CSCI5448 – Object Oriented Analysis and Design

Class Project - Part #2

**Team Members:**

* Chris Pillion
* John Zavidniak
* Akshat Kambli

**Title:**

**Entertainment and Sports Program – Next Generation (ESP-NGen)**

**Project Summary:**

This effort will produce an interactive sports tracking program that allows users to obtain news and information about their favorite sports, leagues, teams, and athletes. Users will be able to create an account, access different aspects of sports information and news, save preferences, and receive customer support. System Admins will be responsible for keeping the statistical information up to date and providing support to other application users. Lastly, external journalists will be allowed to create an account as a reporter, submit stories and articles for approval, and receive support, if needed. The primary objective is to create a single sports related resource for the sports enthusiast application user.

**Project Requirements:**

The project requirements are broken into the four following categories: **Business Requirements, User Requirements, Functional Requirements,** and **Non-functional Requirements**

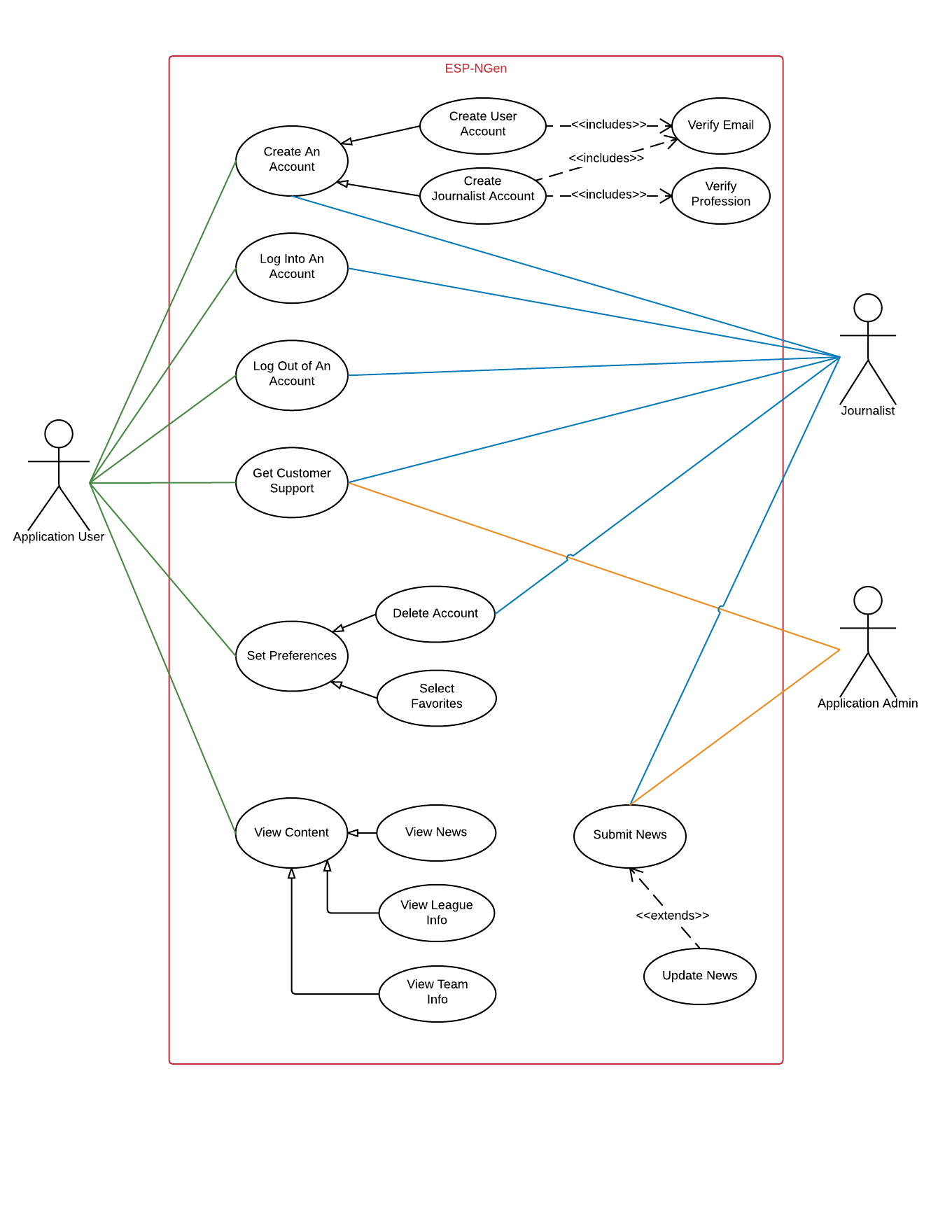
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Business Requirements | | | | |
| ID | **Requirement** | **Topic Area** | **User** | **Priority** |
| BR-001 | Users can only create an account with a verified email address | Authentication | - Primary  - Journalist | High |
| BR-002 | Users can only create at most one primary user account and one journalist account per email address | Authentication | - Primary - Journalist | High |
| BR-003 | Admins must approve all externally submitted news articles and information | Updates | Admin | Med |
| BR-004 | Journalists must create an account with an email from a verified news source domain name | Authentication | Journalist | High |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User Requirements | | | | |
| ID | **Requirement** | **Topic Area** | **User** | **Priority** |
| UR-001 | Primary user shall be able to view sports content related (but not limited) to league, team, and player information | User Content | Primary | Critical |
| UR-002 | Admin shall be able to approve and update submitted news stories from the Journalist | User Content | Primary | Critical |
| UR-003 | Primary user shall be able to access preferences, such as favorites, that will be saved to their account | Preferences | Primary | Low |
| UR-004 | Primary user and journalists shall be able to delete their account at any time | Preferences | - Primary - Journalist | Med |
| UR-005 | Primary user and journalists shall be able to log into and out of their account | Authentication | - Primary - Journalist | High |
| UR-006 | Journalists shall be able to submit news articles to be posted through the application | User Content | Journalist | High |
| UR-007 | Primary users shall be able to view news articles and updates | User Content | Primary | High |
| UR-008 | Primary users and journalists shall be able to receive customer support from a system admin | Support | - Primary - Journalist - Admin | Med |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Functional Requirements | | | | |
| ID | Requirement | Topic Area | User | Priority |
| FR-01 | The System must verify that the username is unique and password meets the set of guidelines at the time of registration | Authentication | -Primary -Journalist | High |
| FR-02 | The System should differentiate between a primary user and a journalist | Authentication | -Primary  -Journalist | High |
| FR-03 | System should detect and display new users in real time | User content | -Primary | Medium |
| FR-04 | Registration will involve two factor authentication through E-mail | Authentication | -Primary  -Journalist | High |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Non-Functional Requirements | | | | |
| ID | **Requirement** | **Topic Area** | **User** | **Priority** |
| NFR-01 | Application should work in adequate internet connection | Performance | -Primary  -Journalist | High |
| NFR-02 | Application should respond to user action within 2 seconds | Usability | -Primary  -Journalist | Medium |
| NFR-03 | Application should handle at-least 5 users at a time | Scalability | -Primary | High |

