

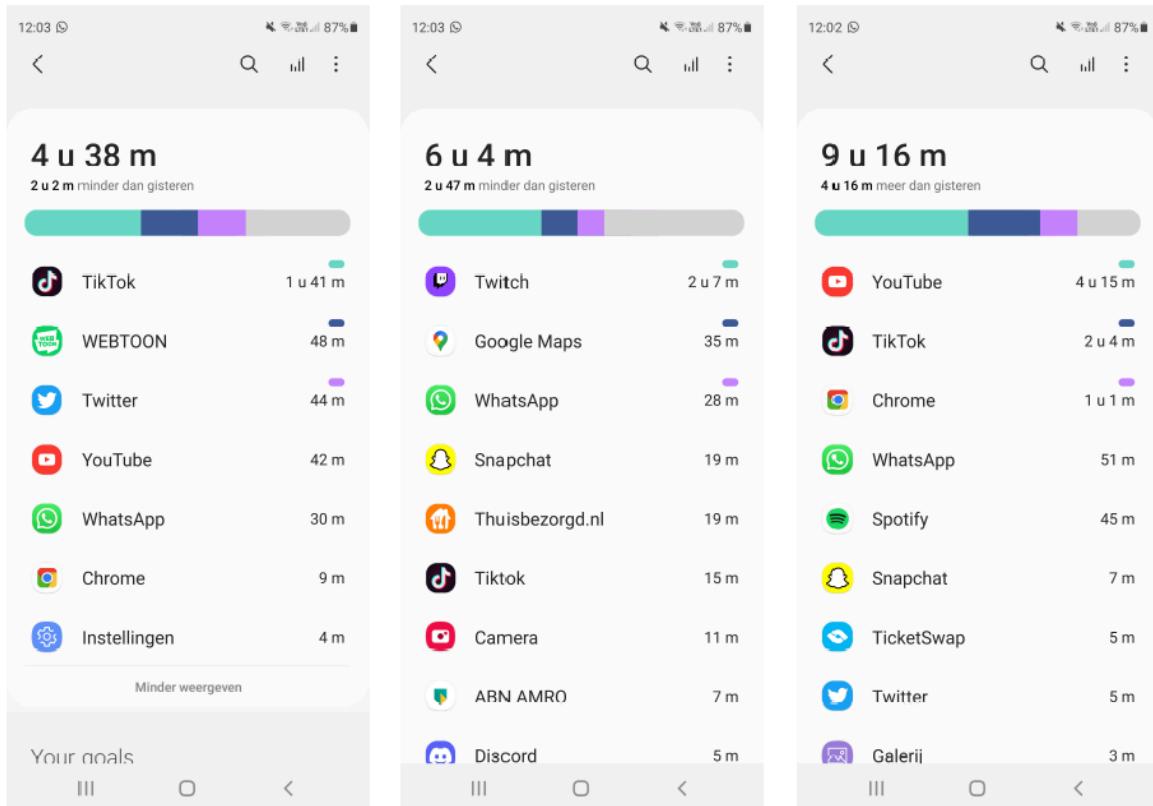
Anouk Vreeburg
Graduation Project DDD
22/08/2022

Technical document

The technical report "Improving cybercrime awareness within mobile applications with the help of interactive data visualisation" is complemented by this one. This technical documentation will assist in describing how the front end and back end of this prototype were created.

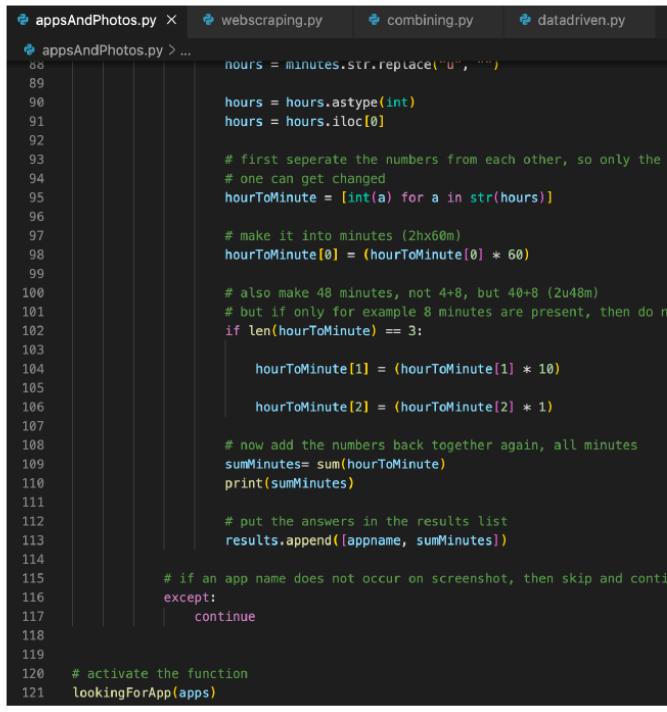
Data collection

Data collection had a rift from the start because there were two ways to battle the request. Understanding the applications a user uses is essential since it makes the experience more individualised and the user will be more engaged. The way of collecting data on screen time was at the top of the list from quite early on. Information on the applications and duration of use is provided via screen time. Even though it sounds perfect, it seems rather challenging to use this data. When it comes to their screen time, Apple and Samsung both have tight privacy policies, making it impossible to share. The idea for a second reply was to ask the user to name all of the applications they frequently use. However, that approach also looked unsatisfactory, since it would need a lot of time and might become much more annoying if many requests were required to keep up with frequent updates. In the end, a novel approach to managing screen time was developed.



You can get Pytesseract, also known as Python-tesseract, as a plugin for the Terminal Engine. This plugin can read and recognize text in photos, providing a solution to the issue. The user could easily snap a screenshot of their screen time and provide it inside the program with the aid of Pytesseract. This process requires less time and work. It's possible that the program might potentially take the screenshot on its own, with the user's consent.

