E-Commerce System: Use Cases

Name	Start-Up
Requirement number	1.1
Actors	User
Parameters	None
Preconditions	1. Machine possesses the latest version of the software. 2. Machines have a stable connection to external services. 3. System has already been booted.
Actions	User runs the client application.
Result	User is considered as Guest, thus receives an empty shopping cart. Homepage is shown to the user.
Input Data	1. User has no connection to the server – Failure. 2. User has an older version of the client.

Name	Start-Up - First Boot
Requirement number	1.1
Actors	User
Parameters	None
Preconditions	1. Machine possesses the latest version of the software. 2. Machines have a stable connection to external services. 3. System is not booted yet.
Actions	1. User runs the client application. 2. User enters system-admin credentials.
Result	Homepage is shown, current user is considered as a Guest, thus receives an empty shopping cart.
Input Data	1. User has no connection to the server – Failure. 2. User has an older version of the client – Failure.

E-Commerce System: Use Cases

Name	Update external service
Requirement number	1.2
Actors	Admin
Parameters	External service credentials.
Preconditions	The new external service is not known within the system.
Actions	Admin added the new external service.
Result	The new external services utilities can be used.
Input Data	1. The external service is already know to the system – Failure.

Name	Payment
Requirement number	1.3
Actors	System
Parameters	1. Payment measure 2. Order details
Preconditions	1. System has stable connection with external payment service.
Actions	The System transfer the payment measure to the external payment service.
Result	If the payment succeeds, a recipe will be provided to the user. Otherwise, an error message will be displayed.
Input Data	1. There is no available external payment service for the system – Failure. 2. Invalid payment measure – Failure. 3. Invalid order details – Failure.

Name	Provision
Requirement number	1.4
Actors	System
Parameters	Delivery details.
Preconditions	System has stable connection with external currier service.
Actions	The System transfer the delivery order to the external currier service.
Result	A tracking number is provided to the recipient. The system is notified when the delivery is supplied.
Input Data	1. There is no registered external currier service to the system – Failure. 2. Invalid delivery details (such as illegal addresses) – Failure.

E-Commerce System: Use Cases

Name	Online notifications
Requirement number	1.5
Actors	System
Parameters	None
Preconditions	There are registered members to.
Actions	The System notifies active members about activities relevant to them such as store owner whose goods has been purchased.
Result	An active user received a pop-up message about activity he concerned with.
Input Data	1. Goods were purchased from a registered, logged in store owner and a pop-message is displayed on his screen – Success. 2. A registered logged in member is given a pop-up message about new message he received.

Name	Postponed notifications
Requirement number	1.6
Actors	System
Parameters	None
Preconditions	There are registered members to.
Actions	The System notifies offline members about activities that occurred while they were inactive.
Result	A user received a pop-up message, after login, about activities relevant to him which occurred while he was inactive
Input Data	1. Goods were purchased from a registered store owner. When the store owner log in, A pop-message about the order will be displayed.

Name	Entering the system
Requirement number	2.1.1
Actors	Guest
Parameters	None
Preconditions	System is booted.
Actions	1. User initializes client application.
Result	Homepage is shown, current user is considered as a Guest, thus receives an empty shopping cart.
Input Data	1. User has no connection to the server – Failure. 2. User has an older version of the client – Failure.

E-Commerce System: Use Cases

Name	Exiting the system
Requirement number	2.1.2
Actors	User, Guest
Parameters	None
Preconditions	None
Actions	1. User closes client application. 2. If the user is currently logged in, the system shall apply logout procedure.
Result	None
Input Data	3. A Member existing the system, his shopping cart is preserved – Success. 4. A Guest existing the system, his shopping cart is discarded – Failure.

Name	Registering
Requirement number	2.1.3
Actors	Guest
Parameters	Username, Password
Preconditions	The username is unique within the system.
Actions	1. User requests to register with some Username and Password. 2. The System Verifies the Password's security level and that the Username does not already exist 3. If one of the previous verifications fails, an error message is shown to the user and the registration abort. Otherwise, a new Member is added to the system.
Result	The new member can now login to the system
Input Data	1. The User enters a Username that's already in use – Failure. 2. The User is trying to use an insecure password - Failure. 3. The User is using a unique Username with a secure password - Success.

E-Commerce System: Use Cases

Name	Login
Requirement number	2.1.4
Actors	Guest, System
Parameters	Username, Password
Preconditions	1. The requested user was registered in the system beforehand. 2. Either guest or system has stable connection.
Actions	1. Guest provides his credentials. 2. If he is already logged in, an error message will be displayed. 3. If the credentials has verified successfully, the system loads member shopping cart and present his postponed notifications. Otherwise, an error message shall be displayed.
Result	The Guest turns into Member. His Shopping Cart is loaded.
Input Data	1. Username is unknown to the system – Failure. 2. Password is incorrect – Failure. 3. Credential are verified – Success.

