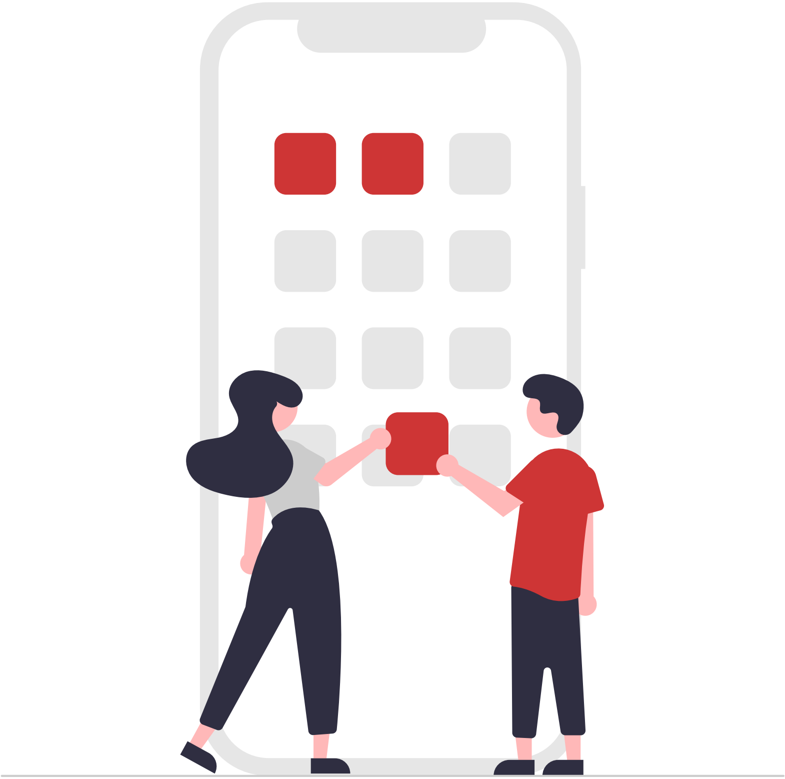
ordine



Stage Four

CPSC 481 Team J

Edward Mah, Minh Hang Chu, Nicolas Urrego, Yunfan Yang

TA - Hessam Djavaherpour

Updated Project Idea

We named the Reserving and Ordering Application to 'ordine'. The System we are designing is still a mobile application that allows customers to make restaurant reservations by selecting date, time, number of people, and seat preference. Customers can sign in as a guest or log in to their accounts. Customers can view a list of restaurants that are grouped by categories. A restaurants information such as address, distance away, rating, review, images, etc. is displayed when clicked upon. Customers can preview restaurants menus to decide if they want to make a reservation. After making a reservation customer can preorder food items and add other accounts/people to a reservation. Customers are also able to write reviews for a restaurant.

List of user tasks

Horizontal

- User can browse, search, and filter restaurants
- User can add restaurant to favourites
- User can swipe horizontally to view more restaurants of the same category
- User can remove a restaurant from favourites
- User can edit a reservation
- User host can cancel a reservation
- User host can add other accounts to the reservation via email
- User host can remove accounts from the reservation
- User can view list of reservations
- User can view restaurant information by clicking on it
- User ca swipe horizontally on restaurant's menu to view menu categories
- User can click on the menu category
- And be instantly scrolled down to that recovery
- User can add a review to a restaurant
- User can give a restaurant a star rating out of 5
- User can view menu prior to making a reservation
- User can view menu when preordering
- User can view dish ingredients

Vertical

- User can make a reservation
  - User can click on the date input to have a calendar view
  - User can swipe vertically to view the dates
  - User can swipe vertically to select the time for the reservation
  - User can swipe vertically to select the number of people for the reservation
  - User can swipe vertically to select seat preference in the restaurant
- User ordering food
  - User can add a dish to the food cart
  - User can modify dish before adding it to the food cart
  - User can increase the quantity of a dish object before the adding it to the cart
  - User can remove a dish object from the food cart
- User paying for food
  - User can choose method of payment and insert credit card details
  - User can insert credit or debit card information
- User can sign in
  - User can sign in as a guest or sign in account
  - User can create an account
  - User can log-out from account
  - User can view and edit information about account

Reflection

What went well with this stage is that we learned how to create a high-fidelity prototype using Figma and using it together to bring our low fidelity prototype to life. We also learned how to do heuristic evaluation. The heuristic evaluation process allowed us to identify many problems with our app. After ranking the problems, we could find the problems that needed to be solved most. Also by looking at current designs such as Skip the Dishes or Doordash we could get ideas of how we should design ours.

What went wrong with the stage is that we tried to implement certain functionalities that couldn't be implemented in Figma. Such as when browsing a restaurant's menu, the user can scroll horizontally to view and select a food category. When a category is clicked then the user would be instantly scrolled down to that category but in Figma we could not implement that.

Heuristic Evaluation Process and Findings

From out heuristic evaluation process we found a few things wrong with our prototype and focused on the problems that were ranked highest.

Consistency and standards

The first thing we found was that originally, we had a few of our buttons are different color to the rest of our program, to emphasize it to the user. However, after doing the evaluation we thought it affected the consistency and standards negatively running the aesthetic of the app. So, we changed the buttons to what every other button in an app looks like.

Help and documentation

Even though are app should be self-explanatory to use we do not have a help section in case a user does not know how to access parts of the app or needs to contact us for help. So, in the account page of the user, we added a help section in the settings. This will help with the help and documentation of our app.

