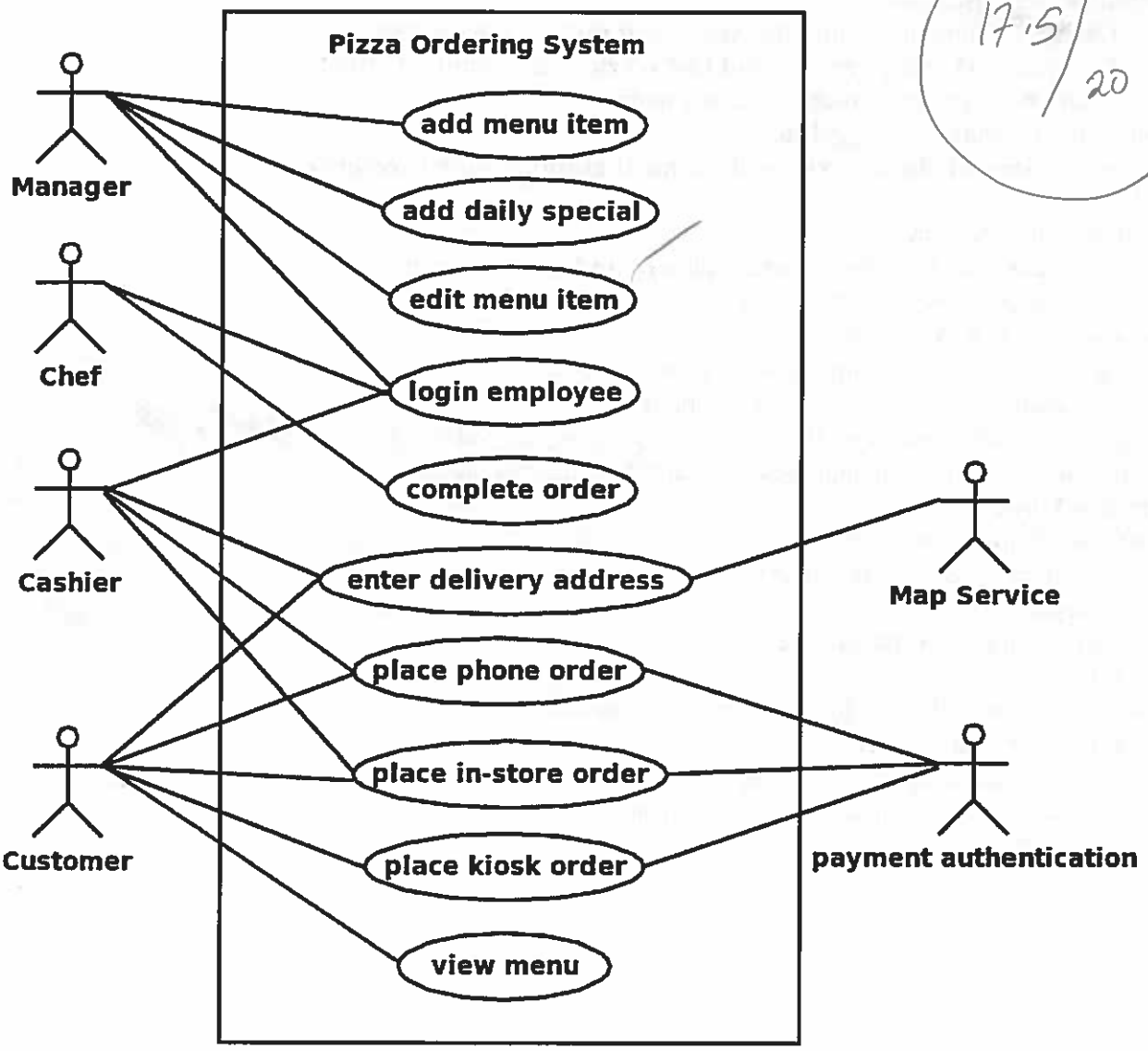
h

CS414: A2
Bryce Myers, Sky Egelhoff, Troy Cogburn and Nate Gillard

17.5/20



login employee not described. -1

Use case diagram 5/5

Use case descriptions

Order including payment 5.5/7

Menu: create, modify, special 5/6

Complete 1.5/2

Other: login, user management: 0.5/1

Use Case: Place Phone Order
Scope: Pizza Ordering System
Level: user goal
Primary Actor: Customer and Cashier
Stakeholders and interests:

- **-Cashier: Wants accurate, fast entry and no payment errors.**
- **-Customer: Wants purchase and fast service with minimal effort.**
- **-Chef: Wants easy to read, accurate order.**

Preconditions: Cashier is logged in.
Success guarantee: Order is saved and payment authorization is recorded.
Basic Flow:

1. ~~Customer calls the store.~~
2. Cashier ~~answers and~~ launches **"Enter delivery address"** use case.
3. Cashier takes order from customer and adds items to the order.
4. ~~System adds items to the order.~~
5. Cashier confirms with customer and submits the order.
6. System calculates price and requests payment information.
7. Cashier enters customer payment. What kinds of payments are accepted? how are they processed (-1)
8. If payment authentication approves payment then order is placed.

Alternative Flows:

5a. Customer changes their mind:
    1. Cashier edits/removes item(s).
    2. Repeat step 3.
5b. Customer requests to cancel order
    1. Cashier cancels order.
7a. Customer cannot afford order or wants to cancel order.
    1. Cashier cancels order.
8a. System is unable to approve payment. Why?
    1. Cashier requests another form of payment.
    2. Repeat Step 7.

Use Case: Place In-store Order
Scope: Pizza Ordering System
Level: user goal
Primary Actor: Customer and Cashier
Stakeholders and interests:
- **-Cashier: Wants accurate, fast entry and no payment errors.**
- **-Customer: Wants purchase and fast service with minimal effort.**
- **-Chef: Wants easy to read, accurate order.**

Preconditions: Cashier is logged in.
Success guarantee: Order is saved and payment authorization is recorded.
Basic Flow:
1. ~~Customer approaches the register ready to order~~.
2. Cashier takes order from customer and adds items to the order.
3. ~~System adds items to the order~~.
4. Cashier confirms with customer and finalizes order.
5. System calculates price and requests payment information.
6. Cashier enters customer payment. see comment on previous page
7. If payment authentication approves payment then order is placed.

Alternative Flows:
4a. Customer changes their mind:
1. Cashier edits/removes item(s).
2. Repeat step 2.

4b. Customer requests to cancel order
1. Cashier cancels order.

6a. Customer cannot afford order or wants to cancel order.
1. Cashier cancels order.

7a. System is unable to approve payment.
1. Cashier requests another form of payment.
2. Repeat Step 6.

steps that do not involve interactions with the system should not be included in your basic steps (-0.5)

Use Case: Place Kiosk Order
Scope: Pizza Ordering System
Level: user goal
Primary Actor: Customer
Stakeholders and interests:

- **-Customer: Wants purchase and fast service with minimal effort. They want an easy to use, fast system.**
- **-Chef: Wants easy to read, accurate order.**

Preconditions: Menu is displayed without previous orders on screen.
Success guarantee: Order is saved and payment authorization is recorded.
Basic Flow:

1. Customer ~~activates the kiosk.~~
2. Launch **"Enter Delivery Address"** use case.
3. Customer submits item(s) to order.
4. ~~System keep add items to order.~~
5. Customer confirms finalized order.
6. System displays calculated price and requests payment information.
7. Customer enters payment.
8. System verifies payment and places the order.

Kiosk orders can only be delivered?

Alternative Flows:

3a. Customer views the menu.
    1. Launch **"View Menu"** use case
5a. Customer ~~changes their mind, and~~ removes item:
    1. ~~System removes item.~~
    2. Repeat step 5.
5b. Customer cancels order.
    1. The order is canceled.
7a. Customer cancels the order.
    1. The order is canceled.
8a. System is unable to verify payment.
    1. System requests another form of payment.
    2. Repeat Step 7.

Use Case: Add menu item
Scope: Pizza Ordering System
Level: sub-function
Primary Actor: Manager
Stakeholders and interests:
-Manager: Wants to be able to easily add menu items and have them appear to customers.
-Customer: Wants to see all available items on the menu.
-Cashier: Wants to see all available items on the menu, so they can complete orders.
Preconditions: Manager is logged in.
Success guarantee: Menu item saved and added to menu.
Basic Flow:
1. Manager requests to add item to menu.
2. System requests name and price of item.
3. Manager enters name and price of new item. and saves
4. ~~System adds item to the menu.~~
Alternative Flows:
*a. At any time Manager changes mind and cancels the add.
1. System returns to main screen.

how and when is menu created? (-0.5
What are your assumptio

duplicate items allowed? (-0.5)

Use Case: Edit menu item
Scope: Pizza Ordering System
Level: sub-function
Primary Actor: Manager
Stakeholders and interests:
-Manager: Wants to be able to easily edit menu items and have them appear to customers.
-Customer: Wants to see all available items on the menu.
-Cashier: Wants to see all available items on the menu, so they can complete orders.
Preconditions: Manager is logged in.
Success guarantee: Menu item saved and updated to menu.
Basic Flow:
1.Manager selects existing item to edit.
2. System displays the given item's information.
3. Manager enters new name or new price.
4. ~~System edits item on the menu~~. Saves edited item
Alternative Flows:
*a. At any time Manager changes mind and cancels the edit.
1. System returns to main screen.

Use Case: Add daily special
Scope: Pizza Ordering System
Level: sub-function
Primary Actor: Manager
Stakeholders and interests:

-Manager: Wants to be able to easily add a special menu items and have them appear to customers.

-Customer: Wants to see all the daily special on the menu.

-Cashier: Wants to see daily specials on the menu, so they can complete orders.

Preconditions: Manager is identified and authenticated.
Success guarantee: Daily special item saved and added to menu.
Basic Flow:
1. Manager request to add a special item to the menu.
2. System requests name and price of item.
3. Manager enters name and price of special item.
4. System adds item to the menu.

can the special be an item already on the menu? duplicates?

Alternative Flows:
*a. At any time Manager changes mind and cancels the add.
1. System returns to log in screen.

Use Case: Complete Order
Scope: Pizza Ordering System
Level: sub-function
Primary Actor: Chef
Stakeholders and interests:

-Chef: Wants to be able to easily see when orders have been completed.

-Customer: Wants their order to be accurate and obtained in a timely fashion.

Preconditions: Order has been placed.
Success guarantee: All items have been completed by chef and are ready for the customer.
Basic Flow:
1. Chef marks an order complete
2. System logs the order as completed.

when does the chef select the order to complete (-0.5)

Alternative Flows
1a. Chef cancels the order at customer or manager request.
1. System cancels the order.

Use Case: View Menu
Scope: Pizza Ordering System
Level: sub-function
Primary Actor: Customer
Stakeholders and interests:

- **-Manager: Wants menu to be displayed clearly so customers know what can be ordered.**
- **-Customer: Wants legible and simple understand menu.**

Preconditions: none.
Success guarantee: All menu items have been displayed.
Basic Flow:
1. Customer requests to see the menu.
2. System displays the available items.

Alternative Flows:

Use Case: Enter Delivery Address
Scope: Pizza Ordering System
Level: sub-function
Primary Actor: Customer or Cashier
Stakeholders and interests:

- **-Cashier: Wants quick way to identify the delivery address of an order.**
- **-Customer: Wants easy way to provide address.**
- **-Delivery driver: Wants minimal mistakes in delivery address.**

Preconditions: none.
Success guarantee: All menu items have been displayed.
Basic Flow:
1. Cashier or Customer submits the customer's address to the system.
2. If the system uses a map service to confirm the address is in the delivery area the address is added to the order record.

Alternative Flows:
2a. If the system can't confirm the address is in the delivery area
    1. Cashier informs the customer and the order is canceled.

Seen