

- Begrüßung
- Hier schon eine Neuigkeit: Wir haben ein neues Logo erarbeitet

TidyShinyFireworks



Kevin Repke
Paul Johne

Audit 3

- heute geplant Vorstellung der Konzeptzeichnungen, wie wir uns das Produkt vorstellen würden
- nochmal das PoC “Download and Visualisation”, da wir daran weiter gearbeitet haben
- sowie den Vertical Rapid Prototype
- zuletzt die Deliverables für den letzten Audit

TidyShinyFireworks

Inhaltsverzeichnis

- vorläufiger Projektplan
- Konzeptzeichnungen
- PoC “Download and Visualisation”
- Vertical Rapid Prototype
- Deliverables für Audit 4



Einleitung



Konzept-
zeichnung



Prototype

4

Audit 4

- in den letzten Wochen haben wir folgende Dinge geplant und umgesetzt
 - mit schon vorgestellten PoCs “Download and Visualisation” und Firework Visualisation” haben wir noch weiter gearbeitet und Anpassungen vorgenommen oder herumprobiert
 - den Vertical Rapid Prototype
 - ein neues Logo wurde erstellt
 - die Konzeptzeichnungen wurden erstellt
 - diese Präsentation wurde vorbereitet

Einleitung

Konzept-
zeichnung

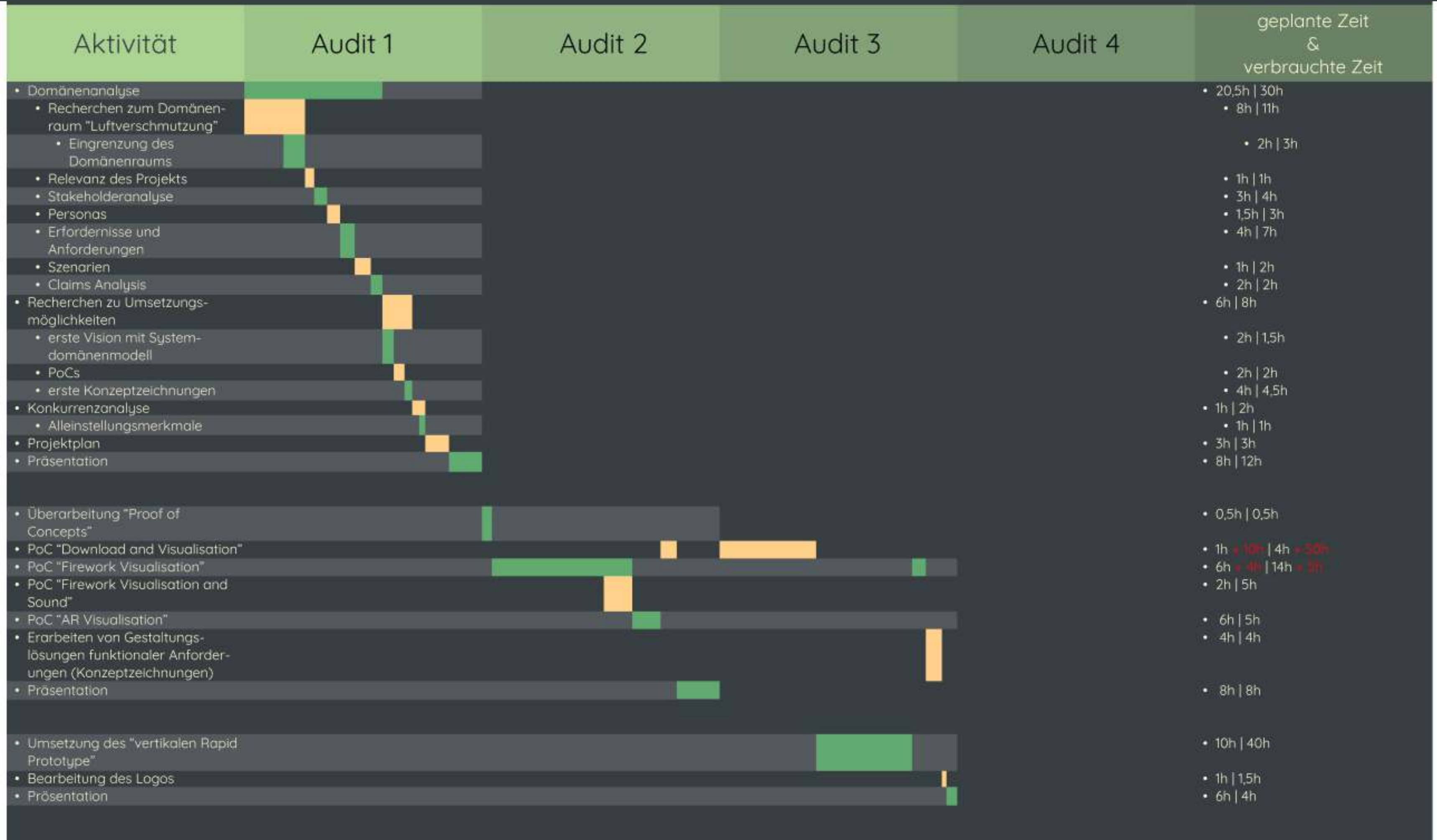
Prototype

4

Audit 4

TidyShinyFireworks

Projektplan



- grobe Vorstellung des Konzepts anhand eines User Journeys mit Screens
 - 1. Als erstes der Home Bildschirm, von dem wir in die Kameraszene wechseln. Die restlichen Buttons wurden schon bedacht, doch konkret noch nicht mit Inhalt gefüllt
 - 2. So sieht der Screen nach Start der Kamera aus. Dem Nutzer ist es möglich zurück ins Menü zu gelangen, die Lampe zu nutzen und einen QR Code zu scannen
 - 3. Hier scannt der Nutzer einen Code, um das Feuerwerksmodell downzuloaden
 - 4. Nach dem Download kann der Nutzer das Modell platzieren. Als Hilfe kommt eine Art Plattform, die die mögliche Position des Modells anzeigt

