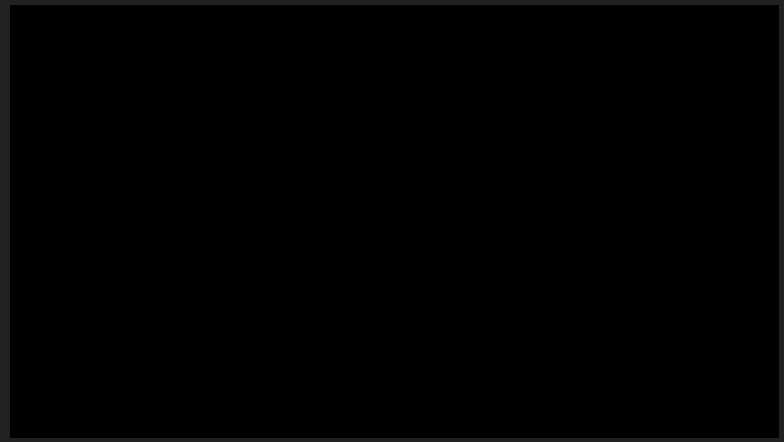
Come Die With Me: Development Slides

Jacob White

Week 1- 14/01/2019 [Before the Pitch]

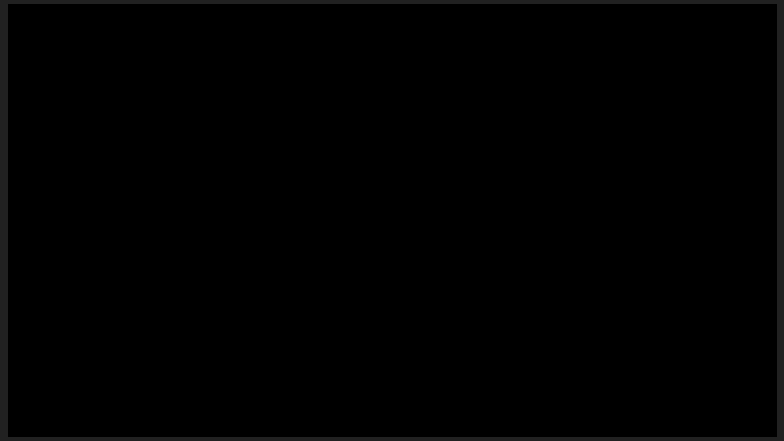
- With the concept for the game set out, being a VR puzzle game in which you combine items together in a pot to fit customers needs in a medieval fantasy setting.
  The main influence for this being the Breath of the Wild cooking minigame [https://rankedboost.com/zelda-breath-of-the-wild/food-cooking-recipes/]
  But with players taking a more direct role with it being in VR.
- In the interests of having a proof of concept I created a non-VR prototype of some of the games mechanics to present during the pitch.
- This consisted of a few food objects in the levels and a cauldron, with players being able to pick up objects and drop them into the pot. If a player drops the correct combination of these objects into the pot it produces a dish, in the prototypes case a Roast Dinner. If the incorrect combination is thrown into it, the pot produces a ‘scrap’ item (likely to be rotten food etc later on).
- In terms of actual coding the pot is just an object with a collision volume inside of it. When an object is dropped into the volume it checks if it belongs to the Food object class then destroys it regardless. If belonging to the food class it records what the object was in an empty array, considering it in the pot.
- This repeats when players throw objects in until three objects are recorded in the pot, at which point it checks through the list of available recipes (AND nodes) If the combination of what's in the pot amounts to a recipe it spawns that dish, if not it spawns a scrap object at which point the pots array is cleared.



[Above] Footage showing the prototype cauldron, producing a roast dinner object and scrap object, shot in editor.
References:
https://www.youtube.com/watch?v=xzCws6d_YgA&feature=youtu.be

Week 1- In Depth; Grabbing and Physics

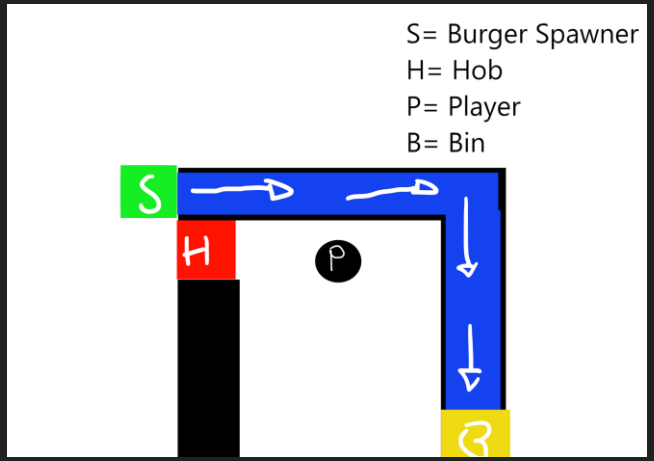
- In addition to the recipe mechanic being prototyped I needed players to be able to grab and manipulate items to do the following:
  A. Throw them into the pot.
  B. Give dishes to the customers.
  As well as being a general source of fun for players.
- As I had no experience with physics based manipulation used the following youtube videos to create a basic template for picking up and moving around objects, editing the distance from which players can do so to fit the players environment at a later date.
  https://www.youtube.com/watch?v=WfZCHG4B9X0
  https://www.youtube.com/watch?v=HnR1Gf5gXcY
- This particular solution did have some issues which will need to be resolved further into the project, namely that as the objects don't currently have a mass, players walking into them kicks them very far across the map.
  However this shouldn't be too big of a concern as the sourced channels have videos on that subject matter.
- A particularly good thing about the current system is that only certain objects can be manipulated, namely those that are physics actors, preventing players from for example trying to pick up an NPC.
- If this project is selected there are some additional manipulation features I'd like to add such as the ability for objects to better gain momentum when players are holding them so they are easier to throw (something that is particularly fun in VR). In addition it would be good practice to make objects be able to rotate in your hand (I am unsure if this is prebuilt into VR), as this could be disorientating in VR if the objects don't respond to your input.



[Above] Footage for prototype of the physics based manipulation, using a line tracer to identify what the player is aiming at, shot in editor.
References:
https://www.youtube.com/watch?v=N0X6DyEZpPI&feature=youtu.be

Week 2: 21/01/2019

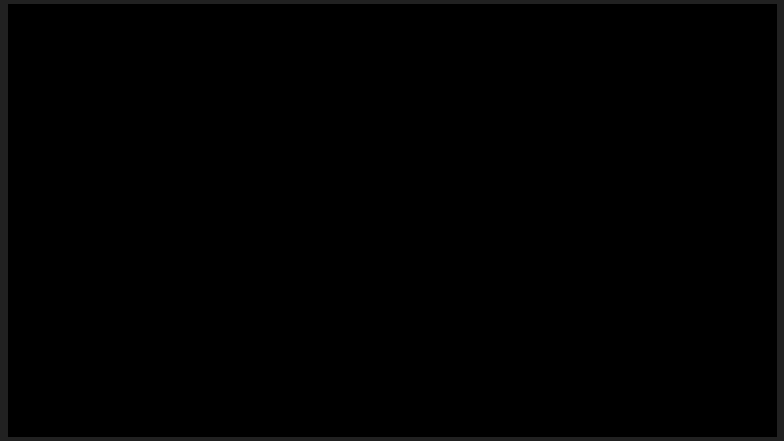
- With the second project (Burger Wars) chosen for use most of the prototypes I made would need to be scrapped, however elements of it could still be adapted to the new project.
  For example the recipe system would potentially be adapted as a method to assemble burgers, changing where the recipe is assembled to simply being the players hand, and clipping the in progress burger to it.
- Now that we were decided on a project using VR I had to do some research into how to best implement it, as well as the limitation regarding VR.
- To begin with setting up the VR, including how to do this with the various available kits (Oculus, PSVR, VIVE, etc) I used the UE VR Cookbook (Mitch Mccafrey) to get a general idea of how VR works.
- From this as well as interactions with the lecturers I learnt that it was best to not take control away from players in regards to their FOV, but conflicting this was that the Oculus in particular has issues with players interacting offscreen.
  Due to the cameras used, if a player for example tries to pick something up off screen the Oculus can have a hard time reading this. From these we decided to use the HTC Vive for now and change if we needed to at a later date.
- In order to make sure we were open to as many platforms as possible I did a short sketch of an area that wouldn't require players to look around them too much, focusing on their forward and sideways FOV's (as shown on the right), to give to level testers.
- Finally we did some small physics tests in VR using the weight settings, as well as seeing how objects could be handled in VR. This was useful for testing as the various burger bombs could have different weights that would make it harder to throw them long distances, making it easier to have different advantages/disadvantages to the weapons.



[Above] Image of a potential level layout using the conveyor belt method of burger bomb spawning (described next page). Everything is within arms reach of the player, burgers spawning from the Spawn point and being deleted when they fall into the bin. The player grabs what burger bombs they want from in front of them and lights the fuse using the hob.
Drawn in Medibang Paint.

