|  |  |
| --- | --- |
| Use Case: | Buy subscriptions |
| Actor: | Customer |
| Precondition: | A customer wants a subscription and has a valid payment method |
| Main Success  Scenario: | 1: The customer gives the command for buying a subscription  2: The customer gives name, address, and email to the program.  3: The program ensures the information is formatted correctly and payment information is validated by the program.  4: The program records this in the database.  5: The customer is given a confirmation message |
| Extensions: | 3a The customer enters an email that already exists in the database.  -program informs the user that the email already is in use  -user is given another chance to enter an email address  3b The customer enters an email that is not formatted correctly.  -program informs the user that the email is invalid and displays the correct format  -user is given another chance to enter an email address  4a. The customer enters invalid payment information.  - program asks for another payment type.  -user can either try another payment type or quit the transaction |

|  |  |
| --- | --- |
| Use Case: | Create articles |
| Actor: | Writers and Editors |
| Precondition: | For editing, there must first be at least one article that exists. |
| Main Success  Scenario: | 1: The writer gives the command for creating an article  2: The program requests a name for the article and the author  3: The writer gives the name and author  4: The program records article name, creation date, last edit date, and author(s).  5: The program enters a writing state, where it accepts and saves inputs as the article until an exit command is given  6: The writer gives exit command and the article is saved.  7: The writer is returned to the initial program state |
| Extensions: | 4a The writer gives a name that’s already been used.  -For unfinished articles, every article must be unique.  - program asks for another name.  -user can either try another name or quit the transaction |