LegoLens: LEGO for the Microsoft Hololens

Thomas Nyegaard-Signori
sfq340@alumni.ku.dk

Tobias Carlos Tvarn
e-mail address

Tor-Salve Dalsgaard
e-mail address

Enes Golic
e-mail address

ABSTRACT
This project set out to produce an augmented reality application using LEGO on the Microsoft Hololens. Using the open source toolkit produced by Microsoft called Holotoolkit a prototype was built. (SKRIV NOGET OM USER TEST).

Author Keywords
Microsoft Hololens; Holotoolkit; LEGO; Augmented reality.

INTRODUCTION
This report focuses on an application for the newly released Microsoft Hololens. The Hololens was released around march 2017, and the fact that the product is in its infant stage and is based on a very new technology opens up possibilities and removes any preconceived notions about which applications and uses the Hololens might have.
The technology is relevant with regards to mobile computing in one very apparent way, in that it is a wearable, computing unit. Other than that, it offers alternate reality (AR) possibilities because of its partly see-through screens and user tracking. Since the Hololens is still a new technology, the mobility and computing power of the product will most likely increase, making it resemble a ubiquitous computer more and more. This evolution of mobile computing was one of the reasons that the Hololens was chosen to develop on in the first place.

The application this report will cover revolves around LEGO. LEGO is a way for kids and adults to build constructions, vehicles and scenery, all in a very physical and three-dimensional way. This, then, seemed like a natural choice for an AR application, since the application layer between the user and the world could expand naturally on the possibilities and limitations of the physical, "real-world" LEGO.
The application in itself should be a sort of digital playground in which a user could interact with LEGO in ways they would find natural. Sticking pieces together the way they do in real life, stacking and constructing, all interactions that the user knows well from having played around with real LEGO. This was done using interaction through a virtual tablet, known as the "Generator board" and simple drag-and-drop with the bricks. We end up with a rough prototype which helped us discover the pitfalls and consideration concerning a LEGO implementation in an AR enviroment.

DESIGN CHALLENGES

How should the user access the application?
A menu screen is an essential part of any application. The menu is something every user has experience with and it is the first thing a user is met with when running an application. This means that the menu has to consist of certain classic elements.

A user needs a way to close the application. On mobile phones nowadays this can be done with 'return' buttons on the phone, but applications generally have a built in exit function.

The user also needs to have some sort of options menu and guidelines. This is a must for this application. Stacking LEGO seems simple and intuitive, but all the operations and possibilities is something that can confuse a potential user.

Lastly the menu needs to have an easy access to the LEGO session itself. It shouldn't be complicated for the user to start a new LEGO session.

The Main Menu

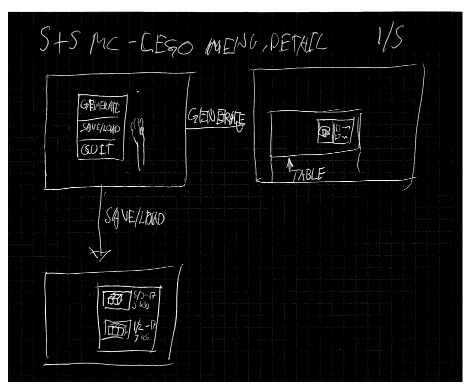

Figure 1. The first idea of a seperate Main Menu screen

We initially though about having a main menu screen. Just as figure1 illustrates. The user would open the Hololens application, and be met with a menu screen, where actions such as generating a play area, loading and saving a game and a way to quit the application.

The sketches were generated following a ’5 plus 5’ approach. Each member of the group drew 5 different ideas of a main menu screen. One common theme in the sketching of the main menu was separation of the design of the menu and the interaction with the menu, and the difficulties with making that separation.