Week 2: In Depth; Burger Spawning

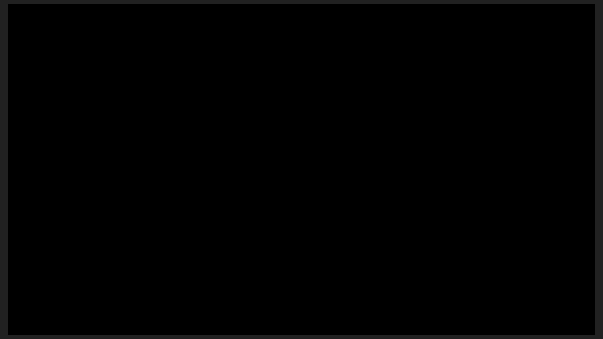
- With tasks being properly set the beginning of the next week I began prototyping some of the very basic mechanics to discuss with the group, testing as much as I could to settle on the core mechanics earlier rather than later.
- In particular this involved how players would get access to the various burger bombs. Whilst three methods were discussed originally, one was quickly discarded for the moment.
- This was the method using the recipe system I had previously made. The issue with using this system now was that with player racing to kill the enemies before they reach the counter, players would have to routinely look away from the counter to assemble burgers. This didn't fit well with what we wanted for the game, with most of the fun coming from throwing the burger bombs this shouldn't be inhibited by assembling every time you want to throw one.
- Instead using this tutorial:
  https://www.youtube.com/watch?v=aqRUe7pm4Ok&t=8s
  I made a basic conveyor belt and spawning object that spawns a burger every few seconds based on percentage probability. For example the most common burger spawning 50% of the time whilst a rarer one might only spawn 10% of the time.
  These values being easily editable meaning these can be changed later on once the full list of burgers is made. Another feature that can be toggled later is what objects can be moved on the conveyor belt, as I have currently restricted it to just the burgers.
- An issue later on could potentially be that this is too random, not allowing players to get the particular burgers they need but this will be discussed during testing.
- I also sketched up an alternative method of spawning using various tubes in front of the player that would provide them with one of each kind of burger, when that kind of burger is taken a timer starts to respawn it from the tube. Allowing for more tactical gameplay.
  Started creating this using the following tutorial:
  https://www.youtube.com/watch?v=dbLFHfIVleg



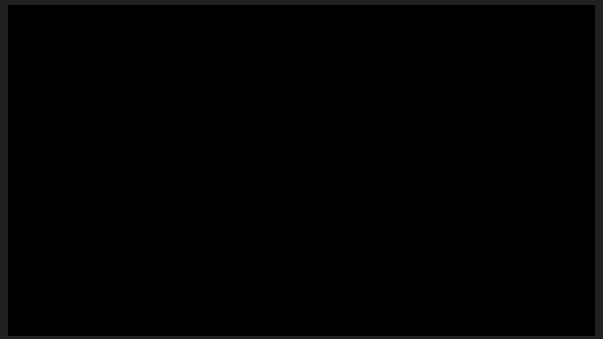
[Above] Footage for a prototype conveyor belt, spawning (semi) random burgers for the player to throw, shot in editor.
References: https://youtu.be/WY0nSaZ011w

Week 3: 28/01/2019

- After my exploration with VR last week I found that I quickly suffered from motion sickness whenever I used it, making testing my features in VR extremely difficult for extended durations. Whilst this wouldn't be a massive issue as another group member held the master version of the game, I would need to be able to test the mechanics I was designing.
  To this end I adapted the system I made in week one with a small change being to include throwing objects using the tutorial below:
  https://www.youtube.com/watch?v=zwjFzkJbqho
  This essentially allowed me to test using the mechanics of VR without using a headset for the majority of the time, only occasionally needing to use VR itself for testing mechanics once they were shifted between the two.
- Continuing on from the conveyor belt concept from last week (which was chosen as the spawning method for no following a team meeting) I added a requested feature which was a bin sort of object at the end of the belt.
  Regardless of what the actual asset was this serves the purpose of removing any burgers the player doesn't use, destroying them when they fall off of the conveyor belt into it.
- An issue I had with this was what to do with objects that weren't burger bombs such as if a player dropped a knife in there. For now the bin simply blocks objects that aren't burgers from entering its radius, but in the future I would like to apply an impulse to these objects to push them out and prevent blockages.
- Leading up to making the burger bombs themselves I added an essential part to the process which was the hob. This simply serves the purpose of providing a trigger box which lights the fuse on the burger bombs, setting off a timer until they detonate. The player having to hold the bombs in this radius which is indicated in game by a fire particle effect. The player is also able to turn the hob on and off by interacting with it, but hopefully this is tied to a button on the hob later on.

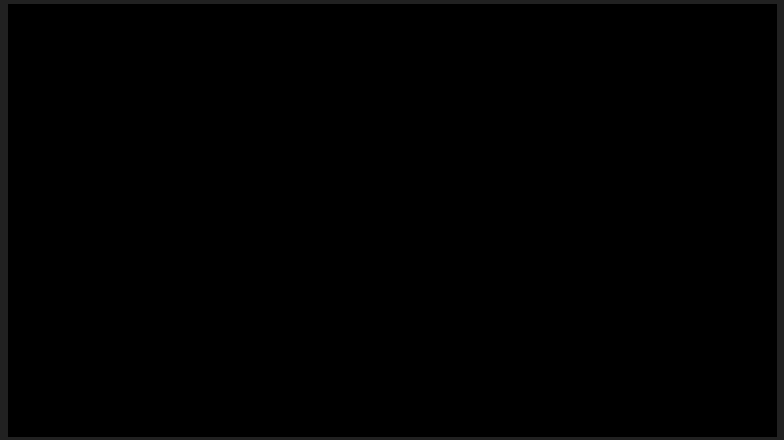


[Above] Video showing the current method for deleting unused burgers.
[Below] Video showing the state of the Hob object, being turned on and off.
Both shot in editor.
References: https://youtu.be/3l7cBNGAs_g
https://youtu.be/G4Ue-D6DdBY



Week 3: In Depth; Explosive Projectiles

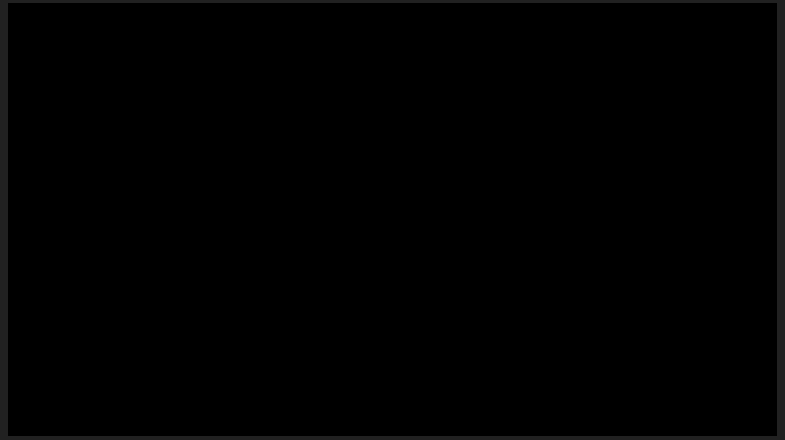
- The main task this week was the blueprinting for the initial burger bomb which needed to include the following:
  A physical explosion to push back enemies as well as destroy them.
  A visual effect to show the area of the explosion.
  And the ability to light the explosive via the hob.
- As mentioned previously the last feature currently works with approximately four seconds passing from the bomb being lit to it exploding, though this is subject to change from testing. When it came to the actual explosions I had to get some help from the below guide:
  https://www.youtube.com/watch?v=HnR1Gf5gXcY&t=568s
  Afterwards changing elements of this to accomodate for the delayed charge on the explosive.
- I did encounter some issues here that whilst won't affect gameplay now may create issues down the line so need to be resolved quickly. Namely the explosions force doesn't affect the basic cubes or other objects built into unreal. It does affect the spawned burgers and the enemies, as well as the player but doesn't register the cubes as physics objects.
  I will likely discuss this with lecturers, if we wish to have parts of the environment be affected in future development.
- The explosion effect itself is currently just a placeholder explosion from Unreal's starter content and will hopefully be replaced by one more suiting the stylised art direction.
- Finally I did some brainstorming into how I might realise the other burger variants:
  Incendiary Burger (El Diablo)- Spawn a trigger box that attaches to an enemy when it makes contact with the burger, causing damage to it and any others within the box.
  Slowdown Burger (Gravy Seal)- The same concept but this trigger box might cause the trigger box to fall and hit the floor, slowing down enemies that pass it and doing the same to those that walk over it whilst the box remains (likely on a timer).