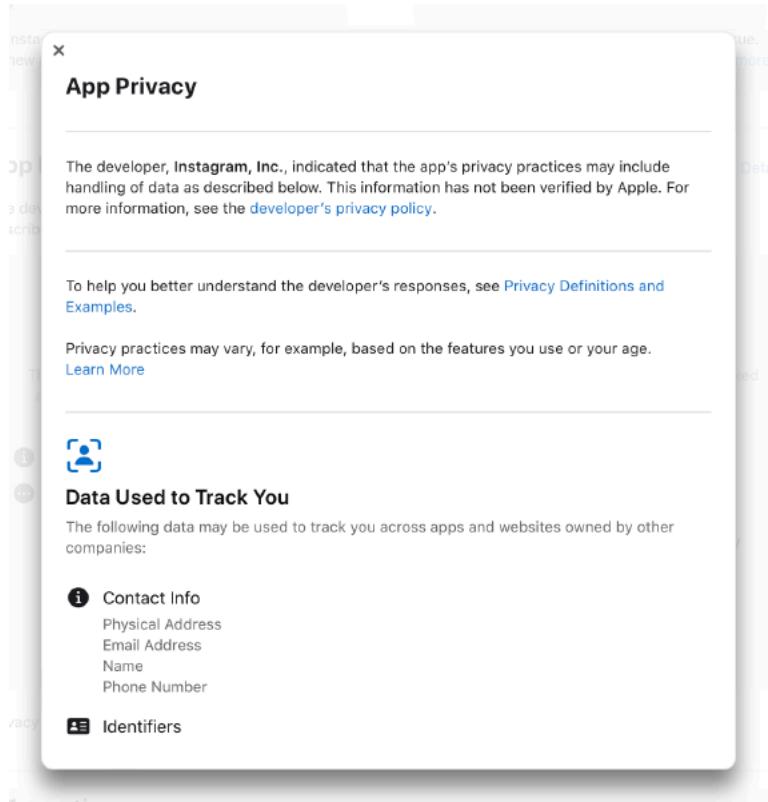
It was challenging to work with Pytesseract because there were several little nuances to consider. While initially inspecting a screenshot, the appropriate information had to be taken at that point. Even when fighting the code, more difficulties kept coming up. The operation was largely successful in the end. With the help of cleaning code, Pytesseract was able to interpret the majority of screenshots correctly. As shown in the picture below, several screenshots looked to waver unintentionally despite appearing to be in perfect condition. It would, however, not scan any words. Another minor glitch involved two-word application names, such as "Microsoft Teams," which were sadly also lost and were therefore unreadable. Ultimately, a clean csv file was able to advance to the following stage of the procedure.



App	Amount of time app used (minutes)
TikTok	523
Twitch	474
YouTube	268
WhatsApp	238
Discord	160
Chrome	114
WEBTOON	107
Zalando	56
Twitter	44
Snapchat	29
Notities	24
TicketSwap	22
Thuisbezorgd.nl	19
Camera	14
Facebook	8
Instagram	7
Spotify	6
Instellingen	4

Webscraping

It is ready for the next phase after obtaining the applications and the quantity of time. A privacy area may be found in every application on the App Store. The user is informed in this area of the application about the type of information that will be collected from them when they use it. Health and fitness to user content are some of these categories' variations. A larger data frame may be created if a list of the data that is gathered could be compiled for each program that the user utilises. Webscraping could be the answer because the App Store is accessible through a web browser. Webscraping is the practice of taking information from a website so that it can be used when coding.



This is where selenium came looking around the corner. In the Terminal of your computer or virtual space, you can download Selenium, much like Pytesseract. Selenium is a strong technology that can automate web browsers and programmatically manipulate them. It works with all browsers, is click-event capable, and has the ability to explore a website in depth. Data from the popup could be gathered using Selenium's assistance. Similar to standard web scraping, finding components by class or id was employed. Additionally, try and except were created since it was crucial to take into account that not all applications in the for loop would have the same categories.

```

AndPhotos.py    ✘ webscraping.py ✘ ✘ combining.py    ✘ datadriven.py
scraping.py > ✘ gettingAppPrivacy
# the main page is the window the driver currently works on
main_page = driver.current_window_handle ## CDwindow-0AB5095C0C91DCA70B4840

# click on the popup window
driver.find_element_by_xpath(xpath).click()

# let the program process it
sleep(5)

# switch to popup window, so the driver now focusses on new window instead
for handle in driver.window_handles:
    if handle != main_page:
        popup = handle
        driver.switch_to.window(popup)

# a empty list for the privacy titles to get into
list = []

# find all the elements that the app takes from the user, they are under the titles
titles = driver.find_elements_by_class_name('privacy-type__data-category-head')

# put every element in the list
for i in titles:
    answer = i.text
    list.append(answer)

```

Finally, it was crucial to summarise which categories were frequently requested and which were not. To obtain a detailed picture.

	index	App	SENSITIVE - Identification method	SENSITIVE - Usage data	SENSITIVE - Purchases	SENSITIVE - Location	SENSITIVE - Contact details
0	0	WhatsApp	1	1	1	1	1
1	1	TikTok	1	1	1	1	1
2	2	Chrome	1	1	0	1	1
3	3	YouTube	1	1	1	1	1
4	4	Spotify	1	1	1	1	1
5	5	Snapchat	1	1	1	1	1
6	6	TicketSwap	1	1	0	0	0
7	7	Zalando	1	1	1	0	1
8	8	Instagram	1	1	1	1	1
9	9	WEBTOON	1	1	1	0	1
10	10	Facebook	1	1	1	1	1
11	11	Discord	1	1	1	0	1
12	12	Twitch	1	1	1	0	1
13	13	Thuisbezorgd.nl	1	1	1	1	1

Finishing up

Both csv files had been finalised and were now ready to be combined into a single, tidy csv file. To get a clear picture of the most utilised programs by users, sort the numbers based on "Amount of time app used (minutes)". The type of data categories the application will accept are displayed next to the time.

App	Amount of time app used (minutes)	SENSITIVE - Identification method	SENSITIVE - Usage data	SENSITIVE - Purchases	SENSITIVE - Location	SENSITIVE - Contact details	SENSITIVE - User content	SE
TikTok	523.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Twitch	474.0	1.0	1.0	1.0	0.0	1.0	1.0	1.0
YouTube	268.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
WhatsApp	238.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Discord	160.0	1.0	1.0	1.0	0.0	1.0	1.0	1.0
Chrome	114.0	1.0	1.0	0.0	1.0	1.0	1.0	1.0
WEBTOON	107.0	1.0	1.0	1.0	0.0	1.0	1.0	1.0
Zalando	56.0	1.0	1.0	1.0	0.0	1.0	1.0	1.0
Twitter	44.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Snapchat	29.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Notities	24.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ticketswap	22.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0
Thuisbezorgd.nl	19.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Camera	14.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Facebook	8.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Instagram	7.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Spotify	6.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Instellingen	4.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
AllAppsTogether	2117.0	14.0	14.0	12.0	9.0	13.0	13.0	14

Now that the data has been formally obtained, it can be send to the data-driven feedback loop. Sadly, the prototype could not now be integrated since Unity, where it was built, demands a different kind of code. This means that the data-driven procedure can only be coded via examples.

Prototype

Along with the python code, the prototype was also developed and enhanced. The prototype was created using the gaming engine Unity, which allows for the programming of augmented reality. Unity is a new field of play because it leverages the C# programming language. Time was running out, and Unity can initially seem intimidating. So, a paper prototype was first created. The prototype was fairly straightforward but succeeded in making its point. The data points were created by making circles of various sizes out of paper. At this point in the project, the data points were applications rather than privacy categories. Points of application. The privacy categories that the application fell under were circled and written on the back of the paper. Initially, it was thought that by having the ability to individually scan through your applications, you could quickly identify any that were harmful and get rid of them. The first iteration round, though, appeared to change that.



