M11 DP Play and Learn Fantasy Book

Students:
Jurrien Brondijk - 1004051
Yixuan Bian - 1479946

Coach:
Erik van der Spek

Contents

Introduction

Educational video games prevail among school education as child education assistance to promote pupils' motivation, because of the high immersion of video games. With new scientific and technological developments such as augmented reality (AR). AR games became the new trend in game development, because it not only has better immersion than traditional video games, but also has a high degree of interaction in reality. Therefore, we wanted to study a new application of the AR technology in kids' English learning.

In this project, we have designed an AR game named Fantasy Book to explore a new, interactive way to help kids between the ages of 10 and 12 to learn English. This game includes a book with QR code and several vocabulary cards. When scanning the QR code or vocabulary cards, the AR game elements will appear on screen. Users will be presented with an incomplete sentence. They can use the vocabulary cards to fill in the blank spots and complete it. This allows them to make the sentence their own. The various interactions between user, book, card and virtual game scenes improve the immersion and quality of kids their English learning.
The high freedom in game encourages users to try different card combinations to repeat practicing with the same sentences, which can effectively lead kids to understand and use the new words. It is an innovative new way to allow kids multiple interactions between virtual game scenes and realistic environments.

Figure 1: complete setup of the current demonstrator

Design Process

Choosing a direction

Before we made the decision of the AR game's subject, we interviewed 2 PhD students in the Play and Learn Squad of TU/e. The goal of these interviews was to understand the previous studies in this field. In Jingya Li's project, she and her team designed a collaborative AR game for an elementary school math classroom, called MathBuilder (van der Stappen et al., 2019). The game includes a set of physical maps, building cards and an AR mobile application. Users can play different roles in the game. Then they need to collect different unique building materials corresponding to their roles by finishing their math exercises. In their user test, they were able to motivate students in math learning activities and encourage them to collaborate with each other.
We also interviewed another PhD student, Tengjia Zuo. Her project focused on how the involvement of fantasy affects the effectiveness of serious games (Zuo et al., 2019). She built an AR card game where users need to use different card combinations to get the right answer of a math problem. In the experiment, her team used different fantasy levels of roles to verify that kids were more likely to choose more fantasy roles during playing.

It is easy to see that both these projects use math questions to propel the game forward. At first, teachers need to give students lectures to impart the related knowledge, such as theories and solving methods. Then students can use the two games as a coursework to practice their knowledge from the class. In another word, the two games have certain mathematical requirements of users.

We considered choosing math as our subject as well, but left the option for other courses open as well. Eventually we decided on English as this as the topic of language learning in AR felt less explored. English was chosen over Dutch as not all the designers working on the project spoke Dutch.

Background research

Playing is a way for kids to learn and explore the word, all while they gather new skills (Acuff et al., 1997). In other words, play is a learning process for kids. Therefore, if kids can train their capabilities while playing, the efficiency of learning will be improved. McArdle said that kids are relaxed, intrinsically motivated, and actively engaged during their playing, which are all behaviors conducive to learning (McArdle, 2001).

According to psycholinguistic and sociolinguistic frameworks, there are lots of previous study works that prove that interaction and motivation have great significance in the foreign language learning process (Richard Donato, 1994, p. 453-464) (Michael H. Long, 1981, p. 359-382) (Michael H. Long, 1983, p. 359-382). Therefore, the high interactive and motivational game has a positive effect on foreign language learning. The characteristics of digital games provide strong immersion feelings for players, so it can create a relaxed and engaging platform for players, where it is a comfortable environment for language learning (James. P. Gee., 2007).

Open-ended play has large numbers of possibilities which can increase the degree of freedom in game. On the one hand, it helps kids to foster creativities in open-ended constructive play environments (Jakobsen et al., 2016). On the other hand, it encourages users to use their imagination in various ways in game (De Valk et al., 2015). Open-ended play is a good attempt in the educational game design to keep their interest, meanwhile it can develop their creativity.

In foreign language learning, improving memory retention is an important aspect. The first stage to this retention is encoding, which means the process of getting information into memory for storing (McLeod, S. A., 2007). There are three types of encoding, including acoustic encoding, visual encoding, and semantic encoding. Acoustic encoding is the process of storing memory via audio, which is often short-term. This method often is applied in the situation where words are repeated to remember them. Similarly, in visual encoding visual elements are remembered. Because half of the human brain is directly or indirectly devoted to processing visual information, visual imagery has strong power in learning. For example, in one experiment, students were asked to remember many groups, three words each. The result showed that students who tried to make visual associations with the three words had significantly better recall than students who only tried to remember the words via repeating them (McLeod, S. A., 2007). The last encoding method is semantic encoding, which is the most long-term storing of the three forms. As opposed to acoustic encoding or visual encoding, semantic encoding processes something (a word, phrase, picture, event, etc.) via the meaning. Compared with the first two forms, semantic encoding looks like a complex and multiple dimensions way to understand the target object. Hence, if we try to combine the three encoding forms in our design concepts, it will help kids improve their memory as efficiently as possible.

We decided to take a look at existing games on the market. From the multitudinous educational games in the current market, we organized them into 3 different types, practicing games, imparting games and practice-imparting games.

The current practicing games' structures are similar with the two PhD students' design mentioned above. Users need to finish the math exercise to drive the game forward, such as collecting materials to build a house/city or feed digital pets, and getting through the game level. Part of English games have the same game structure but change the math exercise to spelling words or speaking sentences. The imparting game includes part of popular English learning games, such as RunFox (RunFox English, n.d.), which use storytelling to teach kids English. It uses funny animation and interaction to replace the traditional classroom teaching. The last type, practice-imparting game, combines the previous two types, which not only teaches kids curriculum knowledge but also deepens kid's understanding via exercises. There are many high-quality programming games falling into this category, such as Swift Playgrounds (Swift Playgrounds, n.d.), which provides a friendly platform for users with no coding knowledge to create scenarios and games. We got inspiration from this programming game, and planned to apply the game structure in English sentence teaching. In Swift Playgrounds, users can control the game elements and roles through adjusting the established code. Controlling roles' behavior through different words in the established sentence became a potential direction of our project.

To see how our game would fit into the current education system, we took a look at the learning goals schools have. In regards to the English learning goals, the SLO (specific teacher education) specifies the requirements for listening, speaking, reading and writing at different grades in primary school in TuLe (intermediate goals & learning lines) for English education in the Netherlands (Appendix B - Translated Learning goals). A translated version of these goals can be found in appendix B.
With these learning goals, we decided on our target user; Dutch students in grade 5 and 6, ages between 8 and 10. It is in these grades that the children are first introduced to reading English. Both the curriculum and textbook content make use of elements surrounding everyday life to teach kids English (such as living somewhere, food and drink, items in a store, etc.). In addition, our target users begin to be taught simple sentences from the top of their head (for example, do you like (ice cream)? Is it (red) or (blue)? It's a (giraffe).) Through playful manners kids are stimulated to show what they know and can say. Thus, we decided to apply AR technology in English sentence's structure teaching in our design. Besides, the elements in the game should be close to daily life.

Formulating our idea

With all this research in mind, we came up with our first design concept, as shown in figure 1. It includes a mobile phone application and a set of cards. When using the phone to scan the cards, 3D game elements will appear over them on the phone. First, users need to use the cards to build their own stage. There are two different types of card, scene cards and decoration cards. When scanning different scene cards, different game scenarios will appear, such as a kitchen, forest, etc. The decoration cards are used to decorate the scenarios, like trees, flowers and furniture. Then users can create the roles in their story and control the role's action via buttons in the phone. After that, users can record their voice as the lines of the role in the game, which is a way to practice their English speaking.

The last step is recording the animation in game and sharing with teachers and friends. In short, users can use the game to act out their own story. This game provides a great degree of freedom for users to create their own story. However, there is a clear weakness to the game as an English teaching tool. The game is only an open innovation platform for kids to play the role-playing game, but cannot impart how to build sentences or whether their voice message is articulate and right or not.