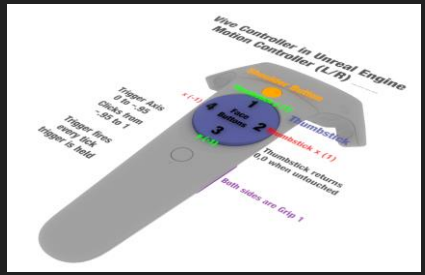
[Above] Video of the explosion for the basic Burger Bomb and its effect on the environment, shot in editor.
References: https://youtu.be/R1r4hWtWB_I

Week 4: 04/02/2019

- My main task this week following a meeting was to change the current system so that players could squirt sauces at the basic burger to change its properties and appearance, as opposed to the previous system where the different burgers would simply spawn randomly.
- In retrospect this new system does work better as it gives players more agency, able to choose the tactics they use through the type of burger they make, as instead of being at the mercy of the random burger generator.
- The first version I made for this mechanic was in an attempt to make the sauces realistic, by detecting the angle the sauce was pointed at and only allowing players to squirt it then. However there were some issues with detecting the rotation of the sauce bottle in the current build of the game. For now I've chosen to use another method of doing this though it could be changed in a later version of the game.
- The version that is currently used functions roughly the same as the initial system, except a collision box is placed at the end of the sauce bottle. If the player activates the sauce bottle when it is held it destroys the burger beneath it and replaces it with the burger type tied to the particular bottle.
- Again a subsystem did have to be made outside of VR to rotate the held objects for the player, this time just snapping them between a sideways and upwards rotation for the sake of saving time.
- When this is converted to VR I will need to perform a few tests on how rotation works and whether it is tied to the object or the players hand. When this is done it should allow me to reintegrate the mechanic of the bottle only squirting when at a downwards angle.
- Two additional things that must be added is the particle effect for the bottle squirting (a simple particle emitter) and spawning decals on surfaces that disappear over time as suggested by a lecturer. This could likely be made by using the below tutorial in addition to adding a basic lifespan to the spawned decal actor:
  https://realtimevfx.com/t/ue4-spawning-decals-at-particle-collision-locations/6677/3
- I did some research (used in the next slide) as well to better understand the layout of VR controllers, where the players hand sits as well as what works best for realism (shown on the right).

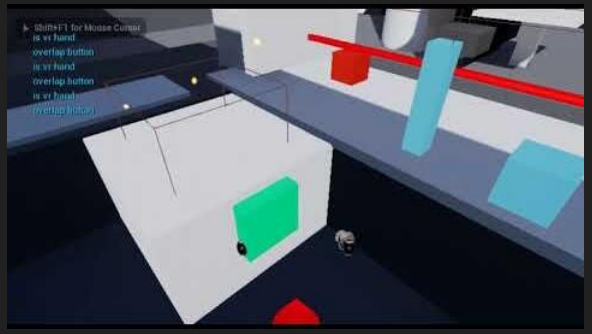


[Above] Video showing the current mechanic of adding sauce to a burger to change its properties, shot in editor.
[Below] Image showing the different controller inputs for vive controllers in UE4.
References:
https://www.youtube.com/watch?v=co9ZZnfH9Pk&list=PLGQ2tt6ABjKFiYaJ1LU8nerYfqYHogNZ2&index=14&t=0s

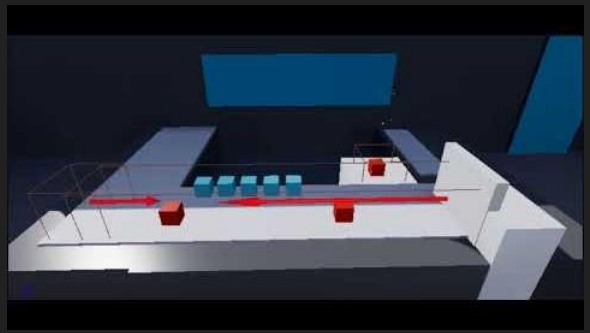


Week 4: In depth; porting to VR

- The second large task that needed starting was the combination of the current level blockout with the current mechanics to be able to test out the game on its intended platform (VR).
  This mainly involved the converting of the current mechanics to suit the VR platform, changing and tweaking small elements that could perform negatively on the platform.
- For example the current system of spawning explosions affected too many objects in the level (all physics actors) including the player.
  With the players limited movement this could knock essential objects (such as the sauces) out of the players reach. In addition going by the previous advice I found in the VR cookbook (see references) moving players around by force can be sickening in VR.
  Resolving this was fairly simple though just by changing the collision presets of the explosion radius.
- In regards to collision that is where most of the errors have come from so far with the port. There was an issue where for whatever reason burger bombs were floating, but this was apparently from in the migration the explosions had been auto enabled, so were persistent as soon as the burgers had spawned.
- With the port to VR being completed some additional aspects have been added. Following some advice from the lecturer the sauce bottles discussed previously can be squirted with the grip button on the VR controller, this being the relative position of the players thumb adding to the realism of the action.
- In addition some extra functionality has been added to the Hob, allowing it to be turned on and off using a button on the front. After some discussion this can only be done when the players fist is balled in the grab state to make it more realistic.
  Whilst the colour of the button also does change when this is pressed, I would like to add the button becoming depressed when hit, moving further into the object.

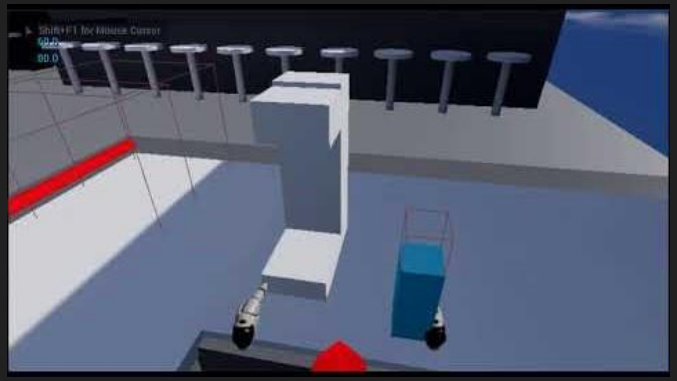


[Above] Video showing the new button on the hob object.
[Below] Video showing the fixed version of the game in the VR template, including the basic layout of the counter, shot in editor.
References:
https://www.youtube.com/watch?v=7UsEUafpWZo&list=PLGQ2tt6ABjKFiYaJ1LU8nerYfqYHogNZ2&index=7&t=0s
https://www.youtube.com/watch?v=aLbhiBTTPbk&list=PLGQ2tt6ABjKFiYaJ1LU8nerYfqYHogNZ2&index=2&t=0s

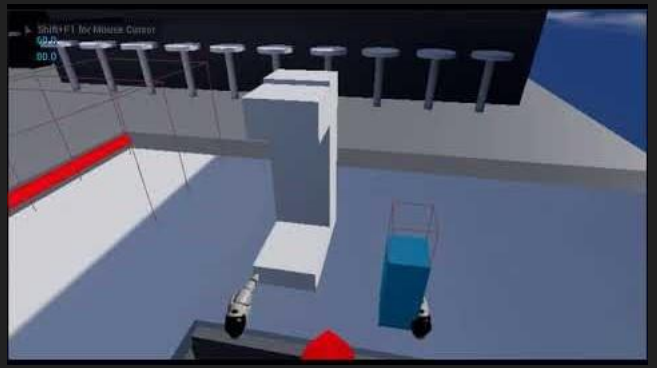


Week 5: 11/02/2019

- This week after the current VR levels were merged into one; allowing the enemies and burgers to spawn in a single version I began work on remodelling the sauce mechanic based on the new brief for it.
  This involved having the sauces fire a projectile which when touching a burger would upgrade it, and place a decal on anything else. In addition the sauce would run out after a certain number of uses, needing refilling by placing in a ‘sauce station’.
- First of all though I wanted to fix an error we had encountered previously, this being that if a player dropped the sauce bottle it would be difficult to pick up with the headset on, and impossible if it went beyond the counter.
  To fix this I placed a simple volume on the floor that would teleport the bottle back to the counter when it was dropped into it. I did have some issues here with the bottle retaining its momentum when dropped, but after discussing with another group who had the same issue it was resolved by turning on and off the bottles physics.
- Making the sauce into a projectile was slightly more difficult even though I had the framework for it available for the previous version (being squirted when in the players hand). I attempted to figure it out using this Unreal tutorial:
  https://www.youtube.com/watch?v=pdjFm7YA8vI
  This allowed me to spawn the projectile but I had some issues getting it to come from the tip of the bottle at the right angle. To solve this I had to do some testing by using the sauces tip as the location, but a separate arrow location to find a forward vector for it to shoot at.
  Whilst a burger does upgrade when shot at, I still need to add the decal spawning options, as well as a potential spread to the sauce shooting so it is not just a single projectile.
- When it came to the refilling mechanic I just made a basic asset in-engine with a volume that when the sauce was held in it would trigger an event to fill it by a percent even half a second, the opposite happening when it is squirted.
  However I did encounter an issue here in that the values range (0-100) I put on the sauces’ capacity were not obeyed allowing for minus numbers as well as those exceeding 100. This could likely be fixed though by adding a branch so the values are not affected if they deviate from the range.
- Finally I would like to add at a later point that the sauce bottle attaches to the beverage machine to make it so the player can’t move the bottle whilst it is being refilled, though this can be discussed with the whole design team.