Iteration 1

Each iteration round starts with a test plan in mind, consisting of motive, research design, questions and potential assignments for the participant.

Motive: Figuring out the feel of the prototype and its reactions

Research design: Paper prototype

Questions: How do you feel about the concept? What do you think of the data points? Would you be interested to learn more? ...

Assignment: No assignments

The paper prototype was tested by 2 participants from the target group, found in Almere. Because it was just a paper prototype, interactions and testing were not very successful. But there was enough of talk to be had, discussions about the potential of the prototype, and how to make the prototype more entertaining and improved. Important questions were also raised. At the moment the goal could not be passed down to the participants, as they did not understand what was that needed to be shown. What do the users need to become aware of? Why do I use the applications as data points, users can already see them floating around on their home screen when opening their phone. It was a thing to think about.

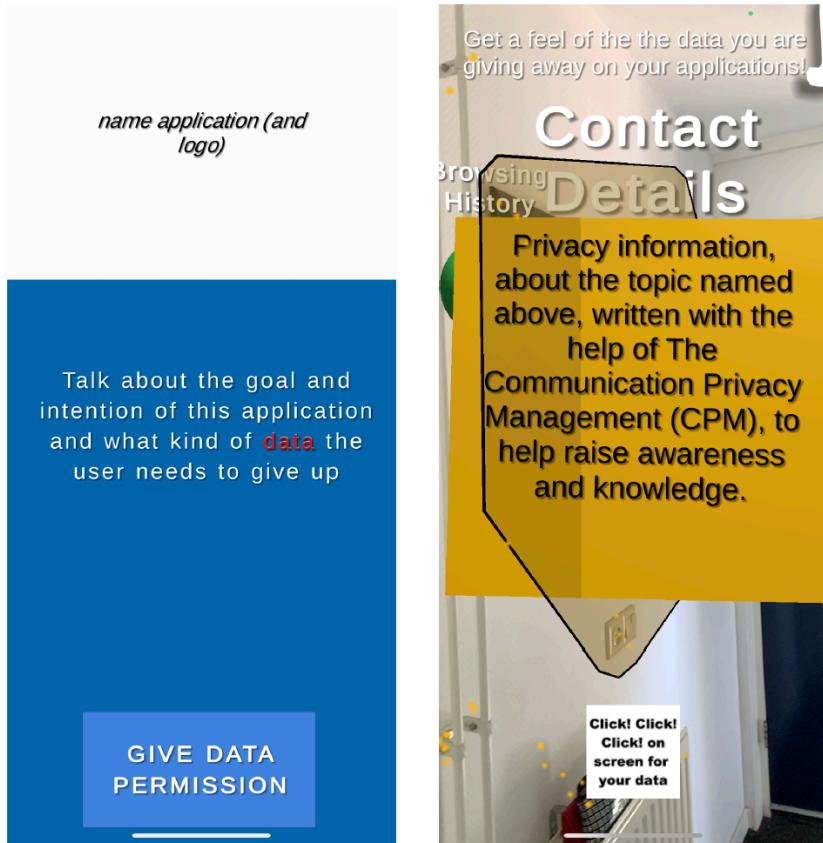
Elements to take with me: (copied from notes taking that day):

- Better to show privacy than application, cause what can you really say about the application other than that
- Really need to start building in unity
- Impressed with the idea
- Make it intractable
- Important questions were raised, what do I want to show? What do I want to user to learn, become aware of? Of the applications that they have, they can already see them float around when opening their phone
- They would be interested

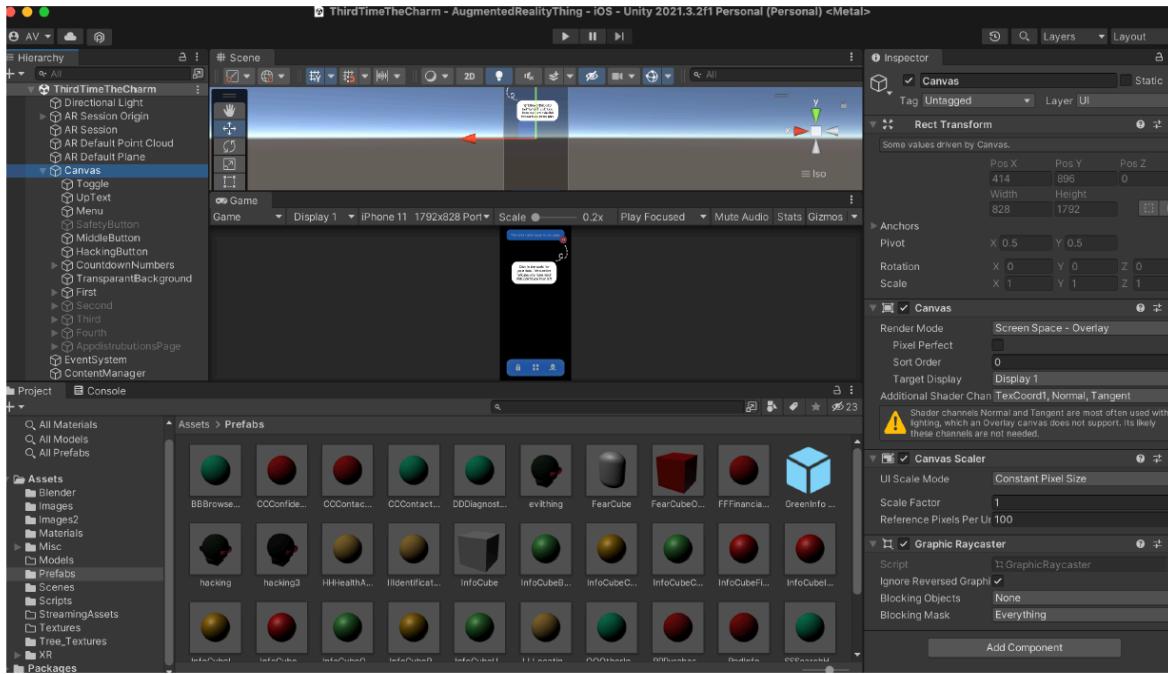
Upgrading

After the initial iteration was successful, it was time to go deep into the work in Unity. Massive advancements had been made. Now, data points may fly all over the place. The user may send "data breaches" toward a chosen data point via a straightforward menu, and upon

touching it, a popup warning of danger would appear. Additionally, data points could now be clicked, which would freeze time and bring the point right up to the user's face, overwhelming them with data.