**Use Case Diagram:**

****

**Use Case Documents:**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-01 | | |
| Use Case Name: | View News | | |
| Description: | Primary user can view sports related news content in the application | | |
| Created By: | Chris Pillion | Last Updated By: | Chris Pillion |
| Date Created: | 13-Oct-2016 | Date Last Updated: | 13-Oct-2016 |

|  |  |  |  |
| --- | --- | --- | --- |
| Actors: | Primary User | | |
| Preconditions: | * User is signed into the application via a user account * User is at the main lobby page * At least one news article been submitted by journalist * At least one news article has been approved and stored in the database for viewing | | |
| Postconditions: | * Primary user can see a list of available news articles * Primary user can view and read articles | | |
| Frequency of Use: | Multiple times daily by Primary User | | |
| Flow of Events: | **Step** | **Actor Action** | **System Response** |
| 1 | Primary user selects option to view news | System loads available news articles |
| 2 | Primary user selects desired article to read | System loads the article content from the database |
| 3 | Primary user can view content associated with article |  |
| Variations: | a | Primary user is not logged in | System notifies primary user that they must log in |
| b | No news articles have been submitted or stored | System notifies primary user that there is no news to show |
| Notes and Issues: | Specific case of the generalization “View Content” | | |
| Developer Notes: | May consider a quick way to bring the user to the main options screen for quick navigation | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-02 | | |
| Use Case Name: | View League Info | | |
| Description: | Primary user can view a sports league info content in the application | | |
| Created By: | Chris Pillion | Last Updated By: | Chris Pillion |
| Date Created: | 13-Oct-2016 | Date Last Updated: | 13-Oct-2016 |

