Endabgabe

Gruppenarbeit von Asya Tetik, Christina Gabler, Lisa Fingerle, Deborah Reinbold

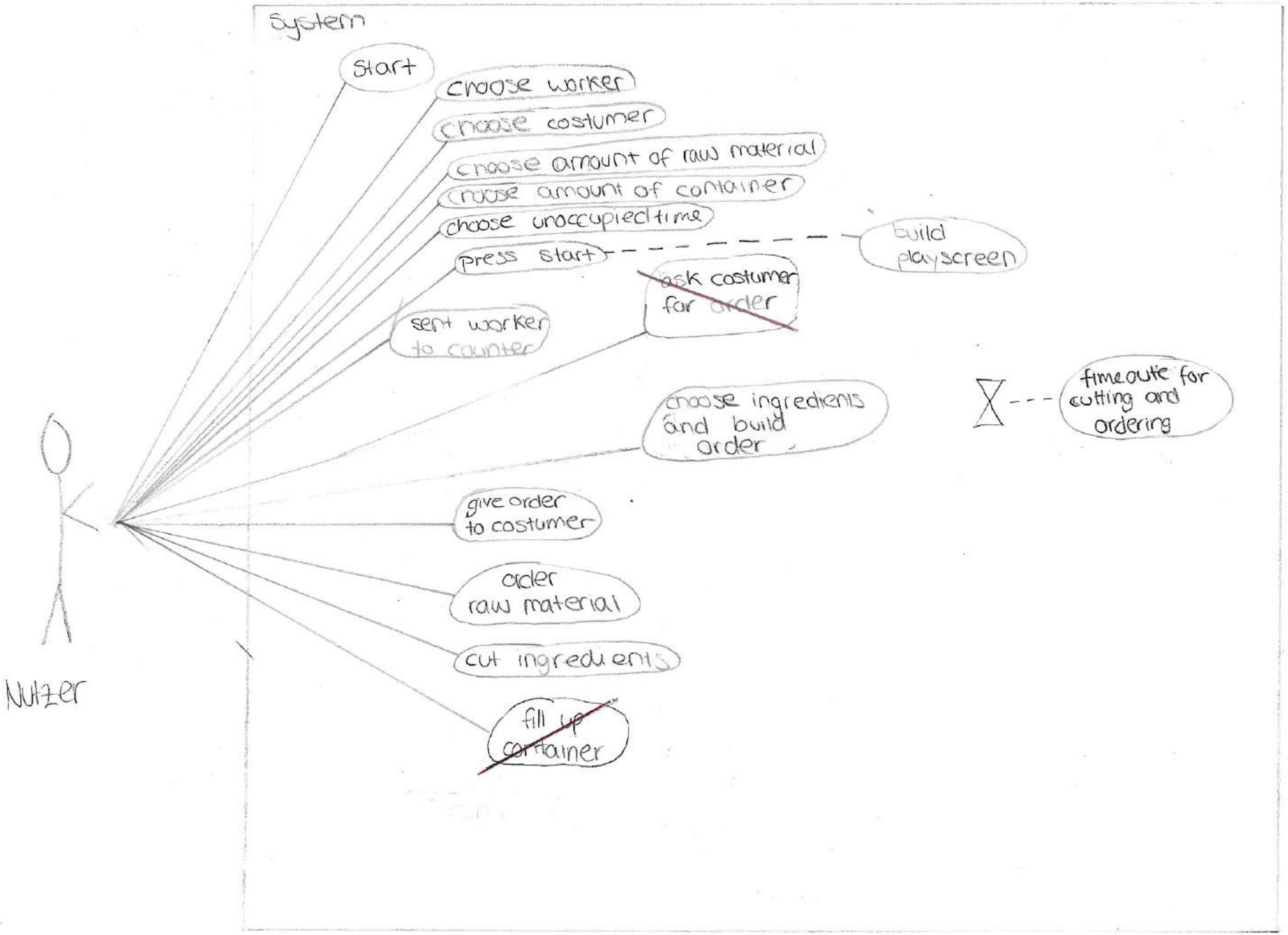
Aus Gründen der Übersichtlichkeit haben wir gemeinsam nur in einem Repository gearbeitet und dort rein gepusht und gepullt und die fertige Endabgabe dann in einzelne Repositories hochgeladen.

Von Christina Gabler, Deborah Reinbold, Lisa Fingerle und Asya Tetik

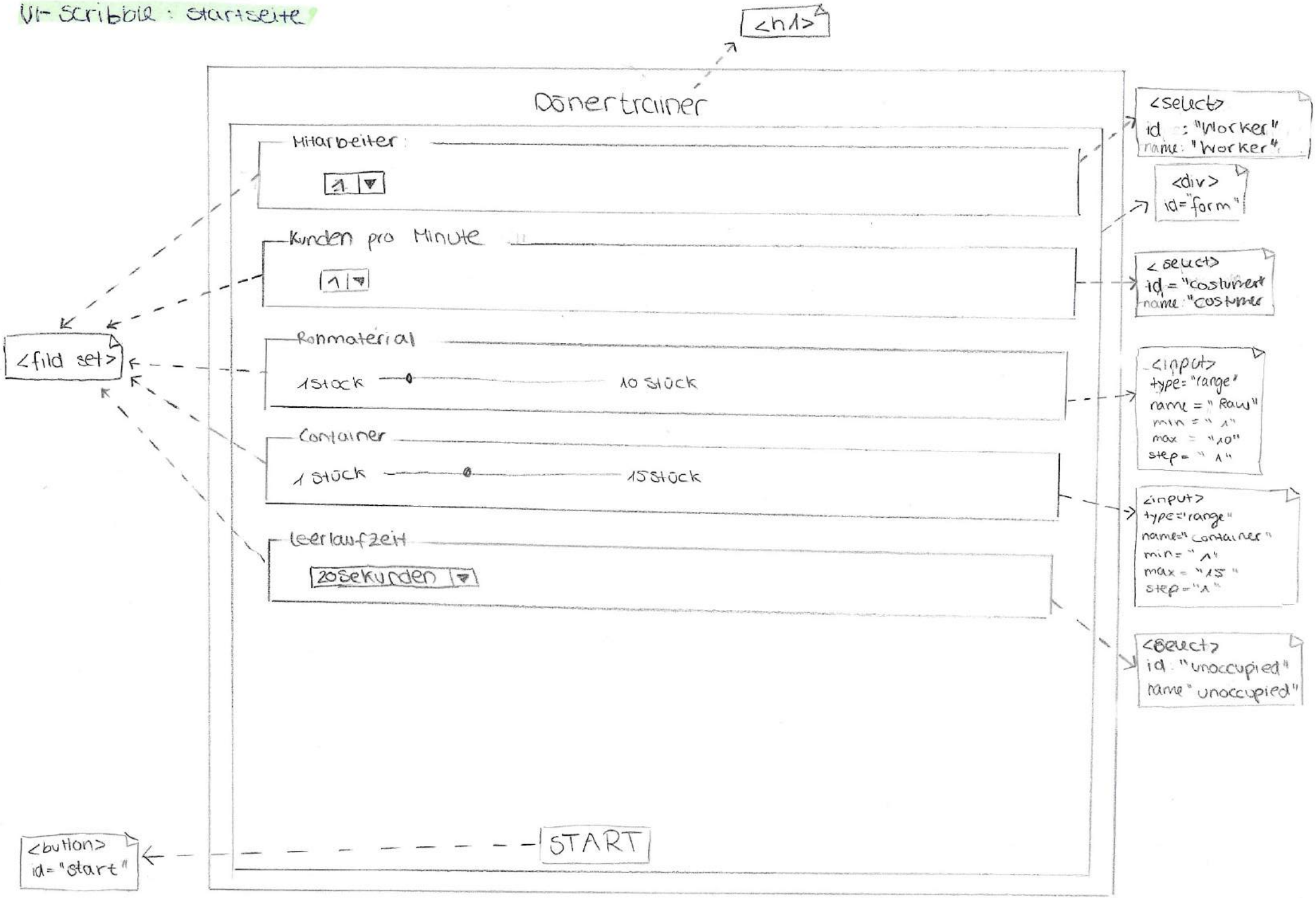
Döner-Trainer Anleitung

1. Zu Beginn kann der Nutzer auf der Startseite die Anzahl der Kunden pro Minute, die Anzahl der Arbeiter, die Kapazität vom Rohmaterial und geschnittenem Material und der Leerlaufzeit der Mitarbeiter bestimmen.
2. Der Kunde gibt seine Bestellung auf und wird vom Mitarbeiter (durch den Nutzer gesteuert) bedient. Um die Mitarbeiter zu bewegen, muss er zuvor durch einen Klick ausgewählt werden.
3. Durch Klicken auf die einzelnen Zutaten wird die Bestellung zusammengestellt und die zur Verfügung stehenden Zutaten werden verringert. Danach kann das Gericht mit Klick auf den Kunden diesem übergeben werden.
4. Die Zufriedenheit der Kunden kann sich durch die Geschwindigkeit und die Genauigkeit des Spielers verändern und das Spiel beeinflussen. Bspw.: Wenn der Spieler zu lange braucht, um ein Gericht zu servieren, wird der Kunde wütend und verlässt die Dönerbude.
5. Auch die Mitarbeiter verändern ihre Stimmung je nach dem, wie hoch ihr Stresslevel ist.
6. Falls die Zutaten an der Theke leer sind, muss der Mitarbeiter in die Küche geschickt werden, um die Rohstoffe zu schneiden. Sind die Rohstoffe leer, können sie nachbestellt werden.
7. Die Zufriedenheit der Kunden und Mitarbeiter sowie die Anzahl der verkauften Gerichte werden fortlaufend angezeigt.

