CookInVR

Game concept

The game to be developed is a **little cooking game** inspired in the VR game Job Simulator and the 2D game Cook, Serve, Delicious. It's thought to be played in **high-end VR headsets like the Vive, Rift or Windows Mixed Reality**, but if it comes up well maybe it can be ported to mobile VR headsets or even standalone as I would like to experiment with Oculus Go. The game is **suitable for casual gamers** and more generally for anyone that wants to spend a good time interacting with different objects in VR in order to make and serve some food.

Roughly speaking, **the game puts the player inside a kitchen where he will have to make different orders** that will come from clients, having to chop, boil and put together the different ingredients needed for the requested order. There will be a timer for each order so to put a bit of pressure on the player and make it more fun, engaging and challenging.

After some reflections on the idea I **decided to stick to the minimum viable mechanic to keep things simple at first**; so no clients or other NPCs, no complex animations, etc. I'll enhance the gameplay workflow as I iterate over it later. So, the basic gameplay workflow I would like to get working for a first version is the following:

1. Next order magically appears in the Order Box (represented as a kind of cartridge or something similar; idea inspired on Job Simulator and First Touch VR games).
2. Grab the order and place it in something we will call the Order Machine which will make the recipe for the order to be shown on a kind of 3D screen in front of the user workstation.
3. Pick the ingredients from the shelves on the sides or the Fridge Box.
4. Place the ingredients one by one on the little Assembly Board that's above the main allowance according to the recipe.
5. Once the order is ready to go, touch the Dispatch Order Button located on the Order Machine.
6. Grab the next order and continue playing.

I think **this is a more simplistic scenario than the one I have initially thought, which will force me to implement prototypes for the most basic mechanics** (grabbing orders, show order information, grabbing ingredients and putting them together) as well as try them together in their most basic form. So I'll stick to this idea for now, and hope I can have enough time to further improve it later.

State of the Art

Surprisingly, I **have found a bunch of cooking/food serving VR games on the Steam Store based on a similar idea**, some of which are listed below. I've checked them all and took some notes about a few possible features and improvements to add to my own.

Job Simulator

[VR The Diner Duo](#)

[The Cooking Game VR](#)

[Counter Fight](#)

[Dead Hungry](#)

[Taphouse VR](#)

Storyboarding

I started the storyboard on paper, taking the chance to iterate a bit about how all the cooking machinery will be used (a bit of interaction design as part of the prototyping process). Once I had a convincible version of each one I put them together to get the main scenario in which the user will be immersed. Later on I thought aloud about the whole workflow of the story and moved all what I had to a vector graphics version. Continued storyboarding the rest of the workflow upon this.

So, first some of the prototypes on paper (I really suck at hand drawing!):
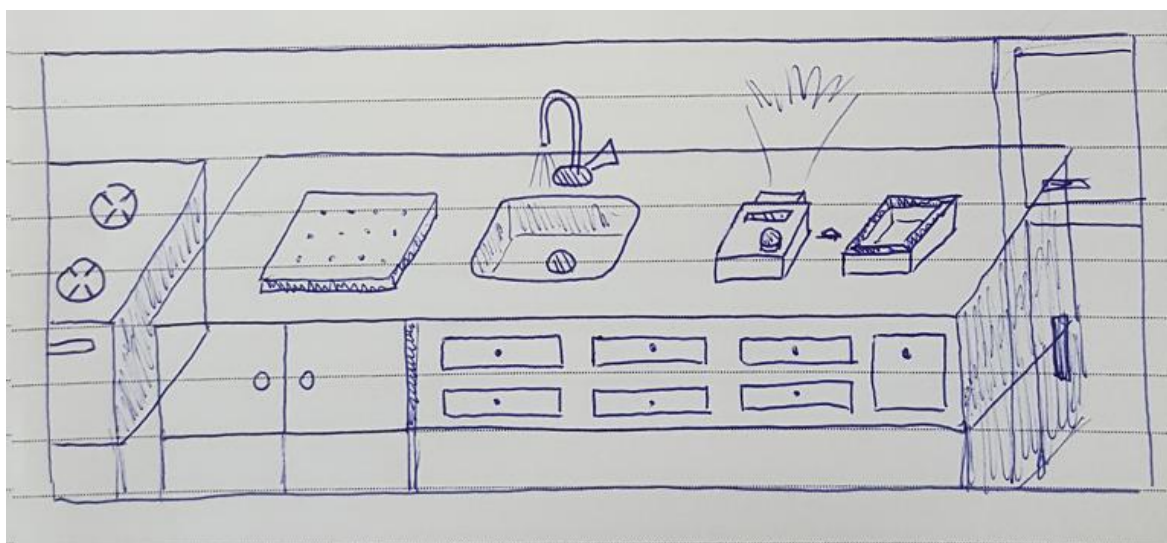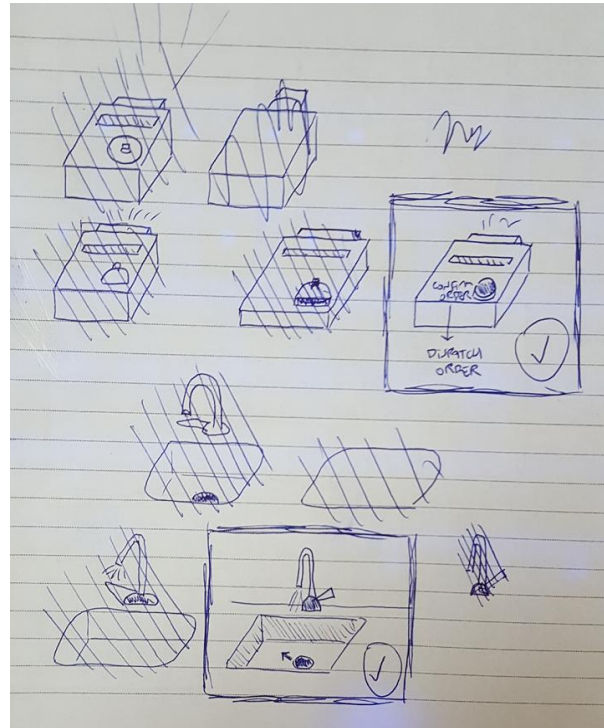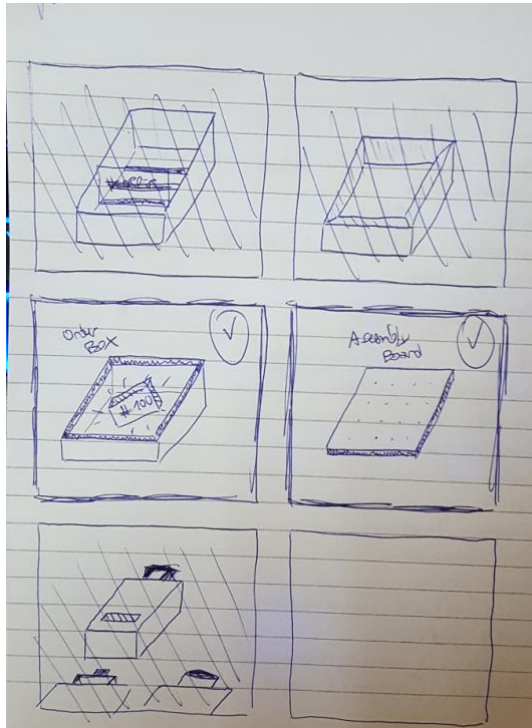
And the final version of the storyboard in digital vector graphics below. I'm not an expert with graphic design either but having come back down to earth the idea with the paper prototypes before, making these took just a few hours of vector graphics tasks mainly:

Order
#76

Dispatch
Order

Order
#77

🍔 **Burger** ✅

WELL DONE!