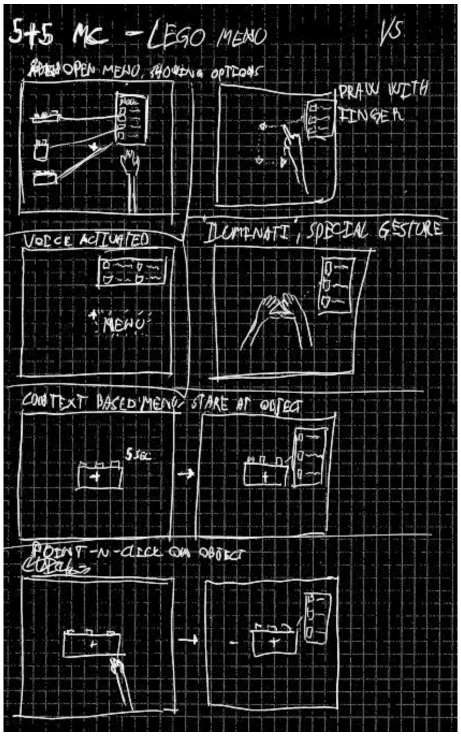

Figure 2. A focus on gestures rather than design, impacted the design. The figure shows 5 different ways of creating an interactive menu for the user to use.

In quite a few of the earlier sketches, interaction with the menu was in focus rather then the design of the menu, just as figure 2 depicts.

The possibilities with the HoloLens gestures and the augmented reality made it apparent that the menu had to be movable, either by closing and opening with the gestures available, or being able to move it around the play area.

The generator board

After discussing menus in the context of the Hololens, it became apparent that there was a need for a menu that could be placed and interacted with in the real world. Using the spatial mapping and spatial understanding capabilities of the Hololens, different designs of the so called ’generator board’ were done. Figure 3 is a sketch of what functionalities the generator board should have. The main idea was to have a generator board that was accompanied with functionalites to alter the blocks, but also to be interactive and movable in the play area.

We designed the generator board as a tablet device, making it more natural for the user to interact with. It follows the idea of the sketch in figure 4, where the generator board is placed on the table. Being a solid object the user knows, makes it more

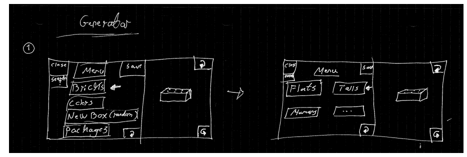

Figure 3. An initial sketch of how the generator board could look like. The menu screen has alot of functionalities, and the actions chosen happen on the right side.

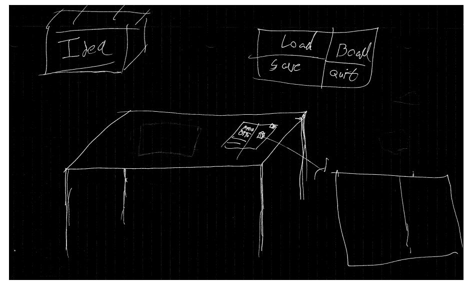

Figure 4. First sketch of the generator board as a movable object. Here placed on the table, spawning blocks on the table.

intuitive and natural for the user to lift, move and place the menu on a surface like a table or the floor. Figure 5 show some of the thoughts made towards the functionality and buttons inside the generator board. Another concern was that each menu needed a ’back’ button so the user could return to the main screen. Colors are also a must to be changed.

Design decisions

During the sketching phase, several design choices were discussed. One of the very first decisions made was the overall look of the menu. The choice of big buttons, clear visual cues and short textual descriptions was present in almost all of the sketches in the early design phase, as seen in figure 6. The generator board needed to have the same simple design. The main menu and the generator board should not be overcomplicated, but not sparse in functions either.

Moving away from a main menu

As the development of the application progressed it became more and more apparent that an actual main menu was not necessary. All the interactions needed for the prototype could be implemented through the tablet looking generator board and could ease user interaction with the application. Instead of going through a main menu and then having the generator which contains the functionality for working with the LEGO bricks, spawning the generator board at the application start up and "cutting out the middleman" seemed as a natural choice for this prototype. Granted, with an eventual increase in functionality and complexity of the application, a root/main menu might prove useful as to not clutter the users experience when they are building as opposed to when they are in the main menu setting options, loading scenarios, downloading templates etc.

