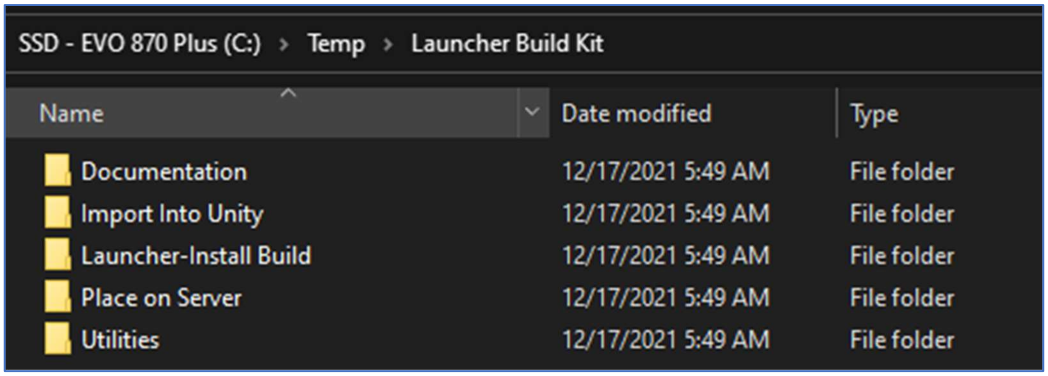
Atavism Game Launcher for Windows Setup Guide

Doc Version:
22.3.27.8

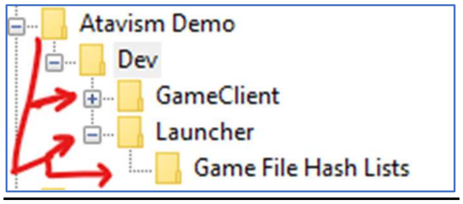


SSD - EVO 870 Plus (C:) › Temp › Launcher Build Kit

Name	Date modified	Type
Documentation	12/17/2021 5:49 AM	File folder
Import Into Unity	12/17/2021 5:49 AM	File folder
Launcher-Install Build	12/17/2021 5:49 AM	File folder
Place on Server	12/17/2021 5:49 AM	File folder
Utilities	12/17/2021 5:49 AM	File folder

Note: The online documentation at https://blog.visuafusion.com/AtavismGameLauncherforWindows may be more up to date than the document files included in the package.

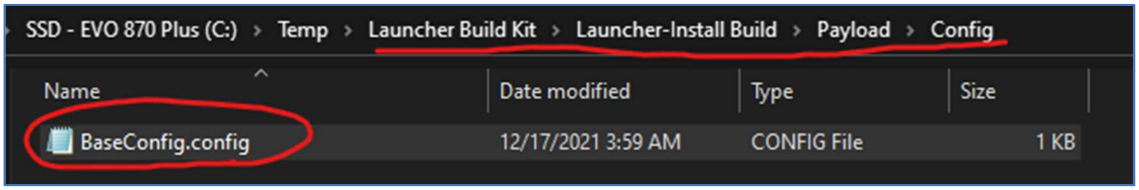
1. **Set Up your Webserver to host the Launcher Solution**
   a. **Prepare web server file hosting directories**
   You will need to configure a game files directory with both "GameClient" and "Launcher" subdirectories on your web server. The Launcher Directory will need a "Game File Hash Lists" subdirectory within it as well. I'd recommend setting this all up under a "<Game Name> -> <Release Branch>" structure, as shown here:



   Note: Other than your game's name and release branch name, the last 3 directory names must match exactly, and the structure must be set up as shown above

   b. **Note down your Web Server's paths Newly create paths from step (a)**
      i. **Launcher Download Root URL** (Launcher Directory from step A) – Example: https://ArcanicaEntertainment.com/Downloads/GameFiles/Atavism Demo/Dev/Launcher
      ii. **Client Download Root URL** (Launcher Directory from step A) – Example: https://ArcanicaEntertainment.com/Downloads/GameFiles/Atavism Demo/Dev/GameClient

2. **Configuring the Launcher Configuration Files and Supporting Texts**
   a. **BaseConfig.config** – This file is the first initial config file that tells the launcher where to go and what to be called. Open Launcher Build Kit -> Launcher-Install Build -> Payload -> Config -> BaseConfig.config with notepad, and configure the following values:



SSD - EVO 870 Plus (C:) › Temp › Launcher Build Kit › Launcher-Install Build › Payload › Config

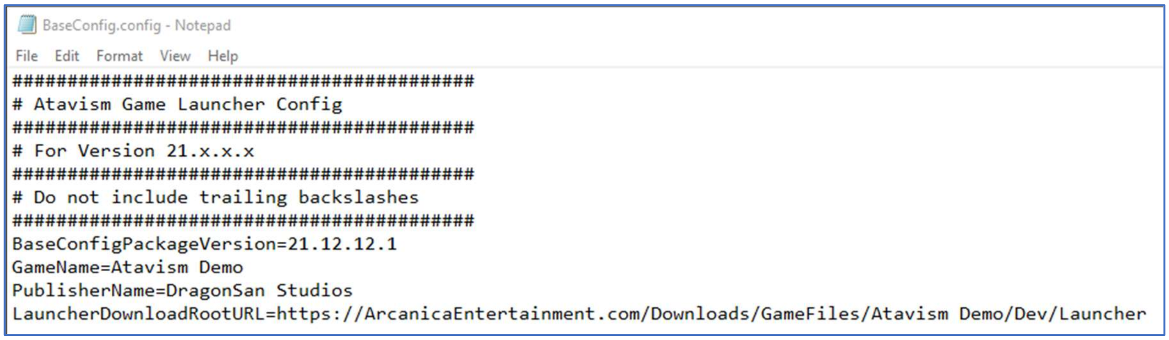
Name	Date modified	Type	Size
BaseConfig.config	12/17/2021 3:59 AM	CONFIG File	1 KB

i. **BaseConfigPackageVersion**: version in x.x.x.x format (YY.MM.DD.<revision> format works great here)

ii. **GameName**: The name of your game (Spaces are allowed in the value)

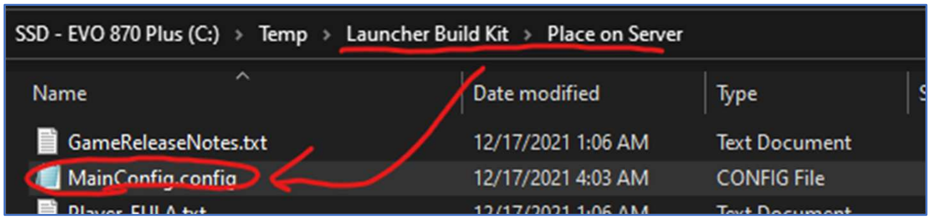
iii. **PublisherName**: Your company name (Spaces are allowed in the value)

iv. **LauncherDownloadRootURL**: The Launcher Download Root URL you wrote down in Section 1, step b.



```
###########################################
# Atavism Game Launcher Config
###########################################
# For Version 21.x.x.x
###########################################
# Do not include trailing backslashes
###########################################
BaseConfigPackageVersion=21.12.12.1
GameName=Atavism Demo
PublisherName=DragonSan Studios
LauncherDownloadRootURL=https://ArcanicaEntertainment.com/Downloads/GameFiles/Atavism Demo/Dev/Launcher
```

b. **<u>MainConfig.config</u>** – This file is the main configuration file that houses most of the values the launcher will use (color scheme, publisher contact info, Button URLs, Game EXE name, etc. Most importantly, this file downloads from your web server each time the launcher starts up, and the package version value in this config file is compared to the package version value in the local BaseConfig.config file – if the BaseConfig.config's package version value is less than the newly downloaded MainConfig.config's version value, a launcher update is triggered (this process is covered later in this document). Open Launch Build Kit -> Pace on Server -> MainConfig.config in Notepad, and configure the following values:



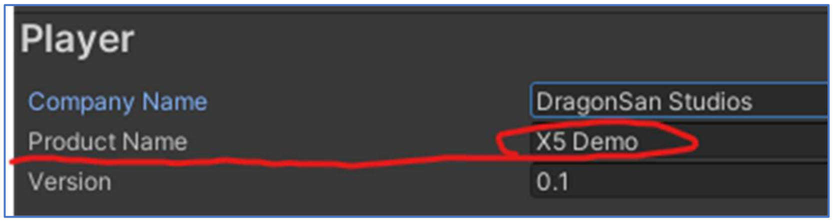
SSD - EVO 870 Plus (C:) › Temp › Launcher Build Kit › Place on Server

Name	Date modified	Type
GameReleaseNotes.txt	12/17/2021 1:06 AM	Text Document
MainConfig.config	12/17/2021 4:03 AM	CONFIG File
Player EULA.txt	12/17/2021 1:06 AM	Text Document

i. **CurrentConfigPackageVersion**: version in x.x.x.x format (This will typically match the version you set up in BaseConfig.config if this is your first initial time setting up the launcher solution)

ii. **ClientDownloadRootURL**: The Client Download Root URL you wrote down in Section 1, step b.