Next order on the way...

Order
#76

Order
#77
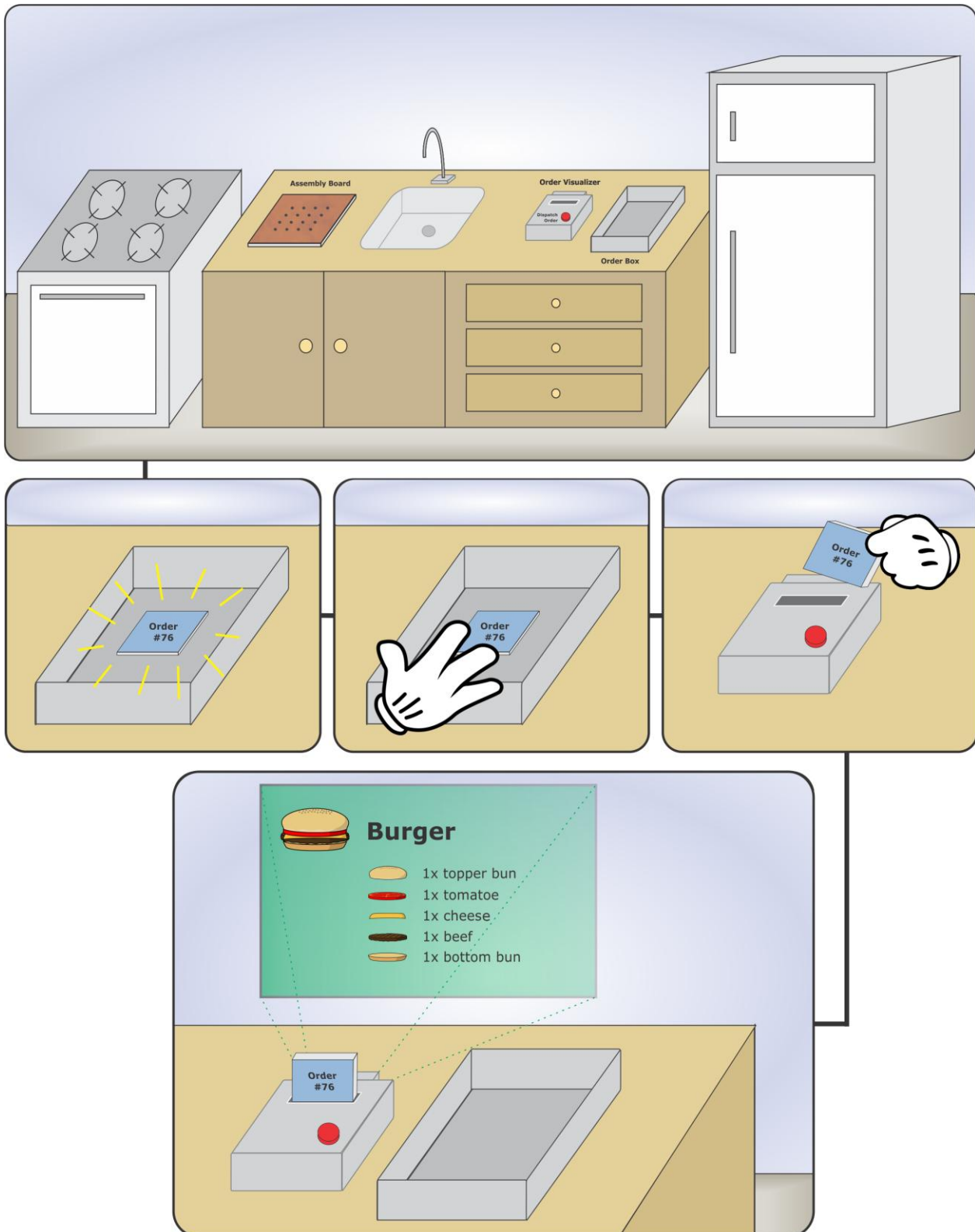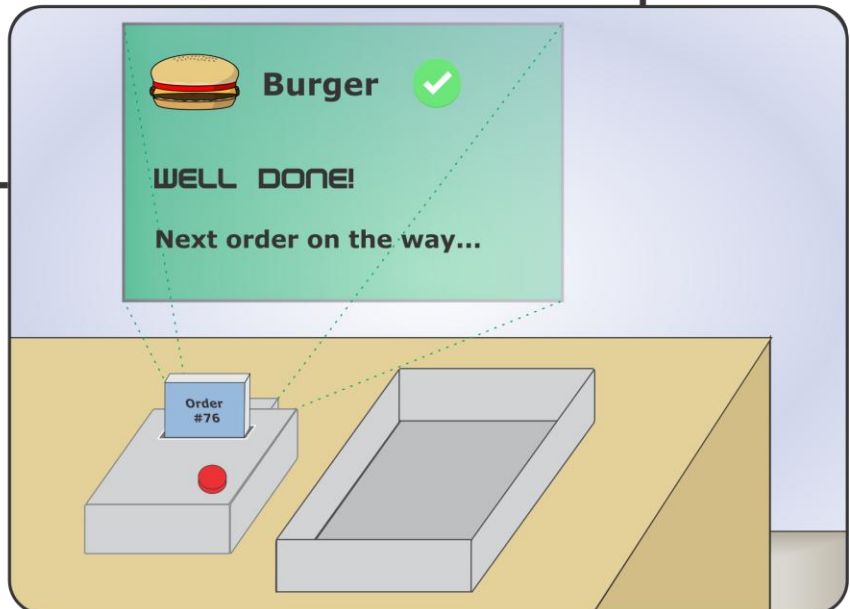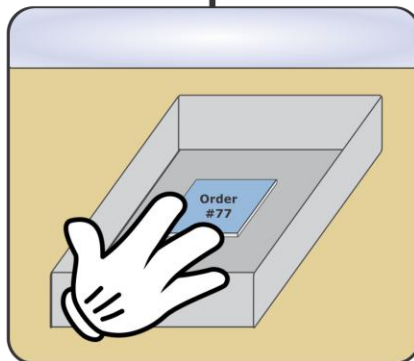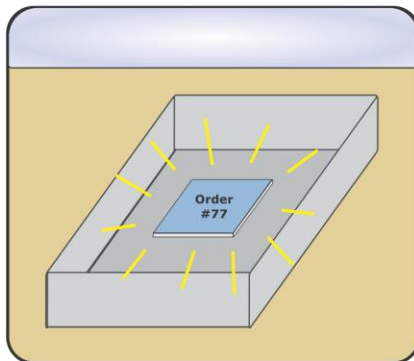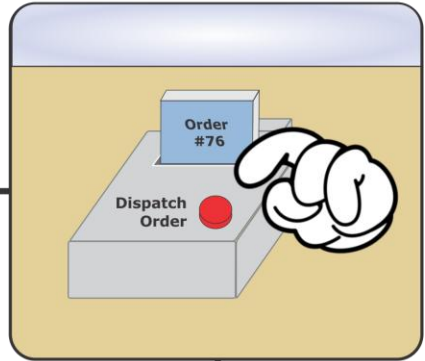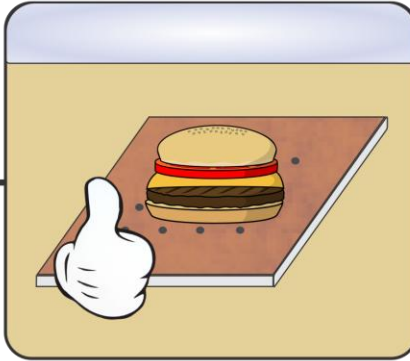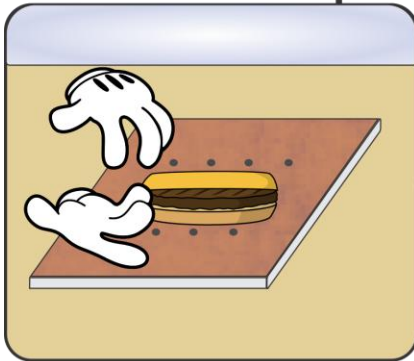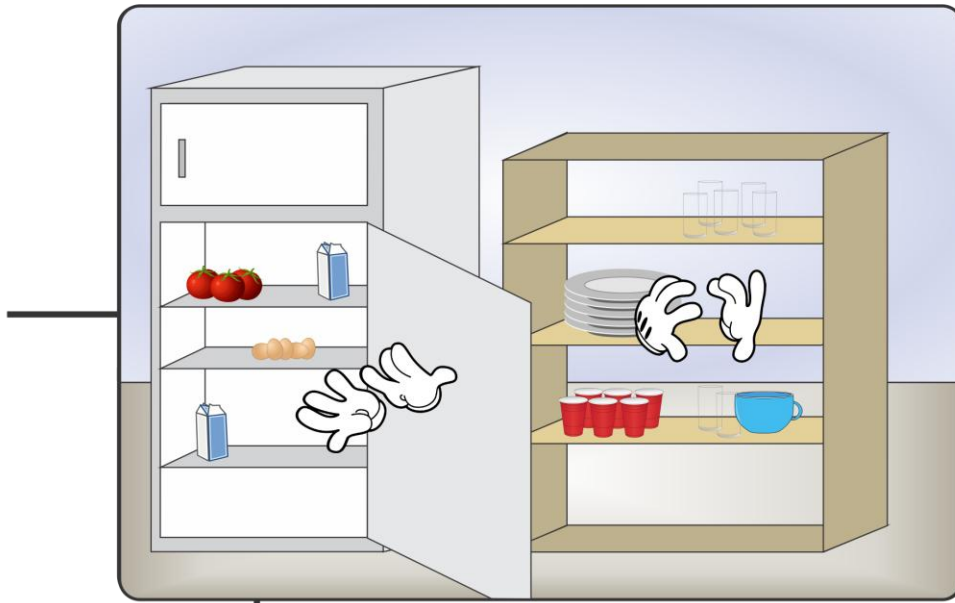
Why is it suitable for VR?

