

Anforderungen Kiosk Vertiefungsarbeit

Abgabe: Sa, 09.06.2018

- Github/Bitbucket Repository
- Generierte JavaDoc
- Präsentation, wenn Folien vorhanden
- UML Klassendiagramm
- UML Activity Diagramm für
 - o Kiosk erstellen
 - o Einkauf tätigen
 - o Artikel bestellen
 - o Inventar

Bewertung

- | | |
|---------------------------|-----|
| - Funktionalität | 35% |
| - Architektur / Codestyle | 25% |
| - JavaDoc | 15% |
| - Unit Tests | 15% |
| - Präsentation | 10% |

Präsentation

- Reflexion ca. 5min
- Was habe ich gelernt?
- Was ist mir gut gelungen, was weniger?
- Wo lagen meine grössten Schwierigkeiten und wie habe ich diese gelöst?
- Was würde ich nächstes Mal anders machen?
- Wie zufrieden bin ich mit dem Resultat?
- etc.

Anforderungen

Es soll eine Applikation erstellt werden, welche einen Kiosk simuliert.

Ein Kiosk hat einen Namen, Standort, Offen/Geschlossen Status, Mitarbeiter, Artikel, Kasse, Lieferant.

Ein Mitarbeiter hat einen Namen und eine Zuordnung zu einem Kiosk.

Alle Artikel haben eine Bezeichnung und einen Preis.

- Es gibt verschiedene Kategorien: Alkohol, Zeitschriften, Snacks, Softdrinks und Tabak.
- Alkohol und Tabak erfordern eine Altersprüfung.
- Zeitschriften haben einen Typ (Lokal, International, Klatsch, Kinder, etc.).
- Snacks haben ebenfalls einen Typen (Sandwich, Süß, Salzig, Früchte, etc.)
- Softdrinks und Alkohol haben eine Mengenangabe.

Der Lieferant hat einen Namen und Artikel.

Folgende Aktionen sollen über ein GUI ausgeführt werden können:

- Kiosk erstellen (Eingabe für Name, Standort, Geld in Kasse, Mitarbeiter)
 - o Zu Beginn ist der Kiosk geschlossen
 - o Wird mit einem fixen Artikel Set initialisiert
 - o Wird mit einem fixen Lieferant initialisiert, welcher mit einem fixen Artikel Set initialisiert wird.
- Mitarbeiter beim Erstellen eines Kiosk hinzufügen (Eingabe für Name)
- Alle erfassten Kioske erscheinen auf einer Übersicht und können hier geöffnet und geschlossen werden. Weitere Aktionen auf dem Kiosk können nur ausgeführt werden wenn der Kiosk geöffnet ist.
- Auf der Übersicht gibt es pro Kiosk einen Button 'Einkauf tätigen'
 - o Eingabe für Kunde erstellen (Name und Alter)
 - o Auswahl aller Artikel im Kiosk
 - o Selektieren, Menge eingeben, Kaufen
 - o Zurück zur Übersicht
- Auf der Übersicht gibt es pro Kiosk einen Button 'Artikel bestellen'
 - o Auswahl aller Artikel vom Lieferanten
 - o Selektieren, Menge eingeben, Kaufen
 - o Zurück zur Übersicht
- Auf der Übersicht gibt es pro Kiosk einen Button 'Inventar'
 - o Erstellt Ein Excel mit dem aktuellen Inventar des Kiosk
- Auf der Übersicht gibt es pro Kiosk einen Button 'Automatischer Durchlauf' (Multithreading Aufgabe)
 - o Erstellt 5 Kunden, fügt Waren in deren Einkaufskörbe, Waren werden verkauft

Technische Vorgaben

- Programmiersprache: Java
- GUI mit JavaFX
- Der Code soll mittels JUnit sinnvoll getestet werden.
- Log Statements sollen an sinnvollen Orten eingebaut werden.

Weiteres

- Wenn mehr Geld vom Kiosk abgehoben wird als vorhanden ist (Artikel bestellen) soll eine eigene Exception geworfen und abgehandelt werden.
- Das Excel fürs Inventar soll alle Artikel listen mit Bezeichnung, Preis und Stückzahl. Sie sollen absteigend nach Preis sortiert sein.
- 'Automatischer Durchlauf': Der Kunde wird als Thread implementiert. Fügt einen *sleep(5000)* beim Bezahlen hinzu um eine Warteschlange von Kunden zu bekommen. Füge auch einen *sleep(2000)* beim Artikel in Warenkorb des Kunden legen hinzu, damit nicht alle Kunden bereits in der Schlange stehen bevor der erste bezahlt hat. Das Erstellen der Kunden soll im Abstand von 1000ms passieren.
 - Die Warteschlange an der Kasse soll detailliert ausgegeben werden. Ein Log wenn ein Kunde die Warteschlange betritt mit Position in der Schlange, ein Log wenn der Kunde an der Reihe ist und ein Log wenn der Kunde bezahlt hat. Beispiel:
 - Kunde1 an Kasse (Warteschlange Position 1)
 - Kunde1 bezahlt
 - Kunde2 an Kasse (Warteschlange Position 2)
 - Kunde3 an Kasse (Warteschlange Position 3)
 - Kunde 1 hat bezahlt
 - Kunde 2 bezahlt
 - Kunde4 an Kasse (Warteschlange Position 3)
 - Usw.

Vorgehensweise

Die Theorie für alle Anforderungen wird im Unterricht behandelt.

Bei Unklarheiten bezüglich Anforderungen nachfragen.

Am Ende jedes Abends nehmen wir uns ca. 30min Um am Semesterprojekt zu arbeiten und Fragen zu klären.

Abgabe: Sa, 09.06.2018