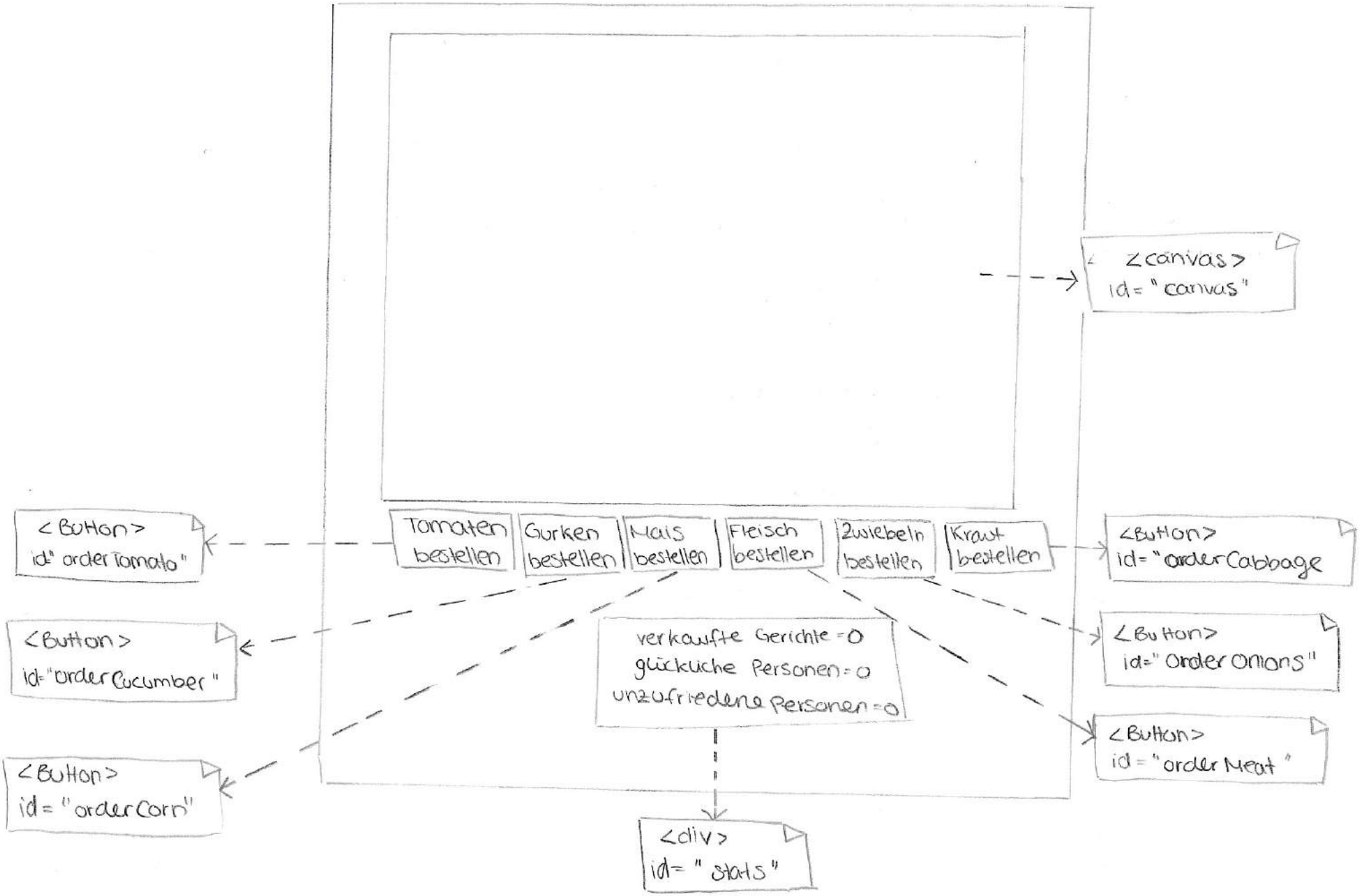
Döner-Trainer : Use-Case-Diagram



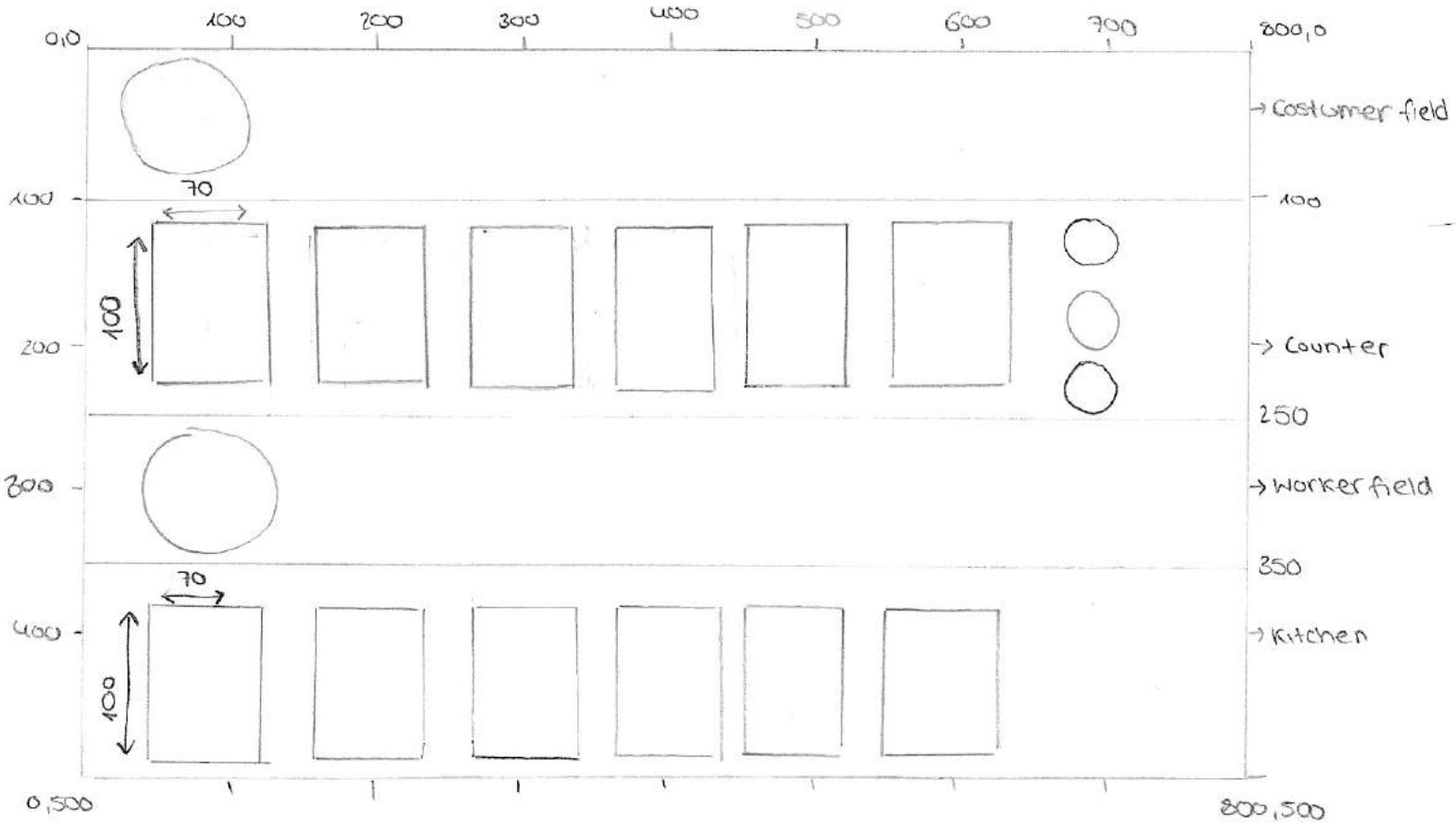
UI-Scribble: Startseite



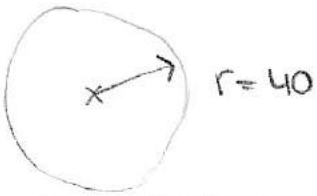
UI-Scribble : Dönerbude



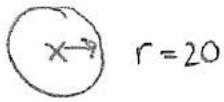
Canvas: Kebab - House



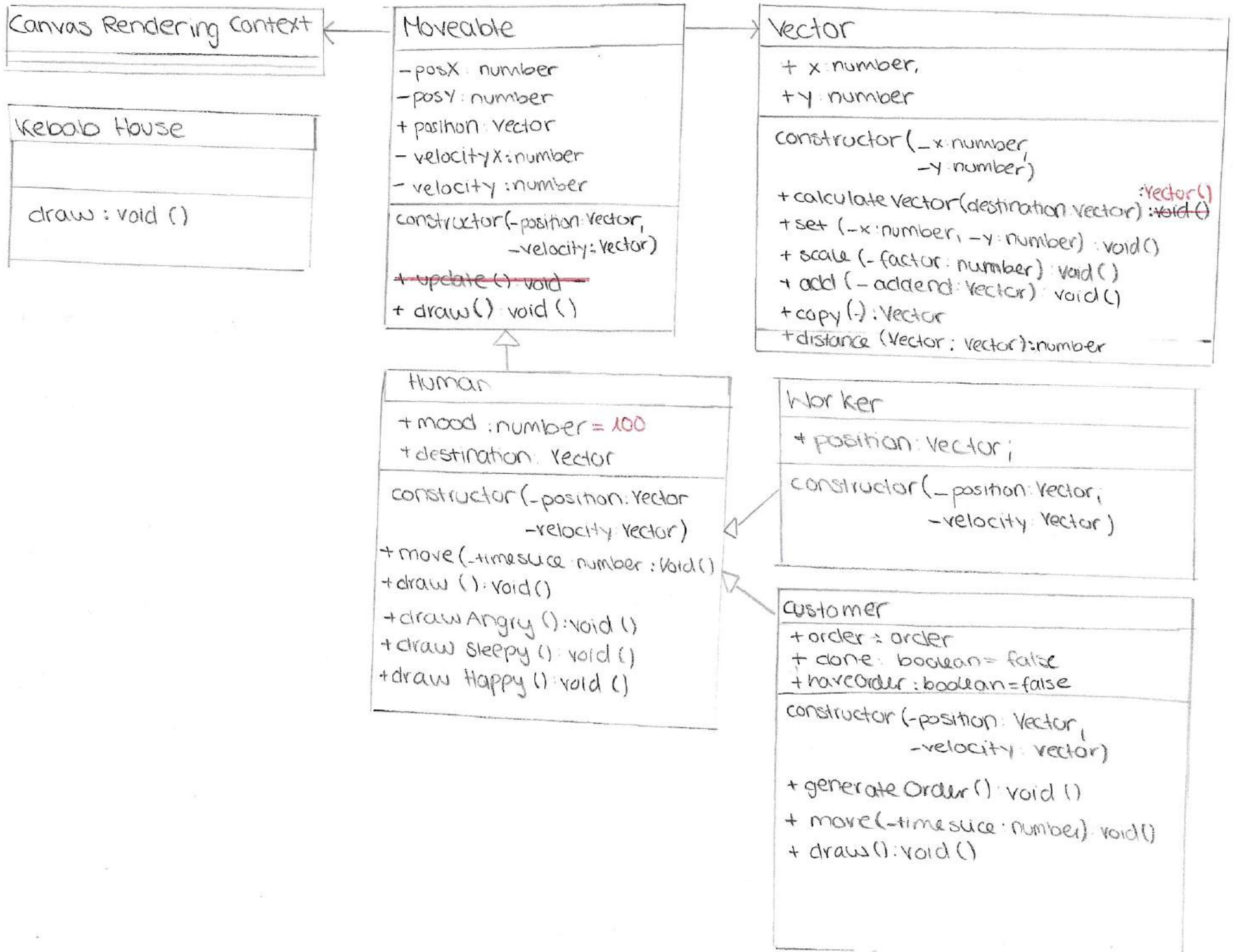
Worker & Customer



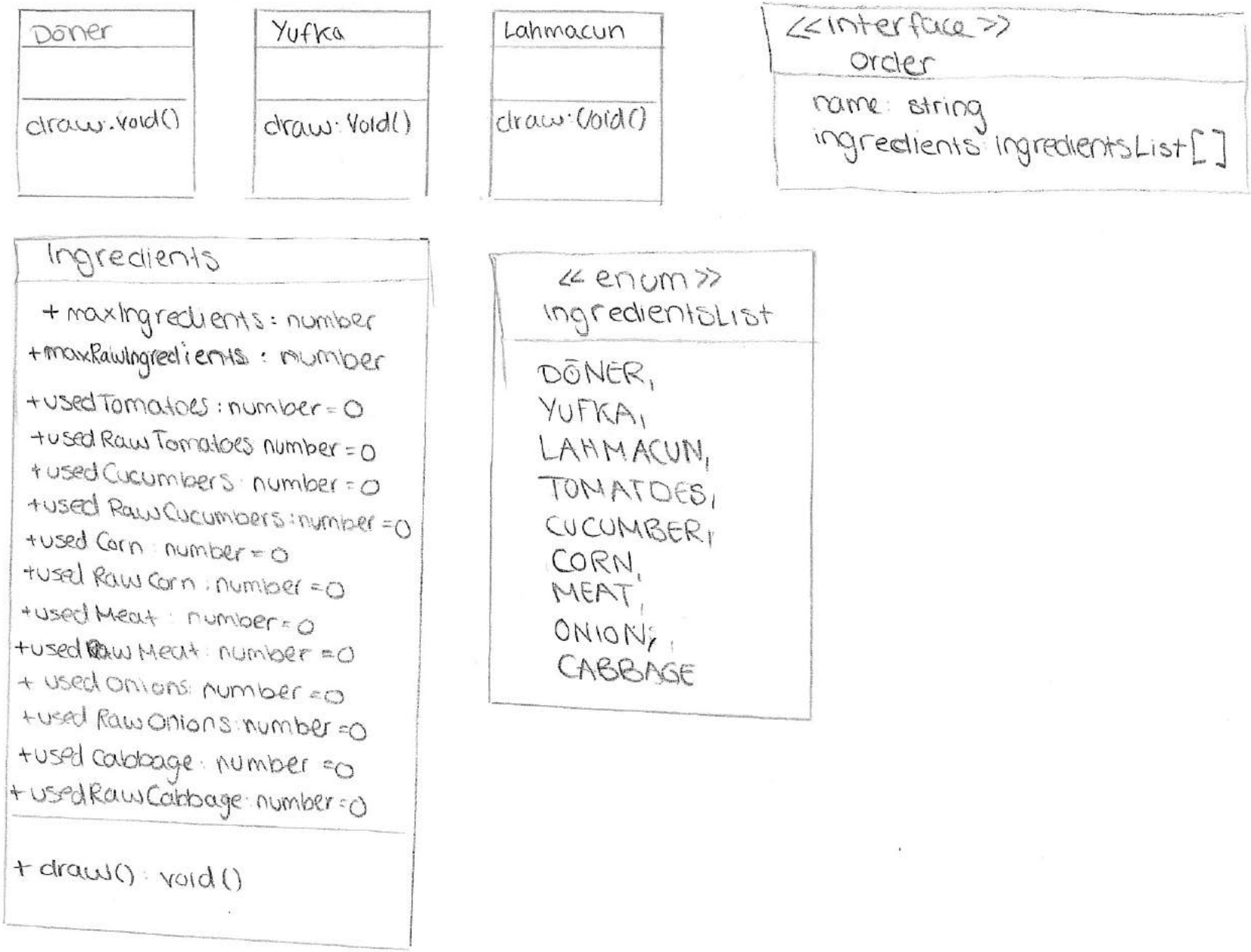
Döner, Yufka, Lahmacun



Classdiagram ②

