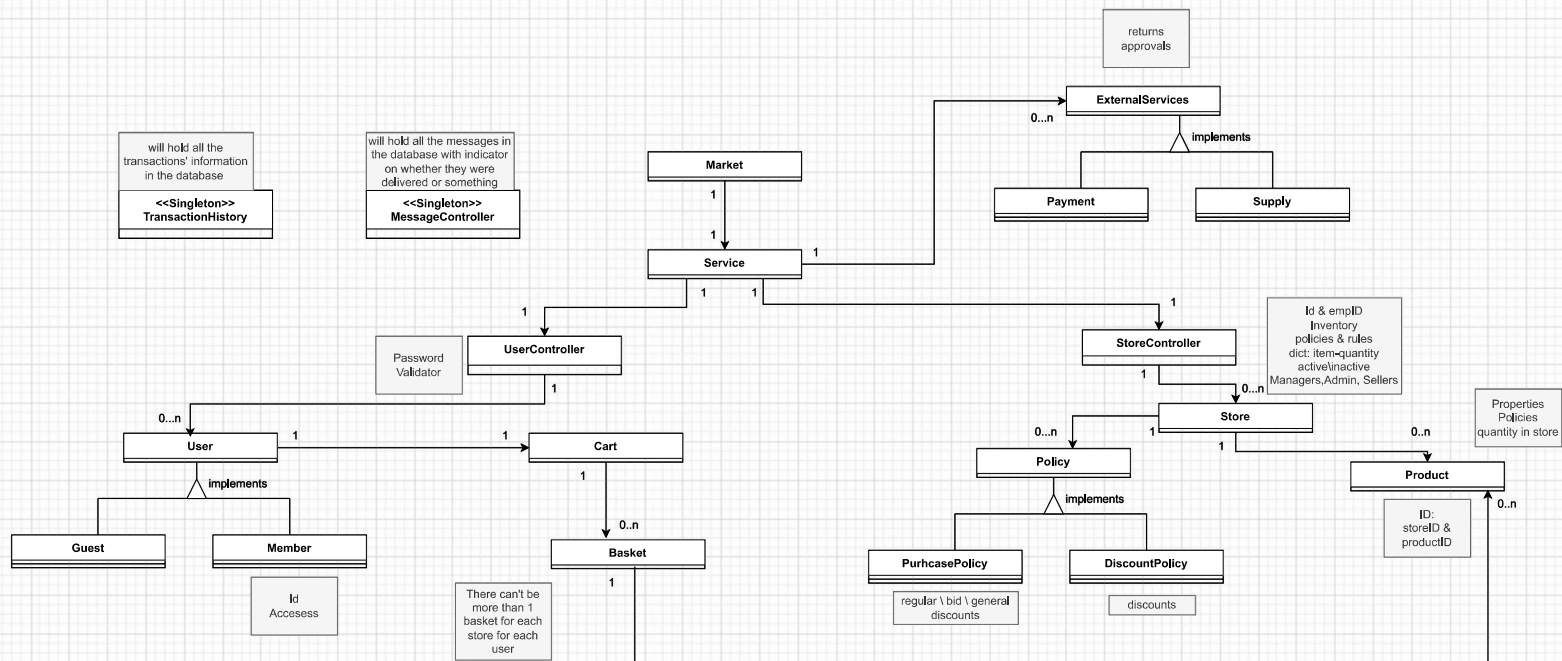