Mainly because of the natural interaction requirement, which is feasible in VR because of the positional tracked motion controllers. So the **player will be able to reach objects with their hands to grab them, put them in another place**, or do whatever they want with them. Doing all this with the more classic ways of interaction in games such as mouse and keyboard or even a gamepad would not be as fun and engaging as with VR.

Graphic Style

The **graphical style will be low poly** as I guess it will give a funnier, colorful and more pleasing aesthetics for this type of game, in addition to conveying a nicer mood to the player. Also, as I'm not an expert with 3D modeling, in case I would have to model something, having a low poly style helps avoid taking care of the subtle details so it will be enough to make some simple models with not so much details. I'm planning to **use Google Blocks for modeling the game assets,** or at least for some of them.

Also, as I won't be using any additional body tracker, I will only have the head and both hands tracked. I'll try to use (find or model) some nice chef's hand models, maybe a pair of kitchen mittens or something, so the player seems himself more like a chef.

Interaction

The interaction will be natural as the player will use the positional features given by the 6 DoF (Degrees of Freedom) **motion controllers to grab and interact with the different objects**. Similarly, the **locomotion technique to be used is just real walking** as the user will be physically walking inside the room to reach the required objects in the virtual environment.

In a future iteration I'm planning to add multiple cooking areas each one with different ingredients and types of food, so in that scenario, the user will **probably use teleportation to move between each cooking station**, making the navigation approach easy and motion-sickness free.

There won't be any social interaction for now, maybe only NPC/agents if I have enough time. An interesting addition in a future iteration if everything goes well would be adding multiplayer support so players can compete somehow. In order to do this I will have to take into account some social interaction techniques (how to represent the avatars and make them socially expressive, how to interact with them, etc) as well as adding all the networking stuff.

Discomfort

As the players will be using the positional tracking features of the PC VR headset, they shouldn't feel much discomfort. As mentioned earlier, the navigation techniques used will be **real walking and (possibly) teleportation**, both of which are proven not to be much nausea inducers.

Prototypes

The idea was to include the following in the initial prototypes:

1. The **basic interaction with orders:** grabbing an order from the Order Box and putting it into the Order Machine.
2. The **basic visualization of orders:** as part of the Order Machine once an order is placed on it, i.e.: show ingredients of the recipe to be made.
3. The **basic interaction with ingredients and recipe making:** grabbing ingredients and stacking them together in the Assembly Board to make the recipe, and confirming the order when it's ready by pressing the Dispatch button on the Order Machine.

Exclude the following:

1. **Any custom 3D model**. Use just native primitives as placeholders or assets from the Unity Asset Store or other 3D model stores instead.
2. **Coding the interaction logic from scratch**. Use a predefined library for interaction instead. Initially it will be the Interaction System from the SteamVR Unity plugin, but I'm also planning to try at least the interaction features of VRTK or even the fully physically-based interaction library NewtonVR, and see which one is more useful/efficient for the required tasks (navigation will be real walking so that feature is not really relevant in either case).
3. **Graphics, audio or any other aspects not relevant for the core interactions mechanics**. Put focus on interaction instead, i.e.: grabbing stuff and making the required tasks.