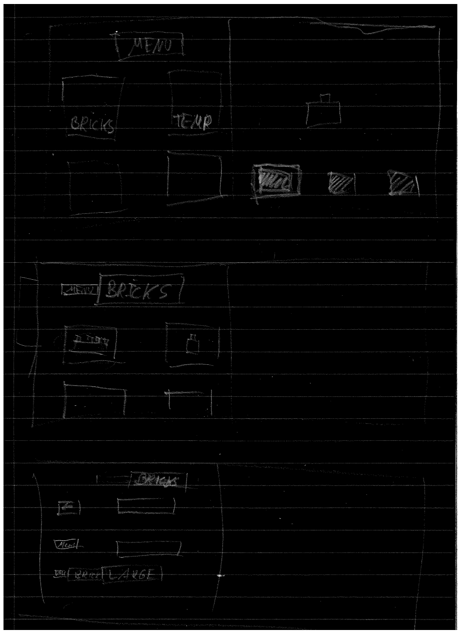

Figure 5. An initial sketch of the functionalities within the generator board. It has various buttons, such as generating a specific block, a template screen with predefined figures etc.

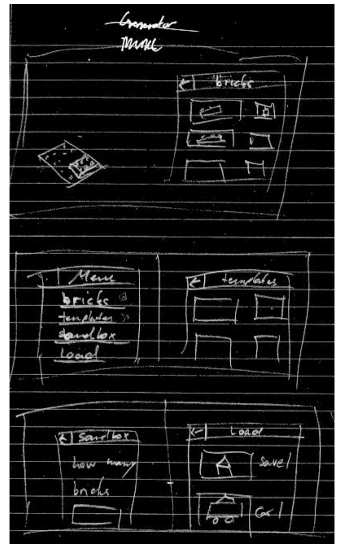

Figure 6. Initial Main Menu screen with big button, simple interaction and short descriptions.

IMPLEMENTATION

Unity

The application was built using Unity and mainly written in C#. Unity provides a real-time interaction with the scenes in an application and has tools to ease the development of AR applications. (FUCKING SKRIV MERE).

Holotoolkit

The main package used for developing for the Hololens in this project was the HoloToolkit-Unity. This package comes with some premade functionality to ease the interaction with the Hololens and is an open source toolkit made by Microsoft to speed up any development for their new platform (quote dem måske?).

The toolkit contains seven feature areas, of those spatial mapping and input were of most interest to us. We used the spatial mapping part of the toolkit to be able to "digitize" the world and make our application able to track surfaces so as to place our generator board on real world surfacs instead of having it float in mid air. This connection between the real-world and digital playground created in our application was essential, both to the experience but also to be able to call our application alternate reality.
The spatial mapping relies on the time-of-flight depth cameras and RGB cameras to provide a robust tracking of the enviroment. This mapping is then readily available to developers through the Holotoolkit and can be applied to object in an application. (SKRIV MERE TEKNISK NÅR VI VED HVAD FUCK DER FOREGÅR)

The input part of the toolkit allows us to track the gaze and the users interaction with the objects in the application, be it buttons, bricks or the generator board. This tracking is done by shooting a ray from the users gaze (the middle of the screen on the Hololens in our case) and checking whether any colliders, object hitboxes, were hit. This raycasting, as its called, is intuitive since it uses the line of sight from the user to any object in the gameworld, so whatever you can see, you can interact with.
More specifically, the orientation and position of the Hololens with regards to the objects in the gameworld is maintained by a GazeManager from the HoloToolkit and the cursor is then placed on the vector originating from the users gaze by a CursorManager. This raycast depends on the Hololens ability to track the user using gyroscopes, accelerometers and computer vision. By tracking the user gaze in the real world and imposing a mapping from the real-world to a virtual-world coordinate system, the user can be mapped in with reference to the virtual objects (QUOTE EN TEKST, FÅ DET TIL AT LYDE MINDRE OSTET).