iii. **GameEXEName**: The name of your games executable file once built, this can be found/set in Unity by going to Edit -> Project Settings and looking at the Product Name – whatever value is here is what the built EXE will be named (in the example shown, the EXE name will be "X5 Demo.exe"



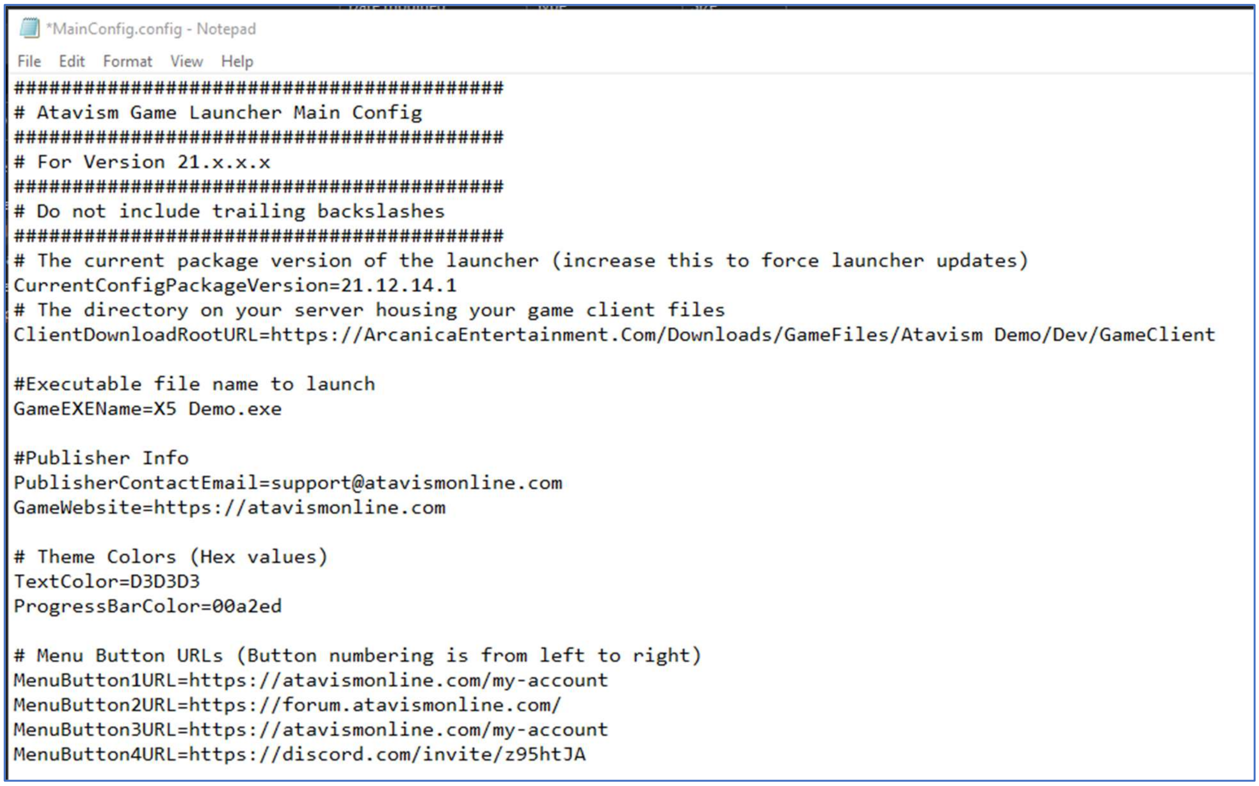
Player

Company Name	DragonSan Studios
Product Name	X5 Demo
Version	0.1

iv. **PublisherContactEmail**: The email you would like people to contact you at for support (IE: support@atavismonline.com)

v. **GameWebsite**: The main website for your game or company (this is used in add/remove programs listing only)

vi. **MenuButton(1-4)URL**: Set up URLs you'd like users to have buttons for here. Buttons are ordered from left to right (1 is the farthest left button on the top of the launcher)



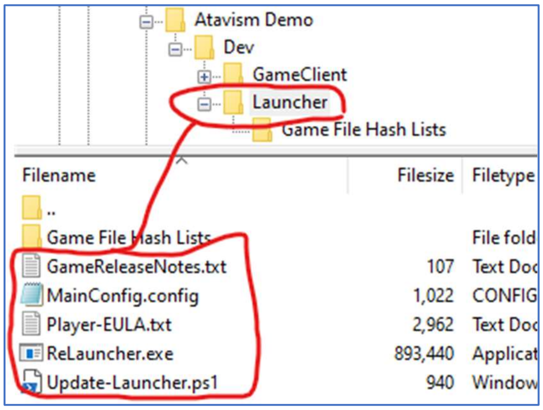
```
##########################################
# Atavism Game Launcher Main Config
##########################################
# For Version 21.x.x.x
##########################################
# Do not include trailing backslashes
##########################################
# The current package version of the launcher (increase this to force launcher updates)
CurrentConfigPackageVersion=21.12.14.1
# The directory on your server housing your game client files
ClientDownloadRootURL=https://ArcanicaEntertainment.Com/Downloads/GameFiles/Atavism Demo/Dev/GameClient

#Executable file name to launch
GameEXEName=X5 Demo.exe

#Publisher Info
PublisherContactEmail=support@atavismonline.com
GameWebsite=https://atavismonline.com

# Theme Colors (Hex values)
TextColor=D3D3D3
ProgressBarColor=00a2ed

# Menu Button URLs (Button numbering is from left to right)
MenuButton1URL=https://atavismonline.com/my-account
MenuButton2URL=https://forum.atavismonline.com/
MenuButton3URL=https://atavismonline.com/my-account
MenuButton4URL=https://discord.com/invite/z95htJA
```

c. **GameReleaseNotes.txt** – edit this document in notepad, include your release notes for your game. This will be something you update as you upload new versions of your game (This will be covered more farther along in this document).

d. **Player-EULA.txt** – edit this document in notepad, include your end user license agreement. This will be displayed to users both during launcher install as well as when choosing a game client location before initially downloading/installing the game from the launcher itself.

3. Upload Initial Server-Side Files

a. Upload the files in the “Launcher Build Kit\Place on Server” directory to your web server’s “Launcher” directory (The Launcher Download Root URL you wrote down in Section 1, step b.)

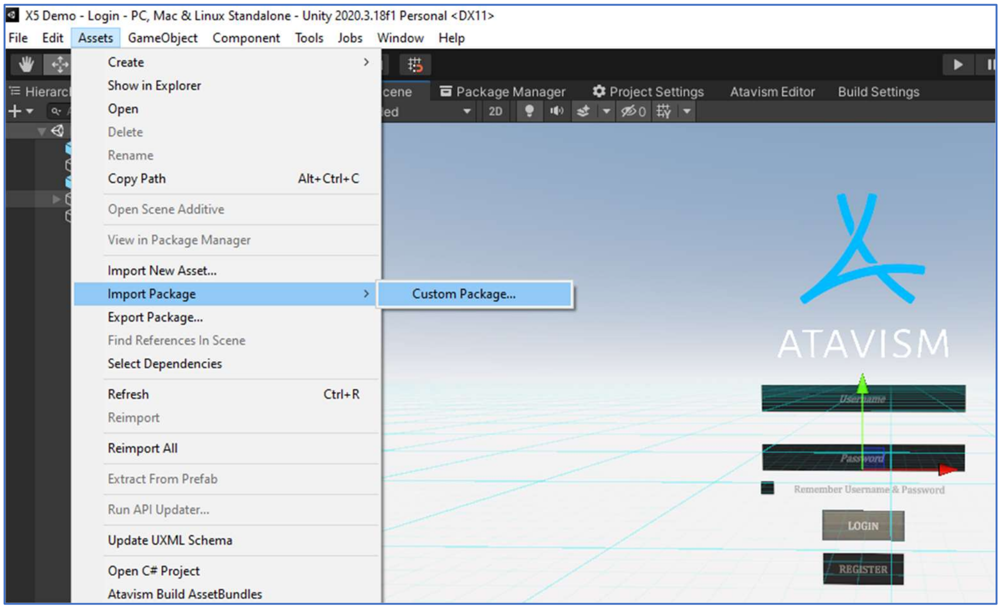


Filename	Filesize	Filetype
..		
Game File Hash Lists		File fold
GameReleaseNotes.txt	107	Text Doc
MainConfig.config	1,022	CONFIG
Player-EULA.txt	2,962	Text Doc
ReLauncher.exe	893,440	Applicat
Update-Launcher.ps1	940	Window

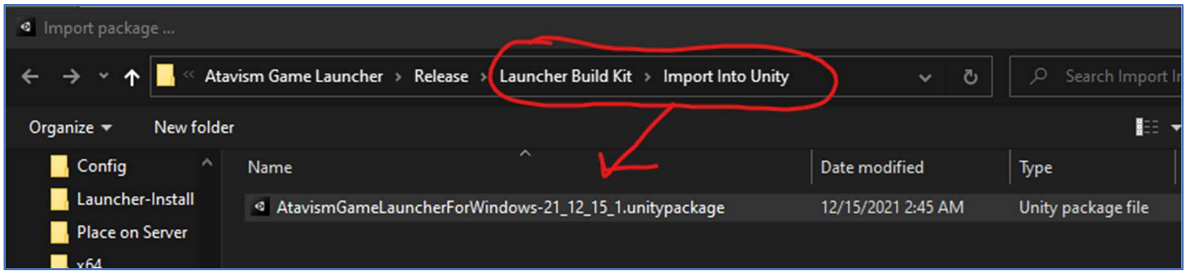
4. Adding Launcher integration to your Atavism Game

a. Import the Unity Package

With your Atavism Project loaded in Unity, Import the AtavismGameLauncherForWindows-XX.XX.XX.XX.unitypackage Package into your unity project by clicking on Assets -> Import Package -> Custom Package.