|  |  |  |  |
| --- | --- | --- | --- |
| Actors: | Primary User | | |
| Preconditions: | * User is signed into the application via a user account * User is at the main lobby page * An initial update (at the very least) for league stats has been performed by the admin and is stored in the database | | |
| Postconditions: | * Primary user can see a list of available sports to view * Primary user can view and read league info for the sport | | |
| Frequency of Use: | Multiple times daily by Primary User | | |
| Flow of Events: | **Step** | **Actor Action** | **System Response** |
| 1 | Primary user selects option to view league info | System loads available sports to view |
| 2 | Primary user selects desired sport to see league info | System loads the league info for the selected sport from the database |
| 3 | Primary user can view content associated with sport league |  |
| Variations: | a | Primary user is not logged in | System notifies primary user that they must log in |
| b | No league info is available because there was no initial update of that info | System notifies primary user that there is no info to show |
| Notes and Issues: | Specific case of the generalization “View Content” | | |
| Developer Notes: | May consider a quick way to bring the user to the main options screen for quick navigation | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-03 | | |
| Use Case Name: | View Team Info | | |
| Description: | Primary user can view a sports teams info content in the application | | |
| Created By: | Chris Pillion | Last Updated By: | Chris Pillion |
| Date Created: | 13-Oct-2016 | Date Last Updated: | 13-Oct-2016 |

|  |  |  |  |
| --- | --- | --- | --- |
| Actors: | Primary User | | |
| Preconditions: | * User is signed into the application via a user account * User is at the main lobby page * An initial update (at the very least) for league stats has been performed by the admin and is stored in the database | | |
| Postconditions: | * Primary user can see a list of available sports to view * Primary user can view and read league info for the sport | | |
| Frequency of Use: | Multiple times daily by Primary User | | |
| Flow of Events: | **Step** | **Actor Action** | **System Response** |
| 1 | Primary user selects option to view team info | System loads available sports to view |
| 2 | Primary user selects desired sport to see associated teams | System loads the teams for the selected sport |
| 3 | Primary user selects a team | System loads the teams info from the database |
| 4 | Primary user can view content associated with the sports team |  |
| Variations: | a | Primary user is not logged in | System notifies primary user that they must log in |
| b | No team info is available because there was no initial update of that info | System notifies primary user that there is no info to show |
| Notes and Issues: | Specific case of the generalization “View Content” | | |
| Developer Notes: | May consider a quick way to bring the user to the main options screen for quick navigation | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-04 | | |
| Use Case Name: | Submit News | | |
| Description: | Journalist can submit articles for approval and posting by Admins | | |
| Created By: | Chris Pillion | Last Updated By: | Chris Pillion |
| Date Created: | 13-Oct-2016 | Date Last Updated: | 13-Oct-2016 |

|  |  |  |  |
| --- | --- | --- | --- |
| Actors: | Journalist, Admin | | |
| Preconditions: | * Journalist is signed into the application via a journalist account * Journalist is at the main journalist lobby page | | |
| Postconditions: | * Journalist will have submitted an article for approval * Admin will have approved or rejected the article for posting to the news content page | | |
| Frequency of Use: | At least once daily by the Journalist | | |
| Flow of Events: | **Step** | **Actor Action** | **System Response** |
| 1 | Journalist selects option to write news | System loads form for writing a new article |
| 2 | Journalist writes article and chooses the option to submit news | System submits the article to storage and sends notification to admin for approval |
| 3 | Admin reviews the article and rejects or accepts the article | System notifies the Journalist of the approval decision. |
| Variations: | a | Journalist is not logged in | System notifies Journalist that they need to log in |
| b | Journalist submits a blank news article | System notifies that the news article cannot be submitted while blank |
| Notes and Issues: | Admin has the option to immediately update the news or defer later | | |
| Developer Notes: | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-05 | | |
| Use Case Name: | Update News | | |
| Description: | Admin can update sports related news content in the application | | |
| Created By: | Chris Pillion | Last Updated By: | Chris Pillion |
| Date Created: | 13-Oct-2016 | Date Last Updated: | 13-Oct-2016 |

