

Generate an Angular project and add one input field to the main component. The input value should get sent to 2 asynchronous external services (Service1 and Service2, which should be mocked) in the following way:

1. Input value gets submitted for processing by pressing ENTER. Upon each submission the input field gets cleared.
2. While each value is being processed, users can submit more values that get processed in parallel.
3. Each submitted value goes into the following processing pipeline:
 - 3.1. If the input value contains only numeric characters, the value gets sent first to Service1. Upon successful response from Service1, value gets forwarded to Service2 together with the response from Service1. If the input value contains other than numerical characters, it skips Service1 and gets sent directly to Service2.
 - 3.2. If call to Service1 throws an exception the processing should automatically retry that call configurable amount of times. Each retry should be delayed with an incremented amount of milliseconds (each retry gets delayed longer and longer).
 - 3.3. If Service2 fails there is no retry.
4. If input value from the input field contains white characters, commas or semicolons, the whole value gets split by the aforementioned characters and each split element should be submitted separately into the processing pipeline as if the user submitted the split element separately. Empty values do not get into the processing pipeline.
5. The status of the processing of each submitted value should be displayed on the screen. It should look like a growing list of processing tasks under the input field.
 - 5.1. Each line of the list should display 'Processing with Service1' or 'Processing with Service2' accordingly. Also errors from respective services should get displayed.
 - 5.2. When Service1 fails after a configurable amount of retries, there should be 2 buttons displayed in the respective line: either to cancel that line of processing or to continue to Service2 without the value of Service1. After cancelling, a message "cancelled" should be displayed in that line.
6. On each error of any line there should be the whole page background flashing red for a moment with an animation, which can be toggled on/off with a radio button on top of the screen or an Observable<boolean> @Input to the component.

Remarks:

- please demonstrate functional and reactive programming techniques
- please mock both Service1, and Service2