I managed to fulfill all the above points, except for a few sound effects I added mainly for debug purposes (i.e.: a debug sound gives some clues of what's happening without the struggle of putting the headset off and checking debug logs on the monitor). Also, as I had some spare time I have also added a very simple environment to start getting sense of how it feels overall.
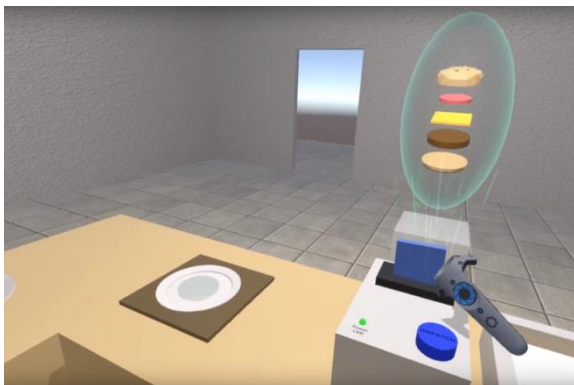
| Order in-slot placement | Recipe making / Ingredient placement |
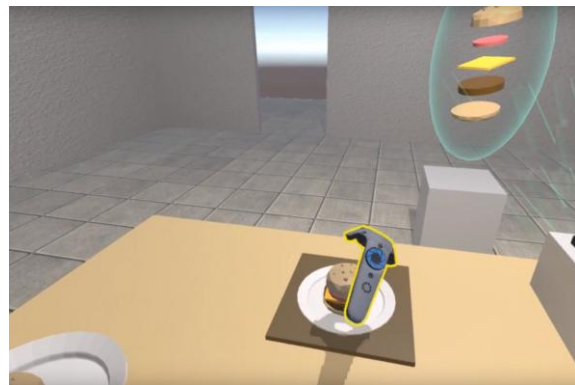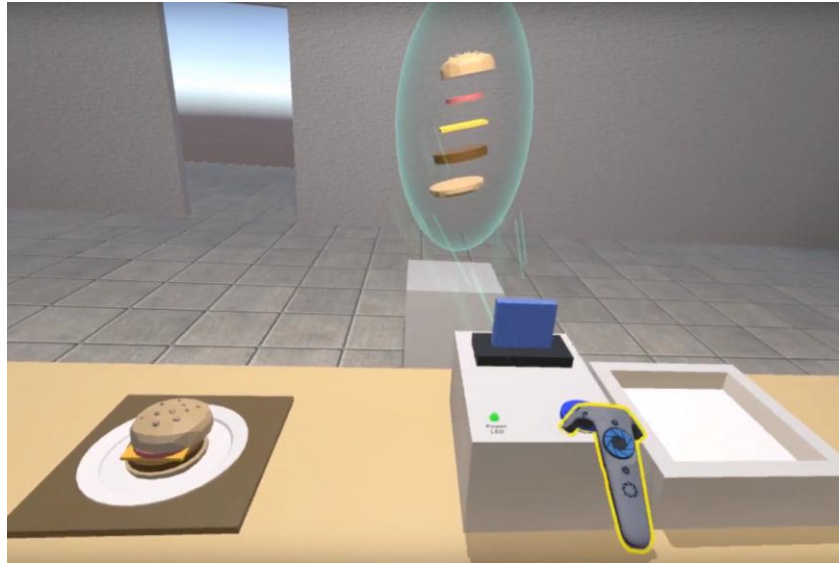
Order dispatch

Because I decided to use the SteamVR Interaction System initially, I **had to implement a few interactions on my own, such as pressing a button, placing an order inside a slot and placing a new ingredient** in the assembly line stack, and test how natural those interactions felt as they can be implemented in a lot of different ways with a lot of different behaviors. These are things that can't be learned or inferred just with a storyboard as with it you can visualize the interaction workflow but you can't check how well it matches the user expectations in practice. For example, the button pressed action should be triggered when the button is down or when it's released after pressing it? And all this kind of more subtle considerations/decisions related with the interaction. User Testing should result in further improvements in this line.

Check the following videos to get a little idea of how this initial prototypes evolved and how this short workflow implemented as part of the prototype works (grab order, place it in order machine, visualize order, make recipe, dispatch it when ready). Anyway, in the *Demos* section there is a table including not only the prototype versions but also the most recent ones.

https://www.youtube.com/watch?v=V5V5sP47X5c

https://www.youtube.com/watch?v=utdpCmk1VuE

https://www.youtube.com/watch?v=kjMlH1FRab0

Testing Plan

The following subsections detail the testing plan to be carried out based on the above prototype.

What to test?

What aspects of the game will you test? Are you comparing two different things? What do you want to find out?

I'll put focus **mainly on the intuitiveness of the different interactions included in the game**. For example, how intuitive it is to realize how to take a new order and where to put it once taken, where to place the ingredients to make the recipe, etc. and how natural it is to interact with the multiple elements in general. Basically I want to find out if all the workflow is user-friendly enough as it is implemented right now, checking whether the users can make a recipe without even knowing how to do it beforehand; just placing the headset and putting him to work. In other words, the idea is to check how self-explanatory the experience is as well as finding possible confusing situations that may occur; I'll be giving directions as needed though. Additionally I would like to find out if the users realize that they can do some juggling with the food while making the recipe (as in the last video in the *Prototype* section) and how easy it is for them, so later I can eventually make the scoring mechanism to take into account a few extra points based on the amount of juggles they make with the ingredients. Anyway, if implemented, this would be just a little extra feature and won't change the main overall mechanic.

I'm **planning to make a different version of how the ingredients are placed on the assembly board to make the recipe**, but I have yet to implement it. Basically, in the first prototype, the user has to put every ingredient in order over a kind of transparent mark that appears for each one in the assembly board (check first prototype videos above!), and as soon as it's placed the ingredient is automatically snapped to the center of the assembly board. With this solution there's no place for errors as the game is telling the user which ingredient to put next and won't show the next one until he places the actual/correct one, so the user will always end with a successful recipe.

The alternative would be a more fully physics-based approach in which the user just puts the ingredients in whatever order they want in whatever position they want on the assembly board, and before dispatching the order the resulting recipe is compared against the expected solution to check if all the ingredients are present, as well as whether they are placed in order and in the correct position or not, in case these are relevant. This method would allow the user to get orders with errors and eventually making him earn fewer points in the future if there are too many mistakes and that kind of things, so I guess it could be more engaging and challenging in a good way. If I'm able to have this alternative approach ready before user testing I'll include it in the evaluations so I can compare both solutions to the recipe-making process.

About the Testers

Who are your testers going to be? Why they are appropriate to your game? Are they as diverse as your potential players?

The testers will be mainly **casual and hardcore gamers from different ages (on their 20's or 30's), already used to how a VR system works**. I will try to find some users with less experience in VR just for the sake of intuitiveness evaluation but I won't stress too much if I don't find them as the game is aimed to people used to VR.

I'm planning to told the testers that the app is a game about making burgers, as for now it's the only recipe that can be made, roughly explain them what elements they have at disposal (order box, order machine, assembly board and different ingredients) and then just put them the headset. The idea is that they find out how to make it through the process with as little as possible information about the elements they have at reach in the scene. So the task at first will be that the user is able to make at least one burger successfully in a certain amount of time.

