FINAL DRAFT:

* Support Forum:
  + Projected time (educated estimate): The Support Forum will take time estimation: 125 + 125\*0.7 ( to accommodate for research) = 212 hours to complete. (unit testing included)
    - Unit testing takes: 52 hours
    - Integration testing. This feature is reliant on 5 other feature, time estimate may take:
  + Possible complications (risk assessment, risk prevention, risk mitigation) will go here.
    - Risk assessment
      * Archived forum section does not show the locked threads.
      * Threads created do not show in the forum section
    - Risk prevention
      * Be sure thread maintainability is enforced and correctly archived for later viewing
      * Be sure that threads created are logged
      * Test to make sure the wanted output is resulted!
    - Risk mitigation (should the risk happen)
      * Show a message that thread was not posted onto the forum as needed.
      * Allow logging feature to implement a quick fix to logging threads.
  + What features logging interacts with will go here.
    - Dependent on the logging feature
    - Dependent on the archiving feature
    - Dependent on login
    - Dependent on network communication
    - Dependent on UI/UX
  + Test pass/ fail criteria:
    - All testing should pass 95% of the time.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_end**

**SUPPORT PAGE:**

**Phases:**

Phase 1: Design the database tables first. Using the links below will give an idea, but the database should be creating to tailor the requirements/ application needs.

* + <http://cisyst.com/Documents/Library/Design%20and%20Implementation%20Of%20A%20Web%20Forum.pdf>
  + <https://vertabelo.com/blog/database-model-for-an-online-discussion-forum-part-1/>
  + <https://laracasts.com/discuss/channels/general-discussion/database-schema-for-forum-like-laracsts>

Phase 2: look for/ research / study open source code to get an idea of what work goes into building a forum software. (here are some:)The reason for choosing to study open source code, is because it is hard to find a guide or best practices when building a discussion forum from scratch. There are too many third party work arounds, so the solution would be to study the approach taken with other web forums to learn from them.

* + <https://code.tutsplus.com/tutorials/how-to-create-a-phpmysql-powered-forum-from-scratch--net-10188>
  + <https://www.simplemachines.org/>
  + We can also download a free .NET forum to study and test how it works:
    - <https://www.jitbit.com/asp-net-forum/asp-net-forum-video-tutorial/>
  + <https://github.com/YAFNET/YAFNET>

Phase 3: Research youtube implementations/ approaches to creating a web forum also:

* + - <https://www.youtube.com/watch?v=eZKSzJhM8xk&t=358s>

Phase 4: Implementation forum structure in code from research and coding it. (after research we should get an idea of how to code design our forum)

Phase 5: implementation of thread maintainability of thread Locking.

* There should also be research on the implementation of archiving forums to be saved to an ‘Archived forum’ section but no longer editable and are just for viewing.
* implement weekly check for thread archiving if no posts within three days & has been posted for 30 days .

Phase 6: User usability and interface.

* Forum user formatting.implementation of thread format of 20,50 or 100 per page.
* Thread email sign up to a specific account level (basic, developer, admin, and vendor).
* Thread max character count implementation.

Phase 7: Implement a search bar for threads in the forum section.

* Research also autofill implementation as well

**Resources**

Phase 1: Design the database tables first to tailor the requirements/ application needs. **Estimated Time: 8 hours** ( taking into consideration restrictions). **Test time: 7 hours.** (insertion of data into tables and gathering of data for testing)

Phase 2: look for/ research / study open source code to get an idea of what work goes into building a forum software. **Estimated** **Time: 30 hours.** ( keeping in mind C# is still pretty new)

Phase 3: Research youtube implementations/ approaches to creating a web forum. **Estimated** **Time: 7 hours.**

**Phase 3: designing forum layout, time: 10hours**

Phase 4: Implementation forum structure in code from research and coding it. **Estimated** **Time: 50 hours. Test Time: 30hours**

Phase 5: Implementation of thread maintainability of thread Locking. **Estimated** **Time: 7 hours. Test time: 5 hours.**

Phase 6: User usability and interface. **Estimated** **Time: 5 hours. Test time: 5 hours.**

Phase 7: Implement a search bar (with autofill) for threads in the forum section. **Estimated** **Time: 10 hours.** **Test time: 5 hours.**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Total time to implement the support page: 115 hours**

8 + 50+15 = 73 hours + 52 = 125 + 125\*0.7 ( to accommodate for research) = 212 hours

**Interdependencies:**

This feature is dependent on Error handling (in the case thread creation, thread archiving fails), Logging and archiving feature, and UI/ UX.

