A FOOD ORDERING WEBSITE

7/12/23

Aryan Haddady,
Cheriton School of Computer Science

UNIVERSITY OF WATERLOO | FACULTY OF MATHEMATICS

Photo credit: @bruce.digital

Motivation

- Designing a functioning website involves paying attention to many small details that are discovered during development

- Practicing requirement specification is helpful

The First Thing

- A Profile
  - Register an account
  - Sign into it

Signup Page

- Fields
  - Full Name
  - Email (used as username)
  - Phone Number
  - Password
  - Repeat Password

What can go wrong?

- Username is already used
- Solution:
  - The system must check whether the username already exists instantly and inform the user if it does.

What can go wrong?

- Entered username is not a valid email address
- Solution:
  - The system must check whether the email is valid via regex and inform the user instantly if it isn’t

What can go wrong?

- Password is not strong
- Solution:
  - The system must tell the user that the password must be
    - 8 characters long
    - Contain at least one number
    - Contain at least one capital letter
    - Contain at least one small letter
    - Contain at least one special character ([,], {, }, &, *, $)
  - The system must check validity instantly and inform the user in case the password is invalid

What can go wrong?

- *Password* and *Repeat Password* don’t match
- Solution:
  - The system must check this instantly and inform the user in case they differ
  - It shouldn’t happen with each entered keyboard as it ruins he UX

What can go wrong?

- *Phone number* is not valid
- Solution:
  - The input for phone number must have three sections separated by hyphen
    - Three digits of area code
    - Three digits of central office code
    - Four digits of line number

What can go wrong?

- User has entered the wrong email address
- Solution:
  - Confirmation Email

- What if the user with the unintended email address deletes the email (as he didn’t want to signup) and decides to sign up later?
- Solution:
  - Confirmation Email only works for two hours. If not confirmed, all info is lost.

How can we make this process more enjoyable?

- Knowing that one or multiple fields were entered wrong in a form after submitting the form is annoying
- Solution:
  - Instant Validation

If everything is ok ...

- It’s time to log in. Let’s Redirect user to login page

Finalizing the Order

- The user wants to finalize the order and receive his orders
- When can we access this?
- Answer: The cart is not empty.

Preconditions

1. The user must be logged in.
    - What if he isn’t?
        - Solution: Redirect him to login page

- The user must have added at least one item to his cart.
    - What if he hasn’t?
        - Solution: Redirect him to the page that shows him his cart.

What should the user be able to do?

- Choose how he wants to receive the order
  - Pickup
  - Delivered to him

He chooses the pickup option

- What should he know?
  - The address of the restaurant. Maps services should be used.
    - Google’s Map service is pretty good.
  - When the order will be ready to be picked up.
    - The restaurant itself must specify this.

He chooses the delivery option

- What should the website know?
    - The address the user wants his order to be delivered to.
        - New address
            - Has its own form and validation criteria
            - User is asked to save this for future orders
        - Previously saved address

A couple of features here

- User must be able to save the entered address
  - How many addresses can he use?
    - Answer: 20 is enough (probably??)
- User must be able to delete previously saved passwords

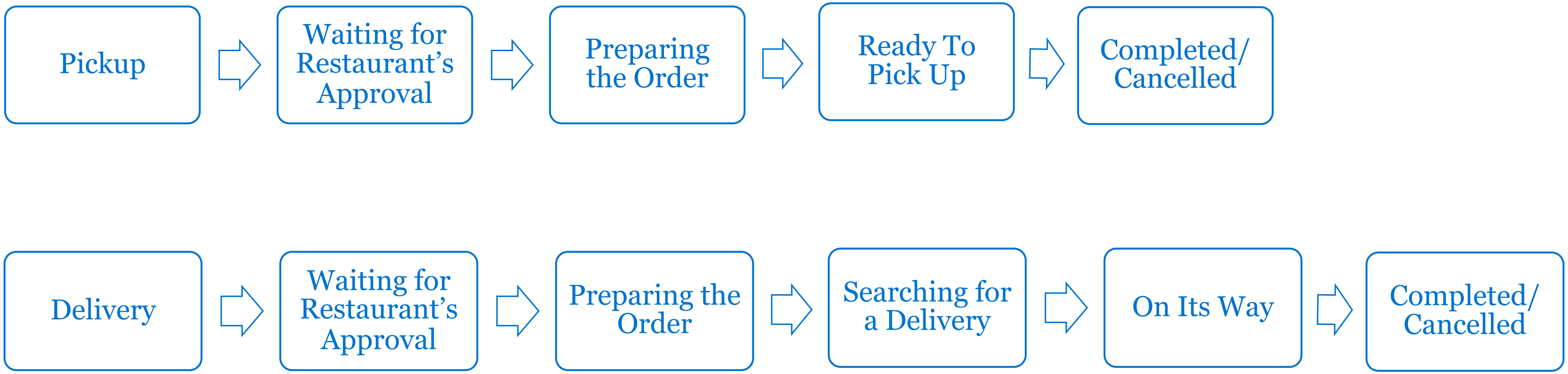
Now finalizing …

- What if user’s balance is less than cart’s price?
  - Solution: He must be informed and asked to add to his balance.
    - There must be a page to add balance to his account.
    - After adding the balance he must be redirected back to the finalize order page.

Everything was successful. What now?

- The user is redirected to the list of orders page.
- Order’s status must be **Waiting for Restaurant’s Approval**

Order Status Changes



Pickup → Waiting for Restaurant's Approval → Preparing the Order → Ready To Pick Up → Completed/ Cancelled

Delivery → Waiting for Restaurant's Approval → Preparing the Order → Searching for a Delivery → On Its Way → Completed/ Cancelled

What about security?

- Security is always important.
- It is especially important here as we are dealing with money.

What are user information we need to protect?

- Password
    - Solution:
        - Store hash results of user passwords on the database rather than the passwords
        - Encrypt password before sending it to the server

What are user information we need to protect?

- User’s location information
  - Solution:
    - Location is not stored in the database.
      - Server obtains the information and throws it away after doing the location-based sorting operations
    - Encrypt location information before sending it to the server.

What are user information we need to protect?

- User’s card information
    - Solution:
        - Encrypt credentials before sending them to the server
        - Replacing first 12 digits of card number with * (E.g., **** **** **** 1111)

Balance issues

- Balance must not be lost
- Solution:
    - Have backup from balance information
    - Only decrease balance after when the restaurant has approved the order
    - Pay attention to the refunding process if the order is cancelled
    - Have a support teams that answers support queries as fast as possible

UNIVERSITY OF WATERLOO

FACULTY OF MATHEMATICS

Thanks!!