X5 Demo - Login - PC, Mac & Linux Standalone - Unity 2020.3.18f1 Personal <DX11>



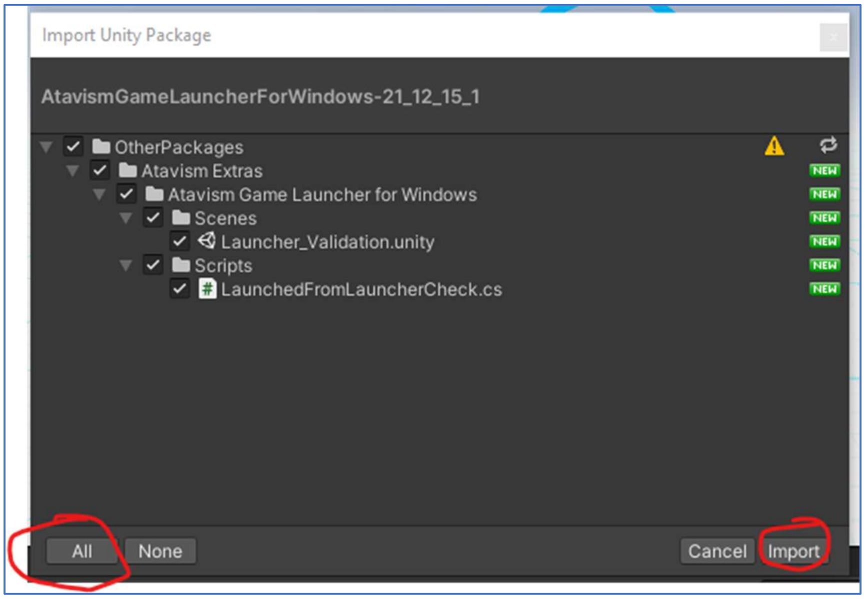
Import package ...

Atavism Game Launcher › Release › Launcher Build Kit › Import Into Unity

Name	Date modified	Type
AtavismGameLauncherForWindows-21_12_15_1.unitypackage	12/15/2021 2:45 AM	Unity package file

b. Select Items for Import

Import all the items in the package into your project by making sure everything is checked and clicking “import”.



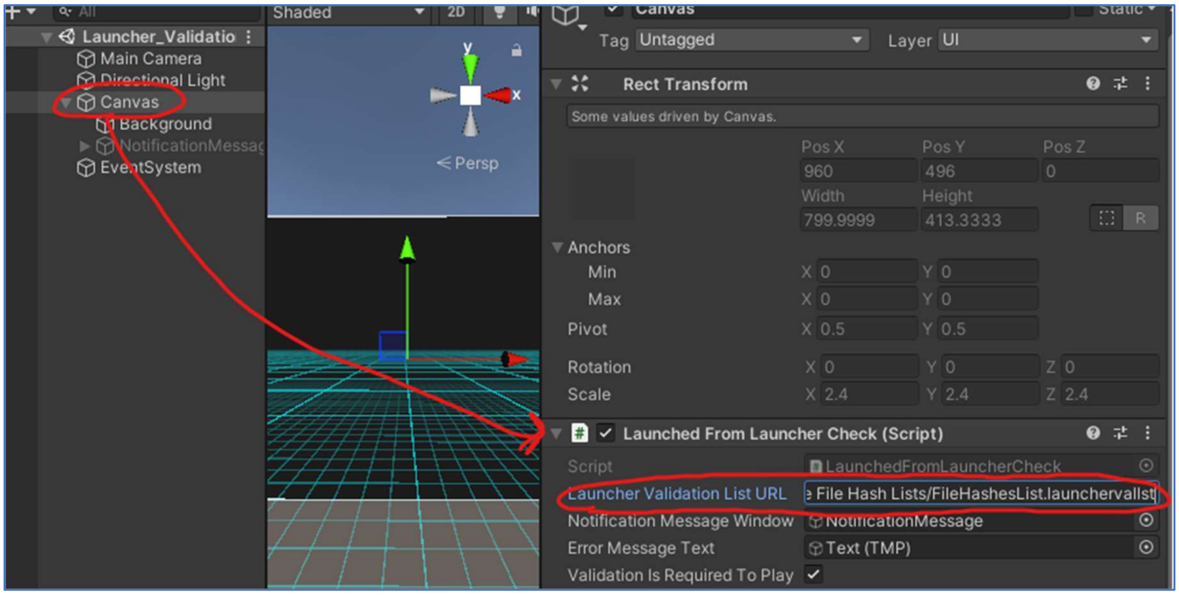
Import Unity Package

AtavismGameLauncherForWindows-21_12_15_1

- ☑ OtherPackages
  - ☑ Atavism Extras NEW
    - ☑ Atavism Game Launcher for Windows NEW
      - ☑ Scenes NEW
        - ☑ Launcher_Validation.unity NEW
      - ☑ Scripts NEW
        - ☑ LaunchedFromLauncherCheck.cs NEW

c. Configure the “Launcher_Validation” scene to use your web server

The script attached to the Launcher_Validation scene’s canvas object needs the Launcher Validation List URL we set up earlier. Include either http:// or https:// as well as “FileHashesList.launchervallst” in the path, this is used to directly download the list.



Launcher_Validatio
- Main Camera
- Directional Light
- Canvas
  - Background
  - NotificationMessag
- EventSystem

Canvas

Tag: Untagged | Layer: UI

Rect Transform

Some values driven by Canvas.

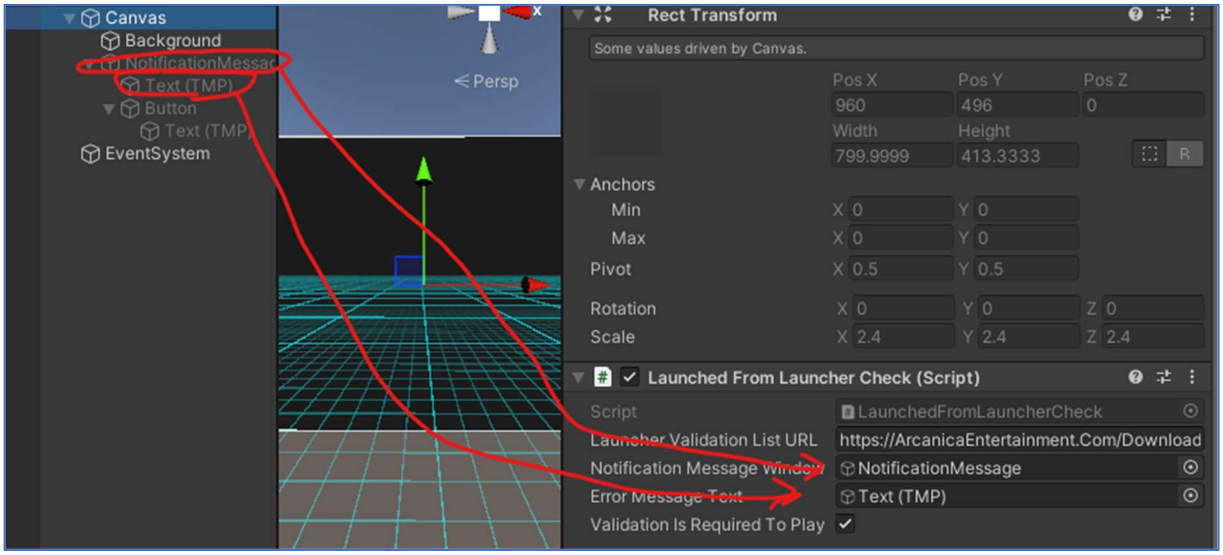
	X	Y	Z
Pos	960	496	0
Width / Height	799.9999	413.3333	
Anchors Min	0	0	
Anchors Max	0	0	
Pivot	0.5	0.5	
Rotation	0	0	0
Scale	2.4	2.4	2.4

Launched From Launcher Check (Script)

Script	LaunchedFromLauncherCheck
Launcher Validation List URL	File Hash Lists/FileHashesList.launchervallst
Notification Message Window	NotificationMessage
Error Message Text	Text (TMP)
Validation Is Required To Play	✓

d. Ensure the UI objects are assigned in Launcher check script

Make sure the Notification Message Window and Notification Message Text objects are assigned in the Launched From Launcher Check script attached the Canvas object



Rect Transform

Some values driven by Canvas.