Although Unity is a challenging program to master, once you get the hang of it, it can also be very rewarding. To describe the objects in the picture, a hierarchy is present on the top left. Scene represents a "level," or an area in which to work. Then, scripts are added to these objects. The magic of coding happens in scripts. After coding, you can click on an object in the hierarchy to access its right-side inspector. All of the object's characteristics can be found in the inspector screen. The created script is one of the components that can be included. When attached, the object complies with the script's coding specifications.



The data point object has the "InfoCube" script attached to it. This script aids in the object's global shooting. Always face the camera when moving and rotating. The scripts instruct the data point to take action when the object is clicked. This and more is available.

```

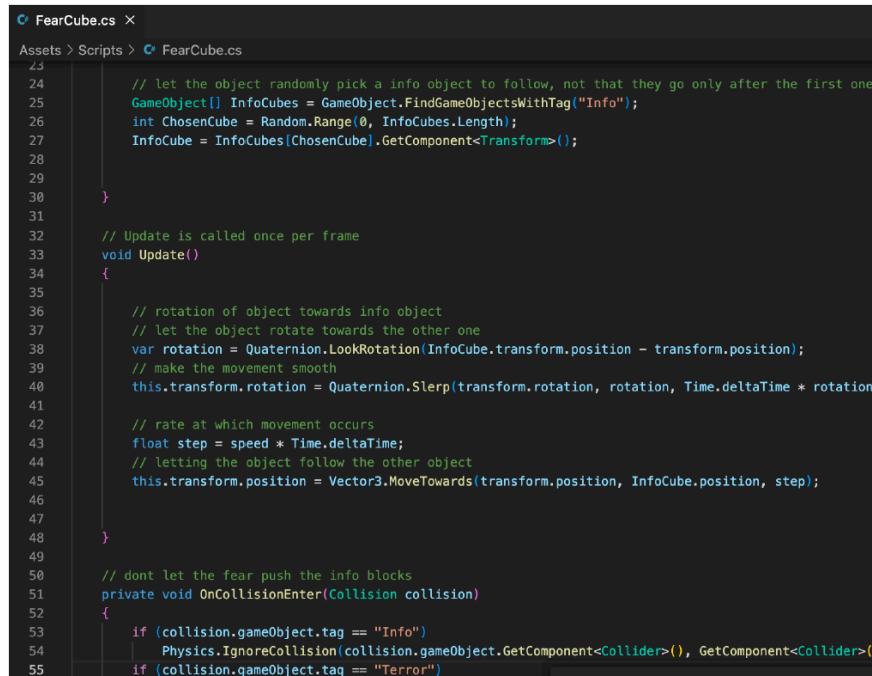
InfoCube.cs

Assets > Scripts > InfoCube.cs

54
55 // Update is called once per frame
56 void Update()
57 {
58
59     // rotation of object towards AR camera
60     // let the object rotate towards the phone
61     var rotation = Quaternion.LookRotation(ARCamera.transform.position - transform.position);
62     // make the movement smooth
63     this.transform.rotation = Quaternion.Slerp(transform.rotation, rotation, Time.deltaTime * rotationSpeed);
64
65     // make the object float around
66     float newX = Mathf.Sin(Time.time * speed);
67
68     // make each info object look towards the camera at all the time
69     GameObject.FindGameObjectWithTag("Info").transform.LookAt(ARCamera.transform);
70
71     // letting the object fly
72     // shoot where user points, some go left, some go right, it is random
73     if(CardIndex == 0)
74     {
75         this.transform.position = transform.position + Vector3.right * speed * Time.deltaTime;
76     }
77
78     if(CardIndex == 1)
79     {
80         this.transform.position = transform.position + Vector3.left * speed * Time.deltaTime;
81     }
82
83
84     // THE OTHER ONE, WHERE THEY ALL WENT IN STRAIGHT LINE THO
85     // transform.position = new Vector3(newX, transform.position.y, transform.position.z);
}

```

Similar characteristics exist in the "FearCube". A random data point is chosen by the script for the data breach to follow, and when touched, events are to occur.



The screenshot shows the Unity Editor's code editor with the file 'FearCube.cs' open. The code is written in C# and defines a script for a game object named 'FearCube'. It includes logic for picking a random info cube from a list, updating its position towards the selected cube, and handling collisions with other objects.

```
23 // let the object randomly pick a info object to follow, not that they go only after the first one
24 GameObject[] InfoCubes = GameObject.FindGameObjectsWithTag("Info");
25 int ChosenCube = Random.Range(0, InfoCubes.Length);
26 InfoCube = InfoCubes[ChosenCube].GetComponent<Transform>();
27
28 }
29
30 }
31
32 // Update is called once per frame
33 void Update()
34 {
35
36     // rotation of object towards info object
37     // let the object rotate towards the other one
38     var rotation = Quaternion.LookRotation(InfoCube.transform.position - transform.position);
39     // make the movement smooth
40     this.transform.rotation = Quaternion.Slerp(transform.rotation, rotation, Time.deltaTime * rotation);
41
42     // rate at which movement occurs
43     float step = speed * Time.deltaTime;
44     // letting the object follow the other object
45     this.transform.position = Vector3.MoveTowards(transform.position, InfoCube.position, step);
46
47 }
48
49
50 // dont let the fear push the info blocks
51 private void OnCollisionEnter(Collision collision)
52 {
53     if (collision.gameObject.tag == "Info")
54         Physics.IgnoreCollision(collision.gameObject.GetComponent<Collider>(), GetComponent<Collider>());
55     if (collision.gameObject.tag == "Terror")
```

It was time for another iteration round to be able to proceed after adjusting to Unity and being able to implement various objects and scripts.

Iteration 2

Motive: Figuring out the effectiveness of the new functions and features, and the element of augmented reality

Research design: Prototype on iPhone

Questions: Do you understand what is happening? Are you able to interact with it? How do you feel about the design? What do you think of Augmented Reality? Do you have any problems with it? Is the goal of the application clear for you? Do you think this raises your awareness about privacy loss? ...

Assignment: No assignments

Three individuals tried the second iteration phase, the two participants from iteration 1 returning as well as a brand-new participant who belongs to the target group. Two of the three

were evaluated in person, while the third was done so via Microsoft Teams. The iteration rounds were quite successful because each participant could offer a variety of feedback. Positive comments include that it is still too confusing, while others raise stern queries as, "What if the user is colourblind? He is therefore unable to perceive the colours red, orange, and green.



Elements to take with me (copied from notes taking that day):

- Needs more explanation, all the buttons, confusing what to do
- Careful that the balls don't go too far when surround by walls
- Clicking on info's is fun that it comes to you, but difficult to comprehend sometimes what is said. Also, a block is not enough to show all the info you want to show
- Not clear what the fear button represents, it attacks and then?
- Like the movement of the blobs
- I see the overall idea is there, but it is not a lot of the moment

- When clicking to find info, the blocks are up front again, which is nice, but when the touch other all hell breaks loose
- What if users are colourblind?
- Upgrade design
- Right icon is confusing, too small to clearly see

Survey

People can perceive patterns and colours differently from others, according to research (Zentner, Covid, & Guevarra, 2019). Researchers might interpret the colour green as denoting infection while teachers might interpret it as indicating correctness. The prototype was now actually taking shape, and the data points needed to be redone. A brief survey was sent out to acquire information about how people perceive images in order to ensure that the colours and patterns are appropriate for the target audience.