|  |  |  |  |
| --- | --- | --- | --- |
| Actors: | Admin | | |
| Preconditions: | * At least one news article been submitted by journalist * Article has been previously approved by an admin | | |
| Postconditions: | * The database is updated to contain new news articles | | |
| Frequency of Use: | At least once daily by Admin | | |
| Flow of Events: | **Step** | **Actor Action** | **System Response** |
| 1 | Admin selects an article from the approved articles | System shows preview of the article |
| 2 | Admin selects the desired posting location (news) | Article is saved to the database for user retrieval |
| a | No news articles have been submitted | System notifies admin that there is no news to post |
| Notes and Issues: | Specific case to extend the action of submit news, if desired | | |
| Developer Notes: | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-06 | | |
| Use Case Name: | Select Favorites | | |
| Description: | Primary users can select the favorite team, league or news | | |
| Created By: | Akshat Kambli | Last Updated By: | Akshat Kambli |
| Date Created: | 13-Oct-2016 | Date Last Updated: | 13-Oct-2016 |

|  |  |  |  |
| --- | --- | --- | --- |
| Actors: | Primary | | |
| Preconditions: | * User is signed into the application via a user account * User is at the main options screen * Primary user can see a list of available sports to view * Primary user can view and read team info for the sport | | |
| Postconditions: | * The database is updated to contain the favorite sports info * The database is updated to contain the favorite team info | | |
| Frequency of Use: | Multiple times daily by Primary User | | |
| Flow of Events: | **Step** | **Actor Action** | **System Response** |
|  | 1 | Primary user selects favorite sports out of the available sports list | Selected sports get updated in the database |
| 2 | Primary user selects favorite team out of the available team list | Selected team get updated in the database |
| Variations | a | Primary user is not logged in | System notifies primary user that they must log in |
|  | b | Primary user selects submit button without selecting the favorite team | System notifies primary user to select the favorite team |
| c | Primary user selects submit button without selecting the favorite sports | System notifies primary user to select the favorite sports |
| Notes and Issues: | Specific case of the generalization “Set Preferences” | | |
| Developer Notes: | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-07 | | |
| Use Case Name: | Delete Account | | |
| Description: | Primary user and journalists shall be able to delete their account at any time | | |
| Created By: | Akshat Kambli | Last Updated By: | Akshat Kambli |
| Date Created: | 13-Oct-2016 | Date Last Updated: | 13-Oct-2016 |

|  |  |  |  |
| --- | --- | --- | --- |
| Actors: | Primary, Journalist | | |
| Preconditions: | * User is signed into the application via a user account * User is at the main options screen | | |
| Postconditions: | * The user account is deleted from the database * The user selected favorites choices are deleted from the databse | | |
| Frequency of Use: | As per requirement of Primary User (Frequently) | | |
| Flow of Events: | **Step** | **Actor Action** | **System Response** |
|  | 1 | Primary user selects option as delete account from the main screen | System prompts user to confirm its selection about deletion of account. |
|  | 2. | User reconfirms the deletion of account request | System notifies database to delete the user account information and selected favorite option |
| Variations | a | Primary user is not logged in | System notifies primary user that they must log in |
|  | b | Primary user rejects the deletion request when asked while reconfirmation | User account is not affected and system returns to the main screen |
| Notes and Issues: | N/A | | |
| Developer Notes: | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-08 | | |
| Use Case Name: | Customer Support | | |
| Description: | Primary users and journalists shall be able to receive customer support from a system admin | | |
| Created By: | Akshat Kambli | Last Updated By: | Akshat Kambli |
| Date Created: | 13-Oct-2016 | Date Last Updated: | 13-Oct-2016 |

|  |  |  |  |
| --- | --- | --- | --- |
| Actors: | Primary, Journalist, Admin | | |
| Preconditions: | * User is signed into the application via a user account * User is at the main options screen | | |
| Postconditions: | * The user request is resolved by the admin | | |
| Frequency of Use: | Frequently | | |
| Flow of Events: | **Step** | **Actor Action** | **System Response** |
|  | 1 | Primary user selects option as customer support from the main screen | System shows details of customer support executive (admin) to user. |
|  | 2. | User sends query to the system | System notifies admin about the user concern |
|  | 3. | Admin gets the query request from the system | System notifies user about acceptance of request |
|  | 4. | User gets notification about the request being processed | System waits for the admin resolution |
|  | 5. | Admin resolves query and informs system about resolution | System notifies user about the resolution |
|  | 6. | User gets notification about resolution of the query | System stores the analysis in the database |
| Variations | a | Primary user is not logged in | System notifies primary user that they must log in |
|  | b | Primary user fills the incomplete details about concern | System notifies user to provide complete details about query |
|  | c | Admin is not able to solve user query on time | User is unable to utilize concern service / application |
| Notes and Issues: | N/A | | |
| Developer Notes: | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-09 | | |
| Use Case Name: | User/ journalist log out | | |
| Description: | Primary users and journalists can log out of the system | | |
| Created By: | Akshat Kambli | Last Updated By: | Akshat Kambli |
| Date Created: | 13-Oct-2016 | Date Last Updated: | 13-Oct-2016 |