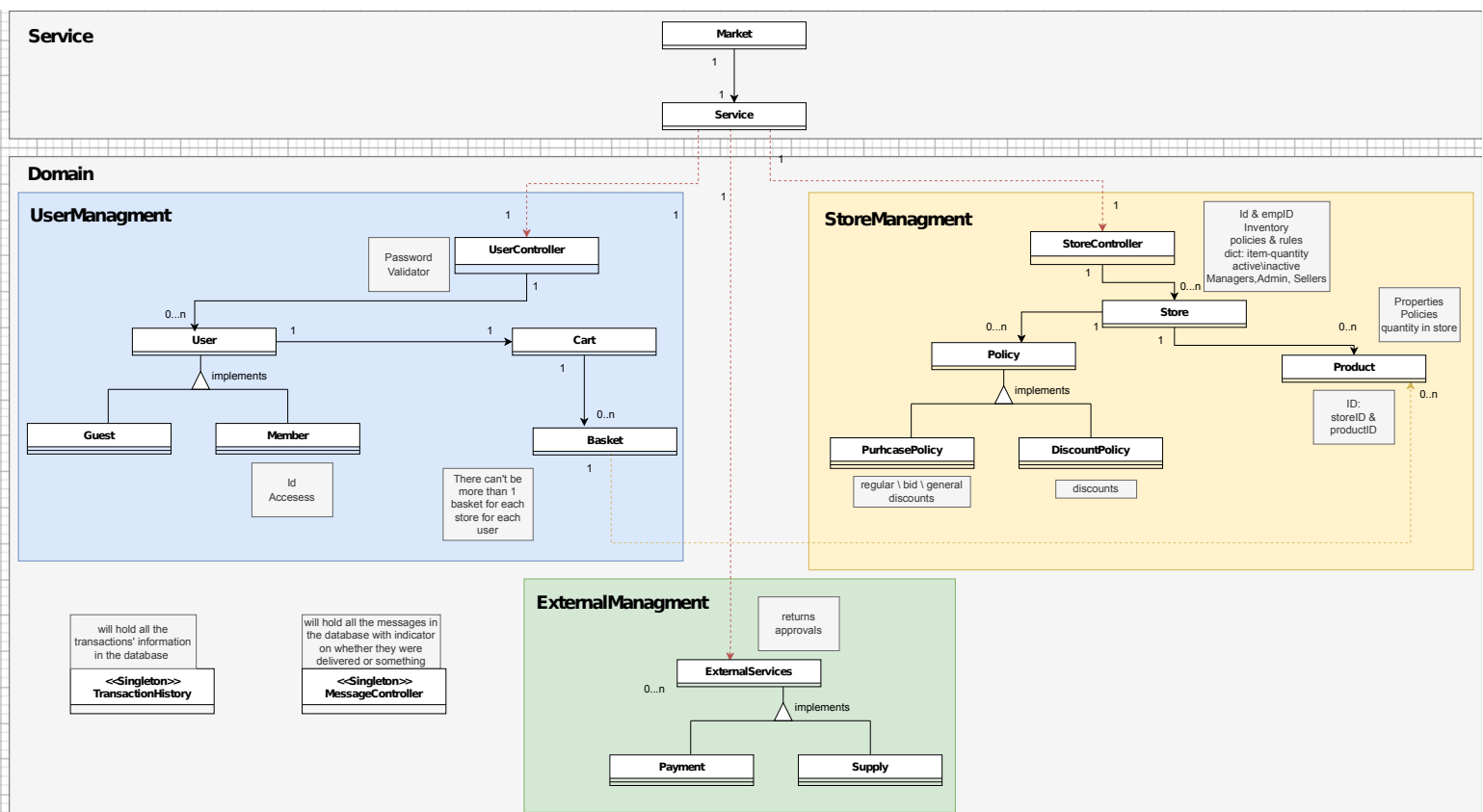