The survey was placed online and received a total of 39 answers. In the survey, there was asked for only people around the age of 16-24. If only people in that age range answered is not completely clear, as an age question was missing, due to forgetfulness.

Questionnaire:

What colour do you recognise danger with?

- 29 people answered red

What colour do you recognise warning with?

- 15 people answered with yellow
- 12 people answered with orange

What colour do you recognise at peace with?

- 25 people answered with green

- 13 people answered with blue

When thinking of data, what image/icon pops in your head?

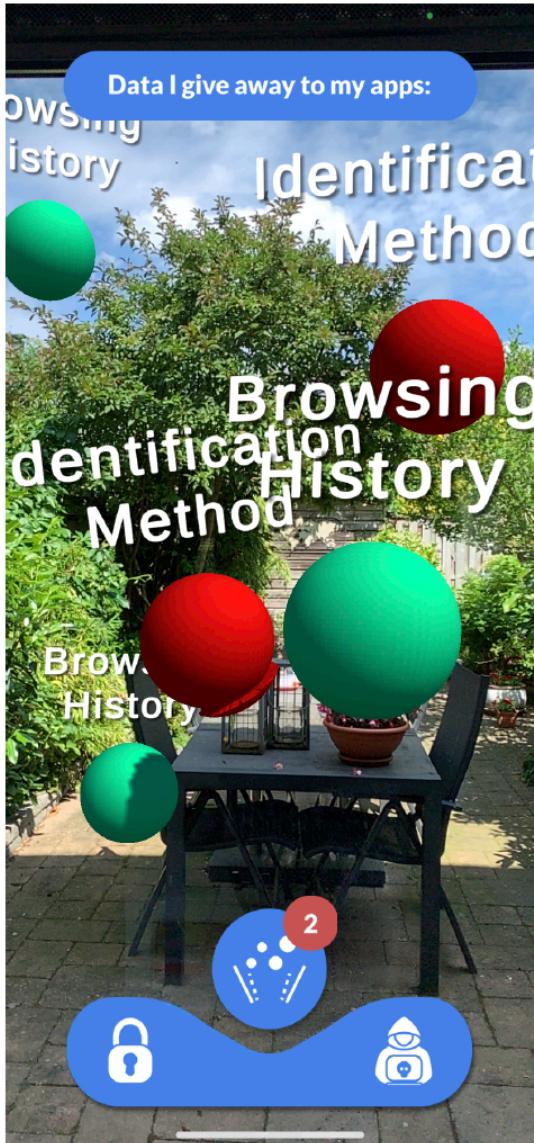
- 16 people answered graphs
- 10 people answered a circle
- 6 people answered zeros and ones

When thinking of cyberattack, what image/icon pops in your head?

- 27 people answered with hacker

Upgrading

The following construction was taking place after more knowledge was gained from the previous iteration and the survey. This time, more attention was paid to clarifying the application, fixing issues, modernizing the look, and upgrading the text within. The entire screen was suddenly covered in information regarding the data points, as opposed to just the block. To request data permission and to introduce the application, an introduction screen was made. The final step was to choose the colors and patterns that will be used depending on the survey findings.



Location

Instagram Facebook

Privacy rules (what is Location data, what are the rules)
“any data processed in an electronic communications network or by an electronic communications service indicating the geographical position of the terminal equipment of a user of a public electronic communications service”

- the data is anonymous; or
- you have the user’s consent to use it for a value-added service, and the processing is necessary for that purpose.

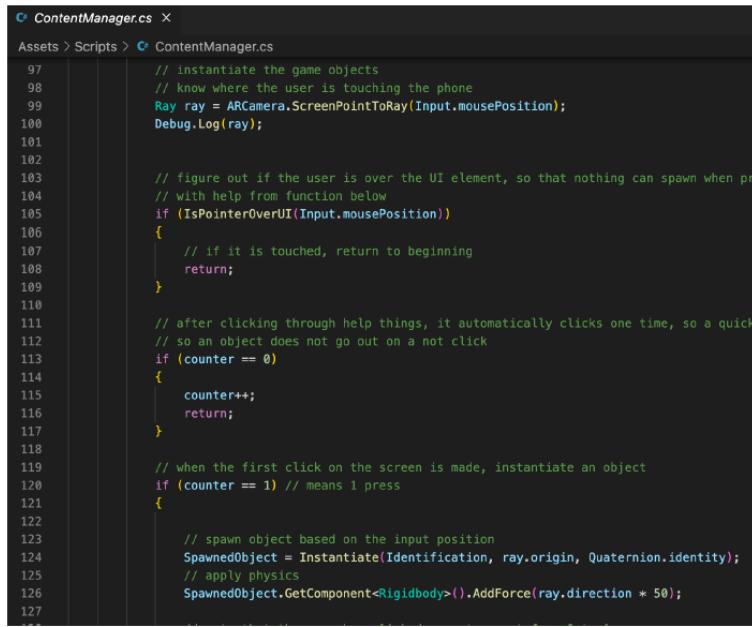
Privacy collectives (where does it go, who has access)

- Mobile phone carriers
- Operating systems
- Apps and App Partners
- Location Analytics Providers

Privacy turbulence (problems that can occur)

Multiple problems can occur when

The largest script in the Unity build is called "ContentManager." The script is able to handle all of the screen interactions and spawn the data points appropriately. By using a few simple coding rules, it can activate and deactivate objects in the hierarchy.



```

ContentManager.cs
Assets > Scripts > ContentManager.cs
97 // instantiate the game objects
98 // know where the user is touching the phone
99 Ray ray = ARCamera.ScreenPointToRay(Input.mousePosition);
100 Debug.Log(ray);
101
102
103 // figure out if the user is over the UI element, so that nothing can spawn when pr
104 // with help from function below
105 if (IsPointerOverUI(Input.mousePosition))
106 {
107     // if it is touched, return to beginning
108     return;
109 }
110
111 // after clicking through help things, it automatically clicks one time, so a quick
112 // so an object does not go out on a not click
113 if (counter == 0)
114 {
115     counter++;
116     return;
117 }
118
119 // when the first click on the screen is made, instantiate an object
120 if (counter == 1) // means 1 press
121 {
122
123     // spawn object based on the input position
124     SpawnerObject = Instantiate(Identification, ray.origin, Quaternion.identity);
125     // apply physics
126     SpawnerObject.GetComponent<Rigidbody>().AddForce(ray.direction * 50);
127

```

A subsequent iteration was planned after corrections.

