Drops

Stage n: 6

Realizado por:

48157, João Reis

52315, Sofia Rocha

52748, Xavier Pacheco

53657, Carolina Lopes

Lab class Nº P1

Group Nº 8

Professor:

Teresa   Romão

Problems pointed out by the heuristic evaluators

Nº	Problem	Our Justification/Solution
1	It's not possible to cancel an order.	Although we didn't think it was necessary for prototype sake, it is a relevant concern and we decided to add the feature to the final version.
2	It's not possible to cancel or edit a credit card.	In the context of the prototype we believe it is an unnecessary feature. Furthermore, if we required actual credit cards we would implement a verification algorithm to validate their authenticity, making the edit button completely pointless.Additionally, we think canceling cards wasn't required at this stage in the prototype.
3	Label of an attribute isn't written in the best way.	There was an extra "_" that we missed. It is fixed now.
4	At the beginning of the order the starting price is "money".	We missed this detail. It is fixed now. We changed the initial value for 0 and added the "€" sign.
5	The terminate order button hasn't feedback.	We removed the button. Indeed it wasn't doing anything. It was there because we use the same model for many screens and it was a typo on our end.
6	Stage of the process isn't necessary.	We politely disagree. We believe it's a nice piece of information for the laundry employees to know in which state the order is in at all times. We think it is a way to avoid misunderstandings because in our prototype the washing/drying/ironing process is just clicking buttons. In the real world not so much.

7	Pickup and Drop off haven't got feedback.	We don't agree with this problem, because on the app there is a message explaining that the containers are unlocked. Also, the containers have a led light that turns green and are physically unlocked, so the distributor knows if the operation succeeded.Of course we cannot simulate this in the prototype.
8	Types of data from credit cards allow insertion of invalid characters.	We agree. We have limited the input types and return an error with invalid characters cases.
9	It's possible that the email of the customer can be a numeric value.	We agree and then we limit the input types and return an error with invalid characters cases.

Problems that we have found

We added a cancel button to cancel the edit process. We didn't have one because, for the heuristic evaluation, we accepted all values of change. It was pointed out by our evaluators this wasn't the best idea, even for a prototype. Thus we decided to add constraints and minimize the error possibilities and, to this effect, we decided to add a cancel button in all Edit Screens.

Our idea is very physical, as in, there are a lot of moving parts in the real world, for example, the driver location, the structures location, the container locking and unlocking, the washing process, etc. This isn't easy to prototype and gets especially hard if we can't set up a theatrical presentation in class. So, in order to minimize this, we decided to take some creative liberties to emulate these behaviours all inside our app. For example, we ditched the map we had in the Marvelapp prototype for a set of cards, each one representing a location.

We also modeled the pickup and drop off process as close to reality as possible, taking the feedback from user evaluation, i.e we unlock the containers (and assume the user physically saw them unlocking) but we also let the container to remove linger in the screen until the user exits the menu. In the real world the app would communicate with the structure and "know" if the container has been removed.

We also decided to create one app instead of three, in order to emulate a distributed system, all offline.

We also found another problem with the alignment of the phone number on the user profile screen and we corrected it.

One thing that we wanted to do in our prototype was the pre selection of a credit card. We tried to do so, but unfortunately, we didn't make it, because of some bugs.

Final Remarks

On an endnote our app was made with Android Studio, with Java and XML. We implemented a MVVM architecture with support from the Jetpack compose Library, as well as the Room Library, to store offline data.

Our website is available here : https://drops-info.netlify.app/