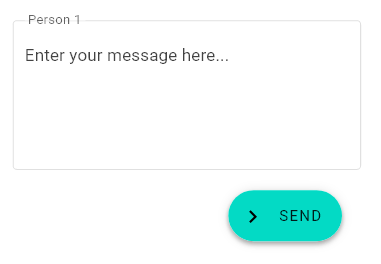
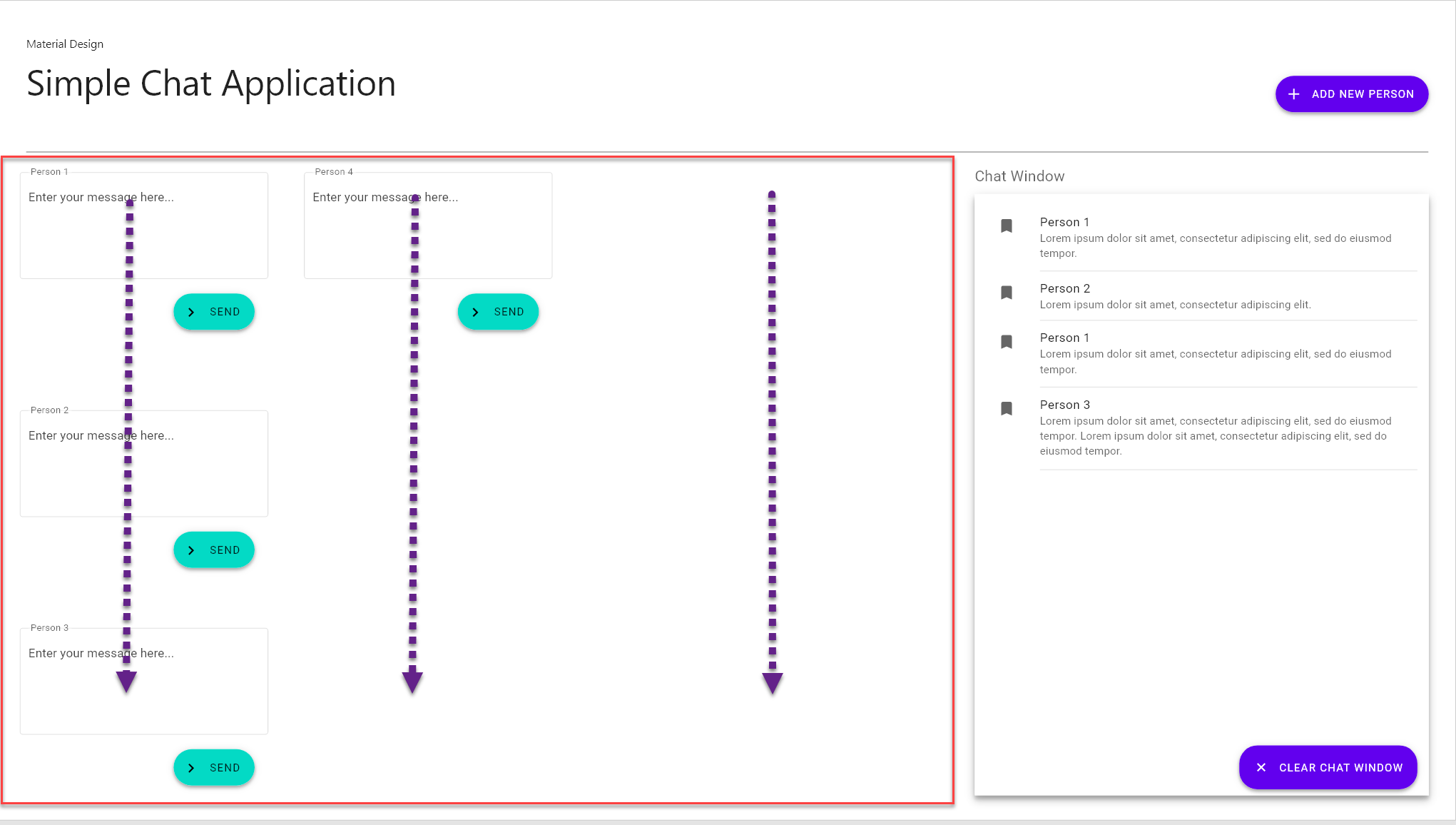
Simple chat application requirements:

1. A person component must be a separate component containing of material text field and a send button. Sending would be performed from within this component (not from a parent app.component.ts). Upon hitting the send button, text will be cleared for the user. User cannot send empty text (disable button, if message is empty).



1. Upon clicking **Add new person** button, located on the top right hand corner, add a new person component with incremented person index (so that they can be easily distinguished in the chat window). For the wrapper element (where person components are being rendered), you need to use the **css grid** with the following wrapping mechanism (3 column layout, as visible on the image below). You don't need to care about scrolling; you can limit number of total people in a chat to 9 (in order to avoid overflow).



1. Chat window should display message from each person in the defined fashion. Please use **mat-list** to achieve that.
2. Upon clicking on **Clear chat window**, you will remove all chat history data. Beware, that this does not serve as stop chat, but only to remove the history. You should still be able to send new messages after clearing the chat history.
3. For chat history and communication **you have to use RxJs**. Chat window should subscribe to Observable / Subject / (whatever you will find appropriate from RxJs family), which would render new message as soon as it's send by the person component.