Iteration 3

Motive: Figuring out the effectiveness of the new functions and features, and see the effect of the upgrades

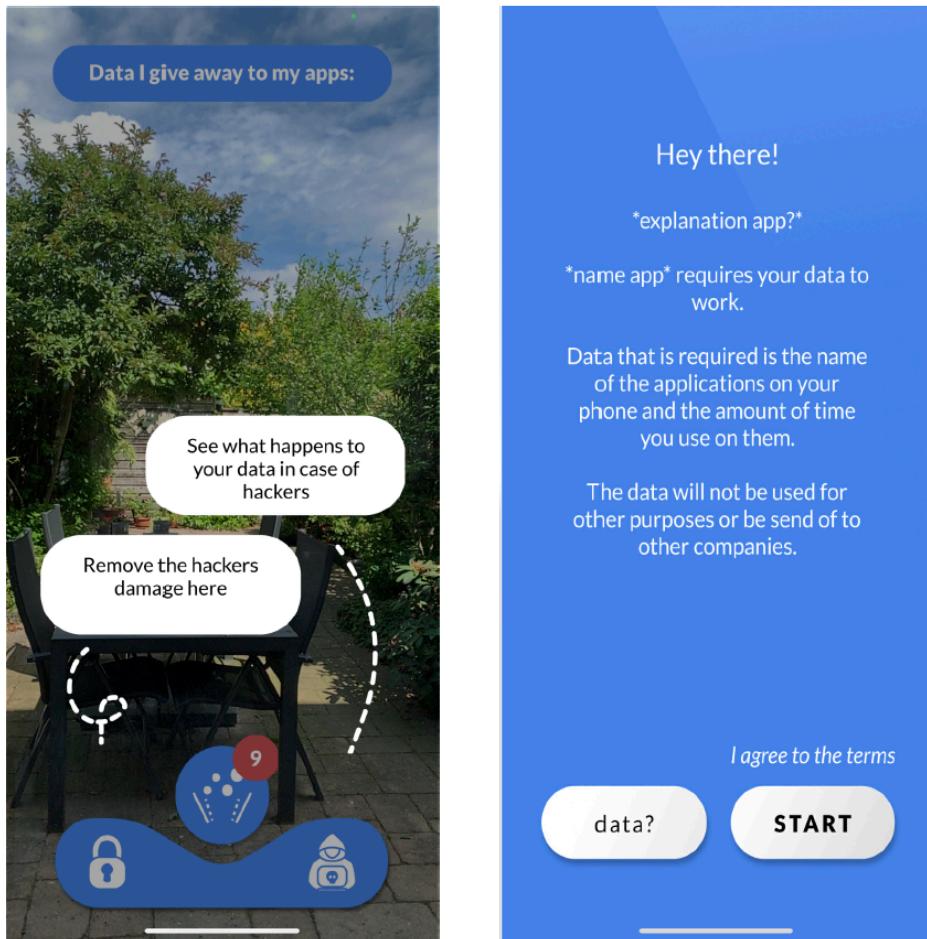
Research design: Prototype on iPhone

Questions: What do you think about the look and feel? How do you feel about the new intro screen? Was everything clear? What kind of information would you like to see more? What would you improve when looking at the screen? Is the menu clear as to what can happen? How do you feel about the reward and punishment system? Is your awareness higher than before opening the application? How could it become even higher? Would you change your behaviour based on the application?

Assignment: Can you go towards the data point page? Can you break a data point? Can you remove the data breaches? Can you start over?

The same individuals from iteration 2 participated in iteration 3. The same as last time, two of the tests could be conducted in person, with the third taking place via Microsoft Teams. This time, opinions were much more in sync, and constructive criticism was offered. Even though

they had higher expectations, participants were beginning to like the rewards and punishments aspect more. The introduction screens made a significant impact by letting participants know what to do and what to expect. There were also still some inaccuracies because each participation encountered this issue with the data points' interactions with one another.



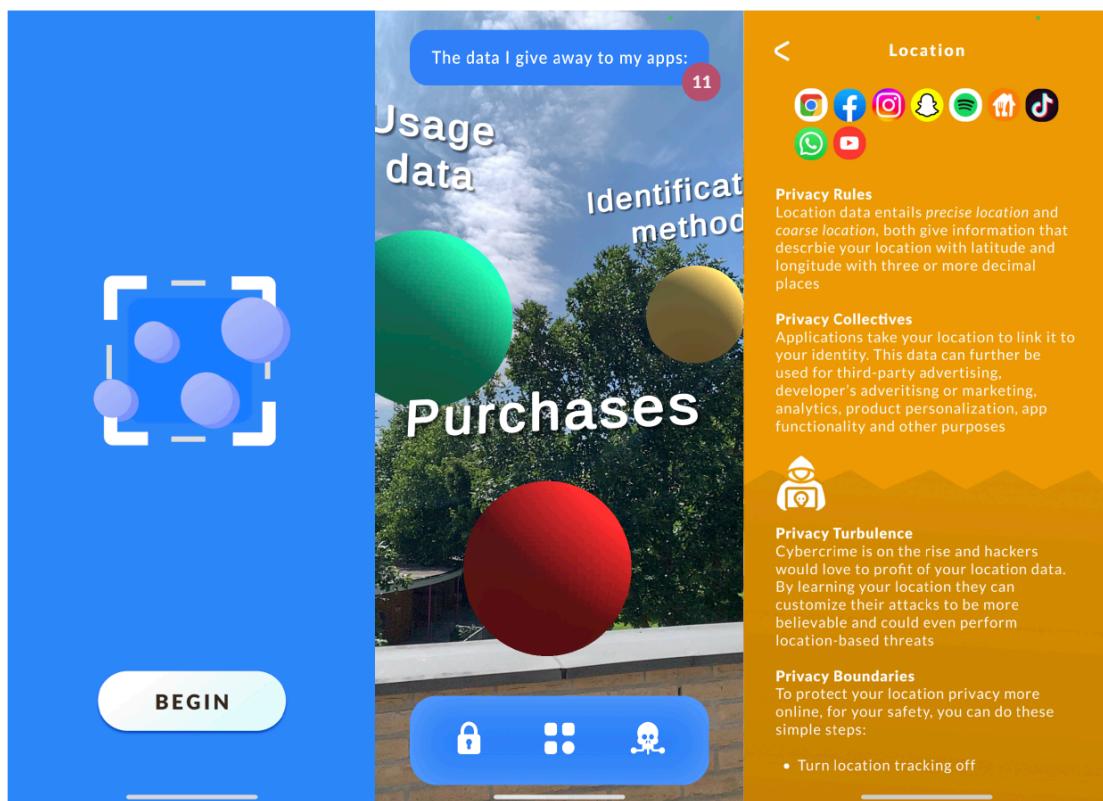
Elements to take with me (copied from notes taking that day):

- Drive the danger/terror parts more home, more emphasise, it feels kind of lame now. I see the visuals but don't understand what is happening, perhaps text for clarification
- Intro screen does not really tell what is happening yet, might be confusing coming into the app
- Really like how they all float around, does go wrong when they touch
- Info page really good, little focus on the cybercrime, what you wanted.
- I also like the new design

- New button information also fun fun
- I like playing with the rewards and punishment, seeing how it changes every time, it is just fun to way the skull go to the random places
- The skull, does feel little out of place
- I like the explanations, really helps with understanding what I can do
- Perhaps have different types of cyber attacks? And healing?