	Pos X	Pos Y	Pos Z
	960	496	0
	Width	Height	
	799.9999	413.3333	
Anchors			
Min	X 0	Y 0	
Max	X 0	Y 0	
Pivot	X 0.5	Y 0.5	
Rotation	X 0	Y 0	Z 0
Scale	X 2.4	Y 2.4	Z 2.4

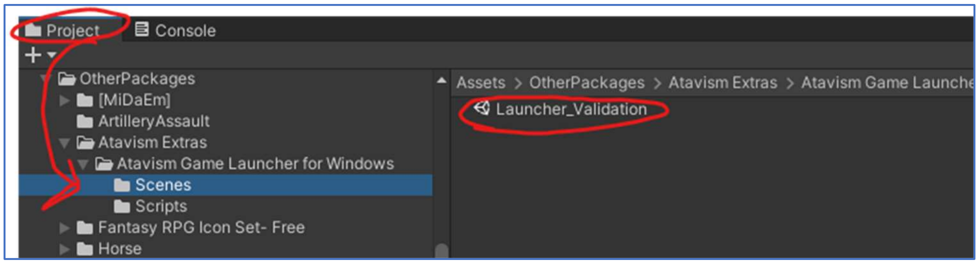
✓ Launched From Launcher Check (Script)

Script	LaunchedFromLauncherCheck
Launcher Validation List URL	https://ArcanicaEntertainment.Com/Download
Notification Message Window	NotificationMessage
Error Message Text	Text (TMP)
Validation Is Required To Play	✓

Feel free to customize the UI of this scene to match your game, just make sure everything above is assigned as shown when you are finished. You may also optionally uncheck the "Validation Required to Play" checkbox, which will allow players to continue to the login scene upon failing validation (Usually, this is not wanted – you don't need to do this to enable play from the editor, the validation script already skips checking if it detects it is running from the editor)

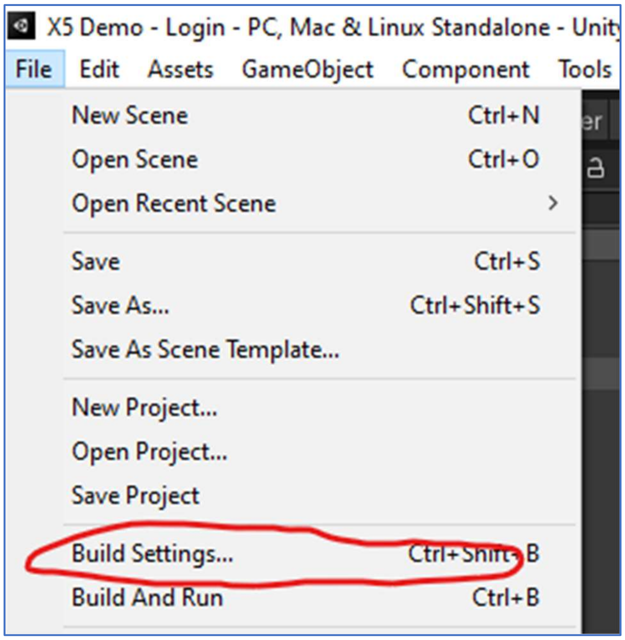
e. **<u>Include the "Launcher_Validation" scene in your build</u>**

The package includes a "Launcher_Validation" scene which needs to be included and set as the 1st scene in your build order. To do this Open the Launcher_Validation scene in Project -> Assets -> OtherPackages -> Atavism Extras -> Atavism Game Launcher for Windows -> Launcher_Validation and double click the scene to open it.

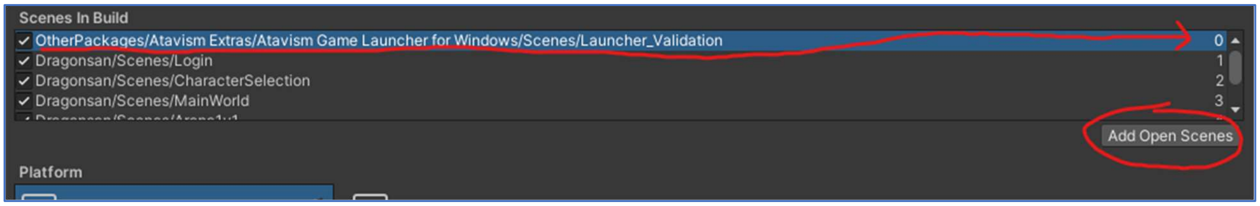

Project
Console
+
OtherPackages
[MiDaEm]
ArtilleryAssault
Atavism Extras
Atavism Game Launcher for Windows
Scenes
Scripts
Fantasy RPG Icon Set- Free
Horse
Assets > OtherPackages > Atavism Extras > Atavism Game Launche
Launcher_Validation

With the Launcher_Validation Scene Open, go to File -> Build Settings and click on "Add Open

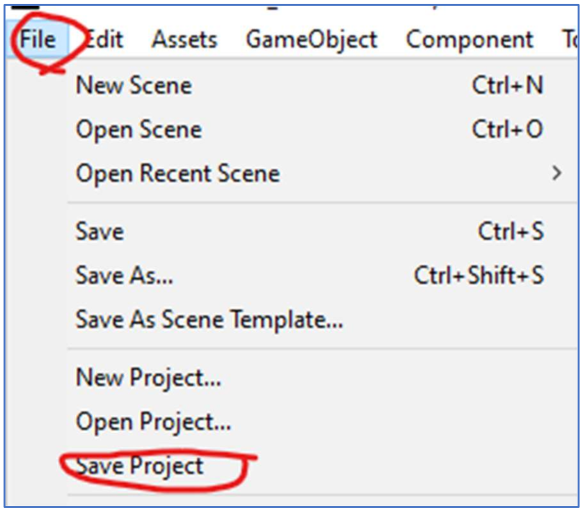
Scenes" under the Scenes in Build list.



f. **<u>Set the "Launcher_Validation Scene as the first scene in your build order</u>**
The newly added scene will be at the bottom of the list at first, you will need to drag it to the top until it is in position 0. Don't forget to Save your project at this point. *It is important that the launcher validation scene is first so that it can verify players have launched the game with the launcher, and that the game is up to date before moving to the login scene!*


Scenes In Build

✓	OtherPackages/Atavism Extras/Atavism Game Launcher for Windows/Scenes/Launcher_Validation	0
✓	Dragonsan/Scenes/Login	1
✓	Dragonsan/Scenes/CharacterSelection	2
✓	Dragonsan/Scenes/MainWorld	3



5. Customize your Launcher to match your game

This step is optional, and you can come back to it later if you just want to get up and running.

If you do come back to this later (or want to update your launcher theme/package), you will need to:

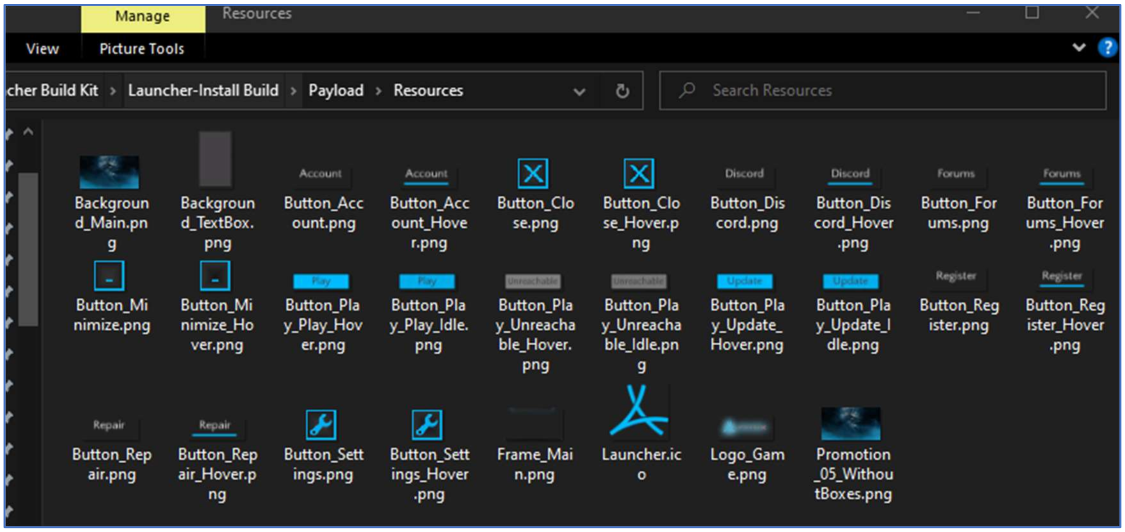
1. update the "**BaseConfigPackageVersion**" setting in "BaseConfig.config" with a higher package version number (this gets compressed and installed with the new build version in the Launcher-Installer Package create in section 6)
2. Update the "CurrentConfigPackageVersion" setting in "MainConfig.config" to the same new version number you placed in "BaseConfig.config" (which you will then need to reupload to your web server's "Launcher" directory (The Launcher Download Root URL you wrote down in Section 1, step b.)
3. Perform this section as well as the "Create a Launcher-Install Package", "" section again

- Doing these things will trigger older launcher builds to see a newer launcher build version on your web server and download it and update, at which point the versions will match and the installed launcher will be the "current" one, along with any customizations you made following this section.