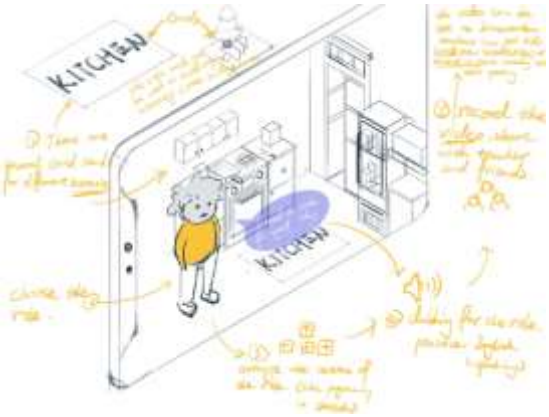


Figure 2: the first design concept

In our second design we focused on teaching English sentence building. As shown in figure 2, the design includes a physical toolkit (a folded pamphlet and a set of vocabulary cards) as well as a mobile phone application. In figure 2, each page of the pamphlet has a big picture of the game scene and fill-in-the-blank sentences. When scanning the pictures, the game scenes will appear in the phone. Users need to use the vocabulary card to complete the fill-in-the-blank sentence to control the roles' action in the corresponding scene. This is a puzzle game, where users can get materials and clues in different game scenes and they need to use them in the right way to promote the plot of the game. For example, users can cook the vegetables from the forest scene in the kitchen scene to get special food, and they can combine the wood and stone from the forest scene in the tool table scene to make the axe. The special food and axe can help users to explore the following story of the game. In this design, we reduced the freedom of the game, which means the storylines causing users being unable to create their own unique story. The storyline we designed was meant to enhance the playability of the game. However, the complex puzzle game may be difficult for our target users (primary school students between 8 and 10 years old) to grasp while they are also required to focus on learning the language.

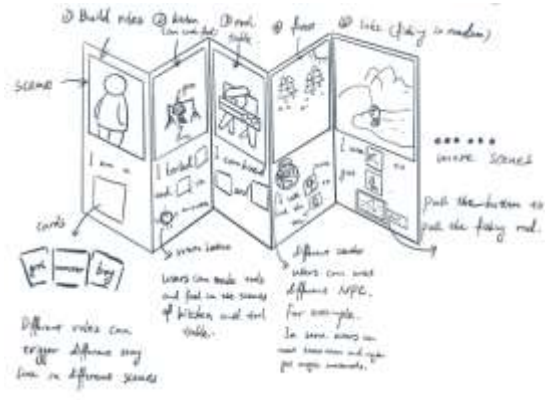


Figure 3: the second design concept

The final concept

Considering the feasibility and level of acceptance by kids, we decide to simplify the second design while taking elements from our first.
We replaced the puzzle aspect with our original story making idea. The open-ended animation combinations of the first idea are replaced with limited storylines, in the shape of incomplete sentences that users have to fill in, similar to the game Mad Libs.
In the game, users fill in the incomplete sentence using the vocabulary cards, after which they are able to see an animation of the sentence they created. We expect that kids will be encouraged to try many different words to see the resulting animations. Through these repeated attempts of different vocabulary card combinations, they can begin to understand the meaning of the sentence. We made a video showcasing the concept, which can be accessed through figure 4 and seen in figure 5.

Figure 4: QR code to first concept video

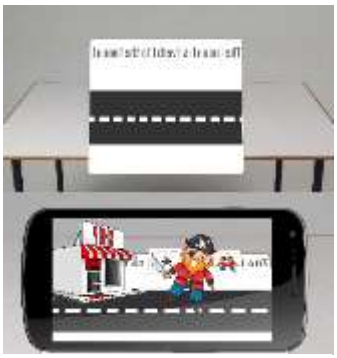


Figure 5: Screenshots of first concept video

We thought this idea had a lot of potential, it seemed to align well with the learning goals of our target group, especially in the aspect of teaching them simple sentences. Therefore, we decided to develop it further and another video was made (figure 6 and 7).
Like the second video shows, the physical toolkit of the second design is still in use, but the folded pamphlet is replaced with a book. The book will consist of several chapters, each containing pages with sentences and pages on which the animations will play. The chapters will be led by a catalogue page, on which each sentence of the chapter is made visible. This catalogue page will display the user's progress and allow them to play an animation of the whole story they created in the chapter so far. The sentence pages have slots for the vocabulary cards. The animation pages have buttons that can be activated by physically pressing on them in the book. These will be registered by the game and play the animation. There is also a button to save the sentence, for when a kid is really happy with whatever sentence they created.
The media device the game is played on got changed from a mobile phone to a tablet, put in a fixed position. This increases the comfortability and convenience of the usage experience. The tablet, placed at the top of the book, acts like a mirror, showing a mirror image of the playing field, revealing the digital elements in this mirror image. This shortens the time spent getting used to the camera.

Figure 6: QR code to second concept video

Figure 7: screenshot of second concept video.

This idea was showcased during our squad's midterm presentation through which we received a lot of feedback on it. Many people thought the idea was good and would certainly help learn the language. There were some questions on the feasibility of some aspects of the design however. In particular, the idea of using the tablet as a mirror was questioned due to the limited viewing angle of the tablet's camera.
It was also suggested to us to take a look at PlayStation Wonderbook (Wonderbook™, n.d.), a game that also uses AR to project game elements onto a book.

After the midterm presentation we started to think about how we were going to realize the idea. At this moment it was still very conceptual and would need further clarification. For example, the idea to use a tablet as a mirror would require an additional camera mounted on top of the tablet.
Taking the midterm feedback to heart, we started to look into the possibilities for realizing our idea.

We found out that using an external USB camera on a tablet with Vuforia is possible (Building and Using the UVC Camera Driver Sample, n.d.). The use of this would help realize the idea as we had imagined it by placing the camera on top of the screen, looking down at the book.
We also decided to take inspiration from PlayStation Wonderbook (Wonderbook™, n.d.), making our book consist of QR codes and filling these in on screen, as shown in figure 3. This is desirable from a business point of view, as it allows the book to be used with a variety of content through software updates, without the need for a new book. Making the product potentially more attractive to schools.

Figure 8: the final design concept

Regarding the structuring of sentences, we decided to only let users enter nouns for the time being. The game is meant to introduce the kids to the English language, and teach them the meaning of words. This is difficult enough without having to explain to them the different classes of words, a concept they are only beginning to understand in their native language, let alone a foreign one.
The use of verbs and adjectives can always be implemented later to increase the difficulty for more proficient and older pupils.

Though we left out the different types of word classes, we did decide that we would split the nouns into different groups of their own. For the purpose of the demo, these groups would be characters, locations and objects.
These distinctions were made based on what functionalities these words could have within a sentence.
A character would be able to animate and interact with any objects, animating when doing so.
Objects would be simple static models to be interacted with, like being picked up and thrown.
Locations would be similar to objects, only with the added feature that they would also be able to set the scene of a chapter. A sentence like "the dog is in the store" would have the next sentences take place inside of a store for example.

The demonstrator

For the demo we decided to make 5 sentences, forming a short story that would showcase each of these functionalities. These sentences became:

> **The (1) walks to the (2)**, which showcases how location cards can set the scene of the chapter by letting the word chosen in (2) set the chapter's scene if a location card is used.
> **The (1) picks up the (2)**, which showcases characters interacting with chosen objects by picking them up.
> **He/She/It puts the (1) on the (2)**, which showcases a character from the previous sentence being used again, by letting the character from the previous sentence perform this action and setting the correct pronoun to suit that character.
> **The (1) kicks the (2)**, which showcases different animations based on words chosen. Kicking an object will send it flying, while kicking a character will cause that character to get upset.
> **The (1) dances with a (2) in their left hand and a (3) in their right hand**, which showcases being able to use all 3 word slots.