Konzeptzeichnungen als Screens 1/2



Einleitung



Konzept-
zeichnung

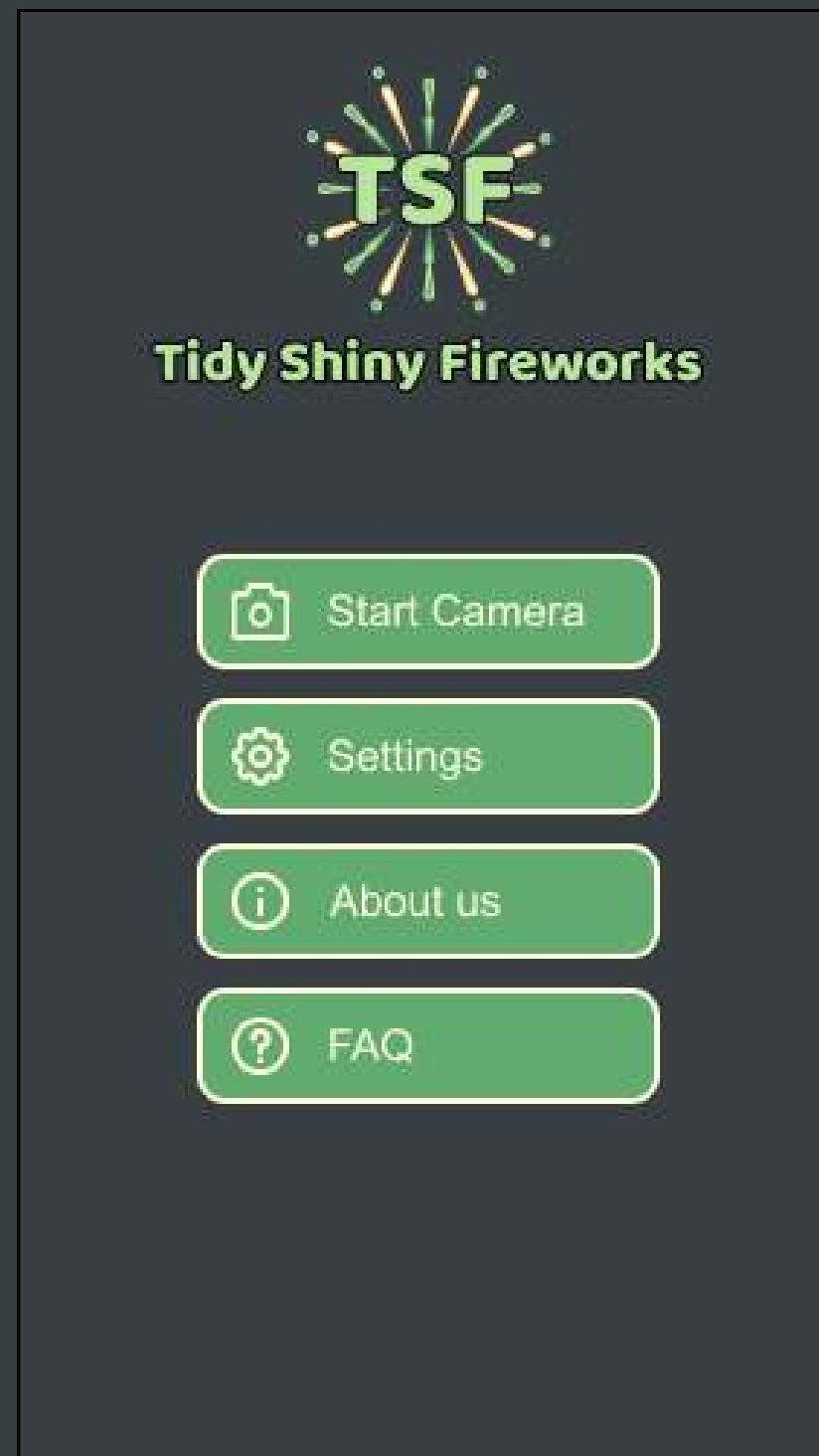


Prototype

4

Audit 4

S. 3



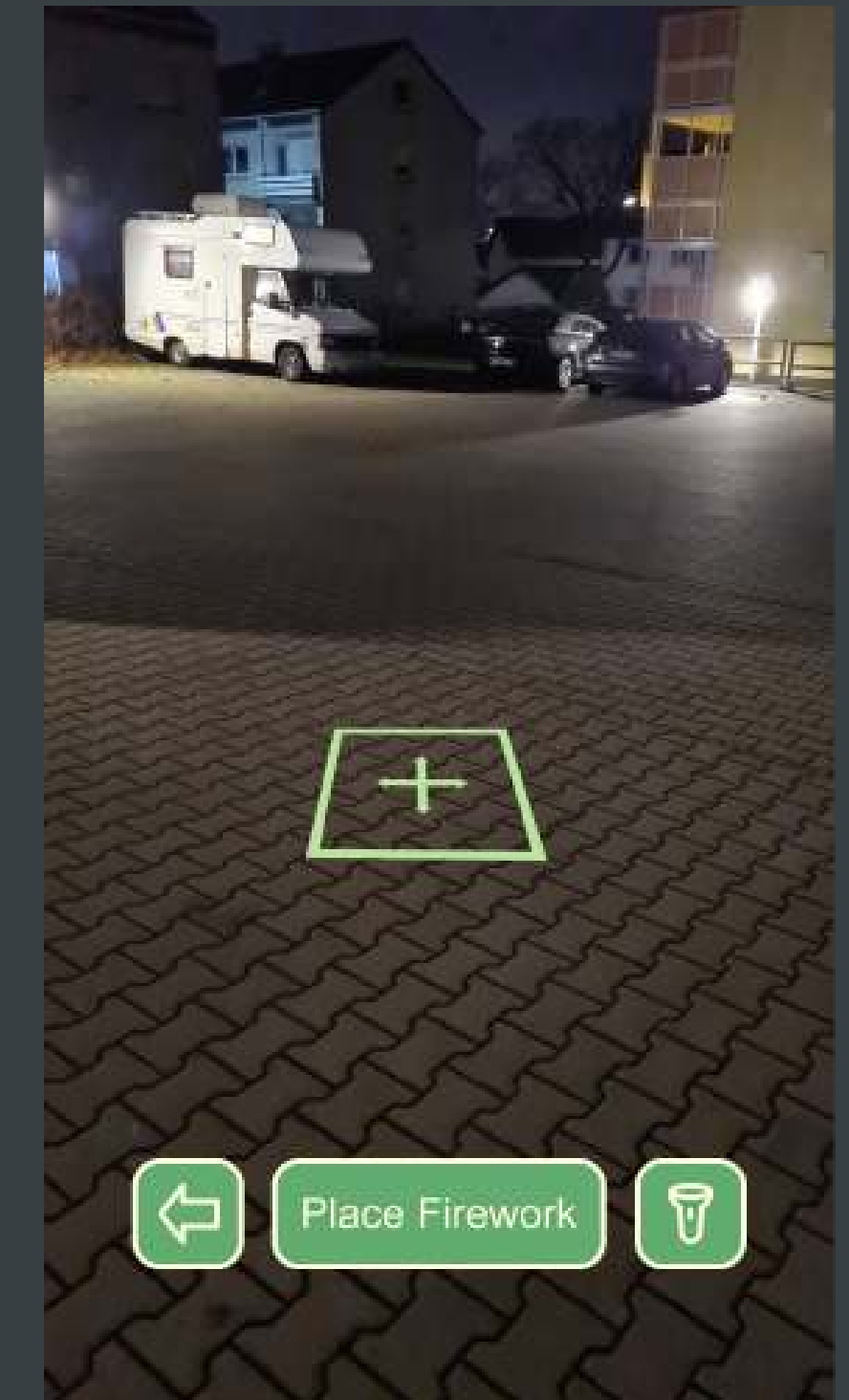
1. Home



2. First View
nach Start



3. QR Code
scannen



4. Platzieren
des Modells

- 5. So würde es aussehen nach dem Platzieren
- 6. Der Button zum Zünden kommt zum Vorschein
- 7. Skizzenhaft würde das Feuerwerk dann so aussehen. Hier ist es eher gephotoshopt

