



ARestaurant

Michelle Pötsch
Markus Müller
Minh Anh Vu

Gliederung

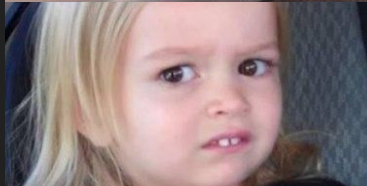
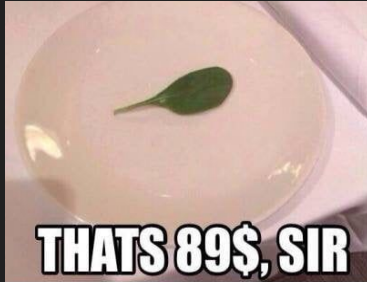
// Motivation

// Idee

// Augmentation und Interaktionsdesign

// Implementierung

Motivation



- einzelne Gerichte oft unklar
 - Gerichte individualisieren
- + Restaurant digitalisieren für Corona

Idee

- Speisekarte Text erkennen oder Image-Targets

Features

- Übersetzung
- Bild / 3D Modell
- Beschreibung
- Zubereitung ändern
- Bestellung

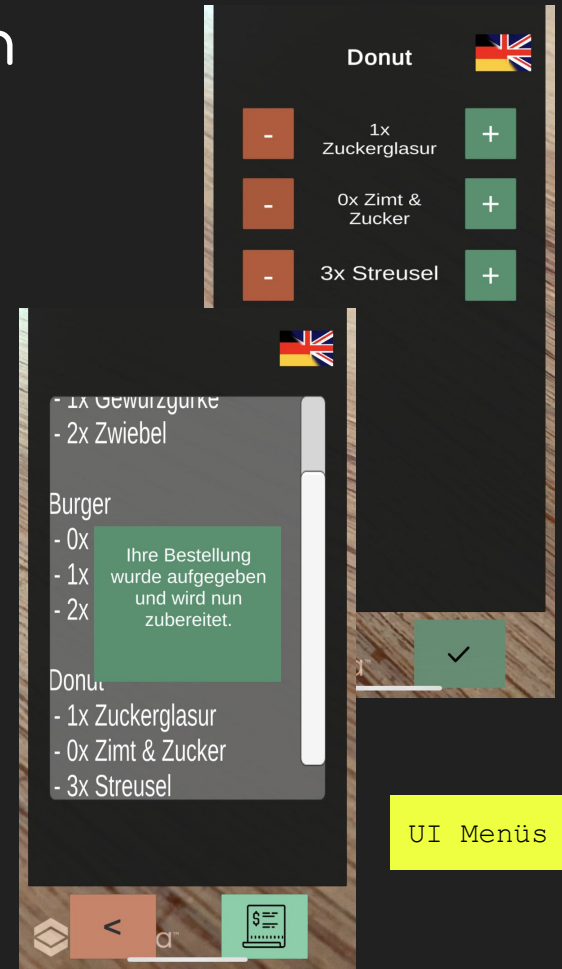
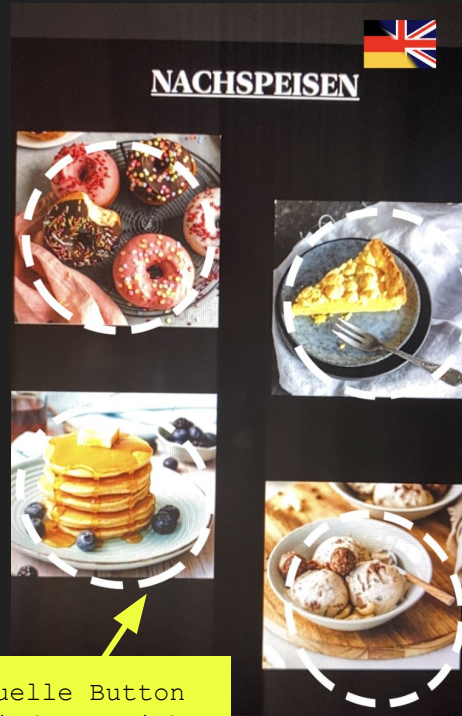
AR