Name	Searching for a Store
Requirement number	2.2.1
Actors	Guest
Parameters	Store name
Preconditions	None
Actions	1. User chooses “Search Store” option from the menu 2. User enters the name of the store he is searching for 3. The System presents the store with the name the user wanted
Result	The store homepage is displayed.
Input Data	1. The User searches for a store which doesn’t exist – Failure. 2. The User searches for a specific known store – Success.

E-Commerce System: Use Cases

Name	Searching for a Product
Requirement number	2.2.2
Actors	Guest
Parameters	Product attributes
Preconditions	None
Actions	1. User chooses “Search Products” option from the menu 2. User requests a Product with specific attributes from the System 3. The System presents all Products from any Store, which apply to those attributes
Result	The system presents list of products (with prices), a quantity input box, an “add to cart” button and “place a bid” button under the list.
Input Data	1. The User is searching for a product that doesn’t exist– Failure. 2. The User is searching for a product that exists– Success.

Name	Adding a Product to the Shopping Cart
Requirement number	2.2.3
Actors	Guest
Parameters	Product attributes
Preconditions	1. The Product is sold by some Store. 2. The User has a Shopping Cart.
Actions	1. The User finds a Product (either from use case 2.2.1 or 2.2.2) 2. The User chooses a product 3. The User presses the “add to cart” button with the product chosen
Result	The User’s Shopping Cart receives the Product from the appropriate store
Input Data	1. The User request amount of a product that exceeds store stock – Failure. 2. The User is adding a product from a Store - Success 3. The User is adding a product found from searching a product – Success.

E-Commerce System: Use Cases

Name	Inspecting Shopping Cart
Requirement number	2.2.4
Actors	Guest
Parameters	None
Preconditions	None
Actions	1. User requests for his Shopping Cart 2. The System presents the Shopping Cart, allowing the user to update products amount.
Result	All previously added products are displayed.
Input Data	1. The User is inspecting his shopping cart when it has no Products – an empty cart is shown – Success. 2. The User is inspecting his shopping cart when it has 1 or more products – Success. 3. The User is inspecting his shopping cart and update his desired amounts from products – Success.

Name	Purchase shopping cart
Requirement number	2.2.5
Actors	User
Parameters	Shopping cart
Preconditions	Shopping cart is not empty
Actions	1. User asks to pay for his shopping using via press on the button “checkout”. 2. The user selects his preferred payment measure. 3. The system verifies the for each product in the shopping cart the following hold: (1) Store from which the product is purchased is still open. (2) The store possesses the desired amount from the product. (3) Verify purchaser is conformed with store policy. (4) Apply store discount policy. 4. The system delegates the payment measure to the external payment service.
Result	The Purchase is completed, the purchaser is notified, stores product’s amounts are updated.
Input Data	1. User checkout an empty shopping cart – Failure. 2. User preferred payment measure is unavailable – Failure. 3. User attempts to purchase a product from closed store – Failure.

	4. User attempts to purchase a product with an amount exceeding store stock – Failure. 5. User attempt to purchase a product which he is forbidden according to store policy – Failure. 6. Payment process failure – Failure. 7. Connections is lost – Failure. 8. None of the above – Success.

Name	Logout
Requirement number	3.1
Actors	Member
Parameters	None
Preconditions	Member is currently logged in.
Actions	1. The Member requests to log out from the System 2. The System saves his / her current Shopping Cart content 3. The System changes the Member's status to Guest
Result	The Member is considered as a Guest by the system. His shopping cart is saved.
Input Data	1. A guest attempts to logout – Failure. 2. The Member is logging out with products in his cart. 3. The Member is logging out with an empty cart.

Name	Opening a Store
Requirement number	3.2
Actors	Member
Parameters	None
Preconditions	Member is currently logged in.
Actions	1. The Member requests to open a Store with some attributes 2. The System verifies the attributes are legal, if it fails it notifies the Member 3. The System makes a new Store and appoints the Member as a Storeowner.
Result	A new Store is created.
Input Data	1. A guest attempts to open a store – Failure. 2. A Member is attempting to open a store with invalid store attributes – Failure. 3. A Member is attempting to open a store with legal attributes – Success.
Name	Update store inventory.

E-Commerce System: Use Cases

Requirement number	4.1
Actors	Store Manager
Parameters	Store, product
Preconditions	1. Store Manager is logged in. 2. The store manager granted permission from the store owner.
Actions	1. The manger loges in. 2. The System verifies the Manager has the right permissions to perform the action. 3. The System adds/removes/update the product.
Result	1. The product is added/removed/updated. 2. The relevant Manager is notified about the change
Input Data	1. The Manager doesn’t have the right permissions – Failure. 2. The Manager trying to add a product that already exists in the store – Failure. 3. The Manager tries to remove/update a product that doesn't exists in the store – Failure.