While the idea was being finalized, we could start focusing on doing some tests for the prototype. We started testing with some drawn mockup cards. This way we could get the main elements of the program working, being able to scan the cards to enter words in a sentence.
The code of the program was written in such a way that it allowed us to very easily change cards or add new ones. Each card object in the game was given a number of variables that allowed the game to work with them; the word itself, the type of word, the model it had to spawn and the pronoun it should have when used in a sentence like sentence 3. To add a new card, we simply had to fill in these variables and it would work.
To create the scanning of the cards, both the card and page were given a trigger. When placed on top of each other, the trigger activates and the page can retrieve the variables stored in the card that was placed on top of it.
Another important test that had to be done was regarding the character animations, we needed to be able to assign an animation belonging to a sentence to any character. Luckily unity has a feature for this, it can standardize character models to share the same bone structure, allowing you to apply any animation to it. This also allowed us to use the animation library Mixamo (Mixamo, n.d.) to gather animations, since any of these animations could now be applied.

Since the scanning of cards now worked, we started to work on designing and printing the final versions of the cards and pages. For the purpose of the demo we wanted to give enough variety in the cards while keeping the scope small, so we created 5 characters, 5 locations and 9 objects. The objects are much easier to create since they only require 1 static model as opposed to the characters, who need to be rigged with a skeleton, or locations, which need to be able to serve as a backdrop as well.
With the final cards created, models could be added to them. Some of these, like all the characters, we created ourselves, while other models were taken from royalty free sources (Quaternius - Game Assets, n.d.) (3D Models for Professionals :: TurboSquid, n.d.).

Figure 10: character, location and object card examples as well as some of the page designs

Figure 9: video of a short playthrough of the completed demo

Finally, the sentences themselves had to be put in the game. The script that scans the cards sends their information out to various other scripts, which update the sentence and spawn in the models. Now each sentence had to be programmed to have characters play the proper animations, attach and detach objects to hands and any other required behavior.

Figure 11: screenshot of the completed demo

User validation

Near the end of our project we found some opportunities to visit two teachers and ask them questions about teaching English, as well as show them our prototype.
In the time before this, we worked off of mostly assumptions and literature, so it was time to validate these. We wanted to know how English is currently being taught, what pain points are encountered when teaching and to what extent schools are making use of digital learning tools at the moment.
On top of these validations, we wanted to discuss our current concept with them, to see whether or not they think it will provide added value and if they think whether or not it needs any altering.

Regarding teaching methods the teachers we spoke and their colleagues mostly make use of songs. One the schools does make use of a method book called “take it easy” (Lesmethode Engels, n.d.). This method makes use of digital exercises, pictures, videos and small animations as well, though the teacher mentioned they often rely on additional material to be able to keep the kids busy. For this they use a tool called Groove (Blink, n.d.). Neither of the teachers think enough attention is given to the subject, given how prominent the English language is in our society.

The biggest pain point encountered by the teachers is the variation in the skill level of the pupils.
Some kids learn a lot from watching movies, playing games or being on the internet, while others hardly know anything. The teachers think digital solutions can be very valuable here, since it can determine the skill level of the kids and assign them exercises suited for their skill level.

Many schools have already started adapting these digital learning methods through programs like Snappet (Snappet NL, n.d.) and Blink (Blink, n.d.). One of the teachers we interviewed had specific times in the day pupils got to work on a computer, their kids have tablets at their schools with which they do the same. Another teacher's school uses Chromebooks, which have to be shared with other classes and pupils, so they are less available.
Overall, the possibilities of implementing our concept are present.

The teachers themselves were very excited about the concept in general. They think especially younger kids will be impressed by the AR technology.
They see it work well as an addition to the current teaching methods. This is very welcome because even the school with the method book runs out of material to cover and relies on additional material to continue teaching. One teacher pointed out she thinks digital learning methods should always be an addition to traditional learning methods rather than a replacement of it, since they don't work well for all kids.
What was appreciated about our concept however was how it allows kids to use many different senses when playing with it. This teacher then brought up multiple intelligences theory (Gardner, 1983), saying that our concept would allow kids of varying intelligences to be engaged by it and therefore improve their learning. The game features words and sentences for lingual thinking, pictures and animations for visual thinking and sound or songs can also be implemented for kids who learn through those.
While we couldn't find any literature backing up that this application of multiple intelligence theory would improve their ability to learn, an argument can be made that covering these different senses does make the game more engaging and motivating for a broader group of kids (Akkuzu & Akçay, 2011).

Another big factor in motivation the teachers recognized in the game is the amount of creativity the game lets children express. The teachers recommended us to let kids form even strange sentences, like "the shelf walks to the store". Seeing those weird sentences played out would be very fun and motivating, encouraging them to try out more words and repeat sentences.

The biggest critique we got was regarding the setup the game currently requires. Though tablets and Chromebooks are available, having to purchase webcams for the game to be playable increases the cost of the product by a lot. Potentially causing it to be too expensive for schools to consider.
On top of that, it will likely take a lot of time to set up, and if cameras break or for any other reason don't work, the school's IT personal will have to fix them. These personnel are often teachers themselves and don't have a lot of time to do this. Therefore, schools rather don't have this many loose parts.
All answers we got from the teachers can be found in appendix C.

After the interviews, one of the teachers let us come back to test our finished demonstrator with their children, ages 5, 8 and 10. In this (informal) experiment we were trying to determine whether the kids understood the game and how much they enjoyed playing it.
The demo only had one sentence page, the sentences were swapped out by the researcher.

Each kid got an explanation of how they could use the cards to fill in the sentences. They then got to play through each of the sentences, getting assistance where needed. Afterwards they were asked some questions about their experience.

The first thing we noticed was that the kids needed a lot of assistance to play the game. In our demo, the game often lost track of the AR cards, causing it to not be very responsive. This made it even more difficult for the kids to interact with the game, so we had to step in and help them complete their actions.

This problem occurred much less with the 10-year-old kid, who actually recognized when the game lost track of a card and would try fixing it when this happened.
Overall, the 10-year-old had the least trouble with the game in general and displayed most of the behavior we expected to see. They laughed at weird sentences like “the balloon picks up the dog” and would excitedly try out different words to see what would happen.
Afterwards they commented on especially liking the characters and letting them do things. They did mention already knowing most of the words beforehand from the things they learnt in school.

The other kids had a lot more trouble. The 5-year-old had only recently learned to read letters, so when presented with the sentences they would try to pick it apart letter by letter, but they would not understand words. The 8-year-old repeated a year and was still in third grade so they too had quite some trouble understanding the written words. This is quite expected when looking at the SLO learning goals (Appendix B - Translated Learning goals), which point out that since kids in these grades are learning to read and write Dutch, they shouldn’t also be learning to do this in English as it can only confuse them.

Something that was present in all the users’ tests was that they all needed to be externally encouraged to try and translate words or sentences. As it is right now, the game does not ask this of them and they did not do it out of themselves.
A summary of the observations made during the user tests can be found in appendix D.

Figure 12: the game being played during our user test

Conclusion

Fantasy Book has the potential to be a welcome tool in the current education system.
Through playful means it introduces children to short written sentences in the English language. This aligns perfectly with core goal 14 of the SLO learning goals (Appendix B - Translated Learning goals) in the 5th and 6th grade of Dutch primary schools, which involves teaching standard sentences through a playful manner.
Our game achieves this by involving a degree of open-endedness that allows for creativity in the game, which enhances the learning (De Valk et al., 2015). There's also a focus on different types of memory encoding, through the use of different ways of presenting the meanings of sentences (words, images and animations), which should help the children remember it better (McLeod, S. A., 2007).
These different presentations can also help motivate and engage a larger number of children(Akkuzu & Akçay, 2011).

Judging by the interviews and tests so far, our game should function nicely as an additional piece of learning material for schools to use besides their method books, similar to ones already being used. Devices needed to play the game area already largely in use.