4. If Keeping your existing configuration files through updating to a new launcher version, make sure to check for new configuration values by checking the provided example config files included with the new launcher version!

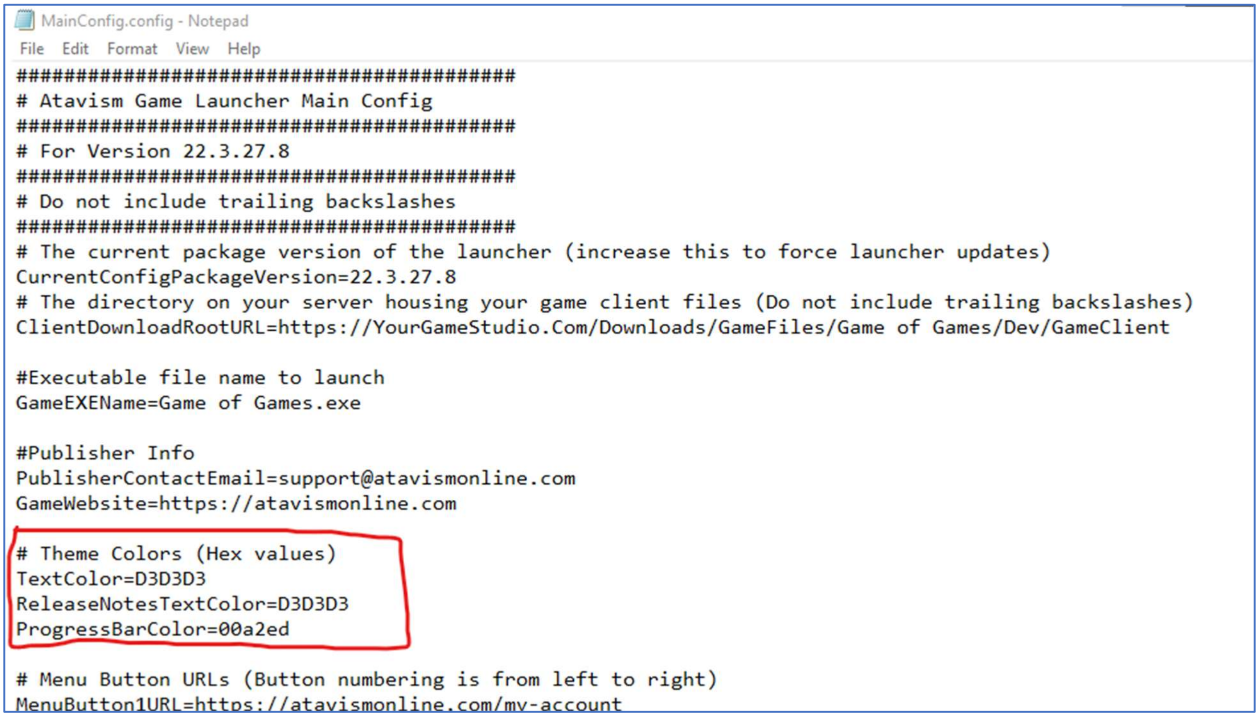
-- Customization Process --

a. **Background, Buttons, Frames**: The launcher loads all it's images used from the Launcher-Install Build\Payload\Resources directory. You can change/modify these as you like, but you must maintain the resolution/sizing in order for everything to fit correctly. The names must remain the same for the launcher to utilize them. There are many graphic editing softwares out there, I personally use Adobe Photoshop Elements (About $99 USD), Krita (Free and awesome), and GIMP (Free and awesome) depending on what I'm doing.



b. **Icon:** The launcher, as well as the installer and update screens will utilize the Launcher.ico file from the Launcher-Install Build\Payload\Resources directory. You can replace it with your own, however, the icon must be named “Launcher.ico” for the launcher to utilize it. It is recommended to create an ico file that includes at minimum the following resolutions: 256x256, 128x128, 96x96, 64x64, 48x48, 32x32, 24x24, and 16x16 resolutions included (all 32-bit is fine). There are many softwares out there to create ico files, I personally use IcoFX (about $40 USD).

c. **Text and Progress Bar color:** Text and Progress bar color can be set in the MainConfig.config file you uploaded to your server’s Launcher Download Root URL you wrote down in Section 1, step b. The values need to be in hex format (you can use https://www.colorhexa.com/ to find those easily), you should leave the # symbol off. Note that if all you are wanting to change is text color and/or progress bar color, you can do this step by itself, there is no need to make a new launcher build as the launcher downloads the MainConfig.config file each time it starts up, so all you have to do is update this file and overwrite the old one on your server. In version 22.3.27.8, a separate color parameter was added for Release Note text color – if it is not set or missing in the MainConfig.config file the launcher downloads from your server, it will default to the TextColor’s value.



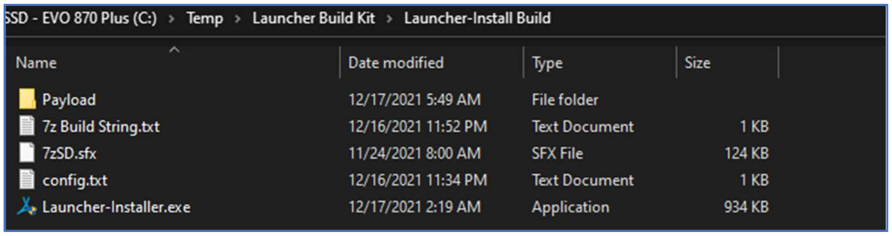
```
##########################################
# Atavism Game Launcher Main Config
##########################################
# For Version 22.3.27.8
##########################################
# Do not include trailing backslashes
##########################################
# The current package version of the launcher (increase this to force launcher updates)
CurrentConfigPackageVersion=22.3.27.8
# The directory on your server housing your game client files (Do not include trailing backslashes)
ClientDownloadRootURL=https://YourGameStudio.Com/Downloads/GameFiles/Game of Games/Dev/GameClient

#Executable file name to launch
GameEXEName=Game of Games.exe

#Publisher Info
PublisherContactEmail=support@atavismonline.com
GameWebsite=https://atavismonline.com

# Theme Colors (Hex values)
TextColor=D3D3D3
ReleaseNotesTextColor=D3D3D3
ProgressBarColor=00a2ed

# Menu Button URLs (Button numbering is from left to right)
MenuButton1URL=https://atavismonline.com/my-account
```

6. Create a Launcher-Install Package

Users will use this to install your launcher, which will then manage downloading your game and keeping it up to date forever after. **You will need 7zip** installed on your dev computer for this part, as we will be creating a compressed self-extracting installer for your end users.

a. Locate The Launcher-Install Build Directory

In the Launcher Build Kit directory, open the Launcher-Install Build directory

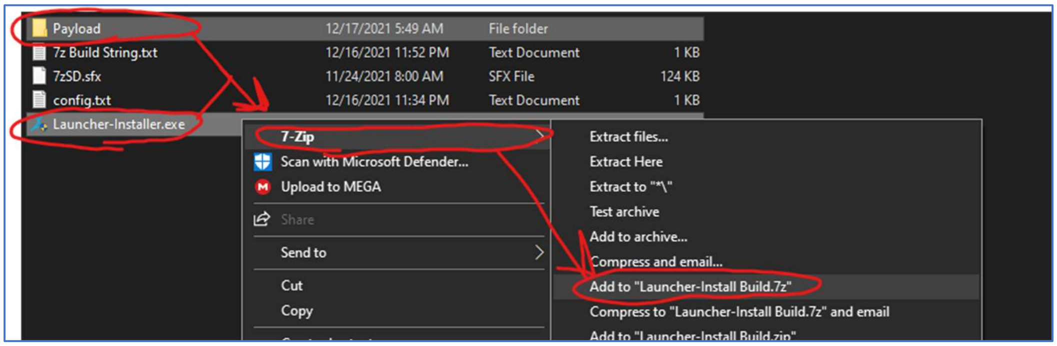


SSD - EVO 870 Plus (C:) › Temp › Launcher Build Kit › Launcher-Install Build

Name	Date modified	Type	Size
Payload	12/17/2021 5:49 AM	File folder	
7z Build String.txt	12/16/2021 11:52 PM	Text Document	1 KB
7zSD.sfx	11/24/2021 8:00 AM	SFX File	124 KB
config.txt	12/16/2021 11:34 PM	Text Document	1 KB
Launcher-Installer.exe	12/17/2021 2:19 AM	Application	934 KB

b. Create a 7z archive with the Payload and the Launcher-Installer.exe

Select the “Payload” directory, as well as the “Launcher-Installer.exe” file (Ctrl+click Them) and right-click on them, select “Add to Launcher-Install Build.7z”. Wait for 7z to finish creating your .7z file.