Upgrading

The third version had been completed for seven weeks at this point. The updated version has taken into account all of the feedback and has been tailored to the preferences of the target group. The user can now access a new button and screen that provides details about the distributions of their programs. It also has a fresh, updated look. Data points can be readily dispersed over the planet without going too far or wide. They can be engaged with and move about on their own. A new and updated screen that appears when you click provides details about the data category.



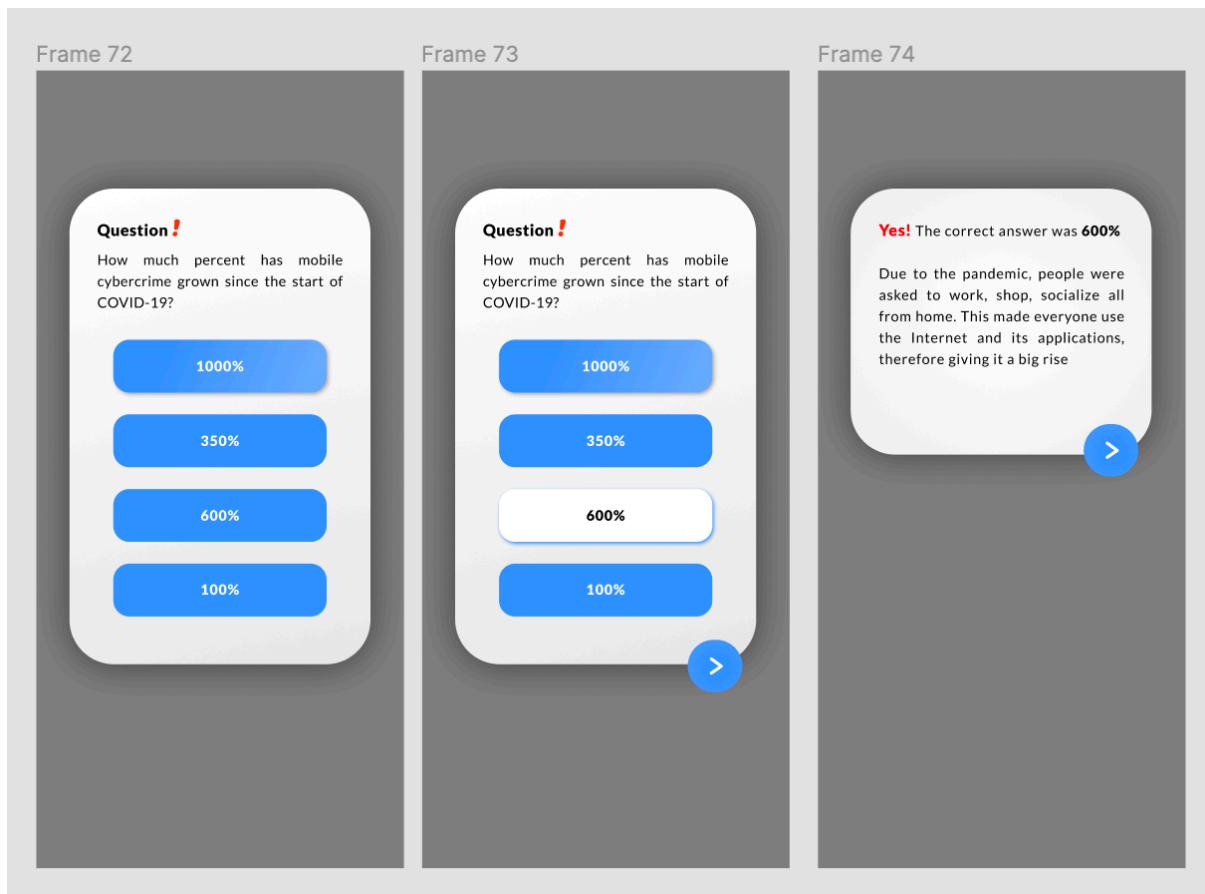
Time-out

The prototype was not quite there, but it was getting there. A brief break was taken to perhaps gain some fresh insights, and then the digging into the subject begane—most crucially, the target group. By doing this, one can hope to offer additional features and something more distinctive, as well as to personalise the design even more for the user. In order to provide users more options than merely watching and reading, which can get quite monotonous very quickly, new functionalities are required.

Upgrading

New avenues were found to spread awareness and information regarding mobile application cybercrime across the target group. First and foremost, the application struggled because it attempted to accomplish several things at once. It attempted to raise users' awareness while simultaneously attempting to change their behaviour. The user found this to be too much and overwhelming, thus the idea has been abandoned. The prototype will initially just concentrate on spreading awareness about mobile applications and cybercrimes.

More attention was paid to the target group's behaviour and awareness process when the research was adjusted to account for it. It is now crucial to work with their behaviour and influence it to our advantage because changing their behaviour is no longer the main focus. The awareness process and application design also need greater research to support them.



The addition of questions when being cyber attacked was one of the initial upgrades. The data point will maintain its shape in the event that a question is appropriately answered. Incorrect responses will cause it to begin to progressively break and it will receive a small pop-up by its name, reminding the user that one mistake is all it can take. These inquiries will be favourably received by the users because they are lively and can provide some form of satisfaction. It also serves as a fresh source of knowledge and understanding.

Besides adding this new feature, the design also got an upgrade.

Iteration 4

Motive: Seeing how the new upgrade will be welcomed

Research design: Prototype on iPhone

Questions: What do you think about the added questions? Does it help raise your awareness? Do you find the questions fun? How do you feel about the effect when answering wrong? Or when answering right? ...

Assignment: No assignments

The same three people who were present for the prior iteration's testing also participated in this one. All three offline. Everyone was delighted with the additional interactive component, expressing that it gave the program a completely new look. Although this was encouraging to hear, turmoil quickly broke out. This caused a lot of questions to surface at once, which was overwhelming, this became because a participant hit the cyberattack button repeatedly in a succession. The new function was nevertheless welcomed and thought to be more instructive than reading from a data point.

After the testing session some brainstorming sessions were also had, which both led to my new notes.

