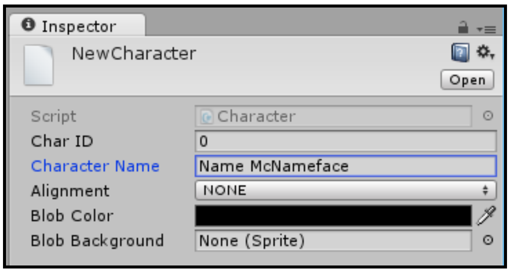


Figure 8: Add character



Inspector

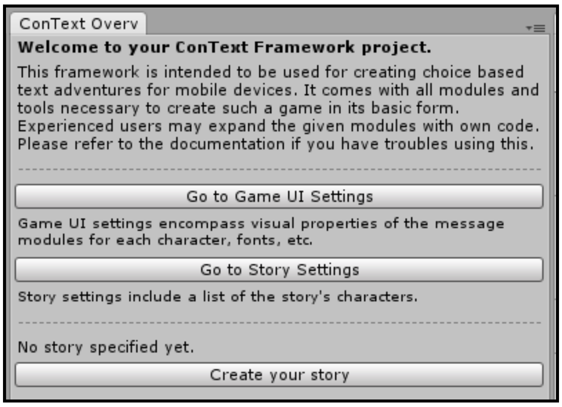
NewCharacter

Open

Script: Character
Char ID: 0
Character Name: Name McNameface
Alignment: NONE
Blob Color
Blob Background: None (Sprite)

Figure 9: Customize character

3. Next, you will create the start of your story. For this, click **Create your story** in the ConText Overview. This will create the first node of your story, which (as of now) will be a Text module. The now created module will be displayed in the Inspector window. There you will need to choose the Character this message should be sent by by clicking on the small circle next to the Character field. Right below that is a text field which will hold the text this message consists of.



ConText Overv

Welcome to your ConText Framework project.

This framework is intended to be used for creating choice based text adventures for mobile devices. It comes with all modules and tools necessary to create such a game in its basic form. Experienced users may expand the given modules with own code. Please refer to the documentation if you have troubles using this.

Go to Game UI Settings

Game UI settings encompass visual properties of the message modules for each character, fonts, etc.

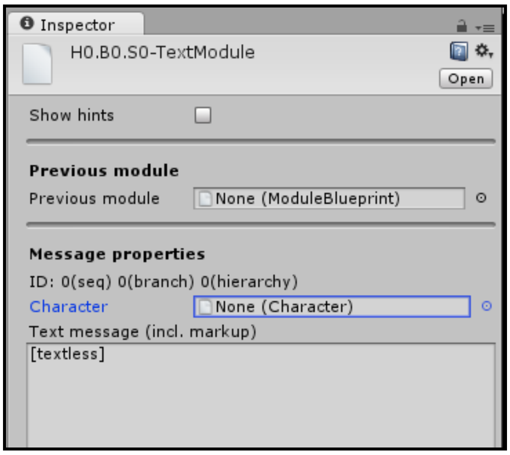
Go to Story Settings

Story settings include a list of the story's characters.

No story specified yet.

Create your story

Figure 10: Create story



Inspector

H0.B0.S0-TextModule

Show hints □

Previous module

Previous module: None (ModuleBlueprint)

Message properties

ID: 0(seq) 0(branch) 0(hierarchy)

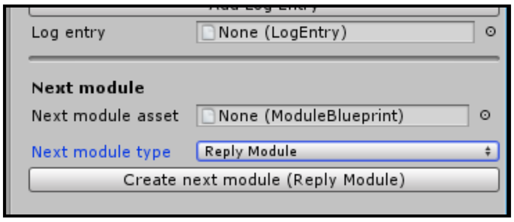
Character: None (Character)

Text message (incl. markup)

[textless]

Figure 11: Message

4. Now your story consists of one message. Within each module, you can create the next module to follow after it (sometimes given a certain condition) through the *Next module* segment in its Inspector (or Replies for the Reply Module). For this tutorial, select the **module type as "Reply Module"** and click **Create next module**. This Create button will always reflect what type will be created.