Discussion

Much like many others, our project faced difficulties this period due to the effects of the Covid-19 pandemic. The pandemic affected our ability to do user testing and in some cases hindered the development of our demonstrator. For example, we weren't able to use university facilities to make our book and cards and therefore had to rely on paper cutouts stuck on cardboard, which was the best we could do at home. Also the development of the game on the digital side was hindered, because we lacked a proper USB webcam and weren't able to buy one due to shortages caused by the pandemic.

We were only able to do limited user testing and only spoke with teachers near the end of our project. Right at the moment our project started to take shape and we had defined our target group, as well as the subject of English teaching, the lockdown began. Incidentally, this was the moment in our project where we wanted to start involving teachers and kids alike to shape our design. One plan we had was to have kids play the game without the digital augmented part, to see what we could learn from that. Even when the measurements were being reduced and schools reopened, the schools still only allowed teachers and pupils to come in. Not even parents were allowed, let alone us.
As a result of this, our concept has mainly been based on literary research and gut feeling.

When testing could be done later on, it was very limited. However, these few tests and interviews did make it clear that certain aspects of our design would have to be changed if development were to be continued further.

It should be noted however that, since only one of the participants fit our target group, the findings are somewhat insignificant. They still indicate potential flaws in the design but much broader user testing would have to be done to make proper claims about the effectiveness of the concept.
Additionally, it would be desirable to develop the demonstrator a bit further and have it be a complete booklet of a chapter consisting of a catalogue page and several sentence pages. This would give the users more control over the game and a much more complete experience.

Future Work

Additional and improved user testing will likely uncover new improvements to be made about the design. From our own experiences, the teacher interview and current user tests we have determined some areas of improvement for our concept if it were to be developed further.

In its current state the game presents kids with an opportunity to playfully experiment with words and sentences in English. While this interaction alone helps them learn, it might be valuable to design the game to encourage the translating of the words and sentences more.
Possible ways for this are to ask kids to enter a translation on the tablet after filling in a sentence or completing a chapter.
Another aspect of the game that is currently missing is audio. By having the game speak selected words out loud, and when playing the animation letting the kids hear the sentence being spoken, they'll be able to not just see but also hear the sentence.

An important point brought up by one of the teachers was the use of a webcam. Having to install a webcam whenever the game is going to be played can be problematic for many schools as it can take time and there is a lot of potential for things going wrong. It is much preferred if the game functions with their current material without any additions.
This would mean sacrificing the idea that the tablet acts as a sort of mirror, instead it will be a sort of lens through which you can view the book's contents. It will make the game less comfortable to play alone, as you will have to put down the tablet to place cards.
It does create interesting opportunities for cooperation, where one child holds the tablet while the other places cards and they can communicate about what is happening.

This does bring up another problem which is the availability of the material at the schools. The game was tested on an Android phone and while many schools use tablets and thus should have no problem running the game, many schools make use of Chromebooks instead. Chromebooks' ChromeOS is not officially supported by Unity (Technologies, n.d.), while Chromebooks should be able to install any Android app, we have not been able to test this with our game.
Another problem with Chromebooks is the lack of a back facing camera, which makes playing the game a lot more difficult. Either a Chromebook version of the game would have to be made or schools with Chromebooks will have to be convinced to purchase tablets or webcams to be able to run our game.

With the added cost of the webcam potentially out of the picture, the game will consist of the app, book and vocabulary cards. This should make the game no more expensive than current additional tools like Groove (Blink, n.d.), which is priced at € 6,50 - € 7,85 per student for the license and € 5,25 per book (which there are several of). Our game can be expanded through software updates without the need of purchasing a new book because of the use of the QR code pages,
The only additional thing that could be required to purchase are additional cards whenever new ones are made.
If it turns out that this is still too expensive for schools, we could make the decision to distribute the books and cards as printouts that school can print and let the children cut out and play with.
This would put the cost of our game at a very comfortable spot for schools.

References

Acuff, D. S., Reiher, R. H., & Free Press. (1997). What Kids Buy and why: The Psychology of Marketing to Kids. Free Press. https://books.google.nl/books?id=eo6dlAEACAAJ

Akkuzu, N., & Akçay, H. (2011). The design of a learning environment based on the theory of multiple intelligence and the study its effectiveness on the achievements, attitudes and retention of students. Procedia Computer Science, 3, 1003–1008. https://doi.org/10.1016/j.procs.2010.12.165

Blink. (n.d.). Lesmethode Engels voor het basisonderwijs. Groove.me. Retrieved 10 June 2020, from https://groove.me/

Building and Using the UVC Camera Driver Sample. (n.d.). Retrieved 10 June 2020, from https://library.vuforia.com/content/vuforia-library/en/articles/Solution/uvc-camera-driver.html

De Valk, L., Bekker, T., & Eggen, B. (2015). Designing for Social Interaction in Open-Ended Play Environments. 9(1), 15.

Gardner, H. (1983). Frames of Mind: The Theory of Multiple Intelligences. https://doi.org/10.2307/3332707

Jakobsen, K. B., Stougaard, J., Petersen, M. G., Winge, J., Grønbæk, J. E., & Rasmussen, M. K. (2016, June). Expressivity in open-ended constructive play: Building and playing musical lego instruments. In Proceedings of the The 15th International conference on interaction design and children (pp. 46-57).

James. P. Gee. 2007. Good video games good learning: Collected Essays on Video Games, Learning and Literacy (2nd ed.) New York, NY: Peter Lang.

Lesmethode Engels | Basisonderwijs | Take it easy | ThiemeMeulenhoff. (n.d.). Retrieved 10 June 2020, from https://www.thiememeulenhoff.nl/primair-onderwijs/engels/take-it-easy

McArdle, P. (2001), Children's play. Child: Care, Health and Development, 27: 509-514. doi:10.1046/j.1365-2214.2001.00230.x

McLeod, S. A. (2007). Stages of memory-encoding storage and retrieval. Retrieved, 21, 2015.

Michael H. Long. 1981. Input, interaction and second language acquisition theory. In Input in second language acquisition, S. M. Gass & C.G. Madden (eds.), Rowley, MA: Newbury House, 377-393 https://doi.org/10.1111/j.1749-6632.1981.tb42014.x

Michael H. Long. 1983. Does second language instruction make a difference? A review of the research. In TESOL Quarterly 17, 359–382. doi:0.2307/3586253

Mixamo. (n.d.). Retrieved 10 June 2020, from https://www.mixamo.com/#/

Richard Donato, Dawn McCormick. 1994. A sociocultural perspective on language learning strategies: The role of mediation. In Modern Language Journal, 78 (Winter, 1994), 453-464. doi:10.2307/328584

Quaternius—Game Assets. (n.d.). Retrieved 10 June 2020, from http://quaternius.com/

RunFox English. (n.d.). Retrieved 10 June 2020, from http://www.runfox.com/

Snappet NL. (n.d.). Retrieved 10 June 2020, from https://nl.snappet.org/

Swift Playgrounds. (n.d.). Apple. Retrieved 10 June 2020, from https://www.apple.com/swift/playgrounds/

Technologies, U. (n.d.). Unity - Manual: Supported platforms. Retrieved 10 June 2020, from https://docs.unity3d.com/Manual/UnityCloudBuildSupportedPlatforms.html

van der Stappen, A., Liu, Y., Xu, J., Yu, X., Li, J., & Spek, E. (2019). MathBuilder: A Collaborative AR Math Game for Elementary School Students. 731–738. https://doi.org/10.1145/3341215.3356295

Wonderbook™: Book of Spells. (n.d.). PlayStation. Retrieved 10 June 2020, from https://www.playstation.com/en-us/games/wonderbook-book-of-spells-ps3/

Zuo, T., Feijs, L., van der Spek, E. D., & Hu, J. (2019). A Classification of Fantasy in Serious Games. Extended Abstracts of the Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts, 821–828. https://doi.org/10.1145/3341215.3356294

3D Models for Professionals: TurboSquid. (n.d.). Retrieved 10 June 2020, from https://www.turbosquid.com/

Appendices

Appendix A - Reflections

Yixuan Bian:

From my personal development plan, I choose this design project to continue my goal that researching the effect of playfulness elements on human. I hope I can design entertainment products to help people learn and explore the world, meanwhile they can feel the satisfaction and happiness of tangible social interaction. In my final year of my undergraduate study, I spent a whole year to do a set of interactive modular toy for kids with cerebral palsy to help them do rehabilitation training every day, as a therapeutic assistance. That project stimulated my interest in how play affects kid's learning behavior, which was also the reason why I chose the project in Play and Learn Squad as my first design project in Tu/e. I planned to learn the methods and implementation technique of game design from this project, AR game for kid's English learning. I really gained a lot from these months of study and practice, but I also realized there were many mistakes and regrets because of my limited design experience on game.