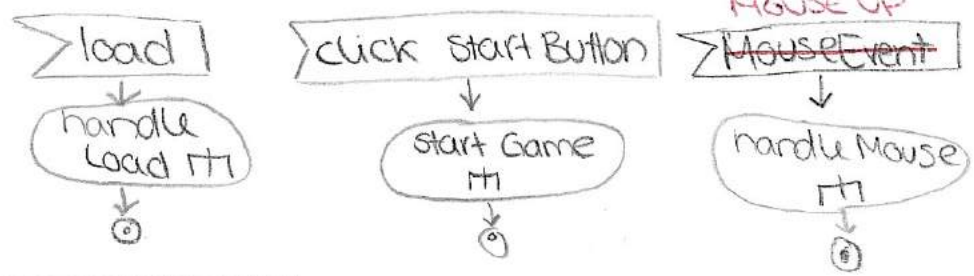


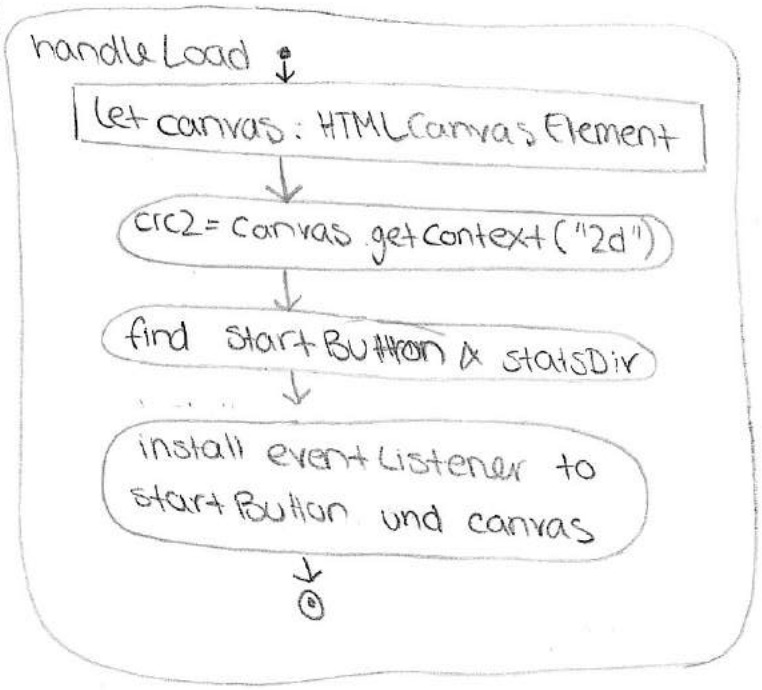
Classdiagram ②



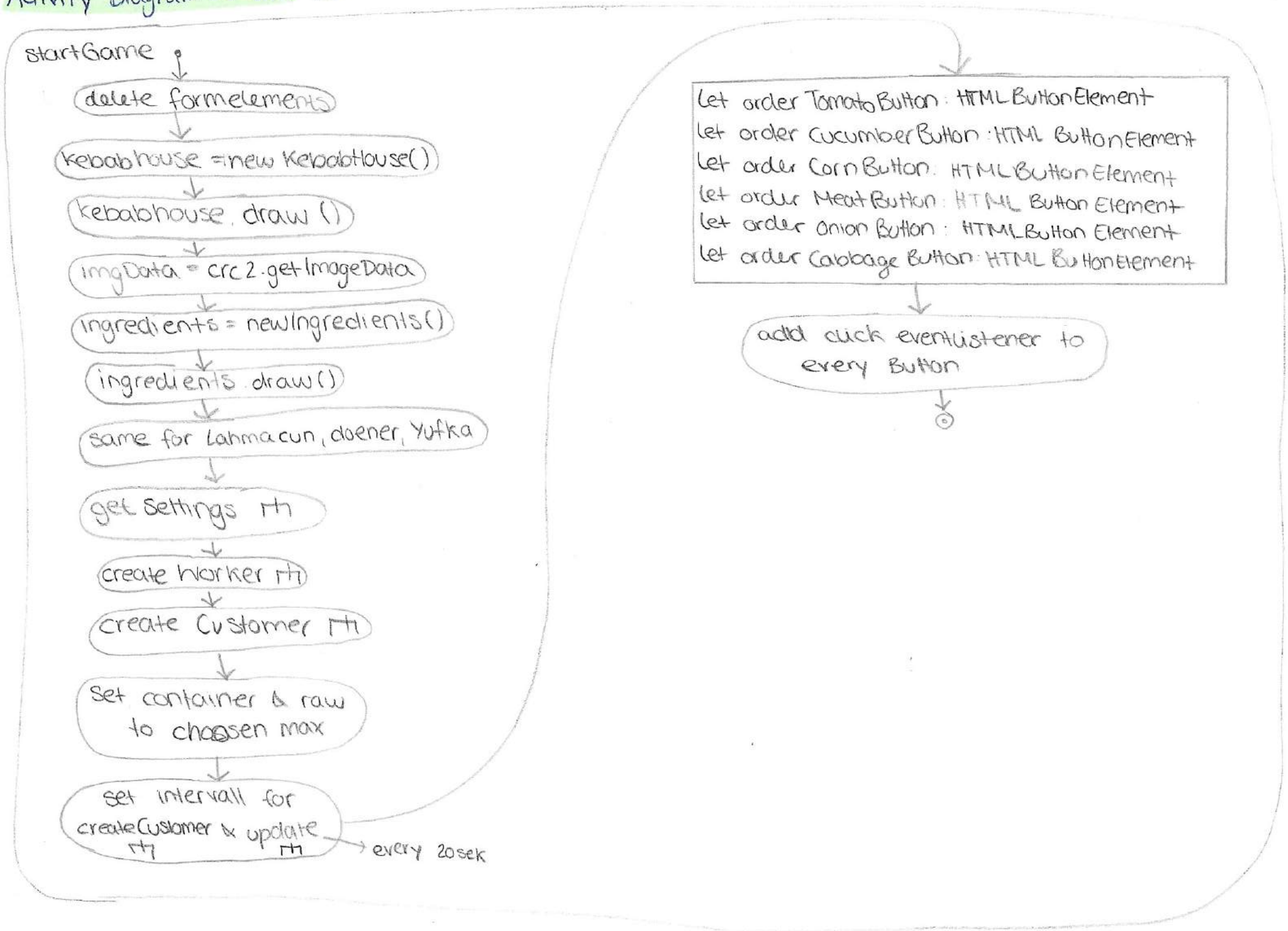
Activity Diagram : Main

```
let crc2: CanvasRenderingContext2D
let startButton: HTMLButtonElement
let kebabhouse: kebabhouse
let ingredients: Ingredients
let doener: Doener
let yufka: Yufka
let Lahmacun: Lahmacum
let worker: number = 2
let customer: number = 3
let raw: number = 10
let container: number = 15
let unoccupied: number = 20
let moveable: Moveable[] = []
~~let animation: boolean = false~~
let allCustomer: number = 0
let imgData = ImageData
let usedIngredients: IngredientsList[] = []
let activeWorker: Worker
let doneOrder: number = 0
let happyCustomer: number = 0
let angryCustomer: number = 0
let happyWorker: number = 0
let unhappyWorker: number = 0
let statsDiv: HTMLDivElement
```

