Sentencer microservice specification

In this exercise you guys will take off the training wheels. Use `mix` to create your own project. Create a module according to the following specification:

- Make a start function that spawns the process.
- When sending a message in the format `{from, {:sentence, array_of_strings}}` return to `from` a message `{:ok, sentence}`.
- the input will be `array_of_strings` which is an array of strings…
- It will return `sentence` which is all the strings from the array concatenated with `,` as a separator and ending in a `.`(dot). The last two strings will be separated by an `e`. example: `array of strings: ["Uriel","Rebores", "Denis"] sentence: "Uriel, Rebores e Denis."`
- There must be a test suit containing these 4 tests
- Starting the service.
  - Test that the start returns the tuple `{:ok, sentencer_pid}` and that sentencer_pid is of the type pid.
- sending an array of size 1.
  - Test that it only puts a dot at the end `["Tuts"] -> "Tuts."`.
- Sending an array to be sentenced.
  - Test that your service works as stated for an array of size greater than 1.
- Sending an empty array.
  - Test it returns `""` as a sentence.

Tips: - Use a structure similar to ping_pong(start, loop with receive) - There is a function `is_pid/1` - `assert_receive/1` will be needed