DISCUSSION

Suitability of LEGO in AR

One of the aspects with the application was the suitability of LEGO in an AR setting. This question was asked rather late in the development process. What became apparent was that the virtual LEGO in an AR setting would not be able to provide the same "finicky" feel that LEGO has, sitting at a table, obsessing over small details in an advanced (?) setup. This is because of the computing limitiations of the Hololens and the technology architecture. The minimal rendering distance of the Hololens is, right now, much larger than the distance one would be from af real-life LEGO project, ie., arms length. This limitation demands a much larger brick size than real-life LEGO and this in turn limits the overall complexity of a virutal LEGO project.

Figure 7. Insert a caption below each figure. Do not alter the Caption style. One-line captions should be centered; multi-line should be justified.

		Test Conditions	
Name	*First*	*Second*	*Final*
Marsden	223.0	44	432,321
Nass	22.2	16	234,333
Borriello	22.9	11	93,123
Karat	34.9	2200	103,322

Table 1. Table captions should be placed below the table. We recommend table lines be 1 point, 25% black. Minimize use of table grid lines.

FUTURE WORK

LEGO size and compexity

To combat the issue with the size of actual LEGO and the complexity available in this scale a virtual magnifying glass could be envisioned. Using a special gesture, a certain area of the LEGO bricks could be brought into view using a secondary "screen", a menu that could show a zoomed in version of the actual view the user has. (Er det lort?)

Brick shortage

We have a severe shortage of bricks available in our prototype. Anyone who considers an AR LEGO application will soon have to think of which blockset or types they would include. The amount of distinctive LEGO bricks is staggeringly high, and this sort of work would probably benefit greatly from working together with LEGO as to get dimensions and oddities right. This work would also make such an application much more attractive, as one of the strongest selling points of LEGO is the variety but ensured compatibility.

REFERENCES

1. ACM. 1998. How to Classify Works Using ACM's Computing Classification System. (1998). http://www.acm.org/class/how_to_use.html.

2. R. E. Anderson. 1992. Social Impacts of Computing: Codes of Professional Ethics. *Social Science Computer Review December* 10, 4 (1992), 453–469. DOI: http://dx.doi.org/10.1177/089443939201000402

3. Anna Cavender, Shari Trewin, and Vicki Hanson. 2014. Accessible Writing Guide. (2014). http://www.sigaccess.org/welcome-to-sigaccess/resources/accessible-writing-guide/.

4. @_CHINOSAUR. 2014. "VENUE IS TOO COLD" #BINGO #CHI2014. Tweet. (1 May 2014). Retrieved Febuary 2, 2015 from https://twitter.com/_CHINOSAUR/status/461864317415989248.

5. Morton L. Heilig. 1962. Sensorama Simulator. U.S. Patent 3,050,870. (28 August 1962). Filed Februrary 22, 1962.

6. Jofish Kaye and Paul Dourish. 2014. Special issue on science fiction and ubiquitous computing. *Personal and Ubiquitous Computing* 18, 4 (2014), 765–766. DOI: http://dx.doi.org/10.1007/s00779-014-0773-4

7. Scott R. Klemmer, Michael Thomsen, Ethan Phelps-Goodman, Robert Lee, and James A. Landay. 2002. Where Do Web Sites Come from?: Capturing and Interacting with Design History. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '02)*. ACM, New York, NY, USA, 1–8. DOI:http://dx.doi.org/10.1145/503376.503378

8. Psy. 2012. Gangnam Style. Video. (15 July 2012). Retrieved August 22, 2014 from https://www.youtube.com/watch?v=9bZkp7q19f0.

9. Marilyn Schwartz. 1995. *Guidelines for Bias-Free Writing*. ERIC, Bloomington, IN, USA.

10. Ivan E. Sutherland. 1963. *Sketchpad, a Man-Machine Graphical Communication System*. Ph.D. Dissertation. Massachusetts Institute of Technology, Cambridge, MA.

11. Langdon Winner. 1999. *The Social Shaping of Technology* (2nd ed.). Open University Press, UK, Chapter Do artifacts have politics?, 28–40.

Figure 8. In this image, the map maximizes use of space. You can make figures as wide as you need, up to a maximum of the full width of both columns. Note that LaTeX tends to render large figures on a dedicated page. Image: 🅭🅯⊜ ayman on Flickr.