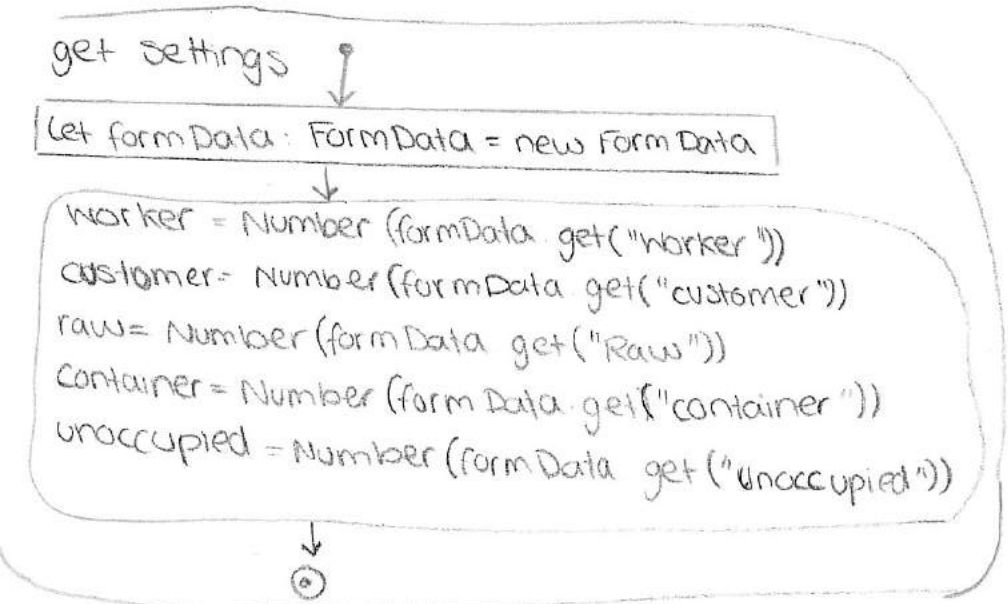


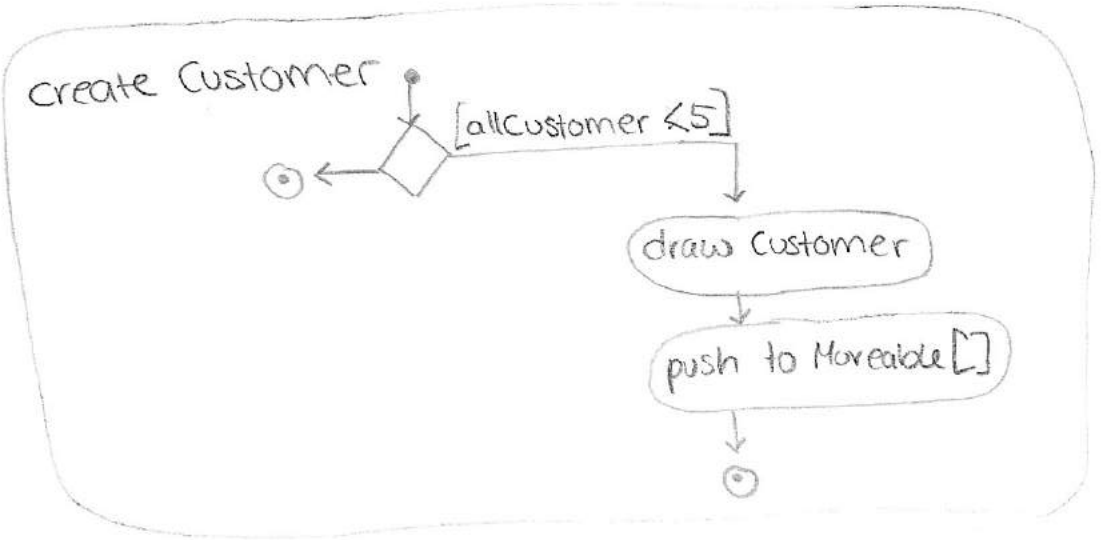


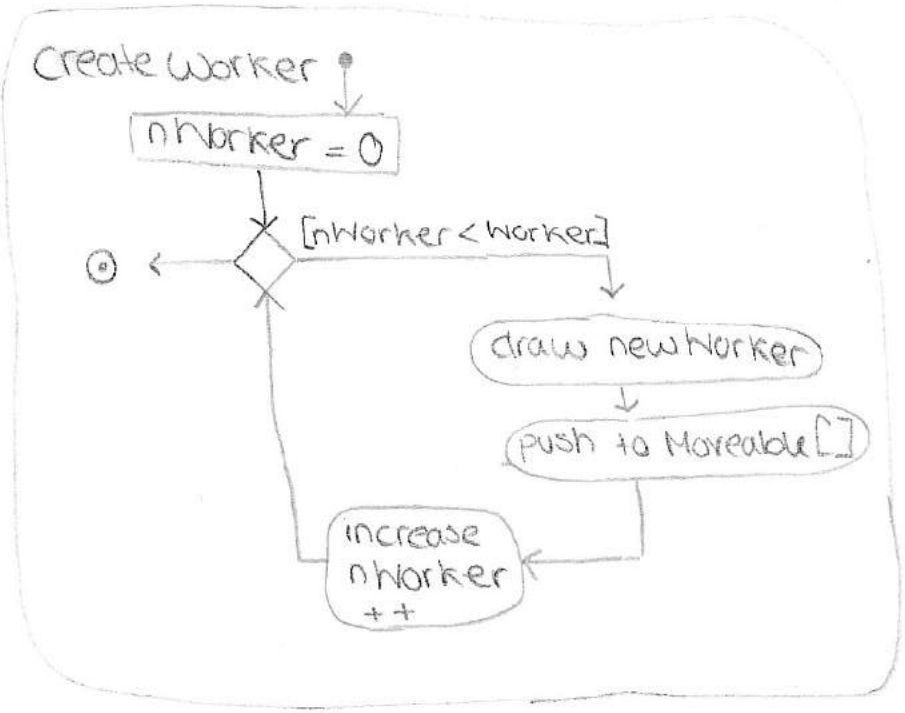
Activity Diagram : Main ②



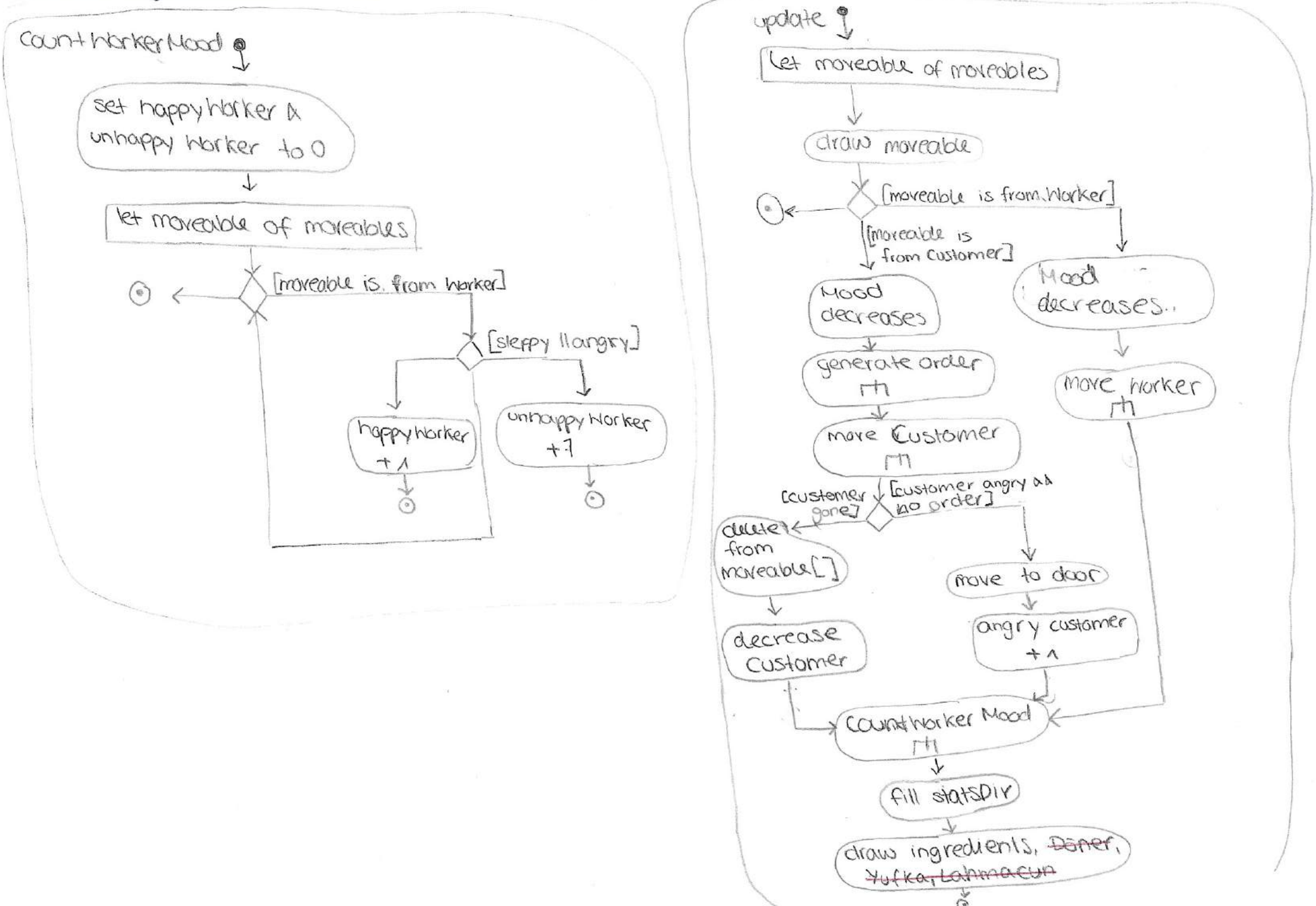