* Constraints TO take into consideration:
  + user must register to edit and all users can view only
  + can sign up for thread notification
    - choose to sign up to notified by ‘specific’ user level
    - a Developer, Admin, Vendor, or other basic users
  + thread maintainability
    - weekly check: if no posts within three days & has been posted for 30 days
      * then: archive thread and save
      * archived threads in an archived forum section.
  + Forum user formatting
    - specified incrementations of 20,50 or 100 per page
    - Accounts can search for specific threads by tile in the Q&A forum section
  + Posting text in no more than 10,000 characters long
* Resources necessary
  + Amount of time it will take for the feature to be completed and start the feature
* Who will be responsible for its execution?
* Identify any interdependencies.
  + Do certain tasks need to be completed b4 others?
  + Do other features need to be completed before this feature?
* set milestones for overall project
* links/ references:

<https://www.youtube.com/watch?v=nItgtJwgV4Y&list=PL3_YUnRN3Uhiz2HomrXKcaEW6b3pDhKTX&index=1>

<https://www.c-sharpcorner.com/forums/how-to-create-forums-in-c-sharp>

<https://github.com/POPWorldMedia/POPForums>

<https://www.findbestopensource.com/tagged/forum-software>

<https://yetanotherforum.net/forum/>

<https://github.com/YAFNET/YAFNET>

* \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### *Support Page*

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

* **Test scenario 1: Test archiving functionality (test scenario)**
  + For particular scenario the expected behavior: (how the system will behave for the scenario)
    - Threads created by accounts will be logged and archived for later viewing after a specified time frame of 30 days with no user activity. The thread will then be held in the forum section for viewing only.
  + Test cases: (breaking down testing if the thread archiving is working)
    - Test that archiving happens after 30 day time period of no user activity
    - Test that once thread has been archived it is correctly moved to the archived forum section
    - Test that threads archived can no longer be editable by the user
  + Successful test scenarios:
    - Users can only view threads that have been archived in the archived forum section
    - Users can no longer view threads that have been archived in the main forum section. (only in the archived section)
    - Users do not have editing functionality on archived threads
  + Failed test scenarios:
    - Thread is not archived but instead there are two copies of the same thread.
    - Users are still able to edit archived threads
    - Threads are not being logged resulting in loss of thread once archived.
* **Test scenario 2: Test user formatting functionality (test scenario)**
  + The expected behavior:
    - Formatting filters such as searching and number of thread incrementations per page correctly display
  + Test cases
  + Successful test scenarios
  + Failed test scenarios
* **Test scenario 3: Test thread creation functionality (test scenario)**
  + The expected behavior:
    - Users should be able to apply the formatting filter
  + Test cases
  + Successful test scenarios
  + Failed test scenarios
* **Test scenario 4: Test user thread editing functionality (test scenario)**
  + The expected behavior:
    - Users should be able to apply the formatting filter
  + Test cases
  + Successful test scenarios
  + Failed test scenarios
* **Test scenario 5: Test user thread notification sign-up functionality (test scenario)**
  + The expected behavior:
    - Users should be able to apply the formatting filter
  + Test cases
  + Successful test scenarios
  + Failed test scenarios
* **Test scenario 6: Test user posting to a thread sign-up functionality (test scenario)**
  + The expected behavior:
    - Users should be able to apply the formatting filter
  + Test cases
  + Successful test scenarios
  + Failed test scenarios

**Will be used for reference:**

**Successful Test scenarios:**

* Forum archiving: ( what am I testing for?)
  + ( What am I testing for?)
  + Test that archived forum has been archived after 30 days of no activity
  + Test that User A cannot access archived threads for editing functionalities, only viewing
* Forum format:
  + Test the user A is able to adjust the number of threads viewed to correctly reflect (20, 50 and 100) increments
  + Test that recently created threads are correctly shown at the top (for more recent)
* Thread creation:
  + Test that thread creation can only be carried out by Accounts
  + Test that empty threads cannot be successfully created
  + Test that Account A did not reach maximum number of threads for the day (5 thread creations)
  + Test that the created thread is shown in the correct sub-forum after completion
  + Test that time stamp of creation is correctly reflected
  + Test that correct owner of thread is correctly reflected
* Thread editing
  + Test that Account A can ‘Delete’ their content in their thread created, and will not delete on cascade all posts made to the thread.
* Thread notification sign-up:
  + Test that Account A can sign up to a thread or post in the forum by receiving an email signup verification.
  + Test that Account A receives an email update on a previously followed thread that it has been posted to.
  + Test that Account A is able to specify who they would like to sign up to be notified of posts from ( “notify me when ‘admin’ responds” )
* Posts to threads:
  + Test that time stamps and User account are correctly accompanied with post within the thread
  + Posts within a thread are correctly timelined.
  + Test that Account A’s post deletion will not trigger a delete cascade on other users replies to Account A’s post.

