Use cases for seller module

1. User opens a store and becomes its main shop owner.
2. Shop owner promotes another user to be a shop owner as well.
3. Shop owner/manager adds a new product to his shop.
4. Shop owner/manager creates a new raffle for a product in his store.
5. The main Shop owner unlists his store from the system.
6. Shop owner removes another shop owner which he promoted previously.

Use case 1:

Actors: User

Pre-Condition: the user is registered.

Parameters: the shop data.

Actions:

* User: sends request to open a new shop and become the main shop owner.
* System: creates the shop according to the given data.

Post-Condition: A shop is created and is given a new id with the user as a main owner.

Acceptance Tests:

* Positive: registered user tries to open a new shop given his shop data.
* Negative: user that is not registered tries to open a new shop.
* Alternative: not sure yet.

Use case 2:

Actors: shop owner, user

Pre-Condition: shop owner is logged in and user is registered.

Parameters: user id, shop id.

Actions:

* Shop owner: sends request to promote the given user to be a new shop owner.
* User: gets a message about the promotion and confirms it.
* System: manages the promotions and makes the user a new shop owner under the original shop owner.

Post-Condition: the user becomes a new shop owner under the original shop owner.

Acceptance Tests:

* Positive: the shop owner tries to promote a registered user to be a new shop owner at his shop.
* Negative: shop owner tries to promote an unregistered user to be a new shop owner.
* Alternative: not sure yet.

Use case 3:

Actors: shop owner

Pre-Condition: The shop owner has permission to add new products.

Parameters: product data.

Actions:

* Shop owner: sends request to add a new product to the shop.
* System: adds a new product to the shop and gives it a new id.

Post-Condition: the product has been added to the shop and can be bought and managed in the system.

Acceptance Tests:

* Positive: shop owner with the right permission tries to add a new product to his shop.
* Negative: shop owner without the right permission tries adding a new product to his shop.
* Alternative: not sure yet.

Use case 4:

Actors: shop owner

Pre-Condition: The shop owner has permission to create a new raffle.

Parameters: raffle data and products ids.

Actions:

* Shop owner: sends request to create a new raffle.
* System: adds a new raffle to the shop with the right products.

Post-Condition: the raffle has been created and the users can interact with it.

Acceptance Tests:

* Positive: shop owner with the right permission creates a new raffle
* Negative: shop owner without the right permission tries to create a new raffle.
* Alternative: not sure yet.

Use case 5:

Actors: main shop owner

Pre-Condition: The shop owner is the main shop owner, and the shop is not active.

Parameters: shop id.

Actions:

* Shop owner: sends request to unlists his shop from the market .
* System: hides the shop from the user apart from shop owners and managers.

Post-Condition: the shop is not visible for the common user.

Acceptance Tests:

* Positive: main shop owner tries to unlists his shop from the market.
* Negative main shop owner tries to unlist an unactive shop.
* Alternative: not sure yet.

Use case 6:

Actors: 2 shop owners

Pre-Condition: The shop owner promoted the second shop owner.

Parameters: shop owner id, shop id.

Actions:

* Shop owner: sends request to remove a shop owner that is under him.
* System: remove the shop owner that was requested and remove any other shop owner or manager that he ever promoted.

Post-Condition: the second shop owner is no longer a shop owner and does have any permissions at the shop.

Acceptance Tests:

* Positive: shop owner tries to remove another shop owner that he promoted.
* shop owner tries to remove another shop owner that is not under him.
* Alternative: not sure yet.

Responsibilities

* User can become a main shop owner.
* Main shop owner can unlist and active his shop.
* Main shop owner can do everything that a regular shop owner can do.
* Shop owner can promote a registered user to be a new shop owner.
* Shop owner can remove another shop owner which he promoted previously.
* Shop owner can remove any shop owner who is under him.
* Shop owner can promote a registered user to be a new manager.
* Shop owner can give permissions to the manager that he promoted or under him.
* Shop owner and manager can manage the store according to the permissions that they have.

glossary

* User – user in the system
* Main shop owner – registered user that created the shop
* Shop owner – registered user that was promoted by an existing shop owner
* Shop manager – registered user that was promoted by an existing shop owner.
* Under him – any user that was promoted by someone that I promoted is under me, meaning I can remove them or change their permissions.