Data to collect

What data will you collect?

I'll try to collect data related to **how much time the user took to find out how to do the different stages of the workflow**; i.e.: where's the order, what to do with an order, what recipe to make, how to make the recipe and how to dispatch it. Also, as I haven't a scoring mechanism in place yet I'll just check **how much burgers they successfully made** during the pre-defined time interval. All this data will be objective and quantitative and it will be obtained by taking notes in parallel with each user session.

On the other hand, I'm planning to collect some data about **how intuitive the user think the process is and whether they came up with any issues** and/or there was something that blocked them. Additionally, I would want to know whether they find the mechanic fun as well as any improvements they think that can be made to enhance the experience despite it being just a minimal prototype yet. This will be more subjective and qualitative data and it will be included in the form of post user-testing interviews.

Interviews & Questionnaires

All of the questions you will ask in interviews or questionnaires. Will you ask them in conversation or will you give a written questionnaire?

The post-testing session interview will include the following questions:

1. <u>Taking an order</u>
   a. Did you find out how to take an order? [qual]
   b. How easy was to find it out on a scale of 1-10? [quant]
   c. Did you find any issues when dealing with orders? [qual]
2. <u>Check recipe to make</u>
   a. Did you find out how to know what ingredients did you need for making the order's recipe? [qual]
   b. How easy was to find it out on a scale of 1-10? [quant]
   c. Did you have any issues with checking recipe ingredients? [qual]
3. <u>Making the recipe</u>
   a. Did you find out how to actually make the recipe? [qual]
   b. How intuitive was the process on a scale of 1-10? [quant]
   c. Did you encounter any issues while making the recipe? [qual]
   d. Can you think of some way of improving it? [qual]
4. <u>Order dispatch</u>
   a. Did you find out how to dispatch the current order? [qual]
   b. How easy was to find it out on a scale of 1-10? [quant]

5. Did you find any issues or blocking situations that did not allow you to complete an order? [qual]
6. How much do you think the scale of the objects in the virtual environment is appropriate on a scale of 1-10? [quant]
7. What did you like most about the game? [qual]
8. What did you like least about the game? [qual]
9. How much did you like the game mechanics on a scale of 1-10? [quant]
10. Did you feel disoriented or experienced discomfort or nausea? [qual]
11. Would you change something in the game? Why? [qual]
12. Why did you stop playing? [qual]
13. Would you play the game again? [quant]
14. Would you recommend the game to other users? [quant]
15. Do you have any other general feedback or advice? [qual]

Consent Form
Things that your testers are agreeing to do before they start the experiment.

As I won't be using video recordings or any stuff that could go against user's privacy, I'll be asking the users **permission to just collect and eventually share some basic data**, such as first name, age, gender, and VR experience (on a scale from 1-10). Also, as the users already will have some level of experience with VR and as the game's locomotion approach is quite natural and the testing session will be rather short, they won't probably feel sickness at all.

> *Name:*
> *Age:*
> *Gender:*
> *VR Experience [1-10]:*
>
> - By signing this consent form I declare to be aware that above information about me may be publicly shared for User Testing purposes.
> - By signing this consent form I declare to be aware that the session might lead to slight nausea and/or any other type of motion sickness.
> - By signing this consent form I declare to be aware that I can end the session at any time.

Why testing this way?
Why you chose the testing methods that you did?

As the testing sessions will be more on the short side due to the prototype's lack of features or at least the chance to make multiple recipes in the actual status, I decided to go with quick spoken interviews at the end of each session. This will make the user testing more dynamic and avoid making the user to lose time writing down his answers.

Anyway, thinking twice about this point, I realized that it make some sense to include the quantitative questions as part of a written questionnaire and just leave the usually more subjective qualitative ones to be asked in person to the user. But just thinking aloud about this, I'll attach with just an interview for now for the sake of simplicity.

User Testing

The idea was to user test the prototypes with at least 3-4 users but due to lack of time and given that the sessions I was able to carry out ended up taking more time than expected, I was able to complete just two sessions. Anyway, I get a lot of insights from just those. The results of each session is detailed in the following subsections.

User 1

Name:	Maru
Age:	26
Gender:	Female
VR Experience [1-10]:	7

- By signing this consent form I declare to be aware that above information about me may be publicly shared for User Testing purposes.
- By signing this consent form I declare to be aware that the session might lead to slight nausea and/or any other type of motion sickness.
- By signing this consent form I declare to be aware that I can end the session at any time.

Data collected during user testing session

Prototype 1 (fixed ingredient placement)

- *Estimated time to locate the order:* 20 sec
- *Estimated time to take the order:* 15 sec
- *Estimated time to realize what recipe to make:* 5 sec
- *Estimated average time to make the recipe:* 10-20 sec
- *Estimated time to dispatch the recipe once done:* 15 sec
- *Number of started recipes:* 6
- *Number of completed recipes:* 5
- *Notes taken:*
    - Couldn't finish one of the recipe because of a bug when making the recipe that messed up the already placed ingredients. I should try to reproduce it...
    - Another bug is that for some reason when placing what seems to be the fourth ingredient of an order, the placeholder turns red despite the ingredient is the correct one (this should happen only if the ingredient is the wrong one). I should try to reproduce it...
    - She grabs and throws random stuff after finishing making orders. Seems everyone likes to do this in VR!

Prototype 2 (flexible ingredient placement)

- *Estimated time to locate the order:* 5 sec
- *Estimated time to take the order:* 5 sec
- *Estimated time to realize what recipe to make:* 2 sec

- *Estimated average time to make the recipe:* 10-20 sec
- *Estimated time to dispatch the recipe once done:* 5 sec
- *Number of started recipes:* 6
- *Number of completed recipes:* 6
- *Notes taken:*
  - A bug when dropping an ingredient just a bit below the assembly dish makes it to disappear inside the allowance. Because ingredient placement is physics-based in this case this behavior is expected, but it should be solved somehow to avoid confusion...
  - The hot dog turned out to be a bit difficult to make because of the physics and the fact that you should place a small ingredient (the hot dog) inside a little slot in the underlying ingredient (the bread). So if you just drop the hot dog inside the slot it works fine, but when you make contact with the bread a lot of weird physics things happens.
  - In general, seems to be a lot of issues related with the physics of the elements. The user wants to put something above an element but it gets actually below it because of the physics. The user don't know exactly how physics works (and indeed he shouldn't!), and that an element it's not supposed to touch or pass through another one when placing it because the physics will activate and mess things up. A way of avoiding this kind of errors should be found...