Konzeptzeichnungen als Screens 2/2



Einleitung



Konzept-
zeichnung



Prototype

4

Audit 4

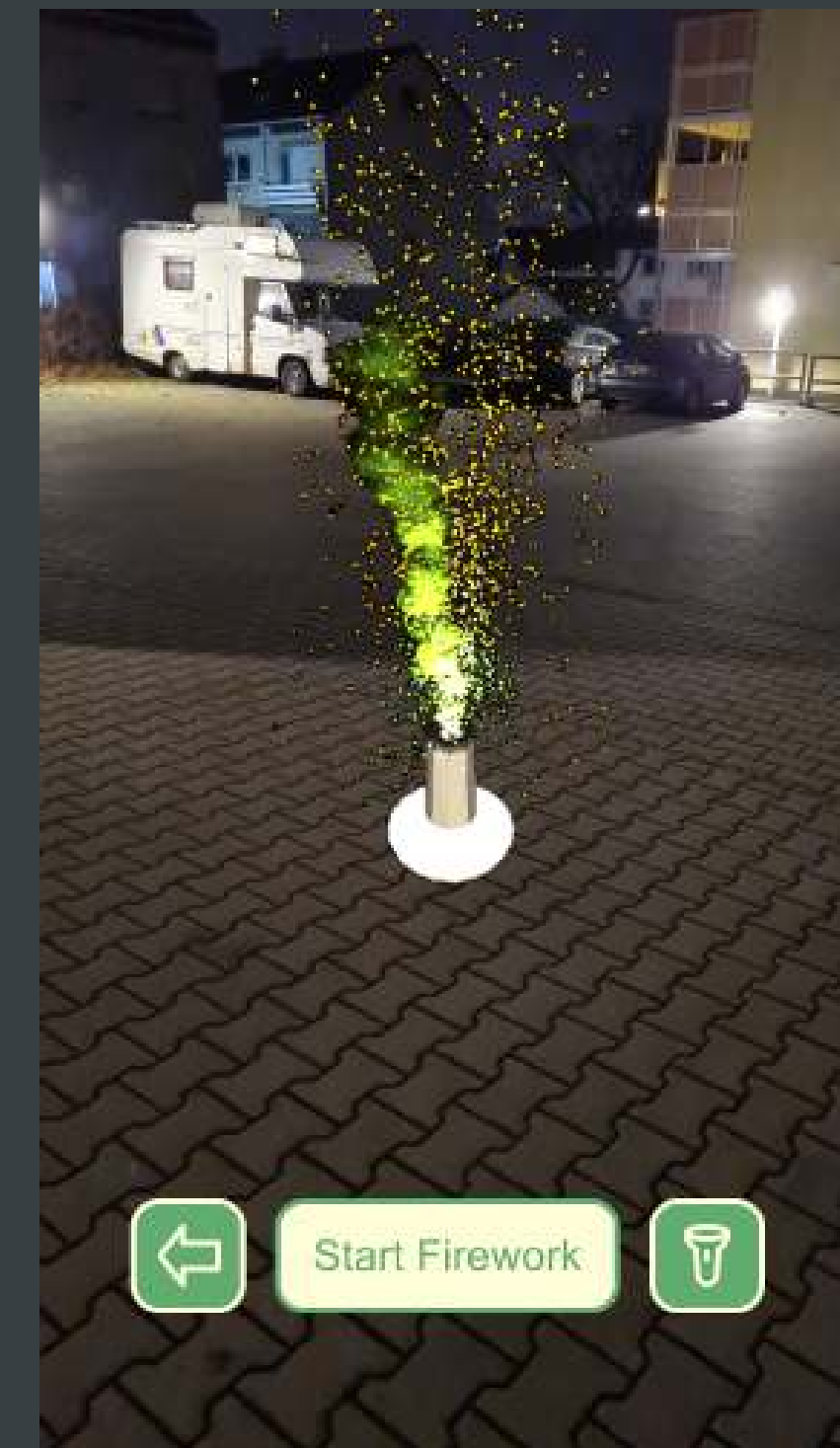
S. 4



5. Platzieren
erfolgreich



6. Zündung
des
Feuerwerks



7. Zünden
erfolgreich

- Rapid Prototype DownloadAndVisualisation nachgeholt
- genutzt, um QR Reading mittel ZXing Library & AR mittel AR Foundation umzusetzen
- aufgetretendes Probleme:
 - Skript zur Speicherung und dessen Abrufs auf Android Plattform nicht 1:1 nutzbar

