*** VISIT HTTP://KUROATO.COM/UAGK FOR THE ONLINE USER GUIDE**

UAGK, User Guide:

Learn how to easily create an awesome Tile-based action game! Create Hack and Slash or Shooter Games, whatever your preference. UAGK, User Guide will help you Get Started with creating your next action game!

UAGK: Important Notes

***Please Note**, if you want to Run the Game, or Test any of your Work with UAGK such as New Stages, you must always begin running the Game in Unity from the "start up" Scene. In the Stage Section we review how to Load the Stages from the "start up" Scene. ***Additional Note**, you must always add all of the scenes to your Build Settings, otherwise the Game will not load, or you will receive errors during run-time. – *UAGK Support*

UAGK: Character Setup (part 1)

Getting Started with UAGK, the first thing we want to learn is how to Setup our Player and Enemy Characters.

1. Lets go to /Resources/Objects/Player and find the**OBJ_ninja_player (melee)** or **OBJ_troop_player (ranged)** prefab, and drop this prefab into an empty scene. *for the following steps we will use "OBJ_ninja_player" but it can but substituted wit "OBJ_troop_player" if you want to create a Shooter Game with Player Setup for Shooter.

2. Now import your Unity Character Asset, and drop it's prefab into the scene as well.

3. Copy the Transform of the OBJ_ninja_player to your Character; try to Scale your Character down to the same size as our “OBJ_ninja_player” if possible.

4. Now simply drag and drop each of the character scripts attached to “OBJ_ninja_player” onto your character. You will also need to copy the Rigidbody and Box Collider*(values)* to your character.

5. Adjust your Box Collider on the Character to match it’s Mesh Size.

6. Animations should now be Updated on your character to match the type of Animations as Setup in our “OBJ_ninja_player” Animation Component. There are 9 Animations that are Supported. Each Type must Match Exactly in the Same Order as We have it Setup in our “OBJ_ninja_player”. Of-Coarse you can substitute Animations, or Re-Use some if you don’t have all the correct Animation Types, but it is Best to Have Matching Animations for Best Results.

7. Take note of Animations 5 – 8 in the List. 5 and 6 are Used for Standing and Attacking Animations, while 7-8 are for Running and Attacking Animations. If your Character does not have Running and Attacking Animations, you can simply Put in Any Standing Attack Animations you have, it should be fine. But when you Load the Game, you will Notice the Player is Moving and Attacking, but the Animation is Not in Sync. So to Resolve this Issue, you will want to Check the Box for “Stand Attack” Under the “**SCR_character Control (Script)**“. This will Freeze your Characters Movement for a Second to Playback the Stand Attack *(animation)* Since you Don’t have a Running Attack Animation.

8. That’s it! Your Player Character should now be Ready! Save/Apply all Changed to your Character Prefab. Next step is to Load your Character, and Test it in Game.

9. Load the “start up” Scene, found under the /Scenes Folder. Now Click on “MAIN” Scene Object. Next Find the “Player Obj” Under the “**SCR_main (Script)**“. Drag and Drop your Player Prefab into the “Player Obj”

10. Finally, Save the Scene, and Click Play in the Unity Editor. You should now be able to Start Game and See your Character in Game along with being able to Move / Run in all Directions with Keyboard, and Attacking / Dodging with Mouse Controls.

UAGK: Character Setup (part 2)

1. If you found that your character is not working as expected from following the above steps, then you may need to tweak a couple things to get it just right. Take note of the following scripts/components: "**SCR_character (Script)**" and "**SCR_character Melee (Script)** or **SCR_character Ranged (Script)**"

2. **SCR_character (Script)** – Use this to Tweak/Fine Tune the Animation Speeds of your Characters Animations. You will Notice the "Anim Speed" List is Set to 9, this goes in order of the Animations we have Setup. So you can Adjust each Animation Speed as Needed. You will also find "Run Speed" *(movement)* and "Roll Speed"*(evade/dodge)* adjust these if needed for your player character.

