

Test

Overview

The sample application maintains a list of telephone calls made by staff. The database is automatically populated every time a telephone call is made and records the user that made the call, when the call was made, which number was called and how long the call lasted for (in seconds).

We now want to extend the application with some additional functionality.

Tasks

Complete the following tasks and submit your completed solution as a zip file.

Task 1

The home page of the web application currently displays a list of all calls. Modify this page so that it displays:

- Calls sorted by User
- Then sorted by Date
- Grouped by User
- If the duration for any user for any given date exceeds a threshold of 30 minutes then all calls for that user for that day should be highlighted in red
- For each user and date add a subtotal row that shows the total duration (in minutes) for all calls for that user on that date

Example:

User Id	Date	Phone Number	Label	Duration
1	28/04/2016 07:39	078000845000	Record 00	1.52 (mins)
1	28/04/2016 08:39	0111624001	Record 01	1.45 (mins)
1	28/04/2016 16:39	0161700009	Record 09	2.78 (mins)
1	28/04/2016 17:39	078010845010	Record 10	1.7 (mins)
1	28/04/2016 18:39	0151753011	Record 11	4.23 (mins)
1	28/04/2016 19:39	078012845012	Record 12	2.6 (mins)
Total duration for user 1 on 28 April 2016				14.28 (mins)
1	29/04/2016 03:39	078020845020	Record 20	4.23 (mins)
1	29/04/2016 06:39	0121652023	Record 23	1.82 (mins)
1	29/04/2016 16:39	0131523033	Record 33	4.43 (mins)
1	29/04/2016 17:39	078034845034	Record 34	4.02 (mins)
1	29/04/2016 18:39	0151529035	Record 35	7.72 (mins)
1	29/04/2016 19:39	078036845036	Record 36	6.78 (mins)
1	29/04/2016 20:39	0171606037	Record 37	7.38 (mins)
1	29/04/2016 21:39	078038845038	Record 38	9.35 (mins)
1	29/04/2016 23:39	078040845040	Record 40	5.73 (mins)
Total duration for user 1 on 29 April 2016				51.47 (mins)

Task 2

Management have decided that the default view should have some simple filtering built in. The filters are:

- Show only calls made in the morning (<= 12:00pm)
- Show only calls made in the afternoon (> 12:00pm)
- Show only calls made to mobile numbers
- Show only calls made to landline numbers

Implement filtering on the home page and make it so that one filter can be selected at one time. You can implement the filtering either client side or server side – it is up to you.

Note: The changes you made in task 1 should persist, and the call threshold now applies to the filtered data. For instance, if you select the “morning calls only” filter and the sum of the duration for a user for applicable calls on a particular date exceed 30 minutes then all the calls should be red.

Example:

Choose a filter				
<div>No filterMorning CallsAfternoon CallsMobile NumbersLandline Numbers</div>				
User Id	Date	Phone Number	Label	Duration
1	28/04/2016 07:39	078000845000	Record 00	1.52 (mins)
1	28/04/2016 08:39	0111624001	Record 01	1.45 (mins)
Total duration for user 1 on 28 April 2016				2.97 (mins)
1	29/04/2016 03:39	078020845020	Record 20	4.23 (mins)
1	29/04/2016 06:39	0121652023	Record 23	1.82 (mins)
Total duration for user 1 on 29 April 2016				6.05 (mins)
1	30/04/2016 06:39	0151599047	Record 47	8.93 (mins)
Total duration for user 1 on 30 April 2016				8.93 (mins)
1	01/05/2016 01:39	078066845066	Record 66	6.25 (mins)
1	01/05/2016 02:39	0151705067	Record 67	4.1 (mins)
1	01/05/2016 06:39	0171699071	Record 71	6.3 (mins)
1	01/05/2016 08:39	0161694073	Record 73	7.7 (mins)
1	01/05/2016 09:39	078074845074	Record 74	1.07 (mins)
1	01/05/2016 10:39	0131564075	Record 75	7.6 (mins)

Task 3

Due to a problem with the recording of some calls from the telephone system we now need the ability to add the details of a call through the web application.

Create a form to add a new call record to the database.

The form should include some validation to ensure that a full record is entered every time.

To assist you in this process the application already includes a `CallDetailsService` which can handle the interaction between the Web Application and the repository layer. The method `AddCallDetailRecord` can be used to add a new record.

In addition, the `AddCallController` already exists and has a view model(`AddCallDetailViewModel`) which can be used.

Example:

Add call

User

Gordon Etherington ▼

Call date / time

29/04/2016 13:09:12

Duration (seconds)

50

Custom Label

This is free text

Phone Number

Please enter a phone number for this call

Save

Task 4

Users of the web application have expressed the need to be able to locate calls that match certain criteria. You need to create a search form that can query the database and retrieve calls that match their criteria.

The search fields that need to be available are:

- User Id
- Custom Label (whole string match only)
- Start date (find calls made on or after the date provided)
- End date (find calls made on or before the date provided)
- Phone number (whole number match only)

To help with this process we already have a search form available ("`~/Views/CallSearch/Search.cshtml`"). The controller has also been built.

You will need to implement the following 2 methods:

- **GetCallsBySearch(CallSearchTermsViewModel)** in **CallDetailsService.cs**; and
- **CallDetailsSearch(CallSearchTerms)** in **CallDetailsRepository.cs**

Task 5

The web application currently contains no unit tests. Write at least 2 unit tests to test any element of the functionality you have included in the previous tasks. Use a unit testing framework of your choice.