White diagram:



Architecture diagram:



System requirements:

1.1 starting the market system

Actor: Admin

Pre-conditions: only admin has system manager premissions, the given user exists in the database.

Post-conditions: None

Flow:

- An admin logging into the system using his password and username of a system manager type user
- The system checks if the user exists in the database
- The system checks if the user is a system_manager type of user
- The system sends a message to payment and supliment services
- a message is being return from those services to the system
- The system sends a message to the the user that he had connected successfully.

Tests:

- Positive: the admin succeed to log into the system.
- Negative: the system says this user doesn't have system manager permissions

1.2 change/replace/add a connection with external services

Actor: Admin

Pre-conditions: at least one external service is connected to the system while the system is running.

Post-conditions: None.

a. change:

Flow:

- The Admin will select a connected external service to change and will press the "change" button
- The system will show all the available fields for changing
- The admin will change the selected fields and will press the "save" button
- The system will update the selected fields.

- The admin will receive a message "The connection has updated successfully"

Tests:

- Positive: the selected fields of the external service connection will be updated
- Negative: the selected fields of the external service connection will not update

b. replace:

Flow:

- The admin will select a connected external service to replace and will press the "replace" button
- The admin will choose a second (not connected) external service and will press the "execute" button
- The system will send a connection request to the second external service
- The system will establish a connection with the second external system
- The system will disconnect from the first external system
- The admin will receive a message "The replacement has accomplished successfully"

Tests:

- Positive: the second selected external service will be connected to the system and the first selected external service will no longer be connected
- Negative: the replacement will fail – the second selected external service will not be connected to the system or the first selected one will still be connected to the system.

c. add:

Flow:

- The admin will select a connected external service to add and will press the "add" button
- The system will send a connection request to the selected external service
- The system will establish a connection with the selected external system
- The admin will receive a message "The connection has established successfully"

Tests:

- Positive: the selected external service will be connected to the system
- Negative: the selected external service will not connect to the system

1.3 payment

Actor: System

Pre-condition: an unpaid transaction details exist in the database, and a payment system is connected to the the market system.

Post-condition: None.

Flow:

- the system checks the connection to the payment system
- The system recive a message from the payment system it's connected to the system
- The system sends to the payment service the transaction details
- The system recive from the payment system a message that the payment has set successfully

Test:

- Positive: The system connects to the payment system successfully
- Negative: The system doesn't recive a message back from the payment system and the connection fails.

1.4 suppliment

Actor: System

Pre-conditions: at least one product suppliment system is connected to the market system.

Post-conditions: None.

Flow:

- The system checks the connection to the suppliment system
- The system recive a message from the suppliment system it's connected to the system
- The system send to the suppliment system a package details and customer details
- The system recive from the suppliment system an approval for the suppliment of the given package

Tests:

- Positive: the system receives an approval message for the supplement of the given package from the supplement system.
- Negative: the system doesn't receive an approval message from the supplement system about the given package.

1.5 real time notifications

Actor: System

Pre-conditions: the shop owner is logged into the system.

Post-conditions: None.

Flow A:

- Customer buys a product from the shop owner's shop
- The system fetch all the logged shop owners
- The system sends an notification to each of them which says "a customer bought a <product name> from your store!"

Tests:

- Positive: each shop owner receives a notification to his screen.
- Negative: at least one shop owner of this shop doesn't receive a notification from the system about this transaction.

Flow B:

- The shop is being shut down
- The system fetch all the logged shop owners of this specific shop
- The system sends an notification to each of them which saying "the shop has successfully closed"

Tests:

- Positive: each shop owner receives a notification to his screen.
- Negative: at least one shop owner of this shop doesn't receive a notification from the system about the shop being shut down.

Flow C:

- A shop has opened
- The system fetch all the logged shop owners
- The system sends an notification to each of them which saying "the store has successfully opened".

Tests:

- Positive: each shop owner receives a notification to his screen.
- Negative: at least one shop owner of this shop doesn't receive a notification from the system about the shop being opened.

Pre-conditions: The store has at least 2 shop owners, the removed shop owner as no nominators under it.

Post-conditions: None.

Flow:

- The system remove the nomination of the shop owner
- The system sends to the former shop owner a message "your nomination in <name of the shop> has been removed".

Tests:

- Positive: the former shop owner receives a notification to his screen.
- Negative: the former shop owner is no longer a owner of this store but no notification was sent by the system.

Actor: User

Pre-conditions: At least 2 users connected to the system

Post-conditions:None

Flow:

- User 1 sends a message to user2 using the system
- The system receives the message and checks if user2 is connected
- The system send a notification with the message to user2

Tests:

- Positive: user2 will receive the message.
- Negative: user2 will not receive the message.

1.6 Delayed notifications

Actor: System

Pre-conditions: a not logged user with delayed notifications in the database

Post-conditions: the user is logged in and with no delayed notifications in the database

Flow:

- User logging into the system
- The system check if there are any delayed messages belong to this uer in the database
- The system fetch the messages and send them as notifications to the user
- The user recives a notification in his screen

Tests:

- Positive: the user recive his delayed notifications.
- Negative: the user have delayed notification but he doesn't recive them.

2 -Users

a) guest

2.1.1 Guest Visit

Actor: User

Pre-conditions: None

Post-conditions: User defined as a guest.

Flow:

- User enters the market.
- User gets a basket assigned.

Tests:

- Positive: User add products to his basket.
- Negative: User gets an error trying to enter the market.

2.1.2 Guest Exits

Actor: User

Pre-conditions: User is in the market as guest.

Post-conditions: User loses his basket.

Flow:

- User closing the website.

Tests:

- Positive: User adds a product to his basket, exiting the market and then returns to an empty basket.
- Negative: Same as the positive test, but the basket stays the same.