Activity Diagram : Main ③



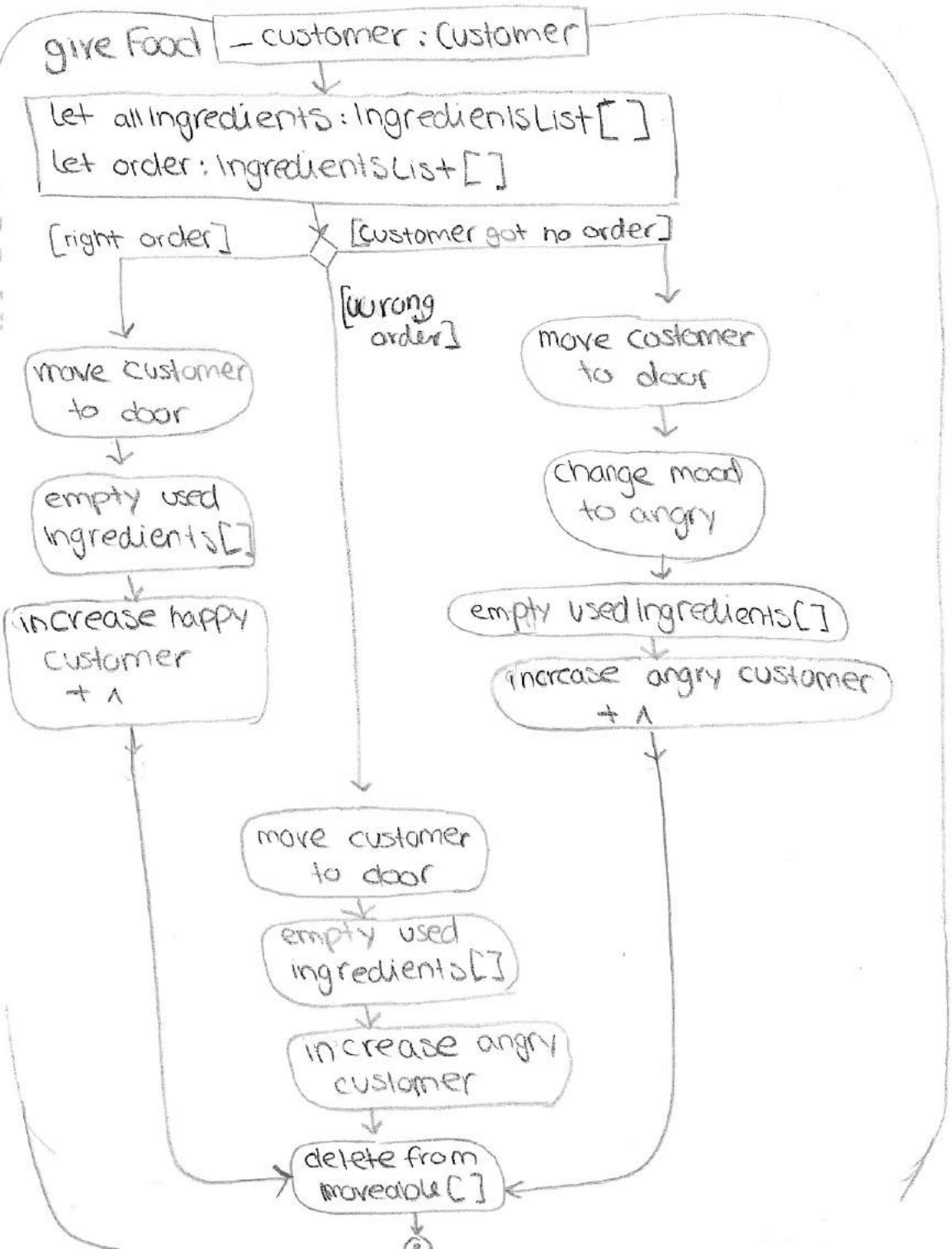




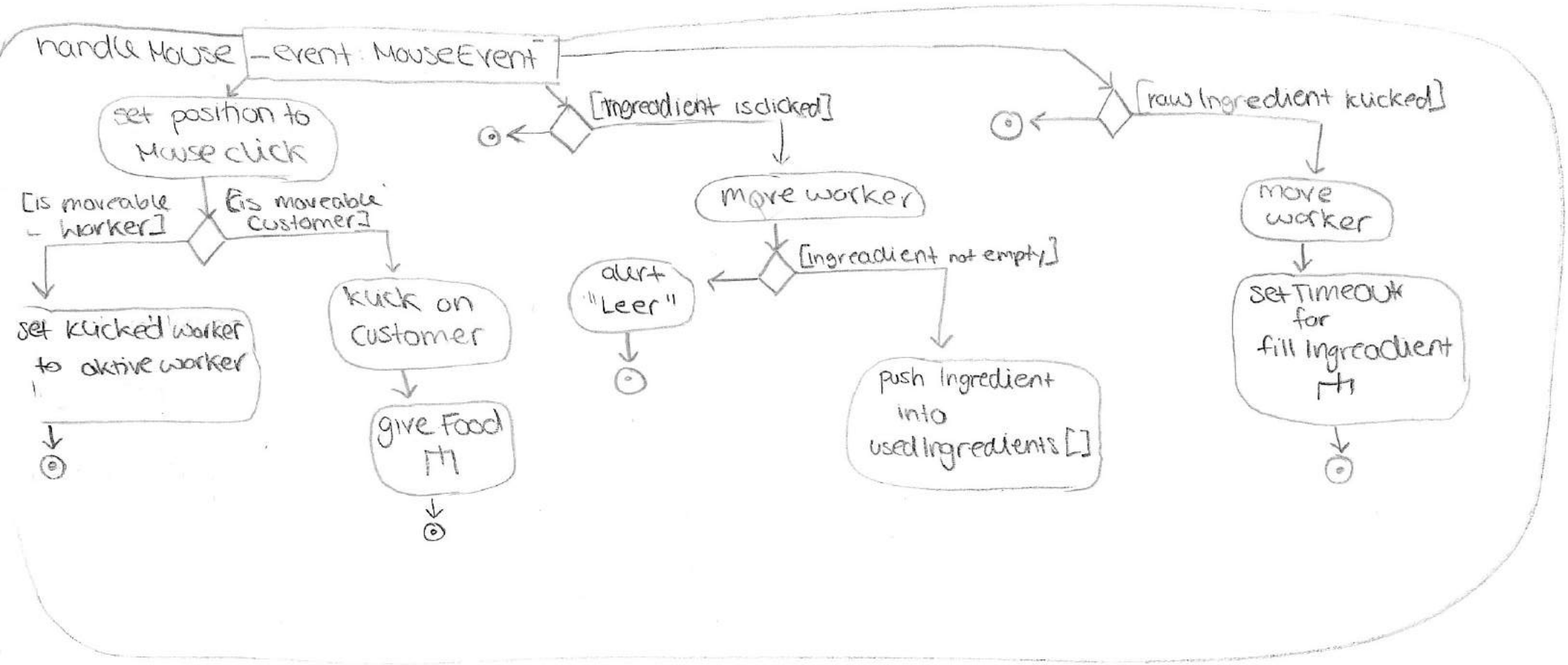
Activity Diagram : Main ④



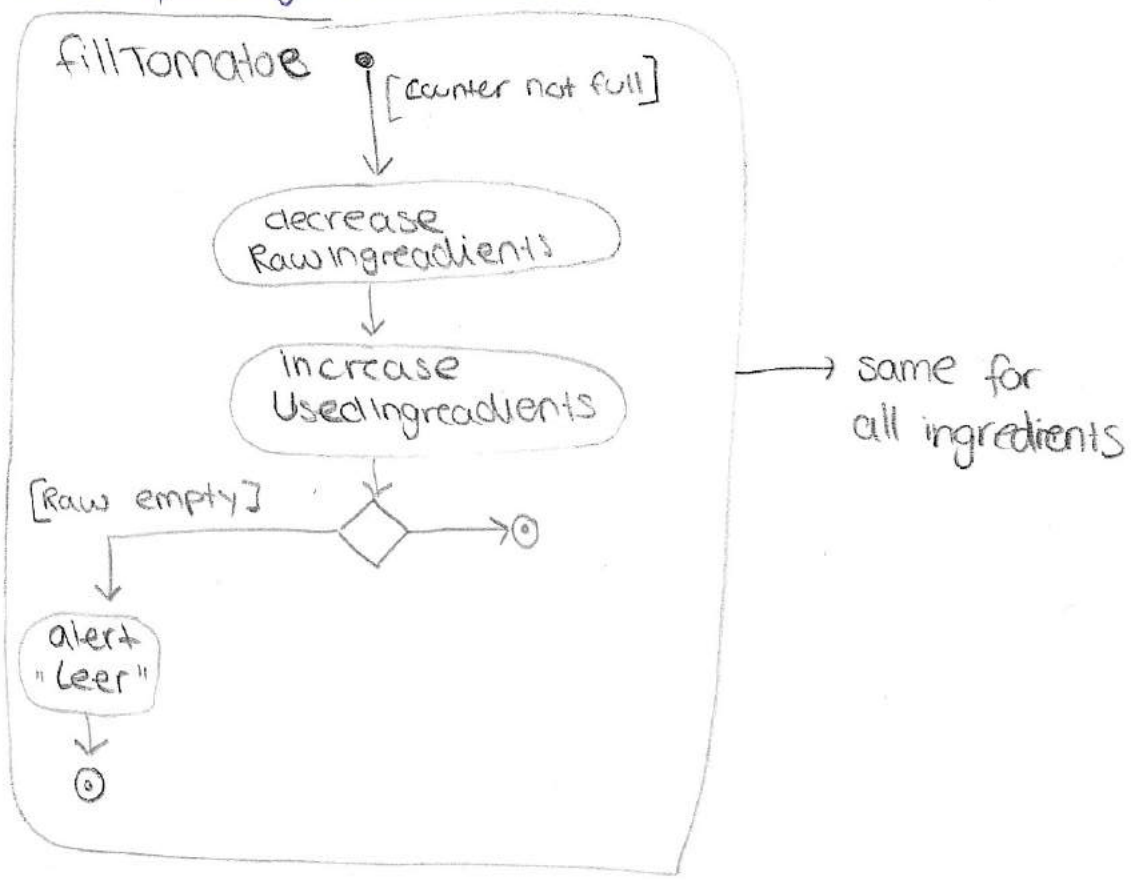
Activity Diagram: Main (5)