Data collected after user testing session

1. Taking an order
   - *Did you find out how to take an order?* Yes, but not so easily. After taking some time to observe the scene I realized that the cartridge must be placed on the machine. Maybe it's good to add some kind of indicator (arrow or something) showing that the order must be grabbed and placed on the machine.
   - *How easy was to find it out on a scale of 1-10?* 4
   - *Did you find any issues when dealing with orders?* Kind of, the one mentioned above.
2. Check recipe to make
   - *Did you find out how to know what ingredients did you need for making the order's recipe?* Yes, it was fast and easy, the hologram showed up as soon as the cartridge was placed.
   - *How easy was to find it out on a scale of 1-10?* 8
   - *Did you have any issues with checking recipe ingredients?* No, but I think maybe adding a better texture for each of the ingredients may help to recognize them.
3. Making the recipe
   - *Did you find out how to actually make the recipe?* Yes, the dish at the left must be used but maybe it could have some kind of indicator or something.
   - *How intuitive was the process on the 1st prototype on a scale of 1-10?* 7
   - *How intuitive was the process on the 2nd prototype on a scale of 1-10?* 7
   - *Did you encounter any issues while making the recipe?* Yes, some ingredients disappeared, and others were superposed with already placed ingredients when trying to place them.

- *Can you think of some way of improving it?* Yes, use arrows or something that the first time shows how to make the recipe. Maybe a good idea is adding some kind of initial tutorial that shows you how to make it, bringing the different ingredients to the main dish and stacking them together until the recipe is completed. Only after that, it's user's turn.

4. Order dispatch
   - *Did you find out how to dispatch the current order?* Kind of. After checking the place in search of what to do with my recipe I managed to understand that the machine button will dispatch the order. Maybe the button could flicker or something so to indicate that the order is ready to be dispatched, or more generally, I think something is needed to indicate that the order can be dispatched. The color of the main dish could be changed as well.
   - *How easy was to find it out on a scale of 1-10?* 6

5. *What version did you find easier and more intuitive?* First one, as it was more guided showing what ingredients to put into the main dish. Also I didn't have too much issues when placing ingredients.

6. *Did you find any issues or blocking situations that did not allow you to complete an order?* Yes, I run out of ingredients.

7. *How much do you think the scale of the objects in the virtual environment is appropriate on a scale of 1-10?* 10

8. *What did you like most about the game?*
   - It was quite easy to make recipes once you get the hang of it.
   - The ingredient's arrangement. It was nice that on the left side of the allowance all the bread-related ingredients were placed, while on the right side the rest of the ingredients can be found.
   - It was nice to be able to grab and handle two ingredients at the same time using both hands.

9. What did you like least about the game?
   - Lack of people waiting for the recipe or at least some furniture in the saloon (tables, chairs, etc).
   - Lack of music and sound.

10. *How much did you like the game mechanics on a scale of 1-10?* 8

11. *Did you feel disoriented or experienced discomfort or nausea?* No

12. *Would you change something in the game? Why?* Things mentioned in Q9, as they would do the game more realistic.

13. *Why did you stop playing?* Because I run out of ingredients.

14. *Would you play the game again?* Yes! After some improvements it will be very fun.

15. *Would you recommend the game to other users?* Yes, sure.

16. *Do you have any other general feedback or advice?*
   - Change color of the order's cartridge (or add an icon or something to it) as soon as an order is assigned so you can identify just with the cartridge what order you will have to make

- Add a free creation cartridge that won't be associated to a recipe but you will have to create it on your own as desired.

User 2

Name:	Nico
Age:	35
Gender:	Male
VR Experience [1-10]:	9

- By signing this consent form I declare to be aware that above information about me may be publicly shared for User Testing purposes.
- By signing this consent form I declare to be aware that the session might lead to slight nausea and/or any other type of motion sickness.
- By signing this consent form I declare to be aware that I can end the session at any time.

Data collected during user testing session

Prototype 1 (flexible ingredient placement)

- *Estimated time to locate the order:* 15 sec
- *Estimated time to take the order:* 10 sec
- *Estimated time to realize what recipe to make:* 2 sec
- *Estimated average time to make the recipe: 20*-45 secs
- *Estimated time to dispatch the recipe once done:* 2 sec
- *Number of started recipes:* 4
- *Number of completed recipes:* 4
- *Notes taken:*
    - The order didn't get properly slotted in the order slot in one of the user's tries to take it, but it worked fine on a second try.

Prototype 2 (fixed ingredient placement)

- *Estimated time to locate the order:* 5 sec
- *Estimated time to take the order:* 5 sec
- *Estimated time to realize what recipe to make:* 1 sec
- *Estimated average time to make the recipe:* 10-20 sec
- *Estimated time to dispatch the recipe once done:* 2 sec
- *Number of started recipes:* 7
- *Number of completed recipes:* 7

Data collected after user testing session

1. Taking an order
    a. *Did you find out how to take an order?* Yes, it was easy.
    b. *How easy was to find it out on a scale of 1-10?* 9

c. *Did you find any issues when dealing with orders?* No, after analyzing the scene it was easy to infer what the order were and how to take an order. A very little issue once was that the order didn't get slotted in the slot, but it worked just fine in the next try.

2. <u>Check recipe to make</u>
   a. *Did you find out how to know what ingredients did you need for making the order's recipe?* Yes, it was seen in the screen right in front as soon as you put an order in the machine.
   b. *How easy was to find it out on a scale of 1-10?* 9
   c. *Did you have any issues with checking recipe ingredients?* No, the order was correctly shown in the visor. The fact that the recipe appears right after slotting a new order in the machine helped.

3. <u>Making the recipe</u>
   a. *Did you find out how to actually make the recipe?* In the 1st prototype, initially I didn't know where to place the ingredients and I placed them just in the box in which the order was, but after completing the order and pressing the button the recipe didn't get correctly dispatched but the order was, and it disappeared. After examination, for the next order I moved all the ingredients to the assembly dish and that did the trick for the recipe dispatchment. I guess that if I would have started with the 2nd prototype I wouldn't have had this issue as a kind of ghostly ingredient is shown where the recipe must be made.
   b. *How intuitive was the process on the 1st prototype on a scale of 1-10?* 6
   c. *How intuitive was the process on the 2nd prototype on a scale of 1-10?* 8
   d. *Did you encounter any issues while making the recipe?* Yes, the one mentioned above in the 1st prototype, as no clues were included related to where to place the ingredients. In the 2nd prototype a transparent version of the expected ingredient was shown in the assembly dish so I get a better sense of where it should be placed.
   e. *Can you think of some way of improving it?* Maybe it would be nice to add some way of showing where to put the ingredients in the 1st prototype. Other than that, all was quite well done.

4. <u>Order dispatch</u>
   a. *Did you find out how to dispatch the current order?* No, I saw that the button on the machine has a "dispatch" label so I just decided to press that button to dispatch the order. Anyway, I was able to also dispatch empty orders, which I guess it's not expected, and sometimes pressing the dispatch button didn't work 100% fine.
   b. *How easy was to find it out on a scale of 1-10?* 8

5. *What version did you find easier and more intuitive?* I found the 2nd version more easier, but I think having the game to always tell you what ingredient must be placed next makes it too much easy and not very challenging for a game. Maybe you can include a timer so the user must make the recipe during certain time block making it a bit more challenging. Anyway, I'm not really sure if not giving the user place to errors on the recipe is a good choice. In that sense I much prefer the 1st prototype in which you can just place the ingredients you want and dispatch them. You will have to think a bit more about it when implementing the rewards and scoring.