UI

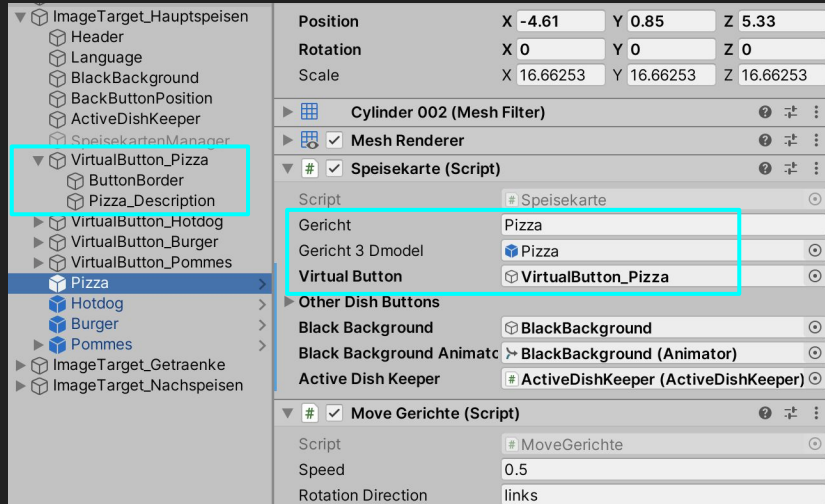


Augmentation und Interaktionsdesign

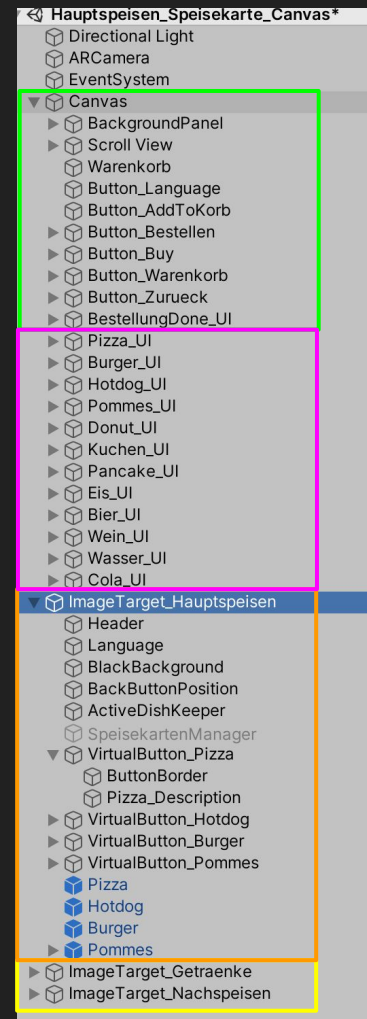
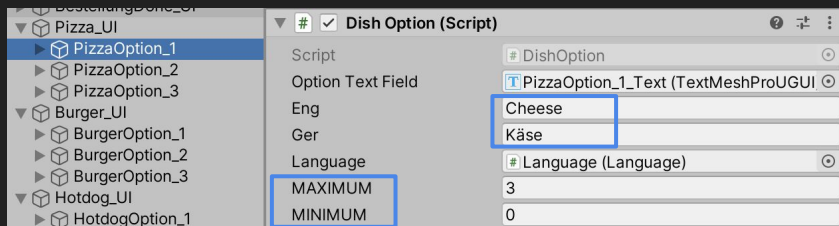


Implementierung

Aufbau pro Gericht



flexible Zutaten



UI
Menü

Individuelle
UIs
pro Gericht

Aufbau pro
Speisekarte

..weitere Speisekarten

Implementierung

Zuteilung Gericht,
Virtual-Button etc.

```
public class Speisekarte : MonoBehaviour
{
    ...

    void Start()
    {
        virtualButton = GameObject.Find("VirtualButton_" + gericht);
        description = GameObject.Find(gericht + "_Description");
        backButtonPosition = GameObject.Find("BackButtonPosition").transform;
        originalButtonPosition = virtualButton.transform;

        virtualButton.GetComponent<VirtualButtonBehaviour>().RegisterOnButtonPressed(OnButtonPressed);
        virtualButton.GetComponent<VirtualButtonBehaviour>().RegisterOnButtonReleased(OnButtonReleased);

        gericht3Dmodel.SetActive(false);
        blackBackground.SetActive(false);
        description.SetActive(false);
        dishActive = false;
    }
}
```