Name	Update store purchase and discount policy
Requirement number	4.2
Actors	Storeowner, store, policy, due date
Parameters	policy, due date
Preconditions	1. Storeowner is logged in. 2. The store has been created and registered to the system. 3. Storeowner is the founder of the given store. 4. Due date is latter than yesterday.
Actions	1. The storeowner loges in 2. The System verifies the Manager has the right permissions to perform the action 3. The System updates policy.
Result	1. The relevant is updated. 2. The relevant Manager is notified about the change.
Input Data	1. The Manager doesn’t have the right permissions – Failure. 2. Invalid due date – Failure. 3. Policy breaches consistency constraints - Failure

E-Commerce System: Use Cases

Name	Approve store owner appointment request
Requirement number	4.4
Actors	Storeowner, Store founder
Parameters	*new_owner, Store*
Preconditions	1. 1. The Owner is logged in 2. 2. Store manager must have proper permission from store owner
Actions	1. The storeowner requests the system to approve the appointment of new_owner 2. The system verifies the storeowner is owns the Store. 3. The system verifies there is an open request to appoint new_owner to Owner 4. The system adds the vote to the list of Owners who approved 5. If all Owners approved the request, then the new_owner is granted Owner permissions
Result	1. The system updates the list of owners who approved the appointment and if all of the owners approved then the new_owner receives Owner Permissions
Input Data	1. The Owner is not the Store owner– Failure. 2. There is no request to appoint new_owner to Owner– Failure. 3. The owner is the store owner and there is a request to appoint new_owner- Success

E-Commerce System: Use Cases

Name	Decline store owner appointment request
Requirement number	4.4
Actors	Storeowner, Store founder
Parameters	*new_owner, Store*
Preconditions	1. 1. The Owner is logged in 2. 2. Store manager must have proper permission from store owner
Actions	1. The owner requests the system to decline the appointment of new_owner 2. The system verifies the Owner is the owner of the Store 3. The system verifies there is an open request to appoint new_owner to Owner 4. The system removes the request and notifies the requesting owner that the request has been declined
Result	The system updates the list of owners who approved the appointment and if all of the owners approved then the new_owner receives Owner Permissions
Input Data	1. The Owner is not the Store owner– Failure. 2. There is no request to appoint new_owner to Owner– Failure.

Name	Appoint store manager
Requirement number	4.6
Actors	Storeowner
Parameters	*new_owner, Store*
Preconditions	1. The Owner is logged in
Actions	1. The owner requests the system to add a store manager to his Store. 2. The System verifies the Store exists. 3. The System verifies the requester is the Store owner. 4. The System verifies the new_manager is a valid member. 5. The System makes the new_manager a manager with default permissions to the store.
Result	The system updates new_manager permissions and notifies the Owner.
Input Data	1. The Owner is not the Store owner– Failure. 2. The new_manager is already the Store's manager / owner – Failure. 3. The owner sets a User who is not the Store's manager / owner - Success

E-Commerce System: Use Cases

Name	Change store manager permissions
Requirement number	4.7
Actors	Owner, Store manager
Parameters	*manager, new_permissions,Store*
Preconditions	1. The Owner is logged in 2. Store manager must have proper permission from store owner
Actions	1. The Owner requests the system to manager permissions 2. The System verifies the Owner is the Store owner 3. The Owner enters the new_permissions
Result	The system updates the manager permissions and notifies the Owner
Input Data	1. The Owner is not the Store owner– Failure. 2. The manager is not the Store's manager– Failure. 3. The owner chooses correct permissions- Success

Name	Close a store
Requirement number	4.9
Actors	Store founder
Parameters	*Store ID*
Preconditions	1. Store exists and in "open" status 2. User is the store's founder
Actions	1. The store founder asks the system to close a store 2. The store founder inputs a store that he founded 3. The system verifies that the store exists and is not in "closed" state already 4. The system verifies the relevant user permissions 5. The system changes the store status to "closed" 6. The system sends the store staff a "Store closing message" 7. The system removes the store's products from the product search view
Result	1. Store's status is changed to "closed" 2. Store is only visible to admins and store owners 3. Store's product are not visible on search 4. "Store closing message" is sent to store's owner and manager
Input Data	1. User enters an invalid store namer– Failure. 2. User enters a store that is already closed – Failure. 3. User enters a store ID of a store he founded - Success
Name	Get purchase history of member
Requirement number	6.4

E-Commerce System: Use Cases

Actors	Admin, Member
Parameters	*Member ID*
Preconditions	Member exists
Actions	1. The admin chooses a member it wants to get its purchase history 2. The system verifies the member exists 3. The system verifies the relevant user permissions 4. The system sends the admin the saved history of the member
Result	Member’s purchase history is provided to the admin
Input Data	1. User enters a member ID that doesn’t exist – Failure. 2. User enters a character that is not a number – Failure. 3. User enters a member ID of an existing member-Success