

<b>Use Case:</b> Access Breath of the Bull app via login
<b>Iteration:</b> To be implemented in third iteration
<b>Primary Actor:</b> Mobile app user
<b>Goal in Context:</b> To allow user to access the system under a created username from an android device.
<b>Preconditions:</b> User/Admin must have previously created an account with a username and password
<b>Trigger:</b> The user decides to log in to the Breath of the Bull application
<b>Scenario:</b> <ol style="list-style-type: none"> <li>1. User opens Breath of the Bull mobile app</li> <li>2. User is brought to welcome screen</li> <li>3. User selects login button</li> <li>4. User enters username and password (each at least eight characters in length, password requires at least one uppercase character and number)</li> <li>5. The system verifies the users login credentials</li> <li>6. The system brings user to home screen with all major functionalities</li> </ol>
<b>Exceptions:</b> <ol style="list-style-type: none"> <li>1. User enters invalid username or password - system will display error message stating that username or password is incorrect and user will be returned to login screen</li> <li>2.</li> </ol>
<b>Priority:</b> Moderate priority, due to time constraints of project, main application functionality should be implemented first
<b>When Available:</b> Third iteration of application
<b>Frequency of Use:</b> Required to use application
<b>Channel to Actor:</b> Via android mobile device
<b>Secondary Actors:</b> Admin
<b>Channels to Secondary Actors:</b> Admin - Via android mobile device
<b>Open Issues:</b> <ol style="list-style-type: none"> <li>1. How will passwords and usernames be securely stored?</li> <li>2. What if user forgets username or password?</li> <li>3. What if user data is corrupted and account cannot be accessed?</li> </ol>

<b>Use Case:</b> User requests to see profile information
<b>Iteration:</b> To be implemented in third iteration
<b>Primary Actor:</b> Mobile app user
<b>Goal in Context:</b> To view user profile information (username, preferences with mindfulness exercises and meditation sessions, statistics on app use)
<b>Preconditions:</b> User must be successfully logged in
<b>Trigger:</b> The user wishes to view profile information
<b>Scenario:</b> <ol style="list-style-type: none"> <li>1. The user selects 'Profile' button</li> <li>2. The user is brought to a screen that contains user information</li> <li>3. The user can view information on various aspects of the application and how the user interacts with them</li> </ol>
<b>Exceptions:</b> <ol style="list-style-type: none"> <li>1. It is the users first time logging in, and there is no information on usage - fields will appear unpopulated</li> </ol>
<b>Priority:</b> Low priority, due to time constraints of project, main application functionality should be implemented first
<b>When Available:</b> Third iteration of application
<b>Frequency of Use:</b> Moderate
<b>Channel to Actor:</b> Via android mobile device
<b>Secondary Actors:</b> Admin
<b>Channels to Secondary Actors:</b> Via android mobile device
<b>Open Issues:</b> No issues with this use case currently

<b>Use Case:</b> User requests daily quote
<b>Prototype Iteration:</b> To be implemented in second iteration
<b>Primary Actor:</b> User
<b>Goal in Context:</b> To receive the current day's quote from a zen practitioner/master
<b>Preconditions:</b> User must be successfully logged in

<b>Trigger:</b> The user wishes to view an inspiring/calming quote from a zen master to contemplate throughout the day
<b>Scenario:</b> <ol style="list-style-type: none"> <li>1. The user selects the 'View Daily Quote' button</li> <li>2. The system selects the quote corresponding to the current day (this will be done by having a list of quotes and iterating to the next quote in the list every time the clock of the device passes 12:00AM)</li> <li>3. The system presents the selected quote on the screen</li> <li>4. The user reads the quote</li> </ol>
<b>Exceptions:</b> <ol style="list-style-type: none"> <li>1. The user manually changes the time on his/her device - there is not much that can be done in this case, user will most likely not have a reason to do this</li> <li>2. The list of quotes has been completely iterated through and there are no new quotes - the system will start at the beginning of the list again</li> </ol>
<b>Priority:</b> High priority, this is a main functionality and should be implemented as soon as possible
<b>When Available:</b> Second iteration of application
<b>Frequency of Use:</b> Frequent
<b>Channel to Actor:</b> Via android mobile device
<b>Secondary Actors:</b> None
<b>Channels to Secondary Actors:</b> N/A
<b>Open Issues:</b> <ol style="list-style-type: none"> <li>1. Will the list of quotes be stored locally or on a database?</li> <li>2. Quote should be presented in a visually pleasing manner</li> </ol>

<b>Use Case:</b> User verifies successful mindfulness exercise
<b>Prototype Iteration:</b> To be implemented in second iteration
<b>Primary Actor:</b> User
<b>Goal in Context:</b> To let the system know the user has completed the mindfulness exercise upon notification from the system
<b>Preconditions:</b> User must be successfully logged in, user has outlined regular times that system cannot notify him/her to complete a mindfulness exercise
<b>Trigger:</b> System notifies user to complete a mindfulness exercise

**Scenario:**

1. User receives notification (vibration or bell/gong sound from mobile device) telling the user to complete a mindfulness exercise (stop and breath, be aware of world around you, thoughts, and what you are doing, etc.)
2. User either denies or confirms that they are doing the mindfulness exercise
3. System logs user input
4. User completes mindfulness exercise
5. This will repeat every hour (within outlined times) so that the user is practicing being present in his/her life each multiple times a day

**Exceptions:**

1. User does not confirm or deny that the exercise has been/is being completed - system will log the exercise as not being done by default after five minutes, this can be changed later in application if necessary
2. User presses wrong button (example: confirms when he/she meant to deny) - can be changed later in application if necessary
3. User has not given any outline of regular times the system cannot notify him/her - system by default operates between reasonable hours (9:00am to 5:00pm)
4. User does not want system to notify him/her at a specific time that is not regular - mindfulness functionality can be enabled/disabled
5. User forgets he/she has left the mindfulness functionality disabled - system sends a reminder if the functionality has been off for a week

**Priority:** High priority, this is a main functionality and should be implemented as soon as possible

**When Available:** Second iteration of application

**Frequency of Use:** Very Frequent

**Channel to Actor:** Via android mobile device

**Secondary Actors:** None

**Channels to Secondary Actors:** N/A

**Open Issues:**

1. Notifications should not being annoying, as to not discourage the user from completing exercises
2. Where will outline of regular times be stored?