PoC Download and Visualisation

Download und Darstellung .obj Datei



Einleitung



Konzept-
zeichnung

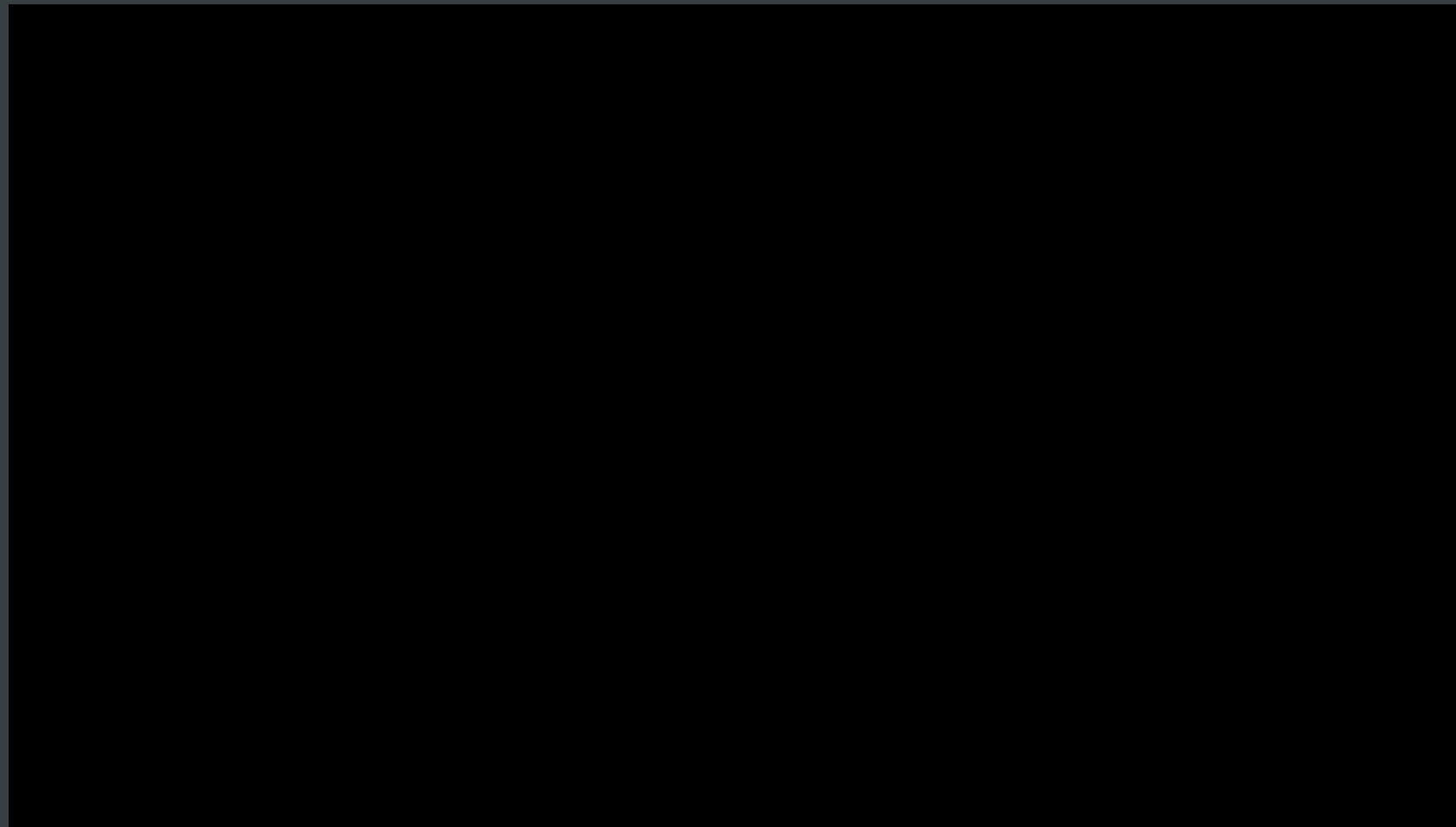


Prototype

4

Audit 4

S. 5



```
filePath = $"{Application.persistentDataPath}/3DModels/";
```

```
private async Task DownloadData(QR_Data data) {  
    debug.text = "Downloading 3D-Model..";  
  
    // creating HTTP GET requests  
    UnityWebRequest requestOBJ = UnityWebRequest.Get($"{rawDriveURL}{data.objID}");  
    requestOBJ.downloadHandler = new DownloadHandlerFile($"{filePath}{data.name}.obj");  
    UnityWebRequest requestMTL = UnityWebRequest.Get($"{rawDriveURL}{data.mtlID}");  
    requestMTL.downloadHandler = new DownloadHandlerFile($"{filePath}{data.name}.mtl");  
  
    var tasks = new Task<UnityWebRequest>[2];  
  
    // receiving data..  
    tasks[0] = StartCommunicationFor(requestOBJ);  
    tasks[1] = StartCommunicationFor(requestMTL);  
  
    // waiting for both webrequests to finish  
    await Task.WhenAll(tasks);  
    print("Passed [ Task.WhenAll(tasks) ]");  
  
    if (tasks[0].Result.result == UnityWebRequest.Result.ConnectionError |  
        tasks[1].Result.result == UnityWebRequest.Result.ConnectionError) {  
        debug.text = "Download failed";  
        print(requestOBJ.error + requestMTL.error);  
    } else {  
        debug.text = "Download successful";  
        print($"OBJ: {requestOBJ.result} | MTL: {requestMTL.result}");  
    }  
}
```

- Rapid Prototype DownloadAndVisualisation nachgeholt
- genutzt, um QR Reading mittel ZXing Library & AR mittel AR Foundation umzusetzen
- aufgetretendes Probleme:
 - ARCore (über AR Foundation) und Vulkan API vertragen/vertrugen sich nicht
 - => Darstellung VFX Graph auf PC durchgeführt
 - => nächster Knackpunkt: Was ist mit OpenGL ES, was wir in URP eh nutzen?
 - generelles Problem mit Mali GPU (Samsung Galaxy) und der Funktionsweise des VFX Graph bezüglich Shader Storage Buffer Objects