2.1.3 Registration to the system

Actor: User

Pre-conditions: User is defined as a guest.

Post- conditions: a new member created in the system with a new member id and the registration information

Flow:

- A guest user press on the Sign-up button fills the information needed and a new member is created in the system.

Tests:

- Positive: A guest user fills the sign-up info sheet with a unique user-name, and a new member account is created in the system.
- Negative: A guest user fills the sign-up info sheet with an already used user-name, and an error message pops and says "this user name already exists."

2.1.4 Logging in the system

Actor: User

Pre-conditions: User is defined as a guest and has already registered to the system.

Post- conditions: The User status is changed from Guest to Member.

Flow:

- A guest user press on the Sign-in button fills the user-name and password bars and log in his account

Tests:

- Positive: A guest clicks on the sign-in button, fills username and passwords bars and successfully log in to his account.
- Negative: A guest clicks on the sign-in button, fills username and passwords bars incorrectly and receives an error message "User name or password are incorrect".

b) user

2.2.1 Guest Information fetching

Actor: User

Pre-conditions: User defined as a guest.

Post-conditions: User gets the information about the stores and products.

Flow:

- User queries information about certain products or stores.

Tests:

- Positive: User search for hair products and gets related results.
- Negative: User search for a specific store but gets a wrong result.

2.2.2 Guest product search

Actor: User

Pre-conditions: User defined as guest.

Post-conditions: User gets the appropriate result according to his search.

Flow:

- User types a certain product name/category/keyword on the search bar.
- User gets the information about the products.

Tests:

- Positive: User searches for specific brand name t-shirts and the search results shows all the t-shirts available for the brand name.
- Negative: User types a specific brand name t-shirts and the search results shows a different brand

2.2.3 Guest adds product to the basket.

Actor: User

Pre-conditions: User defined as a guest.

Post-conditions: User basket filled with the products he added.

Flow:

- User searches for a product.
- User adding a product by pressing "Add to basket" button.

Tests:

- Positive: User adds a product to his basket, when viewing the basket the product appears.
- Negative: Same as the positive, but the product doesn't add to the basket

2.2.4.a Guest views the basket

Actor: User

Pre-conditions: User defined as a guest.

Post-conditions: User basket contains all the products he added.

Flow:

- User clicks on the "Basket" button.
- User can view all the products he added.

Tests:

- Positive: User adds a product, then clicks the "Basket" button where the product he added appears.
- Negative: Same as positive, only that the product doesn't appear.

2.2.4.b Guest changes the basket

Actor: User

Pre-conditions: User have at least 1 product in his basket.

Post-conditions: User changes to the basket success.

Flow:

- User press the "Basket" button.
- User removes/changes the quantity of a product.

Tests:

- Positive: User removes a product, then the product doesn't appear in the basket.
- Negative: User changes the quantity of a product, but the quantity stays the same.

2.2.5.a Guest immediate purchase

Actor: User

Pre-conditions: User basket have at least 1 product that he can buy according to the buying policies.

Post-conditions: User purchase completed successfully.

Flow:

- User clicks on immediate purchase button.
- User fill all the information needed for the payment method.
- User gets a confirmation that his purchase succeeded.

Tests:

- Positive: User adds a product that eligible for immediate purchase, then purchase the product by clicking immediate purchase, then getting a purchase confirmation.
- Negative: User tries to buy a product through immediate purchase, but the product is out of stock.

2.2.5.b Guest bid request

Actor: User

Pre-conditions: User have at least 1 product that eligible for bidding.

Post-conditions: User placed the desired bid successfully.

Flow:

- User choose a product for bidding.
- User place a bid.
- User waits for a response from the shop owner if:

1. Owner accepts the offer – therefore he can buy the product
2. Owner rejects – therefore he cannot buy the product.
3. Owner responds with another offer.

Test:

- Positive: User chooses a product for bidding, the store owner rejects the offer and the user cannot purchase the product.
- Negative: User places a bid, owner accepts the offer but the user can't buy the product.

2.2.5.c Guest immediate purchase with secret discount

Actor: User

Pre-conditions: User basket have at least 1 product that he can buy according to the buying policies.

Post-conditions: User purchase completed successfully.