Elements to take with me (copied from notes taking that day):

- There is no effect when given a right answer, only wrong, that needs to change, as impulsive people need that gratification
- Lessen the questions! More personal questions? Perhaps about their screen time?
- Implement more of the theories that I have been reading
- Remove self clicking when cybercrime attack, maybe have cyberattacks randomly spawn, like in the real world
- More space is needed for the data point information, it also needs to be made more fun to read
- More looking at rewards & punishments, other gamification elements?
- Make the change from showing personal data to showing mobile applications, as the focus of the project has changed
- Replace some of the icons, as they were not understood directly

- Update the intro screen with explanations
- Perhaps make data points more fun to look at, more intractable
- Adding more gamification? Collecting of coins, or something else?
- Focus more on the fear part! That is still important
- Don't forget the data-driven aspect!
- Make it into a simulation? Every few seconds a cyberattack occurs?
- Try to code the data-driven feedback loop, some video's about it online

Upgrade

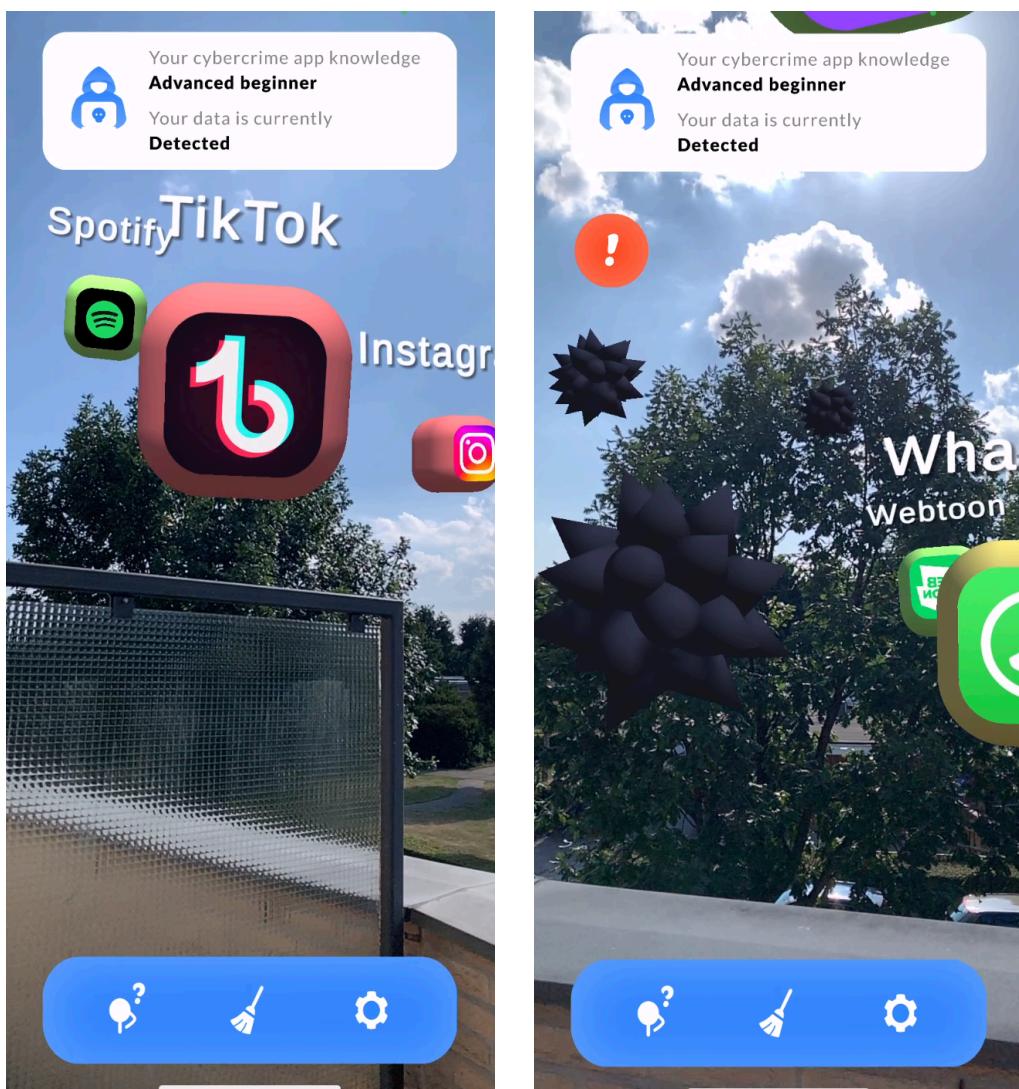
One can no longer click on their own to launch a cyberattack on the data points in order to incite panic further. Now it will automatically sound every 20 or so seconds. It might succeed in accessing a user data point, but it might also fail. As a result, the user has lost control and must now wait to see if it will actually hit. Additionally, this is more in line with how actual cyberattacks operate, thereby instilling fear.

A large red button that screams for the user's attention to respond to a question will emerge on the screen when the user is attacked. One question can only be activated at a time, though, this time. The questions themselves still relate to mobile application cybercrime, however, it is on the writing block due to ethical reasons, but perhaps the questions could also be made more personal.

When the data points were switched from personally identifiable information to applications, a significant difference was made. This modification resulted from mobile applications' transformation into the app's main objective. When this was brought up with a participant in a social setting, they provided some insightful comment, saying that selecting one's preferred application is more enjoyable than selecting a random data term.

Another key aspect was that in iteration 4, results only appeared when a triggered question was answered incorrectly; if the question had been correctly answered, the data point would

not have changed. A new component that emphasises rewards and punishments more will therefore be tried. The user will begin with the level of beginner when they launch the application. They can advance from beginner to expert by correctly answering questions, or vice versa. This enjoyable component can encourage learning and enhance the whole experience.



The final iteration can be read about in the academic paper.

Hey there!

DataSee is an easy to use and fun educational augmented reality app. Your most popular apps are listed in DataSee, along with the general data they would collect about you.

Did you know that cybercriminals would love to get their hands on that kind of online personal data?

Understanding cybercriminals and the things they do is crucial. DataSee aims to make sure you are aware of how exposed you can be due to your apps. DataSee needs some of your data in order to aid you in seeing. To make the experience as individualized and helpful as possible, it needs to collect your screen time statistics every day.

The information will only be used to display to you. It won't be given to other businesses to utilize.

screen
time?

START

Screentime

Your daily screen time can be tracked by phones. Screentime provides details on which applications were utilized that day and for how long.

 Discord 2 h 35 m

 Twitch 1 h 9 m

 TikTok 58 m

DataSee needs this type of information since it can identify the mobile apps that are most at risk. DataSee aims to provide you with a highly individualized perspective of how cybercriminals may affect YOU.

To record your daily screen time, DataSee only needs your consent once. Anytime you want, you can stop this action.