**Use Case:** User begins/ends Meditation Session

**Prototype Iteration:** To be implemented in second iteration

**Primary Actor:** User

<b>Goal in Context:</b> To begin a meditation session
<b>Preconditions:</b> User must be successfully logged in
<b>Trigger:</b> User selects the 'Begin Meditation' button
<b>Scenario:</b> <ol style="list-style-type: none"> <li>1. User begins meditation session</li> <li>2. System brings user to meditation screen</li> <li>3. System guides user through a guided meditation generated from preferences that user has specified as well as inferences of user's preferences the system has made (guided meditation consists of visual instructions and auditory cues)</li> <li>4. User completes meditation session</li> <li>5. User is asked what aspects of the meditation he/she did/didn't like</li> <li>6. System exits meditation screen and returns user to home screen</li> </ol>
<b>Exceptions:</b> <ol style="list-style-type: none"> <li>1. User wishes to end meditation early - user can select the 'Finish Meditation' button to end session early</li> <li>2. User is interrupted during session - user can pause the ongoing guided meditation and resume once no longer distracted</li> <li>3. User accidentally gives inaccurate feedback - preferences can be updated in settings</li> </ol>
<b>Priority:</b> High priority, this is a main functionality and should be implemented as soon as possible
<b>When Available:</b> Second iteration of application
<b>Frequency of Use:</b> Frequent
<b>Channel to Actor:</b> Via android mobile device
<b>Secondary Actors:</b> None
<b>Channels to Secondary Actors:</b> N/A
<b>Open Issues:</b> <ol style="list-style-type: none"> <li>1. Should sessions be played as a video, or consecutive screens?</li> <li>2. How should preferences be stored?</li> </ol>

<b>Use Case:</b> User wants to change/view settings
<b>Prototype Iteration:</b> To be implemented in 3rd iteration
<b>Primary Actor:</b> User
<b>Goal in Context:</b> User wishes to view and change settings such as updating meditation

feedback, changing outline of regular times system cannot notify user for a mindfulness exercise, and enable/disable mindfulness exercises
<b>Preconditions:</b> User must be successfully logged in
<b>Trigger:</b> User selects the 'Settings' button
<b>Scenario:</b> <ol style="list-style-type: none"> <li>1. User is brought to the settings screen by the system</li> <li>2. User can alter current meditation preferences, or change mistakes made when providing feedback after a meditation session</li> <li>3. User can add/remove times from the list of 'Do not disturb' times (will be in the form of a week calender, so weekly times that user knows they do not want to be disturbed can be added)</li> <li>4. User can enable/disable mindfulness exercise notifications</li> <li>5. User saves changes</li> <li>6. System returns user home screen</li> </ol>
<b>Exceptions:</b> <ol style="list-style-type: none"> <li>1. User does not want to save changes - user can return to the home button via a back button</li> <li>2. User wants to clear settings - 'Restore to Default' button can be pressed and system will return all settings to default</li> </ol>
<b>Priority:</b> Moderate priority, due to time constraints of project, main application functionality should be implemented first
<b>When Available:</b> 3rd iteration of application
<b>Frequency of Use:</b> Moderate
<b>Channel to Actor:</b> Via android mobile device
<b>Secondary Actors:</b> Admin
<b>Channels to Secondary Actors:</b> Via android mobile device
<b>Open Issues:</b> No issues with this use case currently

<b>Use Case:</b> Add quote to quote list
<b>Prototype Iteration:</b> To be implemented in 3rd iteration
<b>Primary Actor:</b> Admin
<b>Goal in Context:</b> Admin wishes to add a quotation to the list of zen quotes

<b>Preconditions:</b> Admin must be successfully logged in
<b>Trigger:</b> Admin selects the 'Add new quote' button
<b>Scenario:</b> <ol style="list-style-type: none"> <li>1. Admin is brought to new quote screen</li> <li>2. Admin views current list</li> <li>3. Admin enters new quote</li> <li>4. Admin saves new quote</li> <li>5. Quote is added to the back of the quote list by the system</li> </ol>
<b>Exceptions:</b> <ol style="list-style-type: none"> <li>1. Quote is already on list - admin should view current list before adding new quote to ensure no duplicates are added</li> </ol>
<b>Priority:</b> Low priority, due to time constraints of project, main application functionality should be implemented first
<b>When Available:</b> 3rd iteration of application
<b>Frequency of Use:</b> Low
<b>Channel to Actor:</b> Via android mobile device
<b>Secondary Actors:</b> None
<b>Channels to Secondary Actors:</b> N/A
<b>Open Issues:</b> <ol style="list-style-type: none"> <li>1. Should there be a functionality for this, or should the list only be editable from the developer side?</li> <li>2. Is there a better way to ensure duplicate quotes aren't added?</li> </ol>