From the technology, I have learned much on understand how to make a game (basic operation of Unity) and how to realize AR in a game (basic operation of Vuforia). This was my first time to access to Unity, I had to spend lots of time on self-learning. Fortunately, the PhD student, Tengjia Zuo send me several helpful teaching video, which saved me a lot of detours. Although I paid lots of energy and time on Unity learning, it is still unprecedented difficulty for me to compete my imagination smoothly in Unity. So, my team member was responsible for the most of work in making prototype, and I was responsible for the appearance design, including physical products design and models building. From the cooperation, I learned how to control the team run in clear direction efficiently. This was the first time for me to cooperate with foreign student. Because of the cultural difference, the opinion and expression were different. However, my team member helped me a lot and he was a good leader who knew how to follow the plan to avoid the getting bogged down in inefficient communication.

I enjoyed this cooperation with my team member, because the area of expertise for him and me is totally different. In fact, our division of labor is very clear, so sometimes we look like each did his own work individually and lack of communication. It made our work efficient, but lack of the collision of different views in the second half of our work. But it was for that very reason that I stayed in my comfort zone and lacked a breakthrough in technology part.

The corona virus indeed brought great inconvenience and troubles to our project, especially in user study part. Because I was good at user research, I thought I could be self-worth during user research phase and user testing phase. But in fact, I even cannot close others in this special period, so I had to leave the user relative work to my team member. And because of the few participants, the lacking feedback from our target users cannot state whether our design is needed. In fact, lacking of positive feedback from users during design and difficulty in reaching users discouraged us when working. If there

was no corona virus, our final design will be completer and more mature. The result of our test will be more credible as well.

I really like this project, so I hope I can do more job in summer vocation to improve it and complete it. For example, maybe some interesting storylines will enhance the playability of the game a lot. I believe, after polished, the design will be remarkable.

Jurrien Brondijk:
Over the course of my bachelor at Industrial Design I gravitated towards serious game development, mostly in relation to virtual reality. Starting my masters, I wanted to continue in this direction, which led me to this project. While starting the project, I formulated some goals of areas I wanted to improve on.

I was excited to learn about incorporating augmented reality, a technology I had not worked with before. I also aimed to improve the overall structure and tidiness of my code compared to previous projects.
Not only did I learn a lot about working with AR, I am also very proud of the code I wrote (appendix E). The implementation of AR went smoothly, though there is some room for improvement in making the AR images on the cards more recognizable by the program.
Code wise, I tried my best to keep my code in separate scripts and make it object oriented.
This resulted in code that was easily expandable when, for example, adding new cards to the game.
The past 2 years I have been using a mixture of Unreal Engine and Unity to develop my games. I would often prefer Unreal Engine but in some cases would have to use Unity due to external programs only being compatible with it, as was the case in this project and Vuforia, the tool used for AR.
Over the course of this project I have improved significantly in my knowledge of Unity and I feel as if my abilities with either program are now near equal. I also feel like I have improved at learning new programming languages in general. In my elective, complex adaptive systems, we are using a program called Netlogo, which I managed to get the hang of quite quickly.

Another goal of mine was to improve the communication between me and my peers. I often tend to take a somewhat passive role in a team, whenever I end up in a leading position I have difficulty distributing work and end up taking on a lot of tasks myself.
When I got placed in a team with only one other member I figured there would not really be a need for such a leader role within the project. As the project went by however, I realized that this was not working out. It seemed to me both me and my peer were somewhat passive and it caused things not to run very smoothly. On top of that, the added difficulty of having to work from home and an added language barrier between me and my peer drove me to the realization that I had to take on more of a leader role and start driving the project. I started assigning clear tasks and came up with ways to improve the language barrier. It helped a lot to put things in writing and come up with clear examples of what was meant. This made my teammate and I work more efficiently as a team and allowed us to use our combined strengths to finish our prototype. I am glad I have been able to use my final group project to be able to take this role.

The last goal I had set for myself was to incorporate users much earlier on in the process. This went poorly mostly as a result of the pandemic. Because of the pandemic it became far more difficult to get in touch with schools for our project, it was only later on that I found some connections with teachers with the help of my personal network. It's a shame that I wasn't able to put many user evaluation methods to use, as especially the tests were quite informal.
What I did gain out of this project in this regard is that I now have some connections with teachers who have told me that they are willing to help me again in the future if I do another project relating to primary schools, which is likely.

Another important learning point this project, unrelated to my goals, was in regards to the planning of the project. In the planning we made at the start of the project, we had put a heavy emphasis on getting to know our users at the beginning of the project, mostly because both me and my teammate found it important to involve users early on. What made this difficult however, was that at this point in the project we had no clear idea of who our users were going to be.
While the interviews done at the end of the project would have been very valuable in the beginning phase, it wasn't until the lockdown started that we had decided to focus on English teaching. If we started sooner with conceptualization instead of research, we would have had a clearer idea of what to research. While now we spent a lot of time researching both math and English teaching.
Another point in planning was the making of the demonstrator. I waited with the start of the development until the moment we had completed our concept. This should have been done much earlier on. From the moment we decided we were going to something with scanning cards I could have started programming this behavior, which would have saved time later on.

Overall the process of this project has been a bumpy road due to many external, unforeseen factors. This has also made it a very informative experience. I have learned more than just what I set out to do and I will carry these experiences into upcoming projects.

Appendix B - Translated Learning goals

Common European Framework of Reference (CEFR): https://erk.nl/#home
This is a European standard for different levels of a language people are expected to have.
When starting out in high school students are expected to have an A1 level of skill, this is what primary schools work towards.
This includes:

Listening

Vocabulary	Very simplistic, different sentences divided by long breaks
Tempo and articulation	Speaks careful, slow but clear
Text length	Texts are short and very simple
Listening skills	Can determine probable meaning of texts about familiar subjects

Reading

Vocabulary	Can understand most important words and short and simple sentences
Text structure	Used as visual guidance
Text length	Short, simple texts
Reading strategies	Can determine probable meaning of texts about familiar subjects through use of layout and (when available) (online) dictionaries.

Conversation

Vocabulary	Everyday words, short, simple sentences.
Tempo and articulation	Speaks careful, slow but clear. Long breaks to think about meaning of the words
Aid	The Conversation partner speaks slowly and articulates clearly. Is prepared to repeat what they said.

Writing

Vocabulary	Very basic vocabulary consisting of isolated words and expressions relating to the person or concrete situations.
Grammar	Shows only a limited skill in some basic grammatical concepts.
Production strategies	Can make a decent use of (online) dictionaries and spelling/grammar control in text editing software.

SLO (specifiek leraren opleiding or “specific teacher education”)
created **TuLe** (Tussendoelen & Leerlijnen or “intermediate goals & learning lines”) for english education in primary schools in the netherlands.
https://tule.slo.nl/Engels/F-KDEngels.html
The goals for english are listed here:

Core goal 13:

students learn to obtain information out of simple spoken and written pieces of english text.

Core goal 14:

Students learn to ask for things in english, or talk about simple topics. Doing so they create an attitude in which they feel confident expressing themselves in that language.

Core goal 15:

Students learn how to write certain simple words about everyday subjects.

Core goal 16:

Students learn how to obtain the meaning and spelling of words from a dictionary.