PoC Download and Visualisation

VFX Graph



Einleitung



Konzept-
zeichnung

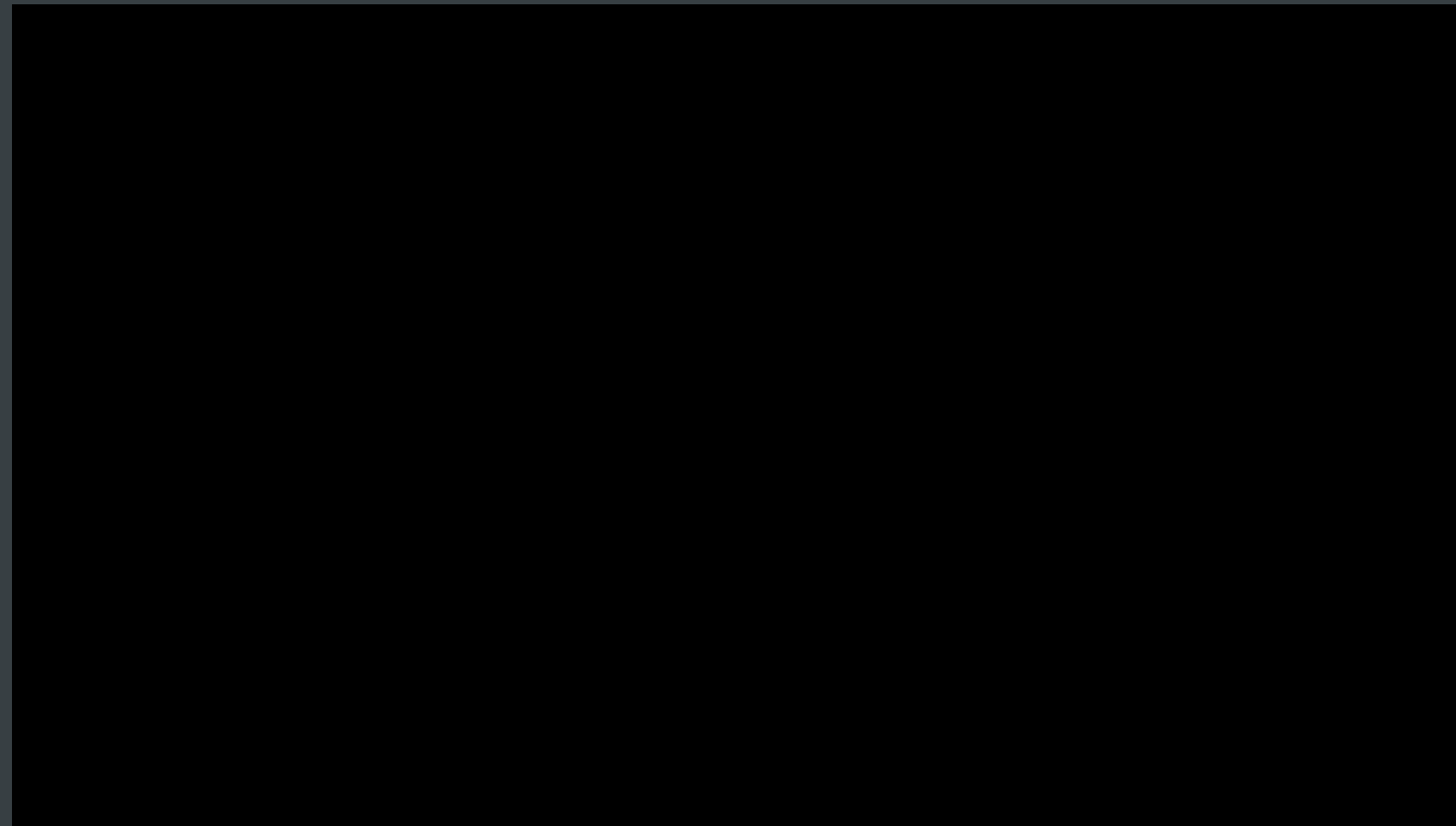


Prototype

4


Audit 4

S. 6



QUELLE:
<https://forum.unity.com/threads/visual-graph-with-opengles-3-android.859261/>

- Warum wird dies extra gezeigt?




Cec
Joined: Apr 7, 2014
Posts: 88

Ok.
Thanks to [@aleksandr](#) who found the reason of the issue in [this thread](#).
It seems that **every single Samsung Galaxy users are unable to use Visual Graph on OpenGL ES 3 (so with ARFoundation actually)** because of its Mali GPU which have only read-only support for ComputeBuffers in stages other than compute and fragment.
Too bad for Samsung Galaxy users ! Very disappointing...