3. **SCR_character Melee (Script) or SCR_character Ranged (Script)** - Use this to Tweak/Fine Tune the Player Character's Attack Features/Properties. Such as:

- "Attack Range", *(range of Attack)*
- "Attack Damage" *(attack Damage per Attack)*
- "Attack/Hit Time" (*time before Damage occurs)*,
- "Attack Hit Object/Offset" *(Melee effect / placement)*,
- "Attack Sound FX" *(set Attack Sound FX)*
- "Force" *(amount of force/push back applied per Attack),*
- "Projectile Object/Effect/Offset" *(controls Ranged Attack effect, projectile object spawned, and placement)*,
- "Rapid Fire On/Interval/Limit" *(controls Ranged Attack speed/rate)*

4. **SCR_character Health (Script)** – is pretty easy to configure, setting the max health, along with the height of the Health Bar Displayed in-Game using the Offset, and Death Sound FX *(when player/character dies).*

UAGK: Character Setup (part 3)

1. After you complete setting up a Player, you will want to Setup and Enemy. Honestly, setting up the Enemy is No Different than Setting up the Player, So Simply Follow all the Same Steps above for the enemy setup, except you will want to Grab the "OBJ_ninja_enemy" Prefab found in /Resources/Objects/Enemy for the Setup.

2. The difference between Player and Enemy Setup is, for the Enemy there is the following: **SCR_enemy AI (Script),** easily understood and configured using Detect Range *(how far can the enemy see before finding the player)*, Points *(amount of points given when defeated),* and finally Aggression *(controls the Enemies Toughness).*

UAGK: Stage Setup (part 1)

1. First step for setting up your Game Stage, first Load the "start up" Scene, found under /Scenes/. Now Click the MAIN Scene Object. Under the **SCR_main (Script)**, you will find "Stage Name" this controls the Size, number of Stages your Game will have; by default we have stage 1 and stage 2. You can change the name of the stage 1 or 2 to whatever you want, just make sure it matches the same name of the Stage Scene in your Project *(if you want to create a new stage, or just rename it).* If you want to add more Stages simply change the value of the Size to any number you want.

2. Since we have a stage 1 and stage 2 already set in place, there is nothing to do here, lets just load the “stage 1” Scene now. Find the **OBJ_Environment, OBJ_Background,** and **OBJ_stageLights** in the Scene. These are all just Parent Scene Objects that hold all of the Stage Objects from the Walls, Floor, Lights, and Environment Objects. You can Rename, or Remove these Scene Objects if you like, they are only there to help you organize the Stage Scene.

3. **OBJ_stageControl**, is the most important Stage Scene Object. Under this Object you will find the following:**SpawnPointCollectable, SpawnPointEnemy, SpawnPointPlayer**, and **stageCorner**. All of the SpawnPoints Objects are of-course SpawnPoints for Collectibles *(items),* Enemies, and the Player, So make sure to leave these as is, and ONLY re-use them, copy them, and/or move their placement on the stage as you wish, “DO NOT CHANGE THEIR NAME or TAGS”, otherwise the game will break.

4. **stageCorner** is also very important! as this tells the camera where it can move, and where it should stop. So also Please be sure to leave as is, *(DO NOT CHANGE NAME or TAGS)* and Move these Objects to the 4 Corners of your New Stage. stageCorners are represented by Yellow Square Objects in the Scene.

5. **OBJ_stageControl,** Now back to this Object.Same applies to this Object as above, “DO NOT CHANGE NAME or TAG” otherwise the game will break. On this Object you will find 3 Components *(Scripts)* attached: **SCR_stage (Script), SCR_enemy Spawner (Script),** and **SCR_collectable Spawner (Script).**

6. **SCR_stage (Script)** - Used to set the Stage Music, and a Skybox Material for that Stage.

7. **SCR_enemy Spawner (Script)** - Controls the Number of Enemies and Types that will Spawn in the Stage. Under the “Enemy Objects”, you can Drop in as many different Enemy Types *(Prefabs)* as you want. With “Enemy Spawn Chance”, you should set the Size Number to the Same Size Number as your “Enemy Objects” then specify the percentage from 0 – 100 of the chance of each Enemy Type to Spawn. for example… if I have 2 Enemies and want the 1st Enemy to Spawn all the time, I set it to 100, if I want the 2nd Enemy to Spawn 1/4 of the time then I set that to 25 *(%, without the % sign)*.