**Failed Test scenarios:**

* Forum archiving:
  + Forums are archived below 30days of no activity
  + User A is unable to edit archived posts
* Forum format by user A:
  + is unable for format thread increments
  + chooses thread increment by x but is not correctly reflected as such
  + thread creation should displayed as most recent thread
* Account A Thread creation:
  + unable to create a thread
  + tries to post empty thread to forum
  + is able to post more than 5 threads a day
  + thread is not under the correct subforum after creation
  + thread created does not display correct timestamp or correct owner of thread.
* Thread editing
  + User As thread ‘deletion’ affects others posts to that thread and in turn deletes all posts made to it.
* Thread notification sign-up:
  + Account A is unable to sign up to a thread
  + Account A does not receive a verification email of thread sign up
  + Test that user A does not receive a email notification upon previously followed thread
  + Test that user A is unable to specify who they would like to sign up to hear posts from ( “notify me when ‘admin’ responds” ) and that user A is incorrectly assigned to hear from another account post not the one signed up to.
* Posts to threads:
  + Test that time stamps and User account are incorrectly accompanied with users As’ post within the thread
  + Test that user A’s post in correctly timelined
  + Test that user A’s post deletion will not delete all posts made to it.

### 

* An account attempts to create a thread.
  + **Expected:** The thread is successfully published under their respective sections in the support page( ex: Q&A forum) With the correct timestamp and username that made the creation.
  + Test cases: (how do you test that the expected is doing what it does correctly??)
    - Test that after the account creates a thread it is shown/ published in the respective forum.
    - Test that the thread creation is stored into the database for retrieval to be shown as ‘published’
    - Test that empty thread prevents the account from posting the thread.
    - Test that non account holders do not have the ability to create a thread
  + **Failure:** 
    - The thread is not created due to an error on our side.
    - An account is unable to post an empty thread.
    - The timestamp is either not correct or not displayed.
    - The username of the account that created a thread is either not correct or not displayed.
* An account tries to edit a thread that does not have replies.
  + **Expected**: The thread will be edited.
  + Test cases:
    - Test that non Account holders cannot edit any threads that are posted by other users.
    - Test that accounts are unable to edit other accounts threads
  + **Failure**:
    - The thread will not be edited
    - The thread will show “(edited)” even though there are no replies on it.
* An account tries to edit a thread that does have replies on it.
  + **Expected**: The thread will have an “(edited)” text symbolizing that the thread has been edited.
  + Test case:
    - Test that parent thread that is edited does not affect posts to that thread.
    - Test that edited thread is correctly displayed of changes made by account.
    - Test that once a thread has been edited an Account cannot revert to previous content.
  + **Failure**:
    - The thread will not be edited.
    - The thread will not show “(edited)” even though there are replies on the thread.
* An account edits a comment on a thread.
  + **Expected**: The comment is edited with an “(edited)” text by it, symbolizing that a comment has been edited.
  + Test cases:
  + **Failure**:
    - The comment has not been edited.
    - The comment does not show “(edited)” by it.
* An account tries to reply to another account’s thread.
  + **Expected**: The reply is successfully submitted under the respective thread. Along with the username of the commenter as well as the timestamp.
  + Test Cases:
    - te
  + **Failure**:
    - The reply is not posted
    - The username is not displayed alongside the comment.
    - The timestamp is either incorrect or not displayed alongside the comment.
* An unregistered user tries to reply to or create a forum post.
  + **Expected**: The user will be prompted to sign up / register or cancel.
  + **Failure**:
    - The user is not prompted to sign up or register.
    - The cancel option is not available for the user to cancel the dialog.
* An account tries to post more than 5 threads in a single day.
  + **Expected**:
    - The account will not be able to create a sixth thread, notifying them that they have reached their daily limit of thread creations.
    - If an account deletes a thread, the thread post is not subtracted from their daily limit count.
  + **Failure**:
    - The account is able to create a sixth thread.
    - The account is not notified that they have reached their daily limit of thread creations.
    - The account gains an additional thread post after deleting a thread.
* An account selects to be notified when replies are made to a thread.
  + **Expected**: The account is notified via email.
  + **Failure**:
    - The account is not notified via email.
    - A reply does not trigger a notification to be sent to the account that opted in for notifications of a thread.