|  |  |  |  |
| --- | --- | --- | --- |
| Actors: | User, Journalist, Database | | |
| Preconditions: | * User is signed into the application via a user account * User is at the main options screen | | |
| Postconditions: | * User account is logged out from the system | | |
| Frequency of Use: | Multiple times each day by the User/ Journalist | | |
| Flow of Events: | **Step** | **Actor Action** | **System Response** |
|  | 1 | Primary user/ Journalist selects option as log out from the main screen | System prompts user to confirm its selection about logging out. |
| 2. | User reconfirms the log out request | System closes the application for that particular user |
|  | 3. | User moves to the main page |  |
| Variations | a | Primary user is not logged in | System notifies primary user that they must log in |
| b | Primary user rejects the log out request when asked while reconfirmation | System returns to the main screen |
| Notes and Issues: | N/A | | |
| Developer Notes: | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-10 | | |
| Use Case Name: | Journalist Login | | |
| Description: | Journalists can login to their journalist account | | |
| Created By: | John Zavidniak | Last Updated By: | John Zavidniak |
| Date Created: | 13-Oct-2016 | Date Last Updated: | 13-Oct-2016 |

|  |  |  |  |
| --- | --- | --- | --- |
| Actors: | Journalist, Database | | |
| Preconditions: |  | | |
| Postconditions: |  | | |
| Frequency of Use: | Multiple times each day by the Journalist | | |
| Flow of Events: | **Step** | **Actor Action** | **System Response** |
| 1 | Journalist enters login information and attempts to login. | System queries the database to check if the account exists. |
| 2 | Database returns account information if the account exists. | System tells the Journalist if the account does not exist. If it does, system checks that the password is correct, the email is verified, and the profession is verified. If any of those fail, system notifies the Journalist. |
| 3 | Journalist moves to Journalist lobby page. |  |
| Notes and Issues: | For Journalists, successful login is dependent on verified email and profession. | | |
| Developer Notes: | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-11 | | |
| Use Case Name: | Create Journalist Account | | |
| Description: | Journalist can create an account | | |
| Created By: | John Zavidniak | Last Updated By: | John Zavidniak |
| Date Created: | 13-Oct-2016 | Date Last Updated: | 13-Oct-2016 |

|  |  |  |  |
| --- | --- | --- | --- |
| Actors: | Journalist, Database | | |
| Preconditions: |  | | |
| Postconditions: |  | | |
| Frequency of Use: | Many times daily as new Journalists sign up | | |
| Flow of Events: | **Step** | **Actor Action** | **System Response** |
| 1 | Journalist enters username, password, and email and clicks sign up. | System queries the database to see if the username is taken. |
| 2 | The database returns an account with the Journalist’s username if one already exists. | System tells Journalist username already exists if it is taken, otherwise system inputs the data to the database. |
| 3 | Database writes the Journalist account into the database, marking it as unverified. | Systems instructs Journalist how to verify their account. |
|  | 4 | Journalist verifies their email and profession | System queries database that the entry should be updated. |
|  | 5 | Database marks the Journalist’s entry as verified. |  |
| Notes and Issues: |  | | |
| Developer Notes: | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-12 | | |
| Use Case Name: | Create User Account | | |
| Description: | Users can create an account | | |
| Created By: | John Zavidniak | Last Updated By: | John Zavidniak |
| Date Created: | 13-Oct-2016 | Date Last Updated: | 13-Oct-2016 |