8. Under the **SCR_enemy Spawner (Script),** we also have “Enemy Spawn Interval”, “Enemy Limit”, and “Enemy Stock Total”. ”Enemy Spawn Interval” should be set to the amount of time in seconds before another enemy will spawn. ”Enemy Limit”, should be set to the amount of Enemies you want on Stage at any given time. ”Enemy Stock Total”, is the Total Amount of Enemies that will be Spawned on Stage before Completing that Stage.

9. **SCR_collectable Spawner (Script),** - works the exact same way the SCR_enemy Spawner *(Script)* does with the number of items and types that will Spawn on Stage, the Chance *(percentage)* they will Spawn, how often, and what is the Limit of how many can Spawn on Stage at any given Time.

UAGK: Camera Setup (part 1)

1. The Camera Setup is pretty easy. Camera’s for the “menu”, and “game complete” Scenes can be easily Configured from those Scene’s by finding the Main Camera and Adjusting the Transform *(position, rotation)* Values as Needed.

2. To Adjust the Stage Camera, you must find**OBJ_camStage** prefab, located under “/Resources/Objects/Stage Setup/”, now on the OBJ_camStage there is only one component *(script)* that we want to adjust, called **SCR_cam Stage (Script).**

3. **SCR_cam Stage (Script),** contains the following settings that will need to be adjusted for a custom camera view, *Note, *view over edge refers to the edge of the stage*:

- Player Follow Speed, *(camera follow speed value)*
- Offset Default *(camera position value)*
- Rot Default *(camera rotation value)*
- Edge Offset Side *(*view over side edge/s)*
- Edge Offset Upper *(*view over upper edge)*
- Edge Offset Lower *(*view over lower edge)*

UAGK: Menu Setup (part 1)

1. To Setup our Main Menu, Load the “menu” Scene.

2. Find the “Main Camera”, with this Camera you can Rotate and/or Position it as you wish to get the Camera View you want for your Main Menu.

3. Next we have “OBJ_menuStage”, this is simply a Parent Object that holds all of the Background, Stage Environment Objects, including the Player Models *(optional)*, and Lights. You are Free to Setup this Scene as you wish to get whatever Background View you want for your Main Menu Scene.

4. **menu control** - is a simple scene object that carries the **SCR_menu (Script)** attached, which controls the Main Menu. It provides only two options for adding Menu Music, and Skybox Material for that Scene.

5. next part of setting up the Main Menu is Loading the “start up” Scene. Find the MAIN Scene Object. Attached you find the **SCR_main (Script)**, you have an Option here to Configure “High Scores On”, it’s an important piece of the Main Menu. This allows you to disable or enable the High Score Board that shows up on the Main Menu.

6. Next on MAIN Scene Object we have **SCR_input (Script)**, this will allow you to define 2 Sound FXs for the Main Menu Buttons, but also applies to Pause Menu Buttons *(in-Game)*, and the Last “game complete” Scene that contains a Menu Button. One Sound FX is for Highlighting a Menu Button, and the Other is for the Select Sound FX *(click on menu button)*.

UAGK: GUI Setup (part 1)

1. *GUI Setup applies to all Menu Buttons, Crosshair, Logos, Images, Text, and Numbers Found throughout UAGK's Game, Stage, and Menu Scenes. Lets Begin by loading the "start up" Scene, and selecting the MAIN Scene Object, and view **SCR_main (Script)**

2. Under SCR_main *(Script),* as mentioned above we have a **High Scores On** Option that you may want to enable or disable depending if you want to use the Score System along with **Points On** Option, which handles the in-game Pop-Up of Points when Killing Enemies.
Next we have **Reticle On + Hide Mouse Cursor** which enables the in-game Crosshair and Hides the Mouse Cursor, both should be enabled together or disabled together. If you would like to Edit the Crosshair Size, Rotation, and Texture, Find the **OBJ_reticle** prefab under "/Resources/Objects/Gui/".

