de.idnow.example

- short description of solution proposed by Maria Barbosz –

Implemented solution:

In my view, the biggest priority is the SLA contract to be respected. So, for example 4, the priority is:

* Identification 2
* Identification 1

For computing the solution, I have implemented IdentificationService. This is algorithm:

1. For all companies that have pending identifications, calculate the slaDiff of percentages like:

slaDiff = CurrentSlaPercentage - SlaPercentage

1. If this difference is > 0 for all companies, then, in that moment, all companies are respecting their SLA contracts.

So, since there is no worry regarding the SLA contracts, from the identification pending list we select the identification that has the longest waiting time.

Also, we should compute the new CurrentSlaPercentage of the company of the identification selected (this is not done by the application because we do not have track of the number of the identifications already handled for this company)

1. If slaDiff <= 0 for some companies, then we select the company with the smalles slaDiff. This company needs the biggest help to achieve and respect the SLA contract.

For the selected company, we select the identification with the longest waiting time.

Then we need to compute the new CurrentSlaPercentage of the selected company, like on previous case.

How to run the tests:

* For running the examples described in the task, access one of the links:

<http://localhost:9000/test1>

<http://localhost:9000/test2>