|  |  |  |  |
| --- | --- | --- | --- |
| Actors: | Journalist, Database | | |
| Preconditions: |  | | |
| Postconditions: |  | | |
| Frequency of Use: | Multiple times daily as new users join | | |
| Flow of Events: | **Step** | **Actor Action** | **System Response** |
| 1 | User enters username, password, and email and clicks sign up. | System queries the database to see if the username is taken. |
| 2 | The database returns an account with the User’s username if one already exists. | System tells User that the username already exists if it is taken, otherwise system inputs the data to the database. |
| 3 | Database writes the User account into the database, marking it as unverified. | Systems instructs Journalist how to verify their account. |
|  | 4 | User verifies their email. | System queries database that the entry should be updated. |
|  | 5 | Database marks the User as verified. |  |
| Notes and Issues: |  | | |
| Developer Notes: | N/A | | |

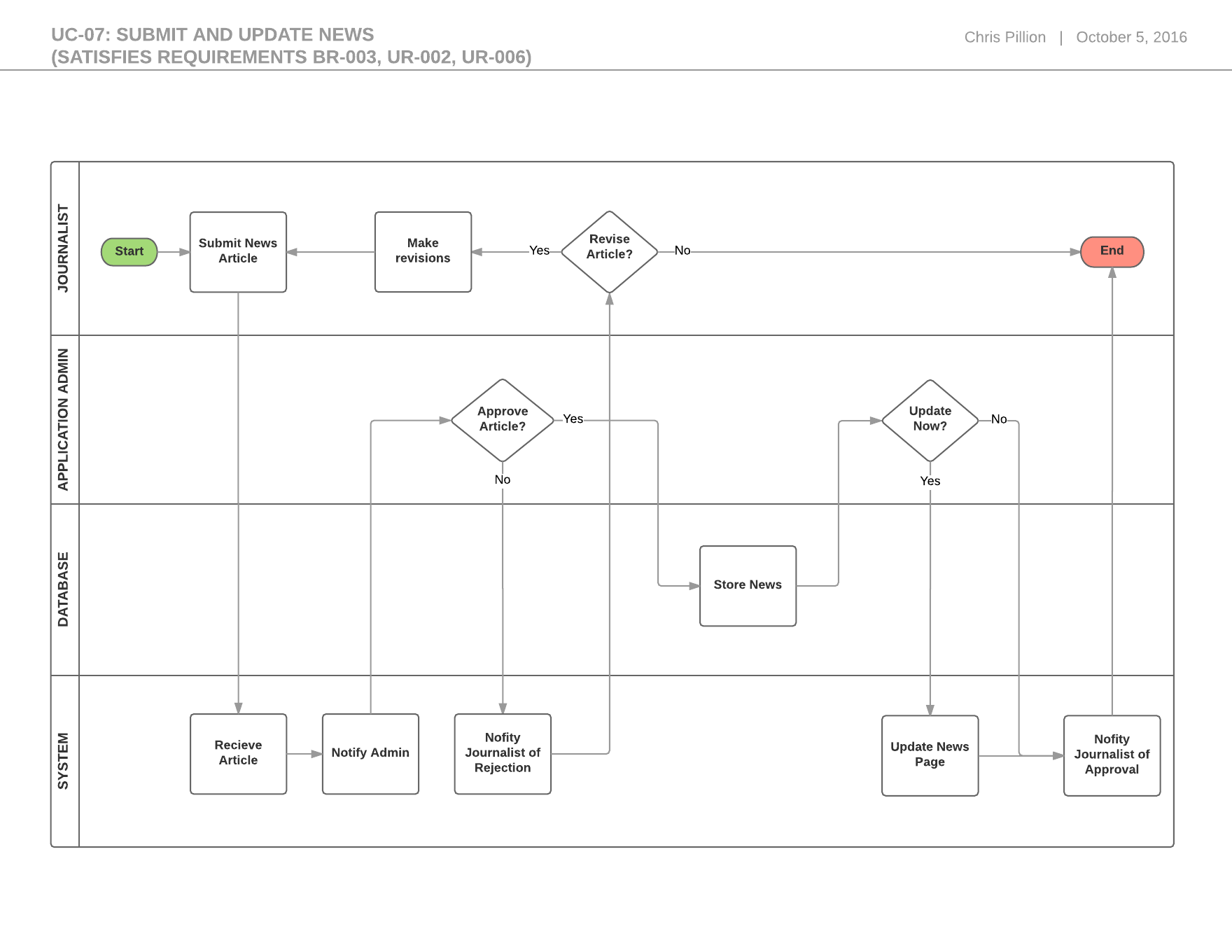
|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-13 | | |
| Use Case Name: | Verify Email | | |
| Description: | Users and Journalists can verify their email | | |
| Created By: | John Zavidniak | Last Updated By: | John Zavidniak |
| Date Created: | 13-Oct-2016 | Date Last Updated: | 13-Oct-2016 |