[Above] Video showing the new source projectile, as well as it running out shown on the print screen.
[Below] Video showing the sauce respawning as well as being refilled by the machine, shot in editor.
References:
https://www.youtube.com/watch?v=nU-opWk0Xuq&list=PLGQ2tt6ABjKFiYaJ1LU8nerYfqYHogNZ2&index=13&t=0s
https://www.youtube.com/watch?v=UOUIK1_hoLw&list=PLGQ2tt6ABjKFiYaJ1LU8nerYfqYHogNZ2&index=10&t=0s



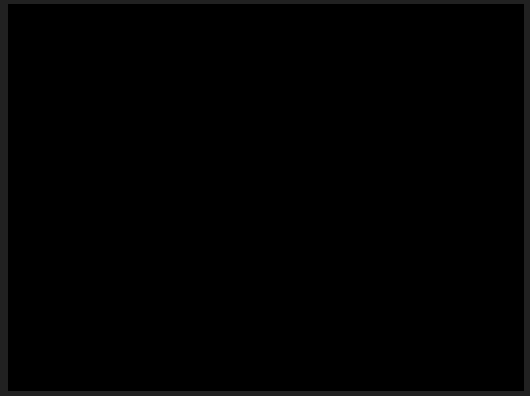
Week 5: In Depth; Dialogue Systems

- One my jobs this week was to set up the basic dialogue system for the other group members to use when implementing the audio at a later date. It is worth noting for now though that the subtitle system used displays the subtitle for as long as the sound clip used, and so won't be accurate until the actual audio is there.
- To begin with I used some tutorials on the basic dialogue system for UE4, as I lacked experience in this area; teaching me how to implement a basic line of dialogue and display a subtitle for it at the same time:
  https://docs.unrealengine.com/en-us/Engine/Audio/Dialogue
  https://www.youtube.com/watch?v=PTZCZQcXdUU
- Whilst this worked for the most part when actually ported into VR there were many issues, I did not know this previously but titles in VR follow the centre of the camera, moving with the headset. Though this seems a practical feature (players being able to see at all times), the text displays at the bottom of the screen. When a player looks down in VR the headset moves slightly making their view blurry and therefore making it difficult to look down to read text, if the entire headset moves with it.
- Even repositioning this text, if it wasn't in the centre of the players screen (obscuring their direct vision) the same effect occurs when players naturally try to centre the text to read it.
  Instead I did some research into different ways of games display information in VR.
  Remembering a playtest session with Wales Interactive I ended up looking into time carnage:
  https://www.youtube.com/watch?time_continue=26&v=FwZKWj4zFhw
  Utilising their solution of building UI into objects in the world I came up with a potential solution with Simon.
- With Simon having learnt how to embed UI elements into the environment we could potentially have subtitles appear in a dialogue box over the players arm. The issue being it wouldn't work in the games world (being in the time period of the 50's). However technically this could be perfect, as the player has control over where the subtitles are displayed (by holding their arm in a position) whilst being able to look at it with their head. This does need discussing with the group though to see if this is worth the additional time that would need going into it.
- For now though the only subtitle implemented is the intro dialogue from Poltroon, which plays shortly after the game starts, triggering the enemy waves starting when it finishes playing. This will be the basic template for the rest of the dialogue that is not played in tandem with events, allowing the rest of the group to use this when more dialogue is to be added.

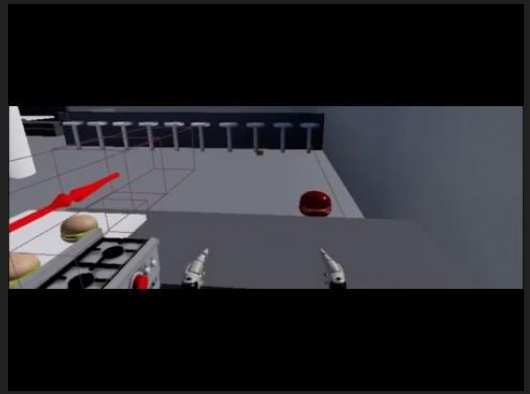
[Above] Image of the current subtitles spawning in engine, shot in editor.

Week 6: 11/02/2019

- With this being the week before the prototype hand-in a lot of the additional work was simple bug fixes and polishes to the system to make sure it was suitable for presentation.
- To begin with after repeated tests in VR the size of the counter was significantly reduced due to us using Standing VR for the prototype (meaning the player cannot move from their space). Whilst aesthetically the counter may not look as realistic now, mechanically it means players don't have to lean to reach any of the objects as they are within arms reach.
  Though it was discussed with the lecturers during a milestone meeting that in the future the size of the counter could be increased to add another layer of challenge.
- After some group discussion it was decided that the sauce system needed refining and cutting down namely so that each sauce had 3 squirts as opposed to the orginal 10. This adds a bit more resource management to the game in having to refill them and just required some tweaking with the refiler to slow it down enough that the challenge carried over to that element as well.
- When creating the additional sauces (discussed below) I made some quality of life changes as well, as soon as the game starts the sauces register their current locations to respawn at when thrown on the floor as talked about previously. This just means that if the level designers choose to move counter elements around the sauces will automatically update their locations, referring the location details into the custom gamemode and respawn correctly without additional script changes.
  A further change with this might be to check if a sauce is currently occupying the respawn location, just so the sauces don't respawn on-top of one another.
- When it came to the sauces I did have to add an additional two for the final burgers, that being the chicken (High Flyer) and the double stacker (Big Bertha), though this was largely the same blueprinting with some overridden functions and additional variables.
  In terms of the burgers themselves it was simply about increasing the blast radius and linear damping (Big Bertha) and decreasing it on the High Flyer to create a heavy and light burger respectively.
  In the future I would like to add a glide to the High Flyer though possibly by placing less gravity on it in order to give the burger some more variance.

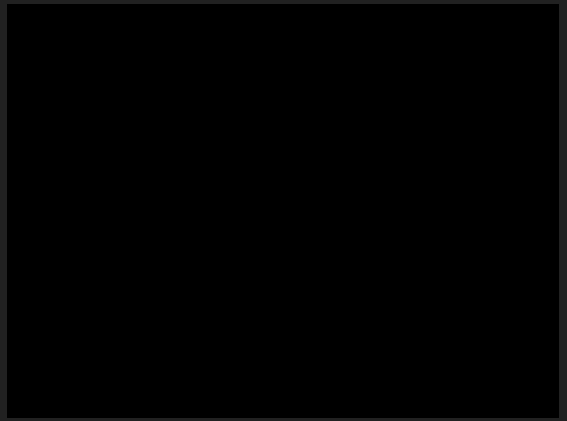


[Above] Video showing the multiple sauces refilling as well as their current projectile usage.
[Below] Video showing the differences in weight between the High Flyer (Left) and Big Bertha (Right, Model in progress), shot in editor.
Sources:
https://www.youtube.com/watch?v=na1J0iu6nl8&list=PLGQ2tt6ABjKFiYaJ1LU8nerYfqYHogNZ2&index=5
https://www.youtube.com/watch?v=B3q_bLwXZEw&list=PLGQ2tt6ABjKFiYaJ1LU8nerYfqYHogNZ2&index=8



Week 6; Continued

- One of the other tasks I was set this week was to do a trial rig of the enemy model and have some basic animations for it. Due to time constraints as well as this being my first time doing this I used the website Mixamo (https://www.mixamo.com/) which helped a lot by making the process quick as well as providing some basic animations for free. However there was an issue in that the most recent zombie model would not work with the rigger and so I has to use the old one, leading to an interesting design decision.
  We had to choose between using the most recent model with no animations, or the original model with. In the end we decided upon using the most recent model as it was more representative of the games current state of using human enemies, and would have to be fixed at a later date.
- When it came to porting the animations into engine I used the following tutorial, but luckily said tutorial is out of date as the current version of Maximo only requires a fraction of this to be done: https://www.youtube.com/watch?v=7vtdJeRscAQ
- I did encounter some issues with this though namely in giving the enemies injured animations. The main example of this is when enemies get caught in range of the El Diablo blast and are set alight. When the enemies did this for whatever reason they would be unable to change back to their original animation and would freeze on the first frame of it. Whilst this was temporarily fixed by using the Play Animation node rather than Set Animation, though I do need to reimport it with the animation set to play in place rather than the model moving forward and clipping backwards.
- My final blueprinting job this week was to teach Tyler had to cue up and import Audio into the game so he could begin putting in the basic VO work. This was just showing him how to import the sounds into array categories in the game mode so that they could be referenced easily from any blueprint using ‘Get Game Mode’ and then how to call them. My only concern here is that this could get messy when importing the version he’s working on into the newest version, as the audio cues will be present in multiple blueprints.