Flow:

- User clicks on immediate purchase button.
- User enter the coupon code that he possesses.
- A message displayed if the coupon code eligible or not.
- User fill all the information needed for the payment method.
- User gets a confirmation that his purchase succeeded.

Tests:

- Positive: User adds a product that eligible for immediate purchase, then purchase the product by clicking immediate purchase, then getting a purchase confirmation.
- Negative: User enters a coupon code, a message appears that the code is correct but the price doesn't change.

3 Registered user

3.1 Logging out the system

Actor: User

Pre-conditions: User is in a member status and is logged in

Post-conditions: The User status is changed from Member to Guest.

Flow:

- A member user press on the Log out button and his status changed from Member to User.

Tests:

- Positive: A member clicks on the Log out button and exits his state as a Member Successfully.
- Negative: A member clicks on the Log out button and receives an error message "Log out has failed"

3.2 Opening a shop

Actor: User

Pre-conditions: User is in a member status and is logged in

Post-conditions: A new shop is created in the system and the Member is defined as its first shop owner.

Flow:

- A member user clicks on the create shop button fills in the info about the shop and a new shop is created and the first shop owner is the member

Tests:

- Positive: A member clicks on the Create shop button, fills the shop information and a new shop is created successfully.
- Negative: A member clicks on the Create-shop button fills the information of the shop and receives an error message "a shop could not be created, please contact an admin".

4.1 Inventory Management

a. adding a new product.

Actor: Shop owner

Pre-conditions: the product doesn't exist in the shop's product list.

Post-conditions: the product does exist in the shop's product list.

Flow:

- The shop owner selects the adding option.
- The shop owner fills the relevant fields of a new product.
- The product is now added to the shop's product list..

Alternative flow:

- The shop owner selects the adding option.
- The shop owner fills the relevant fields of a new product.
- The system alerts the owner that one or more of the fields is illegal and doesn't add the product.

Tests:

- Positive: After filling in the details by the shop owner, the product is now in the shop's item list.
- Negative: After filling in the details by the shop owner, the product is not in the shop's item list.

b. deleting a product.

Actor: Shop owner

Pre-conditions: the product exists in the shop's product list.

Post-conditions: the product doesn't exist in the shop's product list.

Flow:

- The shop owner selects the item he wishes to delete.
- The shop owner selects the deleting option.
- The product is now not on the shop's product list.

Tests:

- Positive: After selecting a product to remove and selecting the removal option, the product is now not in the shop's item list.
- Negative: After selecting a product to remove and selecting the removal option, the product is still in the shop's item list.

c. Changing an existing product.

Actor: Shop owner

Pre-conditions: the product exists in the shop's product list.

Post-conditions: None

Flow:

- The shop owner selects the changing option on the wanted product.
- The shop owner changes the relevant fields of the product with compatible data.
- The product is now changed with the new info from the owner.

Alternative flow:

- The shop owner selects the changing option on the wanted product.
- The shop owner changes the relevant fields of the product with incompatible data.
- The product is not being changed and an error message is displayed to the user.

Tests:

- Positive: Shop owner changed the price to another positive number, change succeeded.
- Negative: Shop owner changed the price to a negative number, change failed.

4.2 Purchase policy and discounts

a. defining purchase policy for the entire shop.

Actor: Shop-owner

Pre-conditions: None

Post-conditions: The required purchase policy is now one of the shop's purchase policies.

Flow:

- The shop-owner chooses the required shop.
- The shop-owner chooses the option of defining a new purchase policy.
- The shop-owner fills out the required details.
- The new purchase policy is now one of the shop's purchase policies.

Tests:

- Positive: The shop-owner fills out a new purchase policy with compatible details and the policy is added to the shop's policies.
- Negative: The shop-owner fills out a new purchase policy with dates that have already passed, and an error message is presented, and the operation fails.

b. defining discount for the entire shop.

Actor: Shop-owner

Pre-conditions: None

Post-conditions: The required discount is now one of the shop's discounts.

Flow:

- The shop-owner chooses the required shop.
- The shop-owner chooses the option of defining a new shop's discount.
- The shop-owner fills out the required details.
- The new discount is now one of the shop's discounts.

Tests:

- Positive: The shop-owner fills out a new discount with compatible details and the discount is added to the shop's discounts.
- Negative: The shop-owner fills out a new discount with a negative discount value (+20%) and an error message is presented, and the operation fails.

c. changing purchase policy for the entire shop.

