

3. Background service simulation

- As a user I want to have an application that simulates execution of a background service that saves logs of working application and frontend page where I can observe this records.

Flow described:

- + - Background task should be added that works on recurrent principle and is invoked in a given time span.
- + - Application should generate and save fake data into a database.
- + - Object to be generated and saved should look like:

long ID
string FirstName
string LastName
string Email
int Age
string IP Address
DateTimeOffset Timestamp

- + - All data should be faked
- + - New portion of 5 records should be added every 5 minutes
- + - As for frontend side on a main page (called Dashboard) I want to have a table where generated and stored data should be shown.
- + - By clickin on an 'View details' button placed for each table record I want to see a details of this record. Possible solutions for example would be: Popup or Expandable row
- + - I also want to have an ability to filter this data by any of displayed columns with a proper ordering principle (aplhabetical for strings; time comparison for Timestamp column, number comparison for Age)
- + - Data should be loaded page by page. Attempt to load all the records and send them to a Frontend will be treated as a failed task.
- + - At the bottom of a table I want to have an ability to select how big the page should be - 25, 50 or 100 items.
- + - At the bottom of a table I expect to see a pager where I can see how many items there are in total, pages buttons should be equal to actual count of pages with given page size

Acceptance criteria:

- Solution should not be based on Tasks and cannot be started from Controller call, Startup.cs etc.
- Data should be generated with exact delay
- I can observe generated items on a frontend
- Server-side pagination, filtering and sorting should be applied

Required technical stack: .NET Core 3, WEB API, Angular 8+, SQL Server

Recommended libraries: Hangfire/Quartz or custom solution based on IHostingService. Also it can be done with a new .NET Core 3 feature: Windows services or Linux daemons. To fake a data we recommend using <https://github.com/bchavez/Bogus> library.

Note: Attempt to start a new Task from running application will be treated as a failure. Solution has to be a background service in exact meaning.

Advanced level requirements:

- Use SignalR to display a newly generated records immediately on a frontend (by showing a notification like 'X new records has been added' and refresh button near)