6. *Did you find any issues or blocking situations that did not allow you to complete an order?* No, except for the already mentioned fail in slotting the order once.

7. *How much do you think the scale of the objects in the virtual environment is appropriate on a scale of 1-10?* 10
8. *What did you like most about the game?* That you can grab almost anything and place it anywhere else, throw it, etc. I also liked the different machines with which you must interact to get an on order taken and dispatched.
9. *What did you like least about the game?* That the ingredients do not spawn automatically as soon as they are used, and that there is no scoring or incentive that make it more engaging.
10. *How much did you like the game mechanics on a scale of 1-10?* 8
11. *Did you feel disoriented or experienced discomfort or nausea?* No
12. *Would you change something in the game? Why?* Some points that come to my mind are the two mentioned in Question 9, and also including more suitable models for the ingredients and more sound effects for the different interactions.
13. *Why did you stop playing?* Because I used all the ingredients and more ingredients were not spawned.
14. *Would you play the game again?* Yes, with a scoring system in place and with more diverse recipes I think it can be quite fun.
15. *Would you recommend the game to other users?* No in this version, but I would when more functionality is added.
16. *Do you have any other general feedback or advice?* It would be nice to add some way of grabbing the whole recipe after finish making the order, I mean, grabbing the burger or whatever and throw it, play with it, etc. Then put it again on its place when you want to dispatch it.

Results

Bugs

(bugs already solved are strikethrough)

1st Prototype	
Issue	**Action**
~~When making the recipe there was something that messed up the already placed ingredients when placing a new one and the recipe couldn't be completed~~	Try to reproduce and solve
~~When placing what seems to be the fourth ingredient of an order, the next ingredient's placeholder on the assembly board turns red instead of green despite the ingredient was the correct one~~	Try to reproduce and solve

2nd Prototype	
Issue	**Action**
Hot dog is a bit difficult to make because of the physics. If you just drop the hot dog onto the	Decreasing or removing physics forces to already placed ingredients? Add physics

bread's it works but if you make contact with it physics can mess things up. More generally, there are some problems with physics with almost all ingredients, which makes them a bit difficult to stack together in case you make contact with an already placed ingredient while placing a new one	constraints for already placed ingredients?

General	
Issue	**Action**
~~Dropping an ingredient just a bit below the assembly dish makes it to disappear inside the allowance~~	Make impossible that an ingredient go through the main allowance by adding some (static) physics to it?
~~Able to dispatch empty orders~~	Validate recipe's status before dispatching the order
~~Dispatch button sometimes didn't work as expected~~	Try to reproduce and solve. Maybe increasing a bit the max button position? Making the button fully physics-based?

UX and Misc Improvements

(user testing result's tasks already implemented are strikethrough)

Prioritized list of tasks to implement:

1. ~~Add some kind of indicator (arrow or something) showing that the order must be grabbed and placed on the machine.~~
2. ~~Add some kind of indicator (arrow or something) showing where to place the ingredients after an order is taken. A good idea is adding some kind of initial tutorial that shows you how to make it, bringing the different ingredients to the main dish and stacking them together automatically until the recipe is completed. Only after user's confirmation, the user can start making the recipe by himself. Also, the transparent placeholders from the 1st prototype seem to work fine to convey this fact.~~
3. ~~Add some kind of indicator showing that the order can be dispatched as well as from where to do it. Maybe the dispatch button could flicker so to indicate that the order is ready, or the color of the assembly dish can change.~~
4. ~~Add an ingredient spawning system so that as soon as you run out of certain ingredient it spawns more units (this was thought to be done as part of Week 5 work).~~
5. ~~Improve textures and more generally, ingredient's models (this was thought to be done as part of Week 5 work). I will try to stick with a low-poly graphics style but there are a few things right now that doesn't feel very low-poly despite being low in polygon count. There is something else I'm missing to achieve proper low-poly aesthetics, so I have to check that.~~
6. ~~Add people or at least some tables and chairs to the saloon so to make the environment more plausible and immersive (this was thought to be done as part of Week 5 work).~~
7. ~~Add a scoring system upon the previous point.~~

8. ~~Add music, background sounds, and more sound effects.~~
9. Change color of the order's cartridge (or add an icon or something to it) as soon as an order is assigned a recipe so you can identify what order you will have to make by just checking the cartridge.
10. Add some way of grabbing the whole recipe once it's finished; i.e.: grabbing the burger or whatever and throw it, play with it, etc. Then put it again on its place to dispatch it.
11. Use the 1st prototype mechanics (fixed ingredient placement with placeholders) for a Time Attack Mode, so you always make the correct recipe but you have to do it in a certain time block, or complete the maximum number of recipes in a given time, and thus it's more challenging. On the other hand, ~~the 2nd prototype mechanics (fully flexible ingredient placing and more "scorable") can be used for an Arcade Mode in which a score is given per recipe depending on the correctness of the placed ingredients and on how well the ingredients are placed~~. Anyway, in the latter case, based on user testing the physics should be checked so the ingredients do not disappear and other stuff like these.
12. Add a Free Creation cartridge that isn't associated to a recipe but you can create it on your own as desired and dispatch it.

Demos

In this section some video demos showing the evolution of the application are included, ranging from very preliminary demos to the last version in which some of the above user testing results improvements have been already made.

Version	Description	Link
1.0	First prototype version including some very basic ingredient handling	https://www.youtube.com/watch?v=V5V5sP47X5c
1.1	Added a very basic environment to give the prototype a little more context and enhanced a bit the handling of orders	https://www.youtube.com/watch?v=utdpCmk1VuE
1.2	Added a new way of making the recipe and handling ingredients by placing them into placeholders that show what's the correct ingredient to be placed next	https://www.youtube.com/watch?v=kjMlH1FRab0
1.3	Added the recipe management system so it's quite easy to add new —stackable— recipes. Added hot dogs and sandwiches. The ingredients should be placed in a predefined order and position so there is no place for errors. This was the 1st prototype/approach tested	https://youtu.be/QfiW2JZQF24?t=1s

	during User Testing	
2.0	Added a new approach to recipe making which is 100% flexible in regards to what can be placed into the assembly dish and how it's placed, so to have a better solution to implement a scoring system based on ingredient's correctness, order, and position upon this. This was the 2nd prototype/approach tested during User Testing	https://youtu.be/QfiW2JZQF24?t=38s
3.0	Improved a bit the overall environment and added the ingredient spawning system which is —almost— in place. A lot of UX stuff to improve yet based on User Testing results	https://www.youtube.com/watch?v=WP0IeZvj7qY
3.1	Ingredient spawning system working properly, improved some textures and materials, added some introductory indicators and tooltips at the beginning based on user testing results, as well as a few general bugs found during testing were solved	https://www.youtube.com/watch?v=8oP8QsAhxFo
3.2	Added scoring system, a few more sound effects, enhanced some graphical elements and solved some bugs. Also a new recipe was added, the Uruguayan Chivito!	https://www.youtube.com/watch?v=SeoVlZpifJ4
4.0	Enhanced and redesigned environment (mainly cooking allowances) so to make space for more ingredients, improved lighting, added toppings support, added interactable jukebox and spatial sound effects, hands rendering, basic timer functionality and a few additional animations	https://www.youtube.com/watch?v=nuWQtAUw7ZI

A video compilation of the evolution from v1.0 to v4.0 can be found here.