3. **SCR_gui (Script),** is also attached to the MAIN Scene Object. SCR_gui controls the font style of all Menu Buttons and Text Found through out UAGK's Game, Stage, and Menu Scenes. This is easily configured by simply choosing/importing a Font of your choice, drag and drop it into the **Text Font** Element. ***Note:** there is an additional option available for you to change the Colors of the Option Menu Buttons, 1 for Primary Color, the 2 for Hover Over *(Highlight)* Color.

***Need more Fonts?** you may also increase the number of supported Fonts, for example if you would like to use different Fonts in different Scenes, you may Increase the Text Font Size, then drag and drop more Fonts into the additional Elements.

4. Continuing on from the ***Need more Fonts?** above, once you have added all of the Fonts you would like into the Text Font Size/List of Elements, your next step is to find the GUI Prefabs *(containing text)* that you would like to change. Every Menu Button *(text),* and or Other On Screen/In-Game Text found throughout UAGK's Scenes can be easily found well organized under "/Resources/Objects/Gui/_Option/" and "/Resources/Objects/Gui/_Text/"

***Example**: Lets say I would like to change the Font of the In-Game Score Text found on the Top Right of the "stage" Scene. I will find that Text Under "/Resources/Objects/Gui/_Text/" in Prefab **Text_ScoreText,** attached to this Prefab we have**GUIText** Component where we can

change the actual Text of “Score:” to whatever other Text we want. Next we have **SCR_text (Script),** this is where you will define what Font Style you will use as you Setup in the previous Step for the Text Font Size/List of Elements. In the “Font Type” Value, you want to enter the Value equal to the Text Font Element as Setup Previously. So if you have Element 0 and Element 1 with different Fonts, simply enter the number 0 or 1 to Set the Font Style for this GUIText / Text_ScoreText Prefab.

There are some additional Options available in SCR_text *(Script)* like Scale Base and Text Position, you are free to Adjust these Values if you would like to move the GUIText to a New Position On-Screen.

5. Finally, if you look under ”/Resources/Objects/Gui/_Icon/” you will find Prefabs of all Logo’s and/or Images of Text found throughout UAGK’s Game, Stage, and Menu Scenes… Such as Main Menu Title’s, and Control Image/Text, In-Game/Stage Count Down Image/Text, Pause Title, and the “complete game” Scene Menu Title.

You may Edit the Logo’s and Images of Text both by Editing the PNG Images each Prefab is Pointing to, and also adjust their Position and Scale On Screen from the inspector under their **GUITexture** component and **SCR_icon (Script).**

UAGK: Music/Sound FX (part 1)

1. Music and Sound FX are very easy to Setup. Each Scene in UAGK can play Music. You can place any music you want in every scene under the scene control object. Here is a List of Scenes with Controls Objects per Scene:

- “start up” Scene: **MAIN** *(set soundfx in SCR_input (Script), Menu Select)*
- “menu” Scene: **menu control** *(set music in SCR_menu (Script), Menu Music)*
- “stage 1” Scene: **OBJ_stageControl** *(set music in SCR_stage (Script), Stage Music)*
- “stage 2” Scene: **OBJ_stageControl** *(set music in SCR_stage (Script), Stage Music)*

- "game complete" Scene: **game complete control** *(set music in SCR_gameComplete, Game Complete Music)*

2. Sound FX are easily configured by placing the Sound FX of your choice in every Prefab that supports a Sound FX. Here is a List of Prefabs and Script/Components that Support Sound FXs:

- **OBJ_bullet** *(set soundfx in SCR_projectile (Script), Impact Sound)*
- **OBJ_ninja_player / …enemy** and **OBJ_troop_player / …enemy**
- *** (set soundfx in SCR_character (Script), Roll Sound)*
- *** (set soundfx in SCR_character Ranged (Script), Shot Sound)*
- *** (set soundfx in SCR_character Melee (Script), Attack Sound)*
- *** (set soundfx in SCR_character Melee (Script), Attack Hit Sound)*
- *** (set soundfx in SCR_character Health (Script), Death Sound)*
- **OBJ_cHealthBoost** and **OBJ_cPowerBoost**
- *** (set soundfx in SCR_collectable (Script), Collect Sound)*