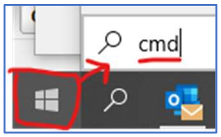


Payload	12/17/2021 5:49 AM	File folder	
7z Build String.txt	12/16/2021 11:52 PM	Text Document	1 KB
7zSD.sfx	11/24/2021 8:00 AM	SFX File	124 KB
config.txt	12/16/2021 11:34 PM	Text Document	1 KB
Launcher-Installer.exe			

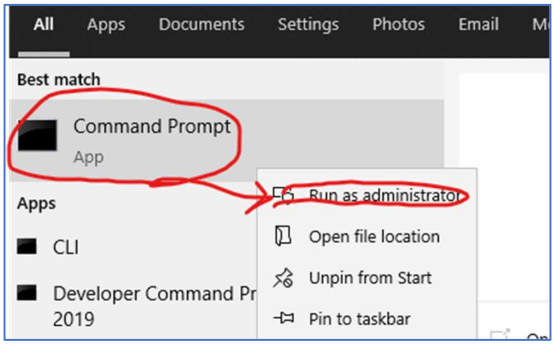
7-Zip
Scan with Microsoft Defender...
Upload to MEGA
Share
Send to
Cut
Copy
Extract files...
Extract Here
Extract to "*\"
Test archive
Add to archive...
Compress and email...
Add to "Launcher-Install Build.7z"
Compress to "Launcher-Install Build.7z" and email
Add to "Launcher-Install Build.zip"

c. Launch an elevated command prompt and navigate to the Launcher-Install Build directory

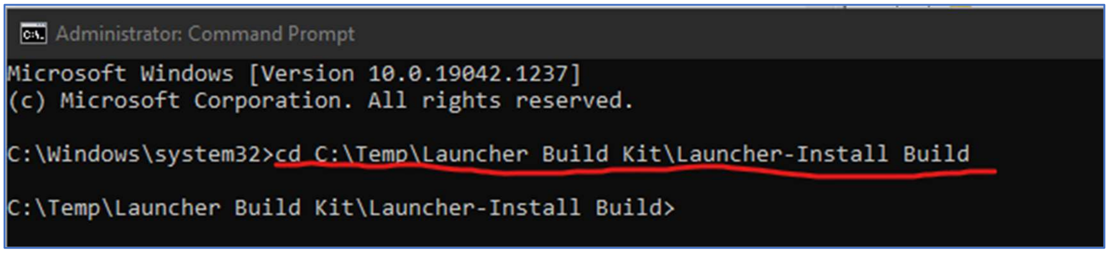
Click on your start menu button type “cmd”



Right-Click on “Command Prompt” and select “Run as administrator”



Type “CD <Path to Launcher-Install Build directory>” and press enter

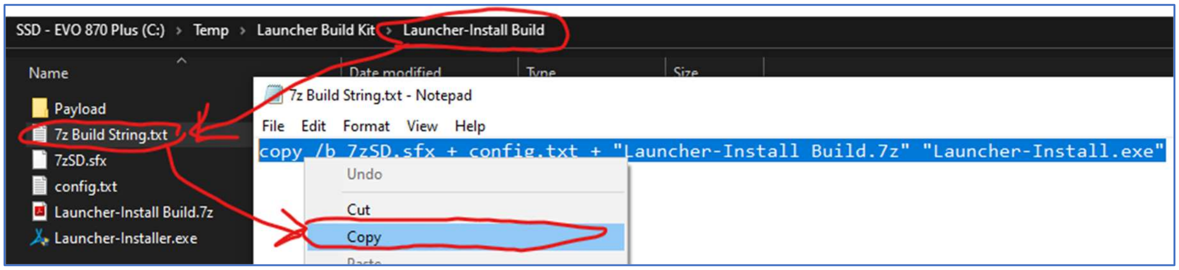


```
Microsoft Windows [Version 10.0.19042.1237]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\system32>cd C:\Temp\Launcher Build Kit\Launcher-Install Build

C:\Temp\Launcher Build Kit\Launcher-Install Build>
```

Note: If the drive of the Launcher-Install Build folder is different that the path showing initially, you will first need to type the new drive name and hit enter to move to that drive: Example – D: (Enter key)

d. **Make a self extracting installer from the .7z archive**

In the Launcher-Install Build directory, open up “7z Build String.txt” and copy it’s contents to your clipboard by highlighting the text, right clicking, and selecting “copy”

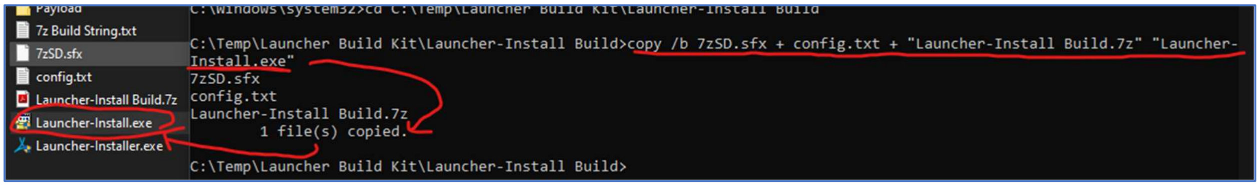


SSD - EVO 870 Plus (C:) › Temp › Launcher Build Kit › Launcher-Install Build
Name
Date modified
Type
Size
Payload
7z Build String.txt
7zSD.sfx
config.txt
Launcher-Install Build.7z
Launcher-Installer.exe
7z Build String.txt - Notepad
File
Edit
Format
View
Help
Undo
Cut
Copy
Paste

```
copy /b 7zSD.sfx + config.txt + "Launcher-Install Build.7z" "Launcher-Install.exe"
```

Note: If you accidentally delete the contents of this at some point, it is: copy /b 7zSD.sfx + config.txt + "Launcher-Install Build.7z" "Launcher-Install.exe"

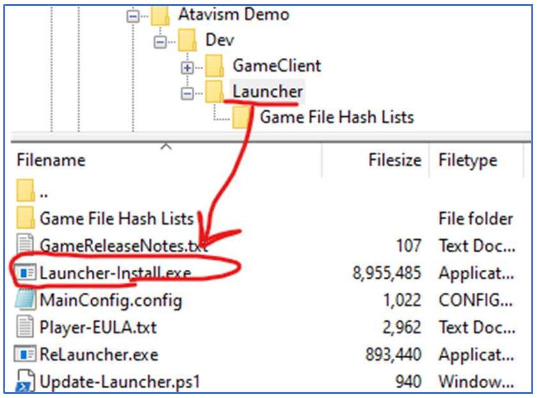
Paste the copied text into the elevated command prompt and press the enter key. Wait for “Launcher-Install.exe” to finish building – it should be slightly larger in size than Launcher-Install Build.7z. If it is not, delete it and go back through part 5 from the beginning. Do NOT change the names of any of the files, everything must be named this way for things to work.



```
C:\Windows\system32>cd C:\Temp\Launcher Build Kit\Launcher-Install Build

C:\Temp\Launcher Build Kit\Launcher-Install Build>copy /b 7zSD.sfx + config.txt + "Launcher-Install Build.7z" "Launcher-Install.exe"
7zSD.sfx
config.txt
Launcher-Install Build.7z
        1 file(s) copied.

C:\Temp\Launcher Build Kit\Launcher-Install Build>
```

e. **Copy “Launcher-Install.exe” to the server**

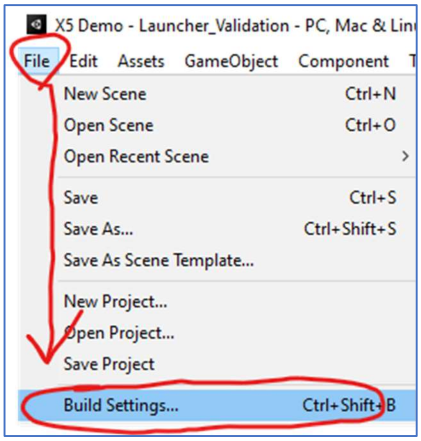
Copy the Launcher-Install.exe you just created to the Launcher Download Root URL you wrote down in Section 1, step b.

