

Warppay (Teil von WarpInfra)

Als Teil des Projektes [WarpInfra](#) wurde die Implementierung eines Shopsystems wieder aufgenommen. Auf der Basis des ursprünglichen Konzeptes wurde hier eine Implementierung umgesetzt.

Die Implementierung besteht aus zwei Komponenten:

- Das Modul WarpPay in [WarpInfra](#) implementiert die API und die Datenspeicherung
- Das Frontend wird durch eine Android-App bereitgestellt:
<https://gitlab.warpzone.ms/infrastruktur/warppay-app>

Die Benutzerkonten werden hierbei aus dem WarpInfra verwendet.

Warppay (Ursprüngliches Konzept, altes Projekt)

Projekt, um ein elektronisches Kassensystem mit Barcode-Scanner einzuführen.

Anforderungen

- Warenkorb
- PrePaid
- Storno
- Aufbuchen: (0,50€, 1€, 2€ eigentlich nur als Ausgleich) 5€ 10€ 20€
- Personenbarcode / Barzahlung-Barcode / Anonyme Karte - Personenbarcode im System mit Namen verknüpfbar
- Touchscreen & Barcode Scannen
- maria/mysql/postgre-Datenbank(!)
- Will Finanzamt die Daten haben?
- In Zentrales OpenHUB Einbauen

Use Cases

- Ware Kaufen
 - Erst (mehrere) Waren Scannen dann Personen(Barcode)
 - Beim Scannen zeigt es den Namen der Ware und den Wert an
 - Es zeigt eine Liste aller gescannten Waren mit Einzelwert und Summe, jeder Scan ist ein einzelner Posten
 - Storno einer Ware löscht letzten Posten (Bis Warenkorb leer sind)
 - 1x Personenbarcode Scannen zeigt Guthaben und ggf. Namen an / Fehlermeldung wenn Guthaben nicht reicht
 - 2x selben Personenbarcode Scannen führt Buchung durch
- Storno (Keinen! Timeout für Offene Posten)

- a) Letzten Posten löschen
 - b) Transaktion abbrechen
- Karteninfo Anzeigen/Ändern
 - Leerer Warenkorb - Personenbarcode Scannen
 - Zeigt Guthaben und Namen an - erlaubt Eingeben/ändern eines Namens
- Guthaben Aufbuchen
 - Geldwert in Warenkorb legen
 - Wie normale Ware Kaufen
- Neue Waren Hinzufügen
 - Über Admin Backend?
- Neuen PersonenBarcode(Karte)
 - Wird über Admin Backend generiert
 - Format: WZK + Fortlaufende Nummer (vorschlag: 6 Stellig)
 - Anlernen über „Karteninfo anzeigen/ändern“
- Auswertung über Admin Backend
 - ...
- Buchungslog
 - Enthält: Artikel, Anzahl, Verkaufspreis, Datum-Uhrzeit, POS(Wo wurde es gekauft)

Weiteres Vorgehen

- Barcodescanner: Tastaturemulation - (im Zweifel seriell per mikrokontrooller)
- Frontent: Webfrontent (Bootstrap)
- Backend(Middleware): Django
- Datenbank Schema: ←insert here→

Aufbau

Frontend

Bootstrap - Dunstkreis, Schneeman

Middleware

Django - Void, (Sand)

DB



Quellcode

git unter [gitolite@warpzone.ms:warpshop.git](https://git.warpzone.ms/warpshop.git)

Nächstes ToDo

- Datenschutz: Nutzer in Transaktion nach x Monaten auf Barverkauf setzen(ggf nur wenn passendes Feld doch/nicht gesetzt)
- Backup: Wie gegen Absturz oder ähnliches sichern? Daten dürfen nicht verloren gehen.

From:

<http://wiki.warpzone.ms/> - **warpzone**

Permanent link:

<http://wiki.warpzone.ms/infrastruktur:warpshop?rev=1491950780>

Last update: **11.04.2017**