UAGK: Items (part 1)

1. Items refer to the In-Game Collectible Items such as Health and Power Ups, which can be found under "/Resources/Objects/Collectable/"

2. Collectible Items have two Components attached, **SCR_collectable (Script)** and **Box Collider**

3. **SCR_collectable (Script),** features are the following:
- Collect Effect Obj *(set particle effect for pick up)*
- Collect Sound *(set sound fx for pick up)*
- YOffset *(Y Axis Offset for spinning the item)*
- Spin Speed *(set the spinning speed of the item)*
- Points *(set points item provides – optional)*
- Health Recovery *(set health points restored)*

- Power Boost *(set attack power boost)*
- Power Boost Duration *(set time for power boost)*

4. **Box Collider,** should be set to "Is Trigger" Always

UAGK: Projectiles (part 1)

1. Projectiles are used for the Enemies or Players with the **SCR_character Ranged (Script),** by default we are using **OBJ_bullet** for the Projectile found under "/Resources/Objects/Projectiles"

2. Attached to OBJ_bullet we will find **SCR_projectile (Script)**, **SCR_projectile Trail (Script), Rigid Body,** and **Box Collider**

3. To create your own custom Projectile simply Create a New GameObject for the Parent Projectile Object similar to OBJ_bullet. We will then use OBJ_bullet to drag and drop the **SCR_projectile, SCR_projectile Trail (Scripts)** onto our new Projectile *(GameObject)* and Copy Paste the Rigid Body and Box Collider Components.

4. Next step is to drop in your custom Projectile Asset, place it under the Projectile *(GameObject) we created.* **Note:** if your Projectile is a Particle Effect, that's no problem, only we still need an actual Projectile Object Mesh, so you can create a Sphere or Cube and turn off the Rendering for it.

5. **SCR_projectile (Script),** features the following that is required to setup a projectile:

- Projectile Tex *(optional, you can set a diffuse texture for projectile)*
- Impact Effect Obj *(set bullet Hit/Damage Effect)*
- MeshStr *(name of the projectile mesh Cube, Sphere or other…)*
- Force *(set the push back force of the projectile's target)*
- Speed *(set the speed of projectile)*
- Health Recovery *(set health points restored)*
- Impact Sound *(set the impact sound fx of the projectile)*

6. **SCR_projectile Trail (Script),** allows you to easily configure a trailing effect following the projectile. Set the Trail Texture, but if you don't have the correct Trail Texture for this, leave it as it is… as you can modify the Trail Width and Color which can give your projectile the trailing effect and look you want.

7. Finally after you have configured and setup your new Projectile Object, Create a Prefab of this Projectile so you can drag and drop it into the **SCR_character Ranged Component (Script).**

UAGK: Score (part 1)

UAGK Provides a built-in High Score System! We have mentioned the High Score Options before as it seems to overlap with the GUI, but we will review it here again.

1. Load the "start up" Scene, and Select the MAIN Scene Object. Under **SCR_main (Script)**, as mentioned before we have a **High Scores On** Option (which displays on the Main Menu) that you may want to enable or disable depending if you want to use the Score System along with **Points On** Option, which handles the in-game Pop-Up of Points when Killing Enemies.

2. Another Part of the Score System is the in-Game Score GUIText that appears on the Top Right of the Screen in the Stage Scenes. Under "/Resources/Objects/Gui/_Text/" we can find the Prefab **Text_ScoreText** and **Text_ScoreNumber.**If don't wish to use the High Score System then you will want to disable these Prefabs to prevent them from appearing in-Game. *(DO NOT DELETE, as you will receive errors due to code that is searching for these prefabs)*

3. Finally we have **Text_GameCompleteScoreText** and **Text_GameCompleteScoreNumber** also found under "/Resources/Objects/Gui/_Text/" you may want to disable these Prefabs also if you do not wish to use the High Score System, as these will Display the Total Game Score in the "game complete" Scene.