Goal 13:	Grade 1 and 2	Grade 3 and 4	Grade 5 and 6	Grade 7 and 8
Goal	Develop a positive attitude towards English. No English class but a few activities a week are aimed at English.	Learning listening skills. No English class but a few activities a week are aimed at English.	Starting a conscious effort towards retrieving information from listening exercises. Careful start in reading. Start with EIBO themes*	Kids can use familiar words from the listening and reading program. Continuation of EIBO themes*
Lessons	Predetermined themes or subjects will be discussed. English is preferably taught by a native speaker or specialized teacher. Listening skills combined with visual. No reading yet.	Continuing with themes and subjects covered in 1st and 2nd grade Listening skills combined with visual. Careful introduction of spelling so as to not confuse it with Dutch spelling.	Reading material relates to simple and familiar subjects about everyday life. Kids become familiar with the role of English in their direct surroundings. Reading and listening skills prepares them for the levels required in 7th and 8th grade.	Reading gets an increased importance There’s a balanced availability of simple reading material At the end of 8th grade kids are able to retrieve information out of text. Reading and listening skills adequate for high school levels.
Material	Picture books, videos, CDs, puppets, other visual material.	More visual material. Method book with no English text for the kids.	More visual material. Method book directed at age group.	All material on the kids level of skill

* EIBO themes (EIBO stands for Engels in het Basis Onderwijs or “English in the primary education”)
The Eibo-thema's are:

- Getting to know someone
- Living somewhere
- Spare time and hobbies
- Food and drink
- Time indication
- Describing a person
- Out on the street
- In a store
- In the classroom
- Parties/Festivities
- The weather

Themes are expanded on every year starting from the 5th grade.
For example the theme “living somewhere”

- in 5th grade the address
- In 6th grade the house and various living quarters
- In 7th grade countries and nationalities
- In 8th grade the furnishing of your room and the region you live in (north east south west)

Goal 14:	Grade 1 and 2	Grade 3 and 4	Grade 5 and 6	Grade 7 and 8
Goal	Develop positive attitude towards English speaking	Continue developing speaking skills	Kids start properly speaking English.	Same as 5th and 6th grade.
Lessons	A few hours a week Listening to and repeating are the most important skills in these grades English is preferably taught by a native speaker or specialized teacher.	Speaking and listening are still the most important skills. (in 3rd grade kids start writing Dutch so it is important not to confuse them with English writing). Kids feel comfortable speaking English. They spontaneously respond in English, not realising the difference between Dutch and english.	Kids develop a positive attitude towards speaking a foreign language. Subjects are basic and relate to everyday life. Mostly learning standard sentences from the top of their head: Do you like (ice cream) Is it (red) or (blue)? It’s a (giraffe) Through playful manners kids are stimulated to show what they know and can say.	Ability to talk English to strangers outside of school Kids develop a positive attitude towards speaking English Speaking tempo is low Grammatical correctness is not important aside from a few things Kids make use of non verbal communication where they can’t communicate in English (pointing at an object they don’t know the word of) Limited vocabulary.
Material	Picture books, videos, CDs, puppets, other visual material.	More visual material. Method book with no English text for the kids.		

Goal 15:	Grade 1 and 2	Grade 3 and 4	Grade 5 and 6	Grade 7 and 8
Goal	Writing does not play a role yet	Writing does not play a role yet, should be postponed due to the kids learning to write Dutch in these grades. Learning to write English might interfere with that.	Some kids are curious to the writing of English words and start experimenting. This can be encouraged	Explore the fun and purpose of writing and communicating in English
Lessons				Kids are familiar having to fill in forms in English (nationality: dutch) Heavier emphasis on writing as a tool and goal.

Goal 16:	Grade 5 and 6	Grade 7 and 8
Goal	Kids look up or check the spelling of words	Kids become familiar using dictionaries as part of language education.
Lessons	Use of dictionaries, lists of words and attention to reading and writing.	Regular use of dictionaries
Material	Dictionaries and lists of words from method books Use of websites or CDs to listen how a certain word is pronounced.	Dictionaries that connect to the courses and subjects of the age and skills levels of the kids. Both Dutch - english and English English sources.

Appendix C - Teacher Interviews

Interview 1

Third grade teacher.
At the moment not a lot of attention for English learning.
For toddlers - mostly use of songs
Third grade:

Also use of songs.
This teacher did not give English lessons, third grade is very busy in learning Dutch (50% of the classes) and math's.
In the afternoons there is time (currently there is a computer moment) in which there could be time for the game.

Many schools already work with tablets.
-downside to this, kids don't get to learn to think as much
Programs like Blink and Snappet Pupil
This works for some students, but not others!
Digital solutions should not replace traditional teaching but add to it.

About the concept:
There should be as little restrictions as possible. If the kids create the sentence "the cupboard walks to the store" let them.
This allows them to be more creative and the game to be more fun. It will also let the game have triggers.
In third grade for example, they do crazy writing exercises like writing in shaving foam on the tables, writing in their air. This is to let the kids use all their senses to express themselves and learn these words. This is in accordance with multiple intelligence theory.

Our game has the capability to use many senses to learn (reading, visual, audio)
It could work from third grade and up, slowly expanding the difficulty as kids progress. Start with a small amount of cards and expand this.
Unlocking levels for motivation, also song and good job animation when finishing a level is very rewarding for kids. (song could include English vocals to also teach the kids that song).

If the schools have to buy a camera separately it is probably too expensive/too much work to set up. Also, if the camera doesn't work this will cost a lot of time.
It's best to use the camera on the tablet. This could also be an interesting cooperative setting where one student looks on the tablet saying what they see while the other puts down cards.

Interview 2

Has been a teacher for 10 years in various schools.
Currently teaching 7th grade special needs kids.

Has experience teaching English, their school uses a method book called “Take it easy”
A colleague also uses an additional tool called Groove.

Senior students get English class, she thinks there should be more attention given to English seeing as how important it is (it’s in movies, news, online, etc.).

The difficulty is the wide variety in skill level. A computer program/app that is able to determine the student’s skill level and assign them tasks based on that would be helpful.
Games also motivate children a lot.

Their school makes use of chromebooks and the learning tool Snappet. But there are more students than they have chromebooks. Some kids need to share and sometimes they are not available because another class is using them.

About the concept:
Overall she was very positive about the concept.
She thinks it’s well suited for junior students like grade 3 and 4. These kids will be especially amazed by how the 3D objects appear on the cards.
For older kids the difficulty would need to be increased.

The game would fit well into the current education system. The method books usually run out of material quickly. Any extra material is welcome.

Appendix D - User study

User 1 (5 years old)
This user was very young, and had only recently learned to read.
When presented with cards and sentences, he recognized the letters, not words. If we want to make the game accessible for kids this age we'll have to make sure there's an audible component to it, or that the kid is at least accompanied by an adult such as a parent.

Playing the game this user needed a lot of assistance, partially due to the demo not always recognizing the cards, making placing the cards quite difficult.

After playing with every sentence they wanted to try out each of the cards to see the 3d model appear. If this somehow also makes the word appear (maybe in an audible way) even this can teach them something.

User 2 (10 years old)
This user understood the game very quickly and was even able to play the game without any help.
Laughed when they constructed a weird sentence (the balloon picks up the dog).
Also laughed when girl kicked the cat

They knew all the words already because in class they cover English as well. They mostly do this by translating songs.

They really liked the characters, and animating them.

They would like the play either alone or with friends

When asked what they would change about the game, they said they would like to have more control over the sentence.

User 3 (8 years old)
This user, although being within our target group age wise, had to do a year over and was still in the third grade as a result. This also resulted in her having more trouble with playing the game then expected. They found it very hard to answer questions or deduce the meaning of words. When given hints and helped she would be able to do it.

They often misidentified words based on the pictures.

They got distracted when answering questions and started to play around with the game instead. (which can be seen as a good thing)

Loose observations
At some point one of the users wanted to set the location and tried to do this by putting a location card on the scene page.