|  |  |  |  |
| --- | --- | --- | --- |
| Actors: | Account (User/Journalist), Database | | |
| Preconditions: |  | | |
| Postconditions: |  | | |
| Frequency of Use: | Multiple times daily as new users/journalists join and verify | | |
| Flow of Events: | **Step** | **Actor Action** | **System Response** |
| 1 | Account opens their email that they supplied during sign up, and clicks the provided link. | The system tells the database to mark the account as verified. |
| 2 | The database updates the account’s entry as verified. | System displays a message to the account saying that they are now verified. |
| Notes and Issues: |  | | |
| Developer Notes: | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-14 | | |
| Use Case Name: | Verify Profession | | |
| Description: | Journalists can verify their profession | | |
| Created By: | John Zavidniak | Last Updated By: | John Zavidniak |
| Date Created: | 13-Oct-2016 | Date Last Updated: | 13-Oct-2016 |

|  |  |  |  |
| --- | --- | --- | --- |
| Actors: | Journalist, Admin, Database | | |
| Preconditions: |  | | |
| Postconditions: |  | | |
| Frequency of Use: | Multiple times daily as new journalists join/verify | | |
| Flow of Events: | **Step** | **Actor Action** | **System Response** |
| 1 | Journalist clicks link sent to the email they signed up with to verify their profession. | System displays a page for them to enter their information. |
| 2 | Journalist enters information about their profession and clicks submit. | System queries the database to store this information. |
| 3 | Database writes the Journalist’s profession information into the database. | Systems notifies Admin that there is a new Journalist who needs verification. |
|  | 4 | Admin verifies that the Journalist’s information is correct, and tells the system to contact the journalist. Or, admin marks the information as unverified, and tells the admin to contact the journalist. | System emails journalist with information specified by the admin. |
|  | 5 | Database marks the journalist as verified. |  |
| Notes and Issues: |  | | |
| Developer Notes: | N/A | | |

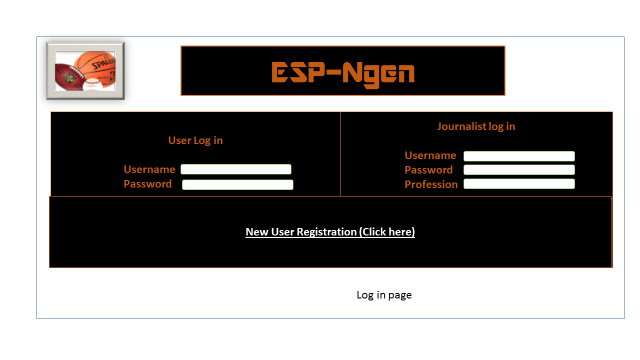
**Activity Diagrams:**

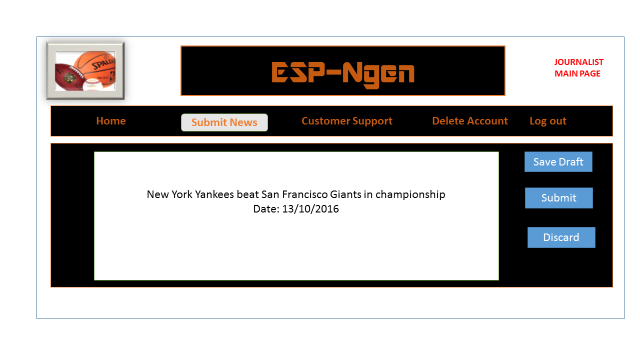
****

**Data Storage:**

Our initial choice for data storage was to use a MySQL database. This could be hosted inside of an RDS instance on AWS free tier, or a variety of other possible locations as well. However, as mentioned in lecture it seems like there is a possibility of Hibernate having advantages over MySQL, especially when used in conjunction with Java. Therefore, we have decided to further investigate Hibernate as well, and once we have a sufficient amount of information the final decision will be made.

**UI Mockups:**

****

****

**User Interactions:**

**Class Diagram:**