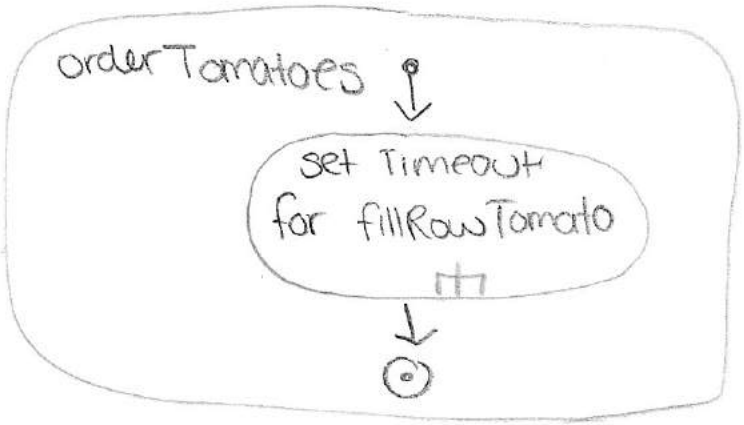
Activity Diagram: Main ⑥

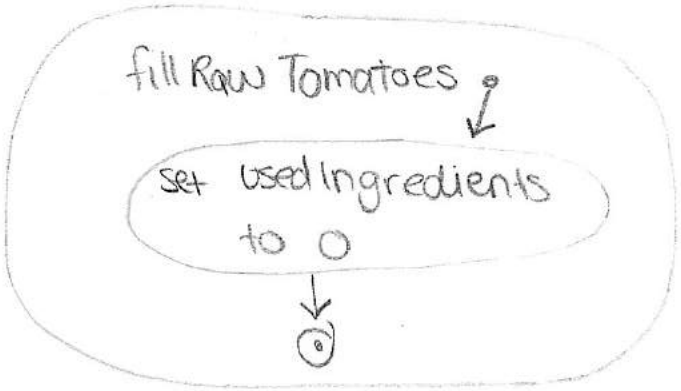


Activity Diagram : Main ⑦

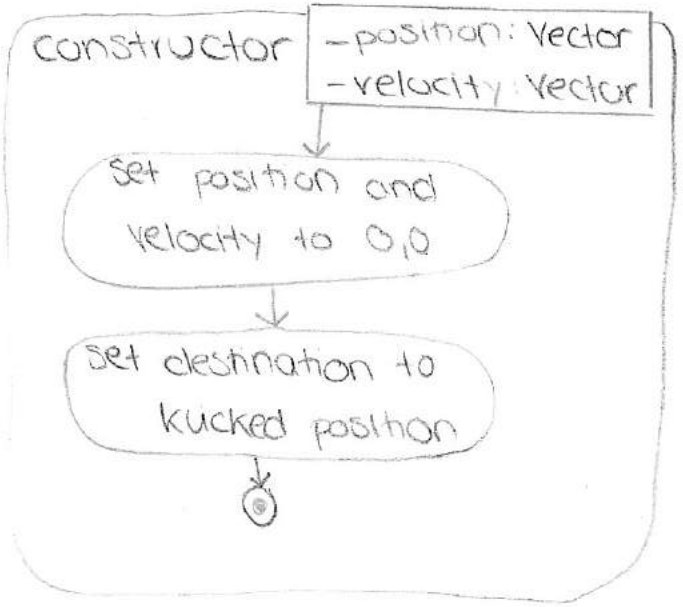


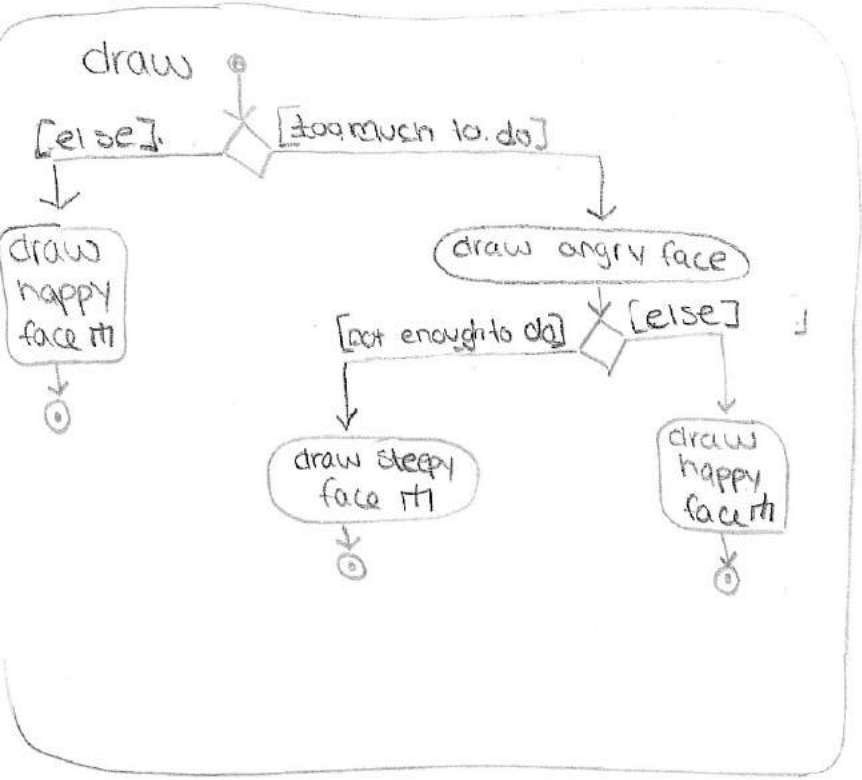
→ same for all ingredients

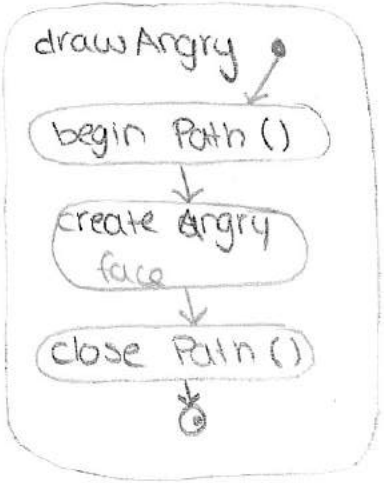


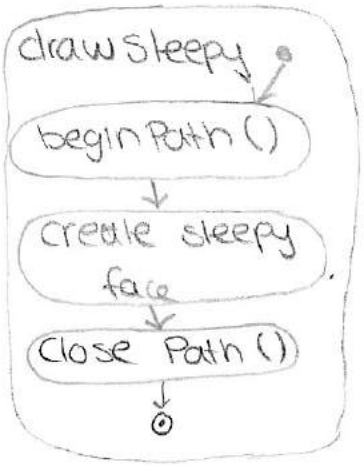


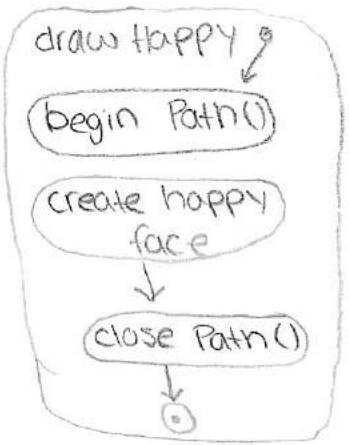
Activity Diagram: Human



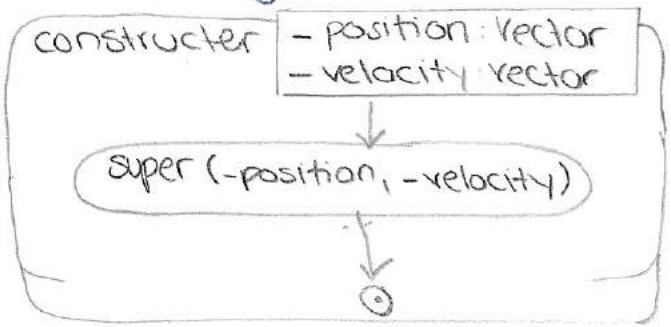




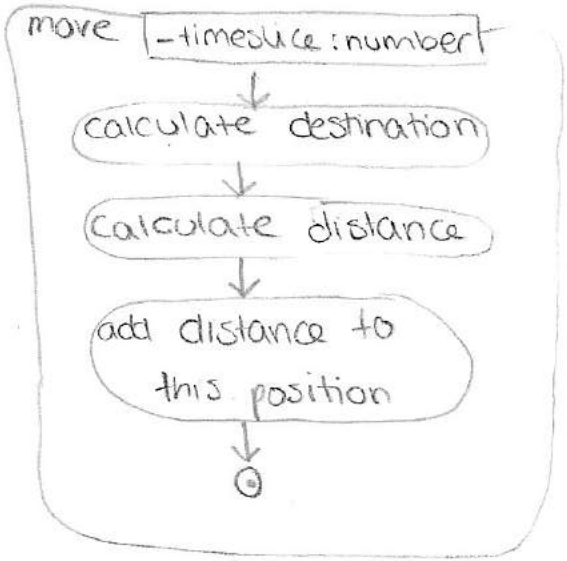