[Above] Video showing some test footage shot on mixamo of a possible animation when enemies are set alight.
Sources:
https://www.youtube.com/watch?v=Xo9qugbfKBg&list=PLGQ2tt6ABjKFiYaJ1LU8nerYfqYHogNZ2&index=3

Week 7- Full Production Plans

- Whilst the prototype model is in a good stage for a prototype, there are a few ideas I have for bug fixes and polish improvements in the final version of the game.
- One of the original bugs from the game is when the hob is turned on, if a burger is already in the ignition volume it will not ignite and will need to be replaced into it to actually trigger. This is due to the use of an Begin Overlap event, as when the box is activated objects within it are not counted as fresh overlap events.
  A possible solution I could use; suggested by Tyler, is to set the collision box as hidden in game until the player turns the hob on, as this would technically count the box as nonexistent until it is set to be shown in game, but will require testing.
  Another solution which I have tested is to check when the hob is turned on if any objects are currently overlapping the ignition radius, but this currently only registers one object so doesn't function if multiple burgers are present in the volume.
- After discussing with Richard a potential mechanic to consider would be that the hob randomly short-circuits and turns off, requiring the player to turn it back on at intervals. This would both fit with the rundown appearance of the diner as well as adding a mechanical use to the hobs button beyond turning it on the first time, adding tension between the enemy waves.
- Whilst not a priority there is a final change I would like to make to the hob which is the addition of button depression; the button physically moving in when the player hits it. Though this could simply be done by having it snap to a pushed in position when turned on and popping back out when it short-circuits, I have looked a more complex but realistic way of doing it online: https://bvisness.me/2017/08/27/ue4-how-to-make-awesome-buttons-in-vr/
  In short this method works by having the button snap to the current position of the players hand when pushing the button, making it look like the players hand is physically pushing the button further down, using a radius to determine the constraints of how much it can move. However it will be up to the group whether we want to put the time into a small feature like this.
- Though this may not be my task when we have the final model (or just a human iteration of it) of the enemies I would like to smooth up the animation a bit more. I would still be using Mixamo for this as it saves time and provides a better quality of animation than we could at this stage, but the animations themselves need to be chosen more carefully to match the now human style of movement, this will need to wait for the new model though. In addition I need to ask for advice on doing animation curves to make the transitions between the animations smoother as it is quite jarring in VR when you can see every detail of the enemies to have their entire body snap to a different pose.
- One of the larger issues I am repeatedly coming across with the prototype build is in regards to the sauce refilling station. There are small errors like the sauce bottles being forced out of it which is simply due to the size of the station being too small and will be fixed when the finished model is introduced. The larger issue is that when placed inside the refilling radius most of the time the bottle is not filled properly, and is ignored by the volume. Whilst this is possibly due to naming errors when merging the versions, these issues have occured before so a fix is needed beyond this. Thanks to a suggestion from Owain on the art team and discussion with Richard I believe this can be solved by using a lever or button on the station to trigger an event to refill a bottle as opposed to a simple overlap event. This would then remove the need to rely on collision as the event is being forced to trigger by the input on the machine, which will hopefully fix the issue.

Week 8- 11/03/2019

- Following the feedback I had a few points of refinement that would need to be made to the current version of the game in regards to my scripting. Whilst mechanically a lot of the solutions I had found worked, aesthetically they didn't make sense especially with the additional scrutiny provided by VR.
- An example of this was the method for respawning the sauces, though functionally it made complete sense; if a player throws the sauce bottles out of their reachable area it is brought back within their reach. However teleporting sauces does not make sense in the context of the game being a real environment in the 1950's theme.
  A suggested replacement by the lecturer was to have either a vent (above the sauces) or a hatch (beneath the sauces) that the sauces would come out of when respawned. The issue here when deciding which of these to do is the vent obscures part of the players view unless it is just on the ceiling (not overhanging), and the hatch has to make it so that players cannot block it.
- Another issue with the sauce is that the lecturers wanted to replace the projectile sauce with a stream, requiring a new particle effect as well as returning the sauces volume to percentages, when I can then also add the decals and depleting bottle mesh.
- A more group related issue that was discussed post-prototype showing was that due to the nature of the counter (modular components with fixed elements), the art team would need to be notified of the changes in the counters proportions. Whilst I did not agree with this as the purpose of modular components is removed by the inclusion of fixed elements and said so in the meeting, I agree this additional bit of communication is necessary if we are including these changes, to ensure all the counter elements fit.
- There were a few fixes that I did this week as I now knew the solution, namely when a bomb goes off in the player hand it locks the hand up. This being the same issue we had with upgrading the burgers, I just needed to force the players hand to release the burger less than a second before exploding. In addition I also fixed up the visual issue of the burger lingering after the explosion, this just reducing the bomb's static meshes lifespan to the lowest possible value so it appears to disappear in time with the explosion.

[Above] Video Showing the fixed burger explosion, [Below]
Sources: https://youtu.be/cTHvvvFUikY

Week 8- In Depth; Sauce Refiller

- A main part of my work following the prototype showing was the refining of the sauce system to remove the majority of the errors. As mentioned in my previous dev slides one of the artists suggested having the refilling event be initiated by a lever as opposed to relying on an overlap event to handle this.
  Discussing this with the lecturers this would be better achieved using a button as levers are difficult to use with VR.
- To begin with I went about making it so the sauce bottle snaps onto the machine when the player moves it within range, ensuring that the sauce bottle will remain in the detection box during refilling. The only issue here was first disabling/reenabling the physics when the player removed the bottle, then making sure the bottle snapped to the straight up rotation when attached. The first issue being easily fixed using the same method as respawning the sauces, and the second through a simple node change.
- Secondly using the button functionality built from the hob (turning on/off), I replicated a similar system in which when pressed the machine checks if the currently held bottle is full, and will continue to refill it until it is full, at which point the machine turns off.
  The machine then also manually turns off when the player removes the bottle, shown by the button changing colour.
- I did have some issues here in that the called refilling event would continue looping after the machine was turned off, allowing the bottle to be refilled if it was replaced in the machine without turning it on. However this error was resolved by applying the following forum post: https://answers.unrealengine.com/questions/431962/in-set-timer-by-event-node-runtime-looping-dosent.html
  Essentially that when using ‘Timer by Event’ looping you need to manually clear the timer to stop it looping.
- There is still an issue though with the sauce refiller sometimes allowing additional bottles to be snapped to it at a time, despite there being a variable there to stop this.
- In addition there are some slight variations I would like to add just to improve the feel of the machine. For example when a sauce has finished filling it might feel better if physics was reapplied to it, making it appear as if it is dropped from the clasps holding it, though this could cause it to fall on the floor or create other issues if another sauce is inserted immediately after.
- Another polish change might be to add a small drizzle particle effect if the sauce is removed early, showing the machine having to manually switch itself off mid pour.

[Above] Video showing the new sauce refiller, refilling a bottle to max and then turning off.
[Below] Video showing the new sauce refiller rejecting a an already full sauce bottle when turned on, and snapping.

Sources:

https://www.youtube.com/watch?v=0XLXMdgDwHI&feature=youtu.be
https://youtu.be/D_rJMwPobPE

Week 9- 18/03/2019

- With the core gameplay mainly being finalised one of the main tasks this week was the recording and implementation of the dialogue. Though we did not have a dedicated voice actor for this with the help of Tyler I recorded some of the basic dialogue and then edited it in audacity (changing pitch, removing background noise etc). The only issue doing this was an additional step in needing to download Lame; a plugin that allows audacity to convert mobile videos to audio.
- Though not of professional quality for obvious reasons, these clips serve the purpose of allowing us to test the length of the dialogue spoken when placing them in game, seeing if the pace is natural and doesn't interrupt gameplay too heavily.
- One of the bigger issues with this was overlapping audio playing whether this is sound effects or dialogue, drowning each other out. For the most part I used a band-aid solution for this in making the audio quieter than the actual dialogue, though this is only a temporary fix. Later on this is most likely to be fixed by cancelling the quip dialogue (in-game dialogue that only plays after certain actions by the player) whilst main dialogue is playing, though this will need a centralised audio player to work properly as opposed to across multiple blueprints like now.
- With the other sound effects I did some testing with attenuation settings, this mainly having to be figured out by myself. This was so that any burgers thrown beyond the counter would have a sound effect that decreases in volume as it gets further away, something that is particularly noticeable in VR.
- A final issue I had with the audio imports was some of them were in 24bit audio files, which unreal doesn't support. This was simply fixed by informing the audio engineer so he knows in the future and getting him to resend them.
- In terms of smaller features the new enemies were imported with fixed attack and movement animations, learning from last time that I need to make them play within blueprints not just setting them when changing them. In addition to this the player is able to die now. The way this works is the counter has a health of 100, this is reduced by a set amount multiplied by the number of enemies currently surrounding the counter until it hits zero and the game ends (a game over screen is yet to be implemented.
  The only issue I will currently have with this is that I need to edit it so that certain enemies do more damage, which will just be adding additional damage amounts and checking for said enemies, if this is what the group wants.

[Above] Video showing the current enemy movement animations.
[Below] Video showing the current enemy attack animations; both shot in editor.

Sources:

https://www.youtube.com/watch?v=cSHBRidKpGY
https://www.youtube.com/watch?v=_7JACgeuYvI

Week 9- In Depth; Bug Testing

- The majority of this week was spent performing bug fixes on the game, testing repeatedly to see if there were errors, noting them down and then fixing them as quickly as possible.
- One of the major issues was with the newly inserted audio for the Hob, in that when the second portion of the audio (the hob running) would play it would not be able to be turned off. I found out that this was again because audio player components don't function well with looping sounds and so disabled this as it was still a 7 minute piece of audio. The only issue in the future here would be if the hob is on for longer than 7 minutes, however with the feature of the hob cutting off being considered this would not be an issue.
- Another of the major bugs was the collision detection between the enemies and the burgers, with the burgers pushing back the enemies or causing them to clip over them. Whilst I still don't know entirely what was the issue as the collision channels became very confusing, sometimes ignoring the burgers entirely and sometimes having them collide. With some help from the lecturer though this issue was resolved so I can return to this and copy the collision settings over if this happens in another object. To make the easier we also added a feature that the bombs now explode on contact with enemies if they are lit, making the accuracy of the physics with the collision less important.
- One of the other issues I encountered was with the size of the burger explosion not encompassing the radius of the particle effect. With there being no clear way to tie the two together I had to just keep incrementing or decreasing the sizes until they roughly fit the boundaries of the particle effect. However there are still some issues with this bomb not triggering correctly, creating the radial force but not damaging enemies.
- Some of the larger issues I've encountered are to do with the construction of the blueprints themselves, for example when editing the enemy spawning. As Jordan received assistance from Dan on making this he didn't know much of how it worked when he asked me to make it so enemies were slowed down in the first wave for the sake of the tutorial. This required an extended period of time trying to figure out exactly how this worked so I could edit it without breaking the system, and whilst eventually done with everything working, in the future we need to ensure that when we receive assistance we still need to know how the system works.
- In addition we reached a compromise with the sauce refilling station, reducing the sauce bottles by half their size still makes the bottle appear to be held in the machine correctly, with the added benefit that they appear to be held within a clasp at the top of the machine.