When asking what a certain word card meant, a user read “object” out loud, since that is at the top of the card. The fact that the actual word was at the bottom had to be pointed out to them.
They also thought the number was part of the word.
Perhaps it’s better to put just the word on the card, or at the very least make it stand out more (especially on the yellow card) and also leave out the number.

Sometimes the kids don’t immediately recognize the picture on a card. Pancakes were mistaken for cakes or pies, mostly due to the cherry on top, and the bedroom mistaken for a couch.

Parent (who is a teacher): I would suggest using a more default font, either hand written or block letters, since this is what the kids are being taught.

Currently the game has no incentive for the kids to actively try to translate words and sentences.

Kids have quite deep expectations of the possibilities of a game. One expected a balloon to pop when a dog picked it up, because it does so in real life. Another expected that giving a man flowers would cause him to plant it (despite the sentence being about dancing).

The kids sometimes missed the animation or thought that the dance characters did after completing a task was part of the animation. It should be clearer these are separate.

Appendix E - Excerpt of code

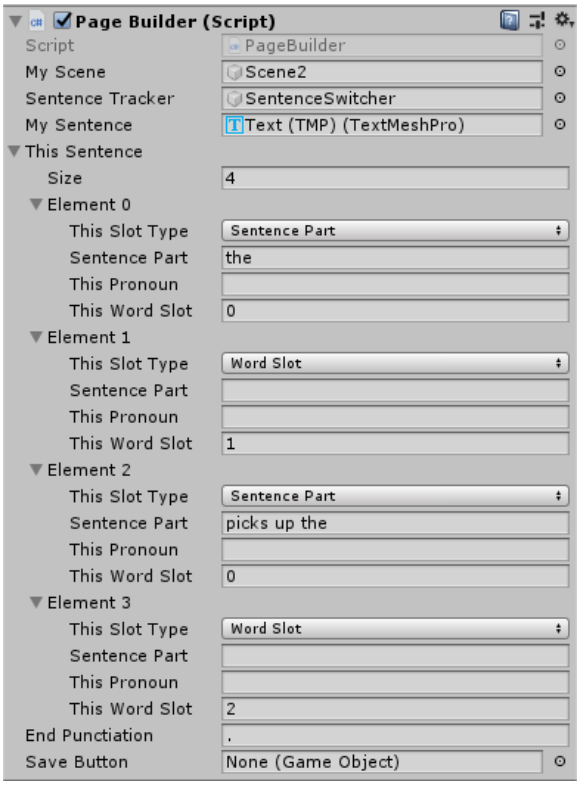
ARCards – script that gives the AR cards their data, which can be set in the editor.

```
using System;
using System.Collections.Generic;
using UnityEngine;

public class ObjectStorage : MonoBehaviour
{
    public enum WordType
    {
        Character,
        Object,
        Place
    }
    public enum Pronoun
    {
        he,
        she,
        it
    }

    [Serializable]
    public class CardData
    {
        public string Word;
        public WordType MyWordType;
        public GameObject Model;
        public GameObject EnvironmentModel;
        public Pronoun MyPronoun;
    }

    public CardData ThisCardData;
}
```

DetectCards – script that retrieves and AR card's data once it's placed on the sentence page. Currently the pieces of code that remove the cards are disabled, because the cards often lost tracking causing them to be removed, which was very annoying when playing.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class DetectCards : MonoBehaviour
{
    public GameObject mySentenceTracker;
    public int slot;

    GameObject myCurrentCard;

    public delegate void CardDetection(int slot, GameObject card);
    public static event CardDetection ReceiveCard;
    public static event CardDetection RemoveCard;
    /*
    void Update()
    {
        if (myCurrentCard != null && !myCurrentCard.activeSelf)
        {
            if (RemoveCard != null)
                RemoveCard(slot, myCurrentCard);
            myCurrentCard = null;
        }
    }
    */
    private void OnTriggerEnter(Collider other)
    {
        if (other.GetComponent<ObjectStorage>() != null)
        {
            myCurrentCard = other.gameObject;
            if (ReceiveCard != null)
                ReceiveCard(slot, myCurrentCard);
        }
    }
    /*
    private void OnTriggerExit(Collider other)
    {
          if (other.gameObject == myCurrentCard)
          {
              if (RemoveCard != null)
                  RemoveCard(slot, other.gameObject);
            myCurrentCard = null;
          }

    }
    */
}
```

Pagebuilder – script that constructs the sentences on the sentence page. In the editor the contents of the sentence can be typed out or selected. As can be seen in the image on the right.
This script stitches all those sections together into a complete sentence and displays it.
When the word detectors send out a new word, this script adds it to the sentence on screen.




```
using System;
using UnityEngine;
using TMPro;
using Vuforia;

public class PageBuilder : MonoBehaviour
{
    public GameObject myScene;
    public GameObject SentenceTracker;
    public TextMeshPro mySentence;

    public enum SlotType
    {
        SentencePart,
        Pronoun,
        WordSlot
    }
    //create settings to be set in the editor
    [Serializable]
    public class SentenceStructure
    {
        public SlotType thisSlotType;
        public string SentencePart;
        public string thisPronoun;
        public int thisWordSlot;
    }

    public SentenceStructure[] ThisSentence;

    public string endPunctiation = ".";

    string [] Words = new string[3] {"(1)" , "(2)" , "(3)"};

    private string sentenceText;

    public GameObject saveButton;

    void Start()
    {
        //update the sentence at the start so it displays the default
        UpdateSentence();
    }
```

```
public void subscribe()
    {
        //subscribe to the card detection events
        DetectCards.ReceiveCard += ReceivingCard;
        DetectCards.RemoveCard += RemovingCard;
    }

    private void OnDestroy()
    {
        DetectCards.ReceiveCard -= ReceivingCard;
        DetectCards.RemoveCard -= RemovingCard;
    }

    void ReceivingCard(int slot, GameObject card)
    {
        //when receiving a card from the card detectors, add it's word to the sentence
and update.
        if
(SentenceTracker.GetComponent<SentenceSwitcher>().currentSentence.GetComponentInChildren<
PageBuilder>() == this)
        {
            Words[slot-1] = card.GetComponent<ObjectStorage>().ThisCardData.Word;
            UpdateSentence();
        }
    }

    void RemovingCard(int slot, GameObject card)
    {
        Words[slot - 1] = null;
        UpdateSentence();
    }

    public void UpdateSentence()
    {
        if (Words[0] == null)
            Words[0] = "(1)";
        if (Words[1] == null)
            Words[1] = "(2)";
        if (Words[2] == null)
            Words[2] = "(3)";

        sentenceText = "";
        for (int i = 0; i <ThisSentence.Length; i++)
        {
            if (i == 0) //on the first word, don't add a space, only the sentence part.
            {
                switch (ThisSentence[i].thisSlotType)
                {
                    case SlotType.Pronoun:
                        sentenceText = sentenceText + ThisSentence[i].thisPronoun;
                        break;
                    case SlotType.SentencePart:
                        sentenceText = sentenceText + ThisSentence[i].SentencePart;
                        break;
                    case SlotType.WordSlot:
                        sentenceText = sentenceText + Words[ThisSentence[i].thisWordSlot
- 1];
                        break;
```

```
                }
            }
            else //otherwise add a space and then the sentence part.
            {
                switch (ThisSentence[i].thisSlotType)
                {
                    case SlotType.Pronoun:
                        sentenceText = sentenceText + " " + ThisSentence[i].thisPronoun;
                        break;
                    case SlotType.SentencePart:
                        sentenceText = sentenceText + " " + ThisSentence[i].SentencePart;
                        break;
                    case SlotType.WordSlot:
                        sentenceText = sentenceText + " " +
Words[ThisSentence[i].thisWordSlot - 1];
                        break;
                }
                if (i == ThisSentence.Length - 1) //when the last word is added, add the
punctiation to the end of the sentence.
                {
                    sentenceText = sentenceText + endPunctiation;
                }
            }
        }
        sentenceText = UppercaseFirst(sentenceText); //make the first letter uppercase
        mySentence.text = sentenceText;