Activity Diagram: Worker

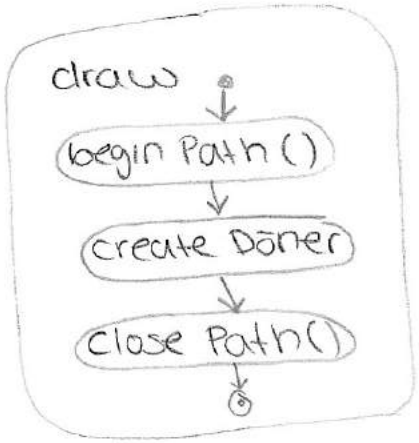


(noch zu Human)

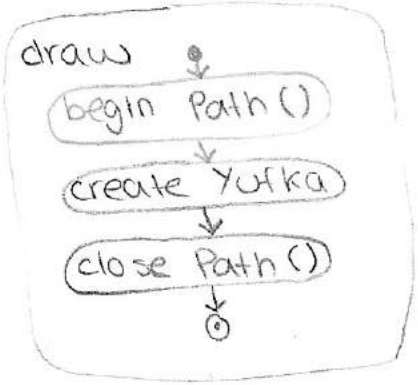


Activity Diagram: Customer

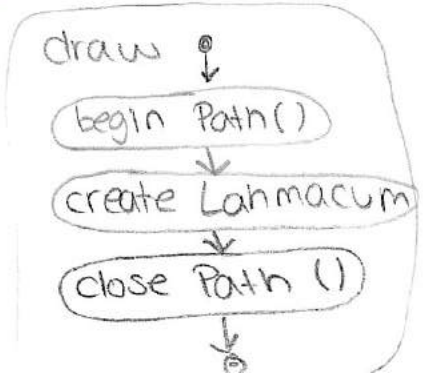
Activity Diagram : Döner



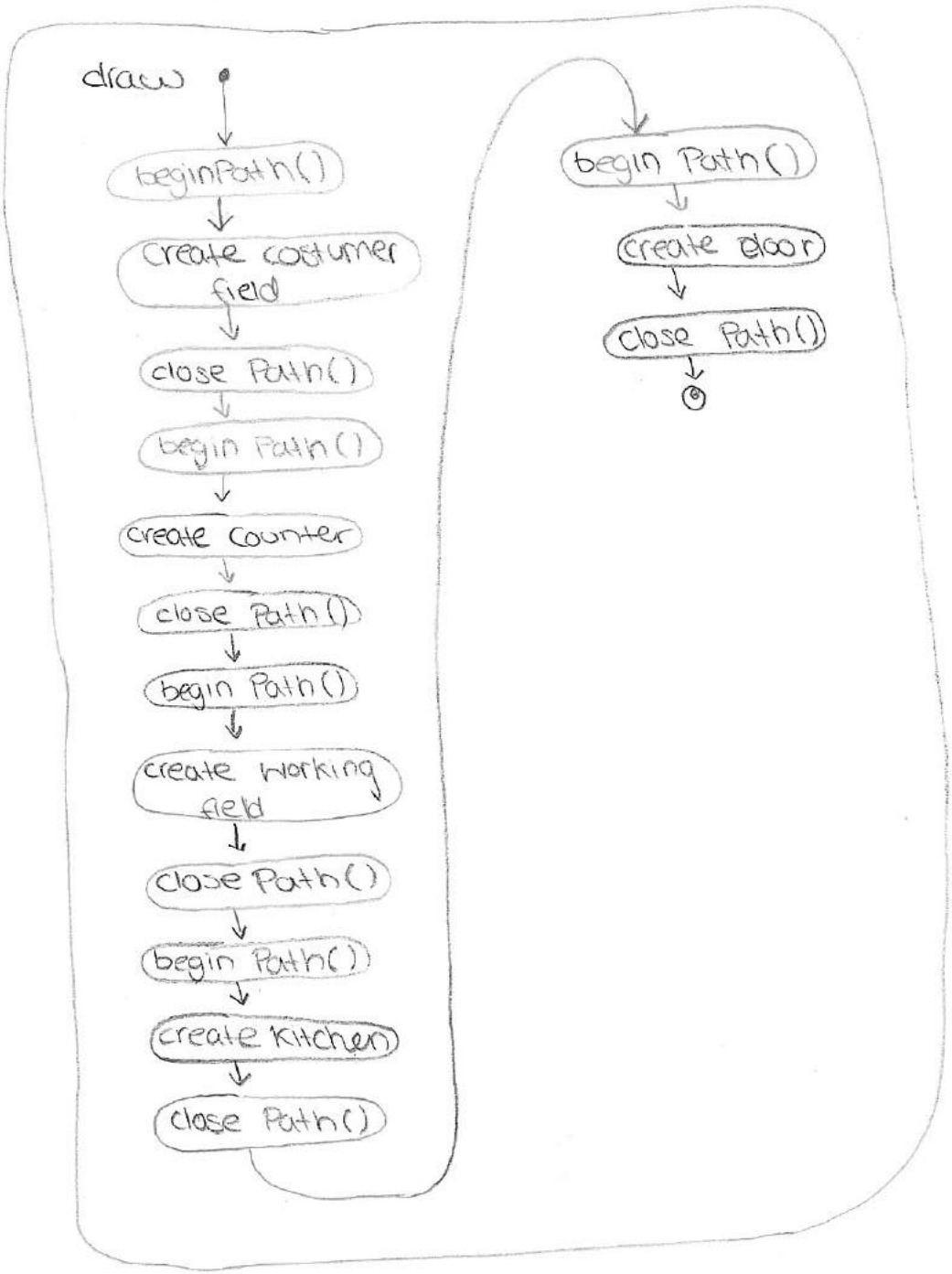
Activity Diagram: Yufka



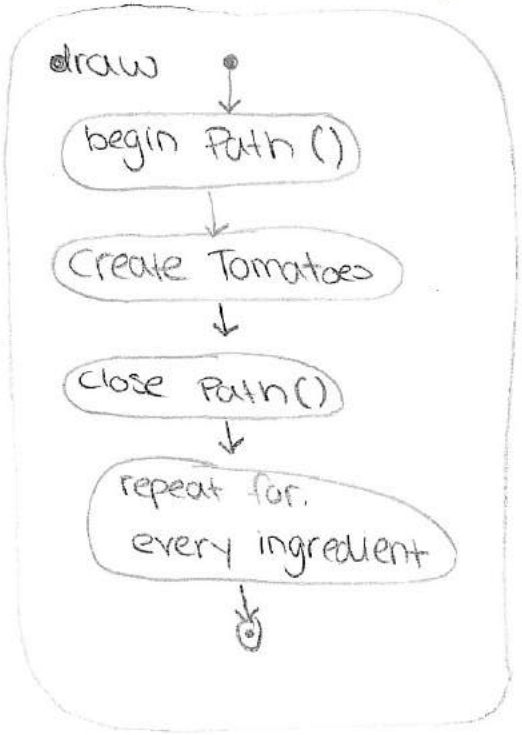
Activity Diagram: Lahmacun



Activity diagram: Kebab-House



Activity Diagram: Ingredients



Activity Diagram: Moveable