Gericht de-/aktiv
setzen

```
// Methoden wenn VirtualButton mit finger getriggert wird
private void OnButtonPressed(VirtualButtonBehaviour obj)
{
    if (dishActive)
    { // DEACTIVATE
        blackBackgroundAnimator.SetBool("dishActive", false);
        StartCoroutine(waitBeforeClosingBackground());
        description.SetActive(false);
        gericht3Dmodel.SetActive(false);
        dishActive = false;
        activeDishKeeper.dishActive = dishActive;
        activeDishKeeper.activeDishName = "";
        activateDeactivateOtherButtons();
    }
    else ...
}

public void OnButtonReleased(VirtualButtonBehaviour vb) ...

private void activateDeactivateOtherButtons() ...
}
```

Implementierung

UI-Management & Warenkorb

Assets > Scripts > MenuManager.cs > ...

```
56 public void Bestellen()
57 {
58     if (activeDishKeeper.dishActive)
59     {
60         // Debug.Log("Bestellen gedrückt mit aktivem Gericht: " + activeDishName);
61         for (int i = 0; i < dishUIsArray.Length; i++)
62         {
63             if (activeDishKeeper.activeDishName + "_UI" == dishUIsArray[i].name)
64             {
65                 Debug.Log("DISH UI WAS FOUND.");
66                 dishUIsArray[i].SetActive(true);
67                 activeDishUI = dishUIsArray[i];
68             }
69             ActivateDishUI();
70         }
71     }
72     else...
76 }
```

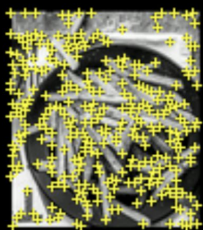
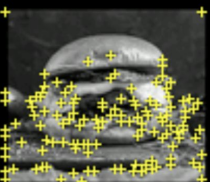
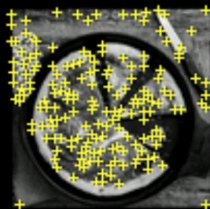
```
91 public void AddToWarenkorb()
92 {
93     Debug.Log("AddToWarenkorb");
94     GameObject[] dishOptions = GameObject.FindGameObjectsWithTag("dishOption");
95     for (int i = 0; i < dishOptions.Length; i++)
96     {
97         // Debug.Log(dishOptions[i].name);
98         optionsAsString += " - " + dishOptions[i].GetComponent<DishOption>().fullText + "\n";
99     }
100     fullBestellung = activeDishKeeper.activeDishName + "\n" + optionsAsString;
101     // Debug.Log(fullBestellung);
102     warenkorb.bestellungen.Add(fullBestellung);
103     ResetBestellungStrings();
104     DeactivateUI();
105 }
107 public void ShowWarenkorb()
108 {
109     ActivateWarenkorbUI();
110     Debug.Log(warenkorb.getAlleBestellungen());
111 }
113 public void BuyBestellungen()
114 {
115     // sende String des Warenkorbs an Küche, damit sie wissen, was sie zubereiten sollen
116     Debug.Log("Bestellung wurde aufgegeben und wir nun zubereitet.");
117     bestellungDoneUI.SetActive(true);
118     warenkorb.emptyWarenkorb();
119 }
```


Image-Targets

HAUPTSPEISEN



HAUPTSPEISEN



Target Name

Type

Rating ⓘ



Hauptspeisen_Schwarz

Single Image

★★★★★

GETRÄNKE



NACHSPEISEN



☐ Target Name

Type

Rating ⓘ



Nachspeisen

Single Image

★★★★★



Getraenke

Single Image

★★★★★



Fragen ?

Vielen Dank für eure
Aufmerksamkeit !

Impressum

Quellen (Helfende Seiten)

- TurboSquid : <https://www.turbosquid.com>
- Unity Asset store : <https://assetstore.unity.com>
- Udemy : <https://www.udemy.com>
- Unity learn : <https://learn.unity.com/tutorials>

HAUPTSPEISEN



GETRÄNKE



NACHSPEISEN