    }
    string UppercaseFirst(string s)
    {
        if (string.IsNullOrEmpty(s))
        {
            return string.Empty;
        }
        char[] a = s.ToCharArray();
        a[0] = char.ToUpper(a[0]);
        return new string(a);
    }
}
```

SceneMaster – generic script that constructs scene of the animation. It places down the models that are assigned to the card that got detected.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class SceneMaster : MonoBehaviour
{
    public GameObject[] SpawnPoints = new GameObject[3];
    public GameObject[] currentWords = new GameObject[3];
    public GameObject[] instantiatedmodels = new GameObject[3];
    public RuntimeAnimatorController[] animationsToSpawnWith = new
RuntimeAnimatorController[3];
    public GameObject ParticleSpawner;

    // Start is called before the first frame update
    void Start()
    {
        UpdateScene();
    }
    public void subscribe()
    {
        DetectCards.ReceiveCard += ReceivingCard;
        DetectCards.RemoveCard += RemovingCard;
    }

    private void OnDestroy()
    {
        DetectCards.ReceiveCard -= ReceivingCard;
        DetectCards.RemoveCard -= RemovingCard;
    }

    void ReceivingCard(int slot, GameObject card)
    {
        if (gameObject.activeSelf == true)
        {
            if (instantiatedmodels[slot - 1] != null)
                Destroy(instantiatedmodels[slot - 1]);
            currentWords[slot - 1] = card;
            if (SpawnPoints[slot - 1] != null)
            {
                //spawn object
                instantiatedmodels[slot - 1] =
Instantiate(card.GetComponentInChildren<ObjectStorage>().ThisCardData.Model);
                instantiatedmodels[slot - 1].transform.SetParent(SpawnPoints[slot -
1].transform);
                instantiatedmodels[slot - 1].transform.localPosition = new Vector3(0, 0,
0);
                instantiatedmodels[slot - 1].transform.localRotation =
Quaternion.identity;
                //if it's a charcter and there is an animation for it, give it that
animation
                if (card.GetComponentInChildren<ObjectStorage>().ThisCardData.MyWordType
== ObjectStorage.WordType.Character && animationsToSpawnWith[slot-1] != null)
                {
```

```
                    instantiatedmodels[slot -
1].GetComponent<Animator>().runtimeAnimatorController = animationsToSpawnWith[slot - 1];
                }
                //create particle effect
                GameObject Particles = Instantiate(ParticleSpawner);
                Particles.transform.SetParent(SpawnPoints[slot - 1].transform);
                Particles.transform.localPosition = new Vector3(0, 0, 0);
                Particles.transform.localRotation = Quaternion.identity;
            }
        }
    }

    void RemovingCard(int slot, GameObject card)
    {
        if (gameObject.activeSelf == true)
        {
            currentWords[slot - 1] = null;
            Destroy(instantiatedmodels[slot - 1]);
        }
    }

    void UpdateScene()
    {

    }
}
```

Scene (4) – Each sentence has it's own scene script that handles the behavior of the animation. This example is of the 4th sentence; The (1) kicks the (2).

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class Scene4 : MonoBehaviour
{
    private Animator objectAnimation;

    void Start()
    {
        SentencepagePlay.StartButtonPressed += playAnimation;
        SentencepageStop.StopButtonPressed += stopAnimation;

        objectAnimation =
this.gameObject.GetComponent<SceneMaster>().SpawnPoints[1].GetComponent<Animator>();
        stopAnimation(true);
    }

    void playAnimation(bool Pressed)
    {
        if (gameObject.activeSelf == true)
        {
            if (Pressed == true)
            {
                stopAnimation(true);

                //animation of first character
                if (this.gameObject.GetComponent<SceneMaster>().currentWords[0] != null)
                {
                    if
(this.gameObject.GetComponent<SceneMaster>().currentWords[0].GetComponent<ObjectStorage>(
).ThisCardData.MyWordType == ObjectStorage.WordType.Character)
                    {

this.gameObject.GetComponent<SceneMaster>().instantiatedmodels[0].GetComponent<Animator>(
).SetBool("Play", true);
                        if
(this.gameObject.GetComponent<SceneMaster>().currentWords[1] != null)
                        {

this.gameObject.GetComponent<SceneMaster>().instantiatedmodels[0].GetComponent<Animator>(
).SetBool("Succes", true);
                            if
(this.gameObject.GetComponent<SceneMaster>().currentWords[1].GetComponent<ObjectStorage>(
).ThisCardData.MyWordType == ObjectStorage.WordType.Character)
                            {
                                //when kicking a character, they become sad

this.gameObject.GetComponent<SceneMaster>().instantiatedmodels[0].GetComponent<Animator>(
).SetBool("Sad", true);
```

```
                            }
                            else
                            {
                                objectAnimation.SetBool("Play", true);
                            }
                        }
                    }
                }
                //animation of second character (if any)
                if (this.gameObject.GetComponent<SceneMaster>().currentWords[1] != null)
                {
                    if
(this.gameObject.GetComponent<SceneMaster>().currentWords[1].GetComponent<ObjectStorage>(
).ThisCardData.MyWordType == ObjectStorage.WordType.Character)
                    {

this.gameObject.GetComponent<SceneMaster>().instantiatedmodels[1].GetComponent<Animator>(
).SetBool("Play", true);
                        if
(this.gameObject.GetComponent<SceneMaster>().currentWords[0] != null)
                        {
                            if
(this.gameObject.GetComponent<SceneMaster>().currentWords[0].GetComponent<ObjectStorage>(
).ThisCardData.MyWordType == ObjectStorage.WordType.Character)
                            {

this.gameObject.GetComponent<SceneMaster>().instantiatedmodels[1].GetComponent<Animator>(
).SetBool("Succes", true);
                            }
                        }
                    }
                }
            }

        }
        void stopAnimation(bool Pressed)
        {
            if (gameObject.activeSelf == true)
            {
                if (Pressed == true)
                {
                    objectAnimation.SetBool("Play", false);
                    objectAnimation.gameObject.SetActive(false);
                    objectAnimation.gameObject.SetActive(true);

                    if (this.gameObject.GetComponent<SceneMaster>().currentWords[0] != null)
                    {
                        if
(this.gameObject.GetComponent<SceneMaster>().currentWords[0].GetComponent<ObjectStorage>(
).ThisCardData.MyWordType == ObjectStorage.WordType.Character)
                        {

this.gameObject.GetComponent<SceneMaster>().instantiatedmodels[0].GetComponent<Animator>(
).SetBool("Play", false);

this.gameObject.GetComponent<SceneMaster>().instantiatedmodels[0].GetComponent<Animator>(
).SetBool("Succes", false);
```

```
this.gameObject.GetComponent<SceneMaster>().instantiatedmodels[0].GetComponent<Animator>(
).SetBool("Sad", false);

this.gameObject.GetComponent<SceneMaster>().instantiatedmodels[0].SetActive(false);

this.gameObject.GetComponent<SceneMaster>().instantiatedmodels[0].SetActive(true);
                    }
                }
                if (this.gameObject.GetComponent<SceneMaster>().currentWords[1] != null)
                {
                    if
(this.gameObject.GetComponent<SceneMaster>().currentWords[1].GetComponent<ObjectStorage>(
).ThisCardData.MyWordType == ObjectStorage.WordType.Character)
                    {

this.gameObject.GetComponent<SceneMaster>().instantiatedmodels[1].GetComponent<Animator>(
).SetBool("Play", false);

this.gameObject.GetComponent<SceneMaster>().instantiatedmodels[1].GetComponent<Animator>(
).SetBool("Succes", false);

this.gameObject.GetComponent<SceneMaster>().instantiatedmodels[1].SetActive(false);

this.gameObject.GetComponent<SceneMaster>().instantiatedmodels[1].SetActive(true);
                    }
                }
            }
        }
    }
}
```