Actor: Shop-owner

Pre-conditions: There's a purchase policy for the shop.

Post-conditions: The changed purchase policy is a part of the purchases policies.

Flow:

- The shop-owner chooses the required shop.
- The shop-owner chooses the purchase policy and the option of update its fields.
- The shop-owner fills out the required details.
- The policy is now with the updated details.

Tests:

- Positive: The shop-owner fills out the updated purchase policy with compatible details and the policy is updated and part of the shop's policies.
- Negative: The shop-owner fills out the updated purchase policy with dates that have already passed. An error message is presented to the user, and the operation fails.

d. changing discount for the entire shop

Actor: Shop-owner

Pre-conditions: The required discount is part of the shop's discounts.

Post-conditions: The required discount is part of the shop's discounts.

Flow:

- The shop-owner chooses the required shop.
- The shop-owner chooses the option of updating a new shop's discount.
- The shop-owner fills out the required details.
- The discount is now with the updated details.

Tests:

- Positive: The shop-owner chooses an existing discount and updates it with compatible details and the changes are saved.
- Negative: The shop-owner chooses an existing discount and updates it with a negative discount value (+20%) and an error message is presented, and the operation fails.

e. defining purchase policy for specific product.

Actor: Shop-owner

Pre-conditions: The shop's product list is not empty.

Post-conditions: The required product includes the wanted purchase policy.

Flow:

- The shop-owner chooses the required shop and the required item.
- The shop-owner chooses the option of defining a new purchase policy.
- The shop-owner fills out the required details.
- The new purchase policy is now one of the product's purchase policies.

Tests:

- Positive: The shop-owner fills out a new purchase policy with compatible details and the policy is added to the product's policies.
- Negative: The shop-owner fills out a new purchase policy with dates that have already passed, and an error message is presented, and the operation fails.

f. defining discount for specific product.

Actor: Shop-owner

Pre-conditions: The shop's product list is not empty.

Post-conditions: The required product includes the wanted discount.

Flow:

- The shop-owner chooses the required shop and the required product.
- The shop-owner chooses the option of defining a new product's discount.
- The shop-owner fills out the required details.
- The new discount is now one of the product's discounts.

Tests:

- Positive: The shop-owner fills out a new discount with compatible details and the discount is added to the product's discounts.
- Negative: The shop-owner fills out a new discount with a negative discount value (+20%) and an error message is presented, and the operation fails.

g. changing purchase policy for a specific product.

Actor: Shop-owner

Pre-conditions: There's a purchase policy for the product.

Post-conditions: The required product includes the changed purchase policy.

Flow:

- The shop-owner chooses the required shop and the required item.
- The shop-owner chooses the option of updating an existing purchase policy.
- The shop-owner fills out the required details.
- The updated purchase policy is now one of the product's purchase policies.

Tests:

- Positive: The shop-owner chooses a purchase policy to update and fill it with compatible details and the policy is changed and is a part of the product's policies.
- Negative: The shop-owner chooses a purchase policy to update and fill it with dates that already passed, and the policy is changed and is a part of the product's policies.

h. changing discount for a specific product.

Actor: Shop-owner

Pre-conditions: There's a discount for the product.

Post-conditions: The updated discount is part of the shop's discounts.

Flow:

- The shop-owner chooses the required shop and the required product.
- The shop-owner chooses the option of updating an existing product's discount.
- The shop-owner fills out the required details.
- The discount is now with the updated details.

Tests:

- Positive: The shop-owner chooses an existing discount and updates it with compatible details and the changes are saved.
- Negative: The shop-owner chooses an existing discount and updates it with a negative discount value (+20%) and an error message is presented, and the operation fails.

4.4 Nomination of a store-owner

Actor: Store owner

Pre-conditions: Nominated user is not a store owner.

Post-conditions: Nominated user is a store owner.

Flow:

- Store owner selects a user to be a store owner.
- User becomes a store owner – if user isn't a store owner already.

Tests:

- Positive: Store owner tries to appoint a new user, not a store owner already, to be store owner.
- Negative: Store owner tries to appoint a new user, that is a store owner already, to be store owner.

4.6

Actor: Store owner

Pre-conditions: Nominated user is not a store owner or a store manager.

Post-conditions: Nominated user is a store manager.

Flow:

- Store owner selects a user to be a store manager.
- User becomes a store manager– if user isn't a store owner/manager already.