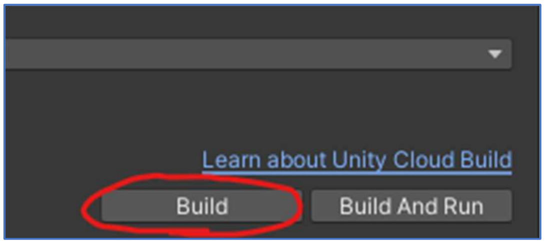


Filename	Filesize	Filetype
..		
Game File Hash Lists		File folder
GameReleaseNotes.txt	107	Text Doc...
Launcher-Install.exe	8,955,485	Applicat...
MainConfig.config	1,022	CONFIG...
Player-EULA.txt	2,962	Text Doc...
ReLauncher.exe	893,440	Applicat...
Update-Launcher.ps1	940	Window...

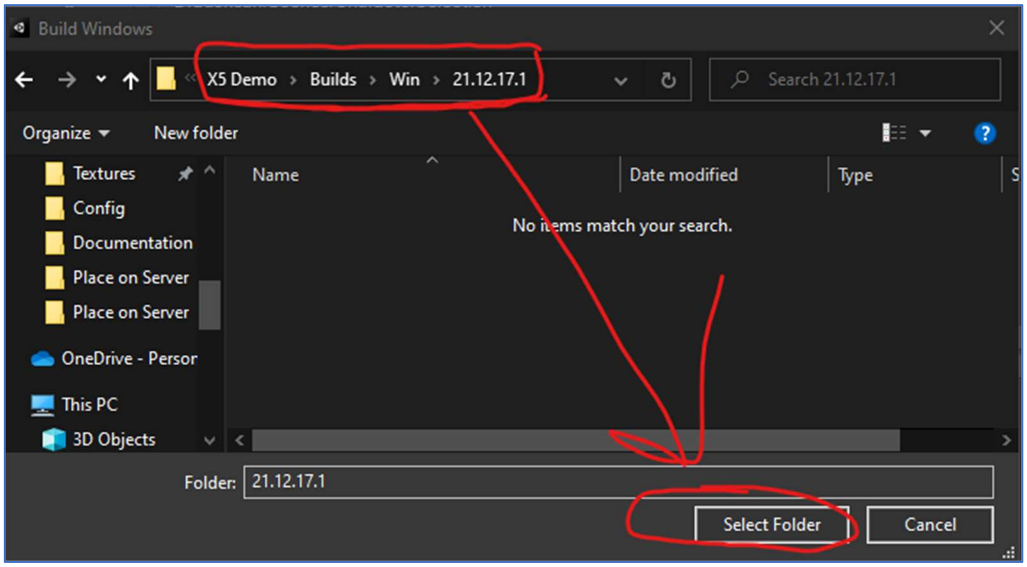
7. Create a Build of your Game in Unity

a. In Unity, go to File -> Build Settings, and click Build. Note that if you use any of the Non-Standard compression methods, the launcher will still work, but clients will redownload the entire game vs. individual files that have changed. Selecting “Default” here gives the best patching experience for your players.





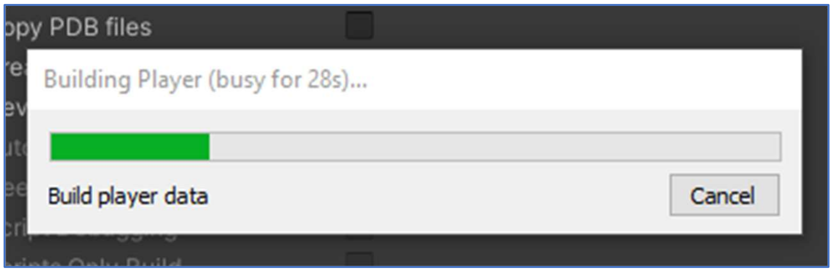
b. Create a directory somewhere for your build (remember where), and click “Select Folder”.



No items match your search.

Folder: 21.12.17.1

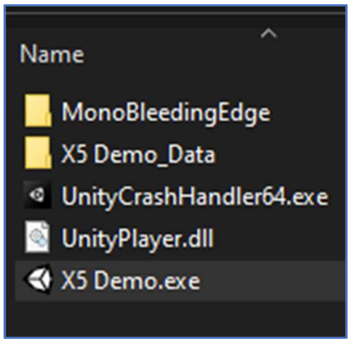
c. Wait for Unity to finish the build



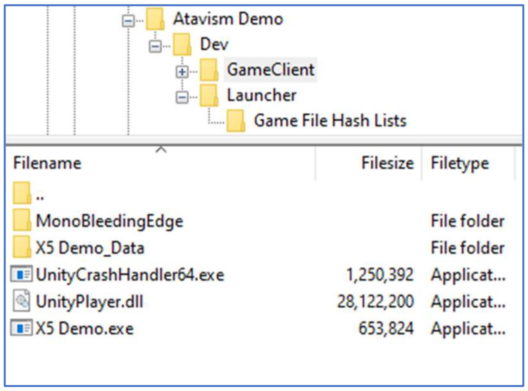
Building Player (busy for 28s)...

Build player data

d. When the build finished, copy the contents of the build folder up to your web server's Client Download Root URL you wrote down in Section 1, step b. Let the upload finish before moving on to the final step of the process.



Name
MonoBleedingEdge
X5 Demo_Data
UnityCrashHandler64.exe
UnityPlayer.dll
X5 Demo.exe



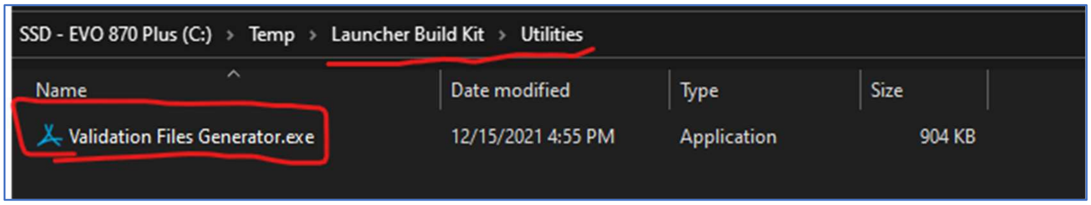
- Atavism Demo
  - Dev
    - GameClient
    - Launcher
      - Game File Hash Lists

Filename	Filesize	Filetype
..		
MonoBleedingEdge		File folder
X5 Demo_Data		File folder
UnityCrashHandler64.exe	1,250,392	Applicat...
UnityPlayer.dll	28,122,200	Applicat...
X5 Demo.exe	653,824	Applicat...

8. Create a File Hash Validation List and copy it up to your web server

The launcher uses a File Hash Validation list downloaded each time the launcher is launched to compare local game files with the latest game build. This file is also used by the Launcher_Validation scene to make sure everything matches as a final check before loading the login scene for the player. A utility is included to generate this File Hash Validation List file for you.

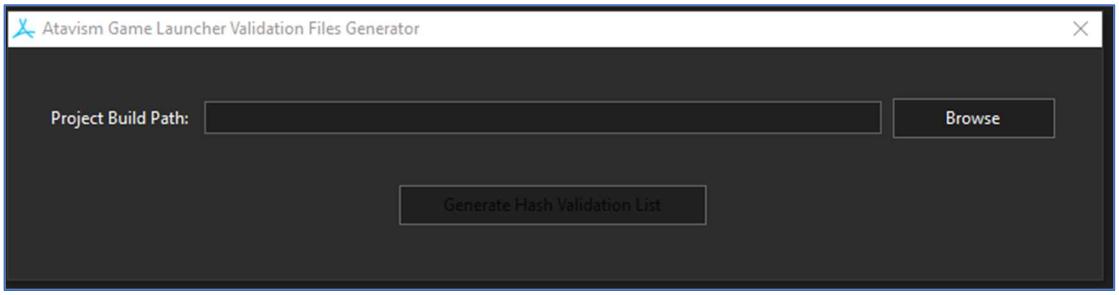
a. Open the "Validation Files Generator.exe" application from Launcher Build Kit -> Utilities



SSD - EVO 870 Plus (C:) > Temp > Launcher Build Kit > Utilities

Name	Date modified	Type	Size
Validation Files Generator.exe	12/15/2021 4:55 PM	Application	904 KB