Log entry: None (LogEntry)

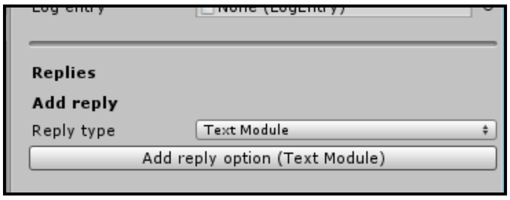
Next module

Next module asset: None (ModuleBlueprint)

Next module type: Reply Module

Figure 12: Next

5. This will have put that Reply module into focus in the Inspector. A specialty about the Reply module is that after displaying its text, it will not automatically send the next message, but leave the player with the respective number of replies to choose from. Each reply may lead to a different branch in the story. For this specific module in the tutorial, set up Character and Text as in step 3., then scroll down to the Replies segment. It works similarly to the Next module segment of the previous module, with the difference being you can add multiple entries. **Add a reply**.



Log entry: None (LogEntry)

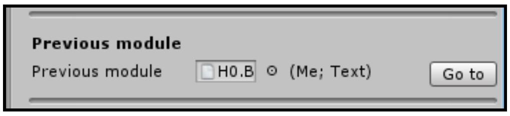
Replies

Add reply

Reply type: Text Module

Figure 13: Replies

6. The newly created module will be in focus then. For the sake of this tutorial, look at the *Previous module* segment at the top of the Inspector. If the previous module is specified, it will have a **Go to** button next to it. Clicking it will naturally take you to that previous module, in this case the Reply module from step 5. There you may change the default description text of each reply, which is so far "new reply x". You may have seen by now that each previous or next module has that **Go to** button, which as of now is a simple way to traverse your story modules.



Previous module

Previous module H0.B ⊙ (Me; Text) Go to

Figure 14: Go to

7. With this, you should know the basic process of creating modules. If you want to manually reconnect nodes in a different way, you can always click the little circle next to one and choose a different existing module. Be aware though, this can easily screw up the structure of the story and might lead to flawed playback.
8. At the bottom of each module's Inspector is a *Delete module* button, which will delete that module and attempt to properly connect the surrounding modules. Be aware, however, that when dealing with modules that have multiple next modules (like Reply modules), this may lose the connection to other branches.
9. As the last step of this tutorial, click on **Go to Game UI Settings** in the ConText Overview. This will take you to the visual properties of the messages and modules, where you might change color, font, font size and more.

Now that you have set up the framework and created your first basic game with it, click on the **play button** at the top center of the tool and check out what it looks like in the preview window.

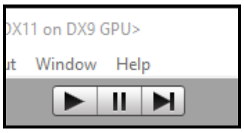
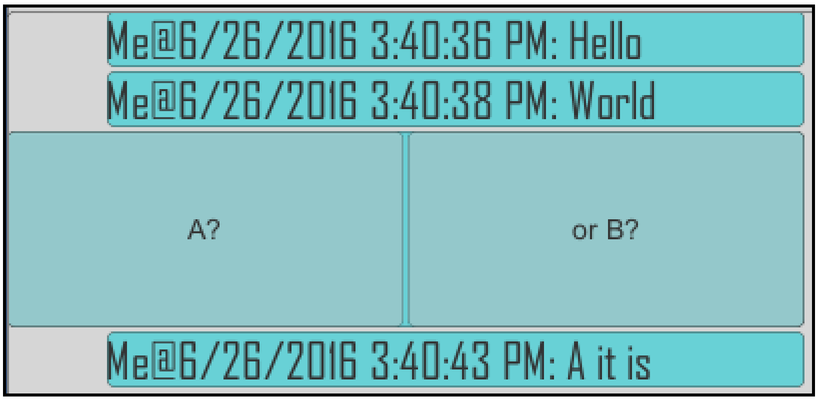


Figure 15: Play!



Me@6/26/2016 3:40:36 PM: Hello

Me@6/26/2016 3:40:38 PM: World

A? or B?

Me@6/26/2016 3:40:43 PM: A it is

Figure 16: Small beginnings

Additional info about importing files into Unity:

It may likely happen that you want to import custom image files to use as background or in an image module or perhaps different fonts.
Files can generally imported into Unity by simply dragging and dropping them into the Project/Asset folder window. So you can just right click into any folder in that Project window, select Create – Folder to add a separate folder for your files, then drag and drop them in there.

In the case of image files, these will by default be imported as Sprites given you project was set to 2D upon creation. If that is not the case and the image is not available for selection as such an image, select the asset and in its inspector, set Texture Type to "Sprite".