How to run it

As mentioned earlier, **the Unity project is thought to be run with a SteamVR-compatible headset such as HTC Vive, Oculus Rift or Microsoft Mixed Reality Headsets** (SteamVR must be installed in the system); in particular, I've tested it with the HTC Vive. The scene to be loaded should be the "*Prototype v3*" scene included in the Scenes folder of the project. Below there are a few considerations when building and running it in different ways. Take into account that the VR Hands Asset is not part of the uploaded project as it is an Unity Asset Store paid asset, so to respect author's copyrights.

Run from the Unity Editor

The project can be run directly from the Editor in case there are no errors with the scripts and the rest of the components (it was created with Unity 5.6.5). Just hit Play from the Editor and considering you already have SteamVR running the game should be up and running in a few seconds.

Build and run

1. Check there are no compiling errors in scripts and/or the rest of the components.
2. Check PC is selected as Build Platform in the Build Settings (File > Build Settings).
3. Check OpenVR is selected in the Virtual Reality Supported list in the Rendering section of Player Settings (Edit > Project Settings > Other Settings > Rendering).
4. Build and Run (File > Build and Run). This will generate the executable file and run it.

TODO

(already done tasks are strikethrough)

- ~~Get a clear idea of the whole workflow for the game in paper (client arrives, place an order, took the order, check order, select ingredients, make food, etc). This will be the most important point I guess as all that will come later will depend on it.~~
- ~~Share the idea with friends so to gather some opinions and early feedback on the abstract idea. Further improve the idea based on this feedback.~~
- ~~Search for similar apps/projects so to check for more ideas. Play Job Simulator or any other VR game strongly based on interaction and take notes~~
- ~~Make some simple prototypes of the workflow on paper. Simple storyboards or something.~~
- ~~Make some very simple isolated prototypes of the required interaction for all this to work (using just cubes and other native primitives initially). Basically grabbing objects and piling them together to make a sandwich, for example, prototype the way in which the recipe with the required ingredients will be shown, etc.~~
- ~~Join all prototypes together to get a first version of the core gameplay of the game working.~~
- ~~Make a Test Plan, including goals of the user sessions as well as questions and interviews to be made to the users.~~
- ~~Implement the other alternative to handling orders and making the recipe (more physics based and more flexible, checking ingredients against the correct recipe after dispatching the order).~~
- ~~Add a few more recipes just to check implemented system's extensibility and take the chance to have a better user testing with more (yet simple) recipes.~~
- ~~Test the prototype with real users (Week 4).~~
- ~~Prioritize the results of user testing taking into account both bugs and improvements (Weeks 5)~~
- ~~Implement some of the high priority tasks from user testing results (Weeks 5)~~
- ~~Complete the gameplay workflow upon this. Implement the scoring mechanism (Weeks 5 and 6).~~
- ~~Find a way of testing the scoring system, so to be able to try quickly (outside VR) multiple scoring scenarios. This was finally done with Unity Test Tools.~~
- ~~Enhance the scoring system. Avoid making below zero scores, and show penalties in red instead of the usual yellow color~~
- ~~Solve bug related to ingredient placement in the assembly. There are some ingredients that ends up repeated for some reason and this mess up the scoring~~
- ~~Add Uruguayan Chivito recipe and add Uruguayan flag so you can place it as a topping. (Weeks 6).~~
- Replace the placeholder primitives with real models. The models will be done inside VR using Google Blocks, so one task will be modeling all the needed assets/ingredients, at least for a few simple recipes (e.g.: hamburger, sandwich and hot dog). Or eventually downloading them from Google Poly so to maintain the low poly graphical style. Put the assets into the game and check if all works fine with those (Weeks 5 and 6).

- ~~Redesign the environment so more ingredients can be accessible, adding kind of shelves below each allowance.~~
- ~~Add recipe name in the display for each recipe.~~
- ~~Solve a bug that makes that when you make some juggling with the ingredients, if there was an ingredient already placed, once you catch the ingredient with which you are juggling, the game incorrectly takes it as placed in the assembly.~~
- ~~Add labels to each spawner corresponding to which ingredient it spawns. Add an animation so when a new ingredient is spawned the label rotates around the spawner's tip hole.~~
- ~~Add a Timer board so as soon as you receive the new order the timer starts ticking.~~
- ~~Add an animation to the order spawner so it's compressed at first, and when a new order is spawned it goes down quickly together with the order, as if it was throwing the order. In addition, this solves the issue related to the occluded timer board.~~
- ~~Add more toppings like the flag for the Chivito recipe so they can be placed at the top of any recipe.~~
- ~~Further enhance graphics and interaction (Week 6)~~
    - ~~Replace controllers with some cartoony hands. I ended up using the paid VR Cartoon Hands asset from the Unity Asset Store.~~
    - Add some nice effects. Maybe some cool shaders or something, but have to think about what effects could work yet.
    - ~~Add missing sound effects, music, background sound, etc.~~
    - ~~Configure spatial audio properly for the different audible elements~~
    - ~~Add a radio or something for attaching the background music and be able to turn it on/off~~
- ~~Eventually, implement the remaining tasks from user testing results as long as there is some time available~~
- Implement rounds based on enhanced timer functionality so as soon as you receive the new order it starts ticking and you have an appropriate time for each recipe, show final scoreboard when time runs out, etc.
- Analyze a strange bug that makes that only in the HMD rendering some ingredients are shown duplicated and with an incorrect perspective (mainly for the cheese and ham as part of the sandwich). In the desktop screen they are shown just fine...

Extra Features

- Add more complex manipulation of ingredients, not just placing them together to make the recipe. For example, cutting, boiling, etc.
- Add chores or other activities beyond cooking (washing dishes, defend against robberies, etc).
- Add a cool menu and some diegetic interfaces. For example, the menu could be attached to the users hand, watch or something.
- Add basic NPC clients walking around the saloon, maybe coming to place an order from time to time, waiting for it to be ready, etc.
- Add a slowed time mode. I would like to experiment with this but don't know where in the game mechanics put it yet.

- Add multiple cooking areas each one with different ingredients and types of food. Implement teleportation to move between each cooking station (easy and motion-sickness free navigation approach).
- Port the game to mobile VR headsets or even standalone if all comes up well. Experimenting with Oculus Go support would be great, but the interaction should be redesigned.
- Add some reflective surfaces such as the oven's front face or a mirror somewhere, so the player can see himself reflected on it (or at least his head and hands moving in sync with his real movements) so to enhance the embodiment illusion.
- Add multiplayer support so players can compete somehow. For this I will have to take into account some social interaction techniques (how to represent the avatars in a socially expressive manner, how to interact with them, etc) as well as adding all the networking logic.