Subject web application: VotingKS

VotingKS is a web-based social network system that allows users to submit or delete assertions and their supportive arguments or evidences. It also allows users to vote other users' assertions.

VotingKS contains three JSPs and one java servlets. The data (users' information, assertions and corresponding arguments or evidences) are stored in xml files.

Specific requirements

Functional requirements

Use Case Name: dispatch
Use Case ID: VotingKS-1
Actors: Web user
Brief Description: The system allows users to login to the system or to register as new users.
Pre-Conditions: None
Main Flow of Events: 1. User chooses to login as a returning user. He/she enters a username and password combination. 2. User presses the 'Submit' button 3. The system forwards the username and password combination to the login use case to validate the user credentials. 4. The system displays a form where user can enter new assertions and a list of all existing assertions along with the voting options.
Post-Conditions: The system displays a form to enter new assertions and lists all assertions along with the voting options.
Alternative Flows and Exceptions: Alternative 1, Insert Point Step 1-3 Alternative flow of events 1. If the user chooses to register as a new user to the system, he/she presses the 'New User' button. 2. User enters a username, password, and confirm password. 3. User selects the 'Submit' button. 4. The system forwards the user information to the register use case to validate and register the user with the system. Alternative 2, Insert Point Step 2 If the user selects the 'Reset' button, the system resets the form (username and password). Alternative 3, Insert Point Step 3 If the system cannot validate the user credentials, it notifies the user about an incorrect username and/or password.
Assumptions: None.

Use Case Name: login
Use Case ID: VotingKS-2
Brief Description: Registered users log into the system using their username and password.
Pre-Conditions: User has registered with the system and a combination of username and password has been entered.

Main Flow of Events:
    1.   The system validates the use credentials.
    2.   The system updates the user status to 'Signed in.'

Post-Conditions: The status of the user has been changed to 'Signed in.'
Alternative Flows and Exceptions: Alternative 1, Insert Point Step 1 If the system cannot validate the user credentials, it notifies the user about an incorrect username and/or password.
Assumptions: None.

Use Case Name: register
Use Case ID: VotingKS-3
Brief Description: Users can register with the system to submit their assertions and vote for other users' assertions.
Pre-Conditions: Username, password, and confirm password have been entered.

Main Flow of Events:
    1.   The system validates the user credentials.
    2.   The system registers the user information (username and password) to the system.
    3.   The system updates the user status to 'Signed in.'

Post-Conditions: The user information is stored in the system. The status of the user has been changed to 'Signed in.'
Alternative Flows and Exceptions: Alternative 1, Insert Point Step 1 If the username is already taken by another user, the system displays an error message indicating that the username has already been taken. Alternative 2, Insert Point Step 1 If the password does not match the confirm password, the system displays an error message indicating the mismatch.
Assumptions: None.

Use Case Name: logout
Use Case ID: VotingKS-4
Actors: Web user
Brief Description: The system allows users to log off the system to avoid unauthorized access.
Pre-Conditions: User has logged in to the system.

Main Flow of Events:
   1.   User presses the 'Log off' button.
   2.   The system updates the user status to 'Logged out.'
   3.   The system displays the main screen where a user can login to the system.

Post-Conditions: The status of the user has been changed to 'Logged out.' The system displays the main screen.
Alternative Flows and Exceptions: None
Assumptions: None.

Use Case Name: voteAssertion
Use Case ID: VotingKS-5
Actors: Web user
Brief Description: The system allows users to vote for other users' assertions. Three voting options are convinced, disagree, and unsure. The system keeps track of the number of each kind of votes.
Pre-Conditions: User has logged into the system. There exists some assertions in the system.

Main Flow of Events:
   1.   User selects the 'Convinced' voting option.
   2.   User presses the 'Vote' button
   3.   The system updates the number 'Convinced' vote of the selected assertion.
   4.   The system re-sorts all the assertions based on the number of positive votes.

Post-Conditions: The system displays all current assertions
Alternative Flows and Exceptions: Alternative 1, Insert Point Step 1-3 If the user selects the 'Unsure' voting option, the system updates the number 'Unsure' vote of the selected assertions. Alternative 2, Insert Point Step 1-3 If the user selects the 'Disagree' voting option, the system updates the number 'Disagree' vote of the selected assertions.
Assumptions: User cannot vote his/her own assertions.

Use Case Name: addAssertion
Use Case ID: VotingKS-6
Actors: Web user
Brief Description: The system allows users to enter new assertions.
Pre-Conditions: User has logged into the system.

Main Flow of Events:
    1.   User enters an assertion and its evidence or arguments.
    2.   User presses the 'Submit' button.
    3.   The system stores an assertion and its arguments along with its owner (username).
    4.   The system displays all assertions that have been entered to the system, sorting by the number of positive votes.

Post-Conditions: The system displays all current assertions. The new assertion is stored in the system.
Alternative Flows and Exceptions: Alternative 1, Insert Point Step 1 If an assertion and/or its evidence is not entered, the system displays an error message requiring the user to enter information before submitting a new assertion into the system. Alternative 2, Insert Point Step 2-3 If the user selects the 'Reset Assertion' button, the system resets the form. No new assertion is recorded in the system.
Assumptions: None.

Use Case Name: deleteAssertion
Use Case ID: VotingKS-7
Actors: Web user
Brief Description: The system allows users to delete their assertions from the system.
Pre-Conditions: User has logged into the system. User has previously entered assertions into the system.

Main Flow of Events:
    1.   User presses the 'Delete this assertion' button.
    2.   The system removes the selected assertion and its arguments from the system.
    3.   The system displays all the remaining assertions that have been entered to the system, sorting by the number of positive votes.

Post-Conditions: The system displays all current assertions. The selected assertion is removed from the system.
Alternative Flows and Exceptions: None
Assumptions: Users can delete only their own assertions.

Other Requirements

Requirement ID	Requirement
SRS_01	The system shall maintain the usernames throughout the sessions.
SRS_02	The system shall allow users to enter unlimited number of assertions.
SRS_03	The system shall allow users to vote unlimited number of assertions.
SRS_04	The system shall allow multiple votes.
SRS_05	The system shall allow users to vote other users' assertions.
SRS_06	The system shall not allow users to vote their own assertions.
SRS_07	The system shall allow users to delete their own assertions.
SRS_08	The system shall not allow users to delete other users' assertions.
SRS_09	The system shall display assertions along with their owners (usernames).
SRS_10	The system shall display assertions along with three voting options (Convinced, Unsure, and Disagree).
SRS_11	The system shall sort assertions based on the number of positive votes.