EDIT : According to [Wikipedia](#), only US Samsung Galaxy smartphone owners can see Visual Effect Graph on OpenGL ES 3 (Adreno GPU chip instead of Mali)...
So for US developers : Carefull not to use Visual Effect Graph with ARFoundation because, the rest of the world Samsung Galaxy users won't be able to see your work.

Last edited: Apr 3, 2020

Cec, Apr 3, 2020 #2



PaulDemeulenaere
Joined: Sep 29, 2016
Posts: 68

Hello !
Thanks for you reporting this issue.

I agree with you, these different processor behind the same brand name are really confusing for users, North America and Asia are shipped with Adreno while the rest of the world are Mali.

It's actually the lack of [Shader Storage Buffer Object](#) at vertex stage on Mali which is problematic for Visual Effect Graph runtime shaders.

You can eventually check this compatibility [programmatically](#).

PaulDemeulenaere, Apr 4, 2020 #3

- zusammengeschnittener Prototyp aus der Logik der Rapid Prototypes
- aufgetretendes Problem:
 - VFX Graph wurde aus Samsung Galaxy (A50) nicht dargestellt

Vertical Rapid Prototype

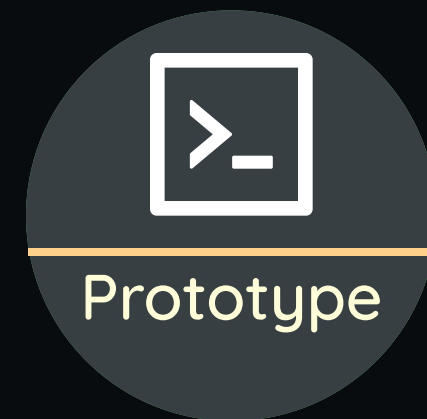
Vorstellung



Einleitung



Konzept-
zeichnung



Prototype



Audit 4

- für den nächsten Audit wollen wir den gerade gezeigten Rapid Prototype überarbeiten und zu einem funktionalen Prototypen umändern
- ein Fazit unsererseits bezogen auf mitzunehmende Erfahrungen bezüglich der Arbeitsweise/
Vorgehen und Inhalte
- eine Reflektion der Aufgabenteilung und Erfüllung der Ziele
- sowie ein Poster, was abgegeben werden soll

Deliverables

für Audit 4

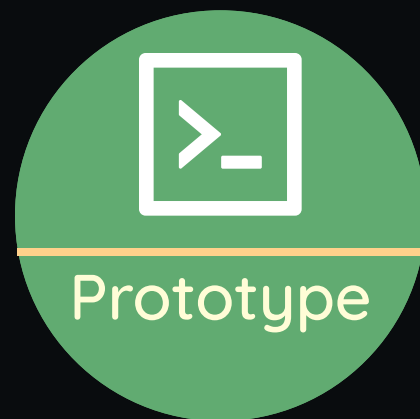
- Umsetzung funktionalen Prototypen
 - benötigte Verbesserungen des vertikalen Prototyps
 - > **Download auf Android** umsetzen
(.obj -> AssetBundles)
(Daten aus QR Code für Downloadlink nutzen)
 - > VFX Graph Ersatz “**Particle System**” für Android ARCore
 - > passende **Audio** für gezündeten Vulkan
 - > **Batterie** als Feuerwerk ???
 - > **AR Multiplayer** ???
- Fazit & Reflektion
- Poster



Einleitung



Konzept-
zeichnung

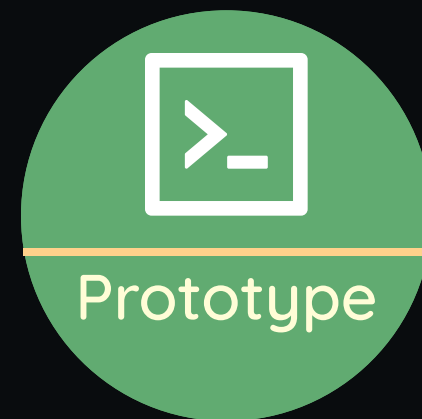


Prototype



Audit 4

- Verabschiedung



Vielen Dank für ihre Aufmerksamkeit!