Error Prevention

Since being able to make reservations is one of our app's main functions, then if the user is at the reservations page and happens to accidentally click the 'X' button which is the delete reservation button then the reservation is will be deleted. So we added a popup message that reassures the user if they really want to remove the reservation.

Appendix

GitHub repository: CloudyYoung/cpsc-481-team-j: Human-Computer Interaction I - Fall 2020 (github.com)

Portfolio: Home | Team J (cloudyyoung.github.io)

Heuristic Evaluation

Evaluator 1

Rule of Thumb	Is this rule being applied? How so?	Is this rule violated? How so?	How can this rule further improve usability, utility and desirability?
1. Visibility of system status	Yes. Such as when a payment is made and when the systems asks if u want to delete a reservation	When trying to order there without making a reservation first users are unable to know that they have to make a reservation first	When users know the current system status, they learn the outcome of their prior interactions and determine next steps. Predictable interactions create trust in the product as well as the brand.
2. Match between system and the real world	The navigation bar. Has the symbols that users should be familiar with	The icon of date has a calendar for calendar view in reservation but it is also the same icon for the reserve section in the nav bar	It's easier for users to learn and remember how the interface works. This helps to build an experience that feels intuitive.
3. User control and freedom	Almost every page has a clear back button	Users who may accidentally press a icon on the navigation bar Have no clear back button to go back instead they have to click the nav bar again	Exits allow users to remain in control of the system and avoid getting stuck and feeling frustrated.
4. Consistency and standards	Has the typical login page along with login in with third party services. Consistent color design most of the time	Some buttons such as the buttons Payment are different color to the rest of the system or not corresponding with what they usually are	Users do not have to wonder whether different words, situations, or actions mean the same thing.
5. Error prevention		A undo button when an item is added to cart would be helpful instead of having to go to cart to remove. There are no popups in case someone wants to delete something.	eliminate error-prone conditions

6. Recognition rather than recall	By adding the food item image in the cart as well users do not have to remember what the name of the food is. The recent searches also helps		Interfaces that promote recognition reduce the amount of cognitive effort required from users.
7. Flexibility and efficiency of use	Having a favorites and recent searches for users. A filter system for the restaurant view	No save filter option for users if they want to filter the same way again	Flexible processes can be carried out in different ways, so that people can pick whichever method works for them.
8. Aesthetic and minimalist design	The restaurant browser menu is designed with only the important parts needed	When a restaurant is clicked the information presented is a lot.	Interfaces should not contain information which is irrelevant or rarely needed.
9. Help users recognize, diagnose an d recover from errors	No option to clear all current filters if needed.	Even with the error prevention popups if a user deletes there is no quick undo. Users must reserve again	These error messages should also be presented with visual treatments that will help users notice and recognize them.
10. Help and documentation	The system is self-explanatory	No help page	Help and documentation content should be easy to search and focused on the user's task to ensure usability

Evaluator 2

Rule of Thumb	Is this rule being applied? How so?	Is this rule violated? How so?	How can this rule further improve usability, utility and desirability?
1. Visibility of system status	Users get alert box and confirmation message for some actions	User can't see confirmation of booking table, processing order and transition between screens are not very smoothly	Adding notification alerts, messages box to inform the system is processing their requests
2. Match between system and the real world	The system uses the languages used by user, no special terminology. The navigation of workflow is clear and logical		
3. User control and freedom	There is return options on some screens.	There is no direct way to exit to browse directly during ordering process.	Adding button allows user to exit right away in the middle of process by going back to previous page.

4. Consistency and standards	Many actions, situations, instances of words in the application are consistent.	The font and size of the application are quite big.	Change sizes of text and color to make it consistent rhougout all pages
5. Error prevention	There are a few alert pop-ups to confirm users action	No errors were encountered. The workflow is very linear and following a same path everytime.	Adding error messages alert (unable to add item, unable to process payment, etc)
6. Recognition rather than recall	Using icon and button makes it clear for user what it means, whih helps reducing the amount of memorization the user has to do. User also do not need to remember details from precious page.	Some icons are not clear in what it indicates and some pages are not clear what user can click	Using different colors instead of only red and black, allowing user to recognize buttons and stay in the workflow
7. Flexibility and efficiency of use	There is some shortcut to remember payment methods and favourite restaurants.	There are no currently options for speeding up processes of ordering, logging in, payment	The system can add saved log in credentials, use last orders, use last method payments.
8. Aesthetic and minimalist design	Most pages achieve minimalist design and please aesthetic. Some pages display only relevant information which help to better focus on what task is supposed to be completed on some pages and make application less overbearing	Some pages do not achieve minimalist design and fairly busy. Some page is not consistent with general design	Filter our information that are most important, make font and size consistent to make page design minimalist.
9. Help users recognize, diagnose and recover from errors	There are a few alert box messages	There are not many error messages in the current application	FAQ section for questions and Help Page would be helpful. User should be able to send questions, concern, bugs for anything to the team.
10. Help and documentation	Buttons are quite clear and restrictly relavant to pages purposes	There is no help indicator thoughout the application to explain fuctionalities.	Adding help indicator to help user stay in the workflow and application and they dont need to navigate elsewhere.

Review 1 Rankings (Left is most important)

- Evaluator 1: 4 9 5 10 6 2 1 3 7 8
- Evaluator 2: 6 4 1 10 5 9 7 8 3 1

Review 2 Rankings (left is most important)

- Evaluator 1: 4 8 2 3 7 1 6 5 9 10
- Evaluator 2: 8 6 2 7 4 10 1 9 5 3