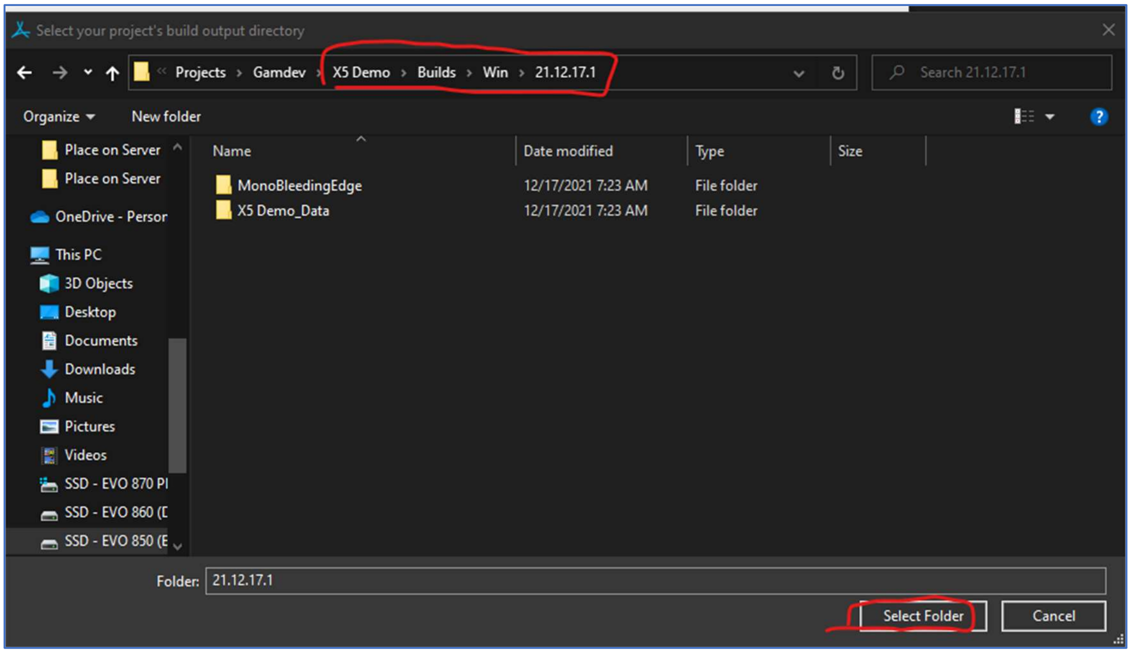
b. Click the browse button



Atavism Game Launcher Validation Files Generator

Project Build Path:

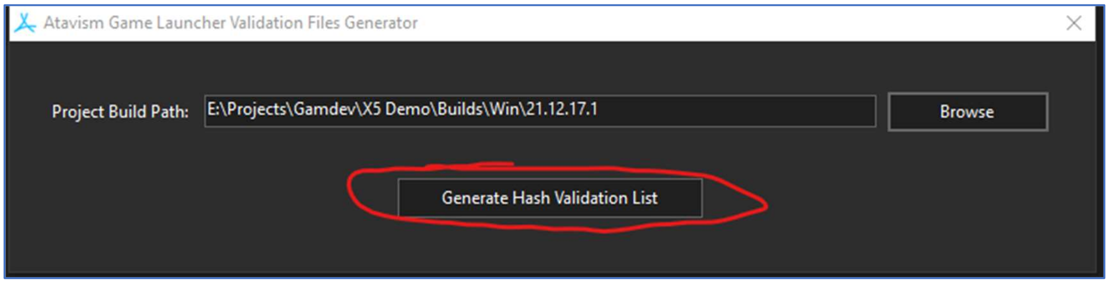
c. Browse to the location you built your game with in Unity (step 7) and click “select folder”



Name	Date modified	Type	Size
MonoBleedingEdge	12/17/2021 7:23 AM	File folder	
X5 Demo_Data	12/17/2021 7:23 AM	File folder	

Folder: 21.12.17.1

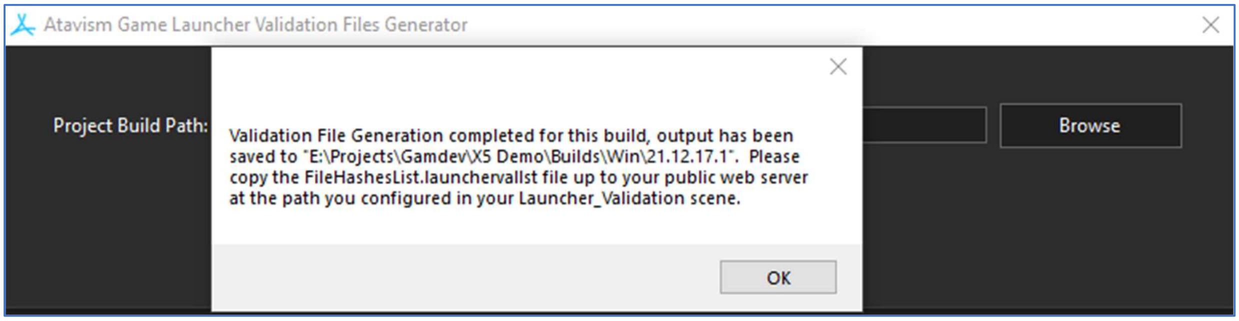
d. The “Generate Hash Validation List” button is now enabled, click it



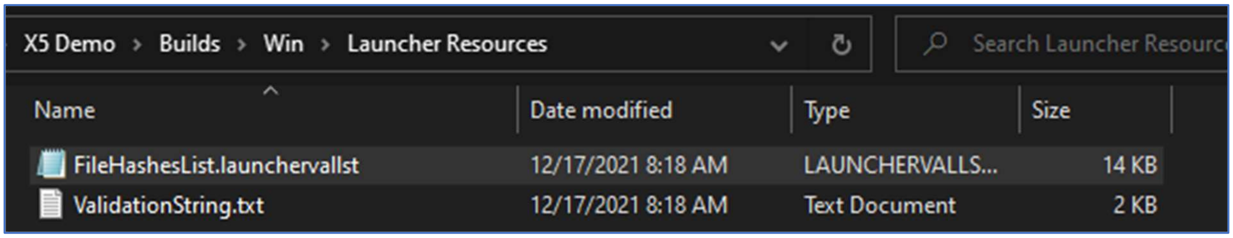
Atavism Game Launcher Validation Files Generator
Browse
Generate Hash Validation List

Project Build Path: E:\Projects\Gamdev\X5 Demo\Builds\Win\21.12.17.1

e. The generator will inform you when it is finished and provide you with the output location, which should be two directories up from your build. It should also automatically open the directory for you, revealing a file named “FileHashesList.launchervallst” – do NOT rename this file. There is also a file called “ValidationString.txt”, which can be used to launch the current build of the game without the launcher if you have a reason to do need to do that (this is usually a file that will just be ignored)

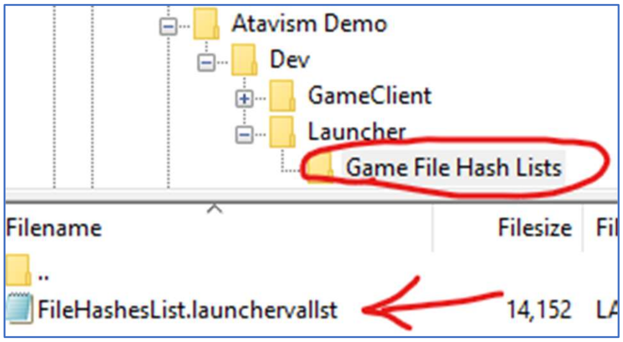


Validation File Generation completed for this build, output has been saved to "E:\Projects\Gamdev\X5 Demo\Builds\Win\21.12.17.1". Please copy the FileHashesList.launchervallst file up to your public web server at the path you configured in your Launcher_Validation scene.



Name	Date modified	Type	Size
FileHashesList.launchervallst	12/17/2021 8:18 AM	LAUNCHERVALLS...	14 KB
ValidationString.txt	12/17/2021 8:18 AM	Text Document	2 KB

f. Copy the "FileHashesList.launchervallst" into the "Game File Hashes List" directory you created on your server underneath the Launcher Download Root URL you wrote down in Section 1, step b. If you do not have this directory, just create it now as shown below, and then copy the "FileHashesList.launchervallst" file into it.



Filename	Filesize	Fi
..		
FileHashesList.launchervallst	14,152	LA

------------------------------------ Other Information ------------------------------------

1. **Updating the Launcher:**
   This is covered in the "Customize your Launcher to match your game" section already, please see there.
2. **Updating the Game:**
   Game updates are done by doing the following:
   * Repeat the "Create a Build of your Game in Unity" section
   * Repeat the "Create a File Hash Validation List and copy it up to your web server" section
   *Doing these two steps will cause the launcher to see the new hash validation list each time the launcher starts, at which point it will know which file hashes changed and download only the needed files from the web server. There is **no need to update the launcher** build by doing any of the other sections **for a game build update, unless you want to change the launcher theme.***
3. **Running the Launcher and finding the game:**
   The launcher installer creates a desktop shortcut as well as a start menu shortcut. There is also an add/remove programs entry created after running the launcher for the first time and selecting the game installation location (the uninstall string for this for the uninstall button in add/remove programs to work is currently not yet implemented, that is coming soon)

4. **Distributing your game to your players**
   Once implemented, all you need to do is provide a download of your Launcher-Install.exe to your users, it should be a very small download, as the launcher will pull down your game and install it once they install it.
5. **Where to find Launcher Logs**
   The launcher comes with logging built in. Logs can be found under the user's AppData\Local\<company name>\<gamename> directory. These can be very helpful for troubleshooting and assisting players with game installation issues.
6. **Support for other languages**
   The launcher has been updated to utilize a LanguageOverrides.config file if present in the Config directory next to the BaseConfig.congig. The presence of this file allows you to override default text with text of your choosing. Lines containing !VARIABLE! strings can utilize the strings to reinject variable values prior to being displayed to the user as well.