[Above] Footage showing the new contact bombs feature of the burgers collision.
[Below] Image showing the fixed sauce bottle size in the refiller, shot in editor.
Sources: https://youtu.be/w7kq57wGG5I

Week 10- 25/03/2019

- With it being the last few weeks working on the game in the University I wanted to get as many sessions in with the VR kits as possible, testing the game in its most accurate reflection before the Easter holidays. The main component of this was getting playtesting sessions in, so far having performed two of them myself.
  We used this form of simple, non leading questions; checked over by one of the notetakers who had done a research methods module similar to mine: https://docs.google.com/document/d/15OCpfTp04wOwFIZ-KV3_WAXTsZf0kgq4f5Sm6-3q7K8/edit?usp=sharing
- The actual playtesting was very constructive in pointing out bugs as well as the narrative not feeling natural enough without the tannoy sound effects, just making it sound like something talking to you in your head.
- One of these issues was brought about by the new collision settings, where the new contact bomb functionality would cause burgers to generate constant hit events when underneath enemies, playing the explosion noise constantly and repeatedly exploding. To fix this I simply set the burgers to only trigger hit events on enemies once.
- We also recorded the second batch of audio this week, though I do want to re-record most of the dialogue again. It followed largely the same process as the first batch: Recording dialogue, cutting and editing it, importing it into engine and inserting it into the games structure. We did received some feedback about this audio though in that it can be unclear due to the pitch change what the dialogue is saying, hence the reason I wish to re-record it.
- This week we received the first batch of particle effects for the game, namely this was the hob fire effect as well as the debris created by the burger exploding. Whilst these were both of a fitting quality the hob fire effect (the one provided by the art students as opposed to Tyler) took far too long to get handed over. Despite discussion and explanation of the process (just wanting to migrate the asset from their version of the game), it took an entire week to receive the zip file as opposed to the day it took Tyler.

[Above] Video showing the particle effects for the hob and burgers residue; shot in UE4.
[Below] An example of some of the questions used in the playtesting document; made in Google docs. Link is in main text.
Sources:
https://youtu.be/b8ke9j0hVLs

1. What did you enjoy the most?

   I enjoyed the satisfying feeling of throwing the burgers and watching them take out waves of raiders, as well as the multiple different ways to attack the raiders through the use of sauces to change the burger type.

2. What did you enjoy the least?

   Some of the audio for the loudspeaker voice was illegible at times and was choppy or cut off other times. This didn't affect the gameplay too much, but was a slight annoyance as it affected my immersion within the game itself.

3. Were there any noticeable bugs? If so, what?

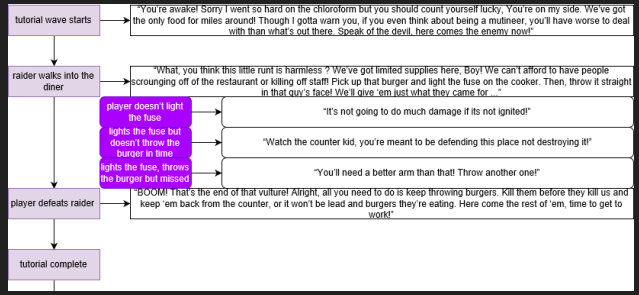
   Disable teleport.
   Burger explosions sometimes don't work (variants).

4. Would you play this in the future?

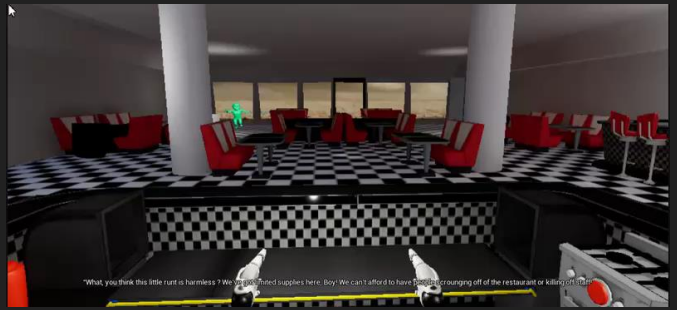
   I would enjoy playing a finished version of this game, as the current version is enjoyable.

Week 10- In Depth; Dialogue Structuring

- My main task this week was the implementation of the recorded dialogue, now that the second batch had been completed (not the final version of it). This mainly involved going through the enemy spawning system and inserting the functions to play dialogue in-between the waves, starting the waves after the dialogue had finished playing.
- The one exception to this was the tutorial ending dialogue which had to be split into two separate segments, triggering when the tutorial ended and when the first wave begins. Though this was fairly simple as it just meant sequencing the two one after the other.
- I did encounter a few issues as I briefly didn't understand that the tutorial wave was not the first wave, but this was just due to a reading error on my part and just required adding another wave into the database.
  Thanks to Tyler's document (shown right) listing out when dialogue needed to be played and at interval I did not encounter a similar issue again.
- Due to the unprofessional quality of the current recording I do have some issues with it, namely that the voice I used is garbled in places as well as abruptly being cut off where I didn't know how to edit it properly. However we have received a trial recording from our sound engineer Kieran so this should be resolved soon.
- As mentioned previously some elements of the counter have to be turned off during the tutorial such as burgers that have not introduced yet. I have managed to get this to work for the conveyer belt, cutting a piece of dialogue in two so the burgers start spawning when they are mentioned. However I did this using an inefficient solution, preventing it from triggering with a branch variable on an event tick which could effect performance. For the other features such as allowing the sauce to appear I intend to set the actor to be hidden in game and then using an event dispatch to disable this hidden setting, learnt in my last module.
- My final concern is how the dialogue system will function with the new accessibility functions, as the video could be missed if not fully in view it needs to loop and was suggested by Richard that the game pause between waves, with a button enabling the next wave to start.
  I have explained to the team that this late into the module this sort of feature is difficult to pull off especially in the two weeks in the Easter holiday in addition to other bug fixes. Instead we will wait until the start of the holiday and see how much work is remaining, and if there is too much there will just be a visual indicator to draw the players vision to the sign language video.



[Above] Image showing the dialogue tree referenced, full version can be found in links.
[Below] Video showing the tutorial elements being enabled, shot in editor,
Sources:
https://drive.google.com/file/d/1qYv0YvnD82ysSW_NqRbYKKV01hNDHRRk/view?usp=sharing
https://youtu.be/cxRf0vVJU5Q



Week 11- 25/03/2019

- One of the remaining issues from last week is that the burger particle effects were too slow in their animation, this being pointed out by the burger explosion (as well as the burger actor) spawning and then exploding before the animation played. Though this was fixed by spawning the particle effect as a separate actor, the issue still exists as it looks unnatural moving so slowly. I have mentioned this issue to Tyler to be resolved.
- Whilst I won't be doing the task myself I suggested that for the Itch.io page we use a layout similar to a 1950's diner menu as this would fit with the aesthetic of the game and make it stand out. My only issue with it is that this layout could obfuscate the information, spreading it out too much on the page. I have mentioned this to Jordan though so hopefully this does not occur, whilst keeping the theming.
- One of the larger changes this week was to the Slow Joe burger; the gravy covered burger with the effect of slowing enemies down. After discussion we decided that having the burger slow down everything in its explosive radius rather than leave a persistent slowing area would be more effective, as there would be no purpose in another area based burger if you could just use the El Diablo burger to destroy enemies rather than slowing them. The new burger variant would compliment the incendiary burger, slowing down enemies in its radius to ensure they would be killed by it.
- The only real change I had to do blueprint wise was integrate a slowing function in the enemies blueprint, and create a unique damage type so the enemy would know it has been hit by the Slow Joe specifically (casting was unreliable as the burger is destroyed immediately).
  For now the enemies' material is changed to show they are being affected, but I would like a simple decal or effect to be applied to them instead and we have contacted the artists regarding this.
- The only issue I am encountering is the exact amount to slow enemies by to make a visual difference but not make them too easy to kill, but this can be fixed in fine tuning. I am also concerned by the purpose of the burger as a normal burger does the same but removes the enemy entirely. To improve this I may increase the size of the gravy burgers radius so that it can encompass more than the basic burger making it better at slowing larger groups that the basic burger couldn't defeat in one hit.
- Another change this week was to double the health for the player, as in all my testing sessions I noticed that it decreased too quickly. Whilst this amount might not be the final value (200), that will be found out through additional playtesting.

[Above] Video showing the new effect of the Slow Joe burger.
[Below] Video showing the burgers new particle effect in action; shot in editor.
Sources:
https://youtu.be/7nwxWMlWcxQ
https://youtu.be/OjF-VfBYlnQ

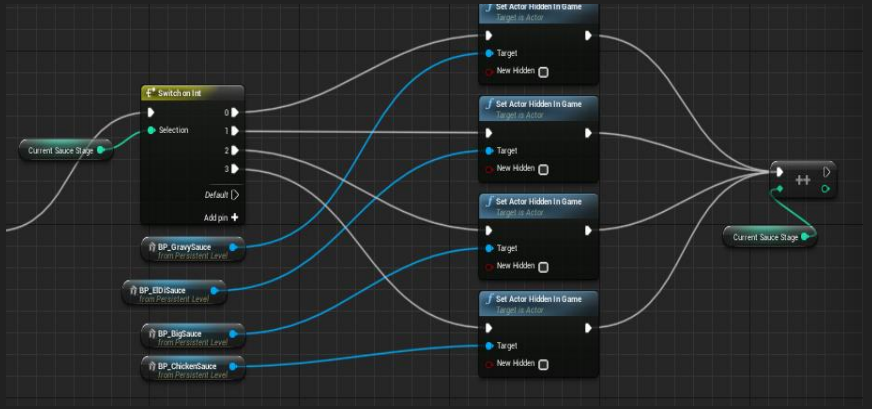
Week 11- Continued

- Most of the work conducted over the Easter holiday and leading up to it was mainly bug fixes and importing updated assets into the game, which only required a small amount of blueprint work to do.
- One of these changes was the importing of the fuse particle effects as well as the edited burger debris effects (changed so that they played quicker). These only required a quick change in making sure the burger particle effects spawned a separate emitter, as they needed to play slightly longer than after the burgers death, then adding the fuse effect as soon as the burger is lit.
- I did encounter a slight issue in that the new burger explosions wouldn't play when the burger hit an enemy and prematurely detonated, but this just meant adding in to activate the particle effect when the burger makes contact as well as after igniting normally.
- In addition this week I added in the new enemy models provided by Chris. I did encounter some issues rigging the new models in that Mixamo would keep displaying an error message saying that the joint markers weren't placed on the model. However after searching for a solution on the forums I came across this thread and many others listing that this was just an error with the software, and to keep reuploading the models until they worked:
  https://forums.adobe.com/thread/2375942
  The only change I made in these models was in regards to animation on the larger enemies' attacking animations, and though I am still unsatisfied with them for now they work better with the speed of the game than the previous slow punching.
- Following advice from Richard I have began packaging the game into a standalone product every few days to check if it runs as a separate package. For now it does run on my laptop but without a VR kit I can't test if it will run as the end product is intended. Something this did make me note though is that there is a lot of clutter in the game currently (assets that aren't used etc). In particular 'Ghost' folders keep appearing, which after some research into forum posts is just where the Engine doesn't update the desktop as to what content is stored in what folders, with no concrete way to fix it without risking losing files:
  https://answers.unrealengine.com/questions/563360/4143-ghost-folders.html

[Above] Video showing the new enemy model an animation.
[Below]
Sources:
https://youtu.be/vXr3-zWl3nU
https://youtu.be/y7jFcKjItGI

Week 12- 08/04/2019

- Continuing with the smaller changes this week I began by re-integrating the inclusion of death; a feature that was turned off during testing so I wouldn't need to reboot the game constantly. Whilst the game currently quits when the players health reaches 0, it does need to exit onto a 'Game Over' screen to prevent sickness from the abruptness of the transition from playing to death. However if we don't receive the asset for this then I will most likely have it return the player to the main menu instead.
- One of the major tasks that was finished this week was making sure the sauces spawned in tandem with the waves they're introduced. I made this using an event dispatcher in the level blueprint to just call the particular sauce to be set Unhidden in Game when called in the dialogue tree, as I had learnt during the previous module.
  Though I originally had separate dispatchers to do this I shortened it down to just one, using a Switch On Int node that increases every time a sauce is spawned to reduce the clutter. This was following advice from Dan on previous features that you should avoid using duplicate functions In-engine.
- In regards to importing work I had import the Set dressing this week (plates, cutlery, napkins) etc, which were imported, assigned materials and then sent to Jordan for adding to the level.
- With it being uncertain if we were going to receive music from the sound engineers, I decided to add in some royalty free music as a safeguard obtained from:
  https://incompetech.com/music/royalty-free/music.html
  Whilst the track does fit with the craziness of the game as well as the setting, I will have to discuss the desired tone of the music with the group to decide what direction we wish to go in. For now I have set it so the music plays from the enemy spawner (as this is where it will be altered the most), with functions to start it at each wave and stop it when dialogue is being played.
- An issue we have had consistently in the project is the appearance of Ghost folders, these being folders that are empty in the editor but contain files when viewed from a File Explorer. Whilst not a massive problem they take up unnecessary space when packaging the game. After a whilst I found the solution using this guide:
  https://answers.unrealengine.com/questions/230431/cant-delete-empty-folder-fixredirects-does-nothing.html



[Above] The condensed function for spawning sauces, advancing to the next sauce to be spawned after each use.

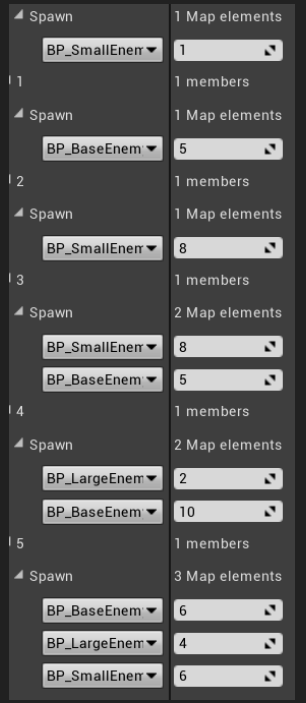
Week 13- 15/04/2019

- Most of the work this week was the continued importing of assets from the art team, applying their textures and handing them off to Jordan for decoration. Time saving here benefitted greatly from a consistent naming convention we used throughout as well as Material Templates (2 Textures, 3 Textures and an Emissive etc) allowing us to copy and paste material then plugging in the new textures instead of repeatedly rebuilding them.
- Reflecting on the re-adding of the Game Over condition last week I realised that the player currently had no clear way of knowing what health they were currently on, as we had scrapped the idea of a health bar earlier on. After thinking about it and discussing with the group I decided to use the pre-existing particle effects Tyler had built as in indication of damage, having them appearing as equipment on the counter. The particle effects are triggered in three batches when the players health reaches a certain percentage, at the amounts 150, 100 and 50.
  At the first mark sparks begin coming off of the burger spawners (being at the front of the counter they are easiest to see), at mark two these advance to smoke, with sparks begin appearing on other appliances on the counter. Once the player hits the final mark the smoke turns to fire, and any sparks advance to smoke. Though not a precise indicator hopefully players will understand the visual language of what these changes mean, as will be tested during playtesting.
- After a suggestion from Richard before the holidays I got around to adding some more utility to the hob cooker, this being that it will shut off at random intervals after being turned on. After the player has turned on the hob, any time in a range between 20 and 50 seconds after this the hob will turn off (only if it is already turned on), meaning they will have to turn it on again. This creates some added difficulty in that the player will have to take time doing this instead of throwing burgers, giving the enemies more time to attach as well as providing a purpose to the hob button beyond a one time use.
  However I do believe we could do with a sound que for this as players may believe this is a glitch without an audio que, though this is late in the module so it may be difficult to obtain.
- One of the larger issues this week was after coming back from resolving the issues with Ghost folders, the Interpreter videos used in the disability setting shad broken overnight. Originally I tried manually changing the file paths for the media sources, containing them within the game instead (so they would be carried through after packaging). This somewhat worked after reimporting the videos again but with only half of the videos functioning as materials, even though all of them could be accessed via a media player in the engine. After this took a great degree of time I handed the bug off to Jordan to fix as he had more experience with this area than I had. If we are required to go back to a previous version we should be okay migrating the fixed elements over, however this is worse case scenario.

[Above] Video showing the different stages of health depletion corresponding with the particle effects changing.
[Below] Image showing the diner after the imported set dressing had been added (not fully up to date).
Sources:
https://youtu.be/PNbSwj03EZY

Week 13- In Depth; Enemy Changes

- A bigger change this week was made to the Slow Joe burger, instead of decreasing an enemies movement speed temporarily it now applies a permanent debuff to the affected enemies. This change was made as the time it takes to build the burger as opposed to the basic burger is longer, it should have a lasting effect, allowing the player to deal with other enemies and making it more impactful. The only change that might need to be made to this is to slow down the enemies animations as it can appear janky when the enemies are moving slower but still physically running.
- In tandem with this I edited the overall movement speed of the enemies to create a significant difference between the enemies; the smaller enemies being faster and larger being slower respectively. I also resolved an issue here in that the slowed movement speed for the tutorial enemy was affecting some other enemies later on in the game, this was easily fixed though by tying the movement change to a variable instead of a timer during the tutorial.
- Continuing with difficulty adjustments, I noticed the difficulty of the game was currently too easy, the easiest way I thought to change this was upping the number of enemies to the adjacent amount. After testing this it now improves the usefulness of some of the burger variants, as players now have to deal with larger mobs of enemies that one burger bomb cannot destroy. Compared to the previous version where the players health would rarely fall beyond 180/200 by the end of the game, I found that regularly the players health would be between roughly 90-50 at that point. However as I have played the games many times at this point, it will be important to see how much this affects new players in Jordan's playtesting.
- This week we also received some enemy variants as well as the final textures for the enemies, there have been some issues with these though. As Chris created the hair and hats for the enemies as separate meshes they aren't part of the original models, this means they are not attached to the skeletal mesh of the enemies. This creates the issue when the enemies are animated as the variant assets don't move with the skeletal mesh which is very jarring. I think this can be fixed by including a socket component to attach the variants to the enemies, at which point I will include a randomiser when they spawn whether the hat or hair is visible.



[Left] Video of the bugged enemy variant assets.
[Right] The adjusted numbering for enemies spawning in each wave.

Sources:

https://youtu.be/2ISh04whkQc

Week 14- 22/04/2019

- A good deal of the work this week was arranging to take screenshots for the games promotional material, to do this I duplicated the most finished (at least visually) of the builds to create the photobooth in. Creating the screenshots in-game would have been extremely difficult (opening and closing the game to capture moments). Instead I placed assets in the level as well as animations in situations that would be similar to the actual game as not be misleading. With Jordan's request I made these as manic as possible to get across the chaos of the game, though I did have to redo them twice due to the new lighting build coming in.
- One issue I did encounter with the new assets being imported was what I originally thought was an error with the blueprinting for the enemies, in that waves would not progress past the second. I discovered though that due to the increased number of enemies, when they would spawn simultaneously some enemies would be pushed out of the navigation area and were getting stuck outside the building, preventing progression. To solve this I moved the spawn points and have tested multiple times with all the waves if this occurred again (as enemies choose from the spawn points randomly), so far there haven't been any further issues.
- In relation to this I did have to downgrade the number of enemies in certain waves, namely the number of small enemies in later waves. The issue here was due to their speed they are already significantly more difficult to deal with, so the most obvious solution was to reduce their numbers in comparison to other enemies.
- With the larger enemies I had to do some fixing as well, as they would die on the first hit with a burger bomb (as they were meant to take two to kill). There were two issues here, that I had used the wrong node with the number of hits needing to kill being lower than 2 as opposed to the desired greater than 2. Secondly the damage would naturally trigger twice from a single burger bomb, due to using the AnyDamage and AnyRadialDamage together. To fix this it was as simple as removing the AnyDamage node and switching to the latter.
- After considering this change I also got approval from Jordan to change the Big Bertha bomb to destroy the larger enemies in one hit, giving it a more specialised use. To do this I created a special damage type produced by the Big Bertha which overrides the multiple damage function, killing the larger enemies immediately.

[Above] One of the promotional screenshots with the new lighting build.
[Below] Video showing the larger enemy now dying in multiple hits, both shot in editor.
Sources:
https://youtu.be/Wc8aX2mATgg

Week 14- Continued

- After discovering the issue with the enemy accessories last week I did some research into how socketing worked in UE4 using this article:
  https://answers.unrealengine.com/questions/80647/how-can-i-attach-a-static-mesh-to-a-bone-with-anim.html
  Whilst I had believed the issue was the enemy mesh being inherited preventing me from attaching to sockets, it turned out I just needed to make the accessory a direct child of the enemy mesh and this worked as shown adjacent.
- Following this I also did the same with an Unreal asset pack a friend had told me was free currently which I have sourced here and on the Itch.io page:
  https://www.unrealengine.com/marketplace/en-US/slug/melee-weapons-pack
  The process with much the same here, aligning the object in the correct place though there are some issues with the enemy not gripping it correctly when in the running pose, it is at an acceptable level currently though.
- In order to vary up the enemies appearances I set up a randomiser to determine their appearance and weapon using an EventBeginPlay node to do it as soon as they spawn. This works by using a Switch on Int node, picking a random value between 0-2 (3 variants) and hiding the required assets to get the desired appearance, repeating this with the weapons. A modified version of this was used for the smaller enemy (just changing up the assets used in places) then removed entirely for the larger enemy.

[Above] One of the promotional screenshots with the new lighting build.
[Below] Video showing the larger enemy now dying in multiple hits, both shot in editor.
Sources:
https://youtu.be/lW1k3L9511o
https://youtu.be/EfadLUO2tZA

References 1

Week 1:

- https://rankedboost.com/zelda-breath-of-the-wild/food-cooking-recipes/
- https://www.youtube.com/watch?v=WfZCHG4B9X0
- https://www.youtube.com/watch?v=HnR1Gf5gXcY
- https://youtu.be/xzCws6d_YgA
- https://youtu.be/N0X6DyEZpPI

Week 2:

- Unreal Engine VR Cookbook; Mitch Mccarfrey.
- https://www.youtube.com/watch?v=aqRUe7pm4Ok&t=8s
- https://www.youtube.com/watch?v=dbLFHflVleg
- https://youtu.be/WY0nSaZ011w

Week 3:

- https://www.youtube.com/watch?v=aqRUe7pm4Ok&t=8s
- https://www.youtube.com/watch?v=HnR1Gf5gXcY&t=568s
- https://youtu.be/3l7cBNGAs_g
- https://youtu.be/G4Ue-D6DdBY
- https://youtu.be/R1r4hWtWB_I

Week 4:

- https://realtimevfx.com/t/ue4-spawning-decals-at-particle-collision-locations/6677/3
- http://fargesportfolio.com/wp-content/uploads/2018/07/ViveMapping.png
- https://www.youtube.com/watch?v=co9ZZnfH9Pk&list=PLGQ2tt6ABjKFiYaJ1LU8nerYfqYHogNZ2&index=14&t=0s
  https://www.youtube.com/watch?v=7UsEUafpWZo&list=PLGQ2tt6ABjKFiYaJ1LU8nerYfqYHogNZ2&index=7&t=0s
  https://www.youtube.com/watch?v=aLbhiBTTPbk&list=PLGQ2tt6ABjKFiYaJ1LU8nerYfqYHogNZ2&index=2&t=0s

Week 5:

- https://www.youtube.com/watch?v=pdjFm7YA8vI
- https://docs.unrealengine.com/en-us/Engine/Audio/Dialogue
- https://www.youtube.com/watch?v=PTZCZQcXdUU
- https://www.youtube.com/watch?time_continue=26&v=FwZKWj4zFhw

Week 6:

- https://www.youtube.com/watch?v=7vtdJeRscAQ
- https://www.mixamo.com/
- https://www.youtube.com/watch?v=na1J0iu6nl8&list=PLGQ2tt6ABjKFiYaJ1LU8nerYfgYHogNZ2&index=5
- https://www.youtube.com/watch?v=B3q_bLwXZEw&list=PLGQ2tt6ABjKFiYaJ1LU8nerYfqYHogNZ2&index=8
- https://www.youtube.com/watch?v=Xo9qugbfKBg&list=PLGQ2tt6ABjKFiYaJ1LU8nerYfqYHogNZ2&index=3

Week 7:

- https://bvisness.me/2017/08/27/ue4-how-to-make-awesome-buttons-in-vr/

References 2

Note: Gameplay videos are in Green.

Week 1:

- https://youtu.be/cTHvvvFUikY
- https://www.youtube.com/watch?v=0XLXMdgDwHI&feature=youtu.be
- https://youtu.be/D_rJMwPobPE
- https://answers.unrealengine.com/questions/431962/in-set-timer-by-event-node-runtime-looping-dosent.html

Week 2:

- https://www.youtube.com/watch?v=_7JACgeuYvI
- https://www.youtube.com/watch?v=cSHBRidKpGY
- https://youtu.be/w7kq57wGG5I

Week 3:

- https://docs.google.com/document/d/15OCpfTp04wOwFIZ-KV3_WAXTsZf0kgq4f5Sm6-3q7K8/edit?usp=sharing
- https://youtu.be/b8ke9j0hVLs
- https://drive.google.com/file/d/1qYv0YvnD82ysSW_NqRbYKKV01hNDHRRk/view?usp=sharing
- https://youtu.be/cxRf0vVJU5Q

Week 4:

- https://youtu.be/7nwxWMIWcxQ
- https://youtu.be/OjF-VfBYlnQ
- https://youtu.be/vXr3-zWl3nU
- https://youtu.be/y7jFcKjltGI
- https://forums.adobe.com/thread/2375942
- https://answers.unrealengine.com/questions/563360/4143-ghost-folders.html

Week 5:

- https://incompetech.com/music/royalty-free/music.html
- https://answers.unrealengine.com/questions/230431/cant-delete-empty-folder-fixredirects-does-nothing.html

Week 6:

- https://youtu.be/2lSh04whkQc
- https://youtu.be/PNbSwj03EZY

Week 7:

- https://youtu.be/Wc8aX2mATgg
- https://youtu.be/lW1k3L9511o
- https://youtu.be/EfadLUO2tZA
- https://answers.unrealengine.com/questions/80647/how-can-i-attach-a-static-mesh-to-a-bone-with-anim.html
- https://www.unrealengine.com/marketplace/en-US/slug/melee-weapons-pack