Tests:

- Positive: Store owner tries to appoint a new user, not a store owner/manager already, to be store manager.
- Negative: Store owner tries to appoint a new user, that is already a store owner/manager, to be store manager.

4.7 Permissions addition/change of store managers

a. Adding permissions for a store manager

Actor: Store owner

Pre-conditions: The store manager has been assigned by the store owner (actor).

Post-conditions: Added permissions for the store owner.

Flow:

- Store owner selects a store manager that he appointed.
- Store owner adds permissions to that store manager.

Tests:

- Positive: A store owner selects a store manager that has been appointed by him and adds permissions to him.
- Negative: A store owner selects a store manager that hasn't been appointed by him and adds permissions to him.

b. Changing permissions for a store manager

Actor: Store owner

Pre-conditions: The store manager has been assigned by the store owner (actor).

Post-conditions: None.

Flow:

- Store owner selects a store manager that he appointed.
- Store owner selects a permission to change.
- Store owner changes permissions to that store manager.

Tests:

- Positive: A store owner selects a store manager that has been appointed by him and changes his permission.
- Negative: A store owner selects a store manager that hasn't been appointed by him and changes his permission.

4.9 Store closing

Actor: Store-founder

Pre-conditions: Store is opened.

Post-conditions: Store is in-active.

Flow:

- Store founder requests to close his store.

Tests:

- Positive: Store founder requests to close his store.
- Negative: Store founder requests to close a store that was opened by another user.

4.11

a. Request store staff info

Actor: Store owner

Pre-conditions: None

Post-conditions: None

Flow:

- Store owner requests info about the staff of the store.
- Store owner receives the info if no error occurs.

Tests:

- Positive: Store owner requests staff info of his store.
- Negative: Store owner requests staff info a store that he isn't store owner of.

b. Request store managers permissions

Actor: Store owner

Pre-conditions: None

Post-conditions: None

Flow:

- Store owner requests the permissions of the store staff.
- Store owner receives the permissions if no error occurs.

Tests:

- Positive: Store owner requests managers permissions of his store.
- Negative: Store owner requests manager permissions of a store that he isn't store owner of.

4.13 Receiving purchase history of a store

Actor: Store owner

Pre-conditions: None

Post-conditions: None

Flow:

- Store owner requests the store purchase history.
- Store owner receives the purchases if no error occurs.

Tests:

- Positive: Store owner requests purchase history of his store.
- Negative: User requests purchase history of a store that he isn't an owner of.

5. Making actions based on privileges

Actor: Shop manager

Pre-conditions: None

Post-conditions: None

Flow:

- Shop manager chooses an action to perform that is allowed by his privileges.
- Action is performed.

Alternative flow:

- Shop manager chooses an action to perform that is not allowed by his privileges.
- Action is not performed.

Test:

- Positive: Shop manager tried to perform an action that is in his privileges.
- Negative: Shop manager tried to perform an action that is not in his privileges.

6.4 Obtain purchase history of a store and buying users

a. Obtain purchase history of a store

Actor: System Manager

Pre-conditions: None.

Post-conditions: None.

Flow:

- System manager requests purchase history of a certain store.

Test:

- Positive: System manager requests purchase history of an existing store.
- Negative: System manager requests purchase history of a non-existing store.

b. Obtain purchase history of a buyer

Actor: System Manager

Pre-conditions: None.

Post-conditions: None.

Flow:

- System manager requests purchase history of a certain buyer (user).

Tests:

- Positive: System manager requests purchase history of an existing user.
- Negative: System manager requests purchase history of a non-existing user.

Terms Dictionary

User – User of the system, could be a guest, a member or an admin.

Guest – a User that hasn't logged in the system and does not have the same privileges and access of system functionality that member has.

Member – a User that logged in the system and could benefit of the full system functionality as a member.

Store founder – a state of a member who has opened a store, which lets him manage his store to its full functionality.

Store owner- a state of a member who got permission from a store founder to be an owner at his store, doesn't have all the store founder functionality.

Store manager – a state of a member who got permission from a store owner to be a manager at his store. The state lets the member certain access to the store management functionality.

Cart – an object used by the User to hold Baskets.

Basket – an Object that gathers products from a specific store.

Product - an Item that can appear in stores to be sold.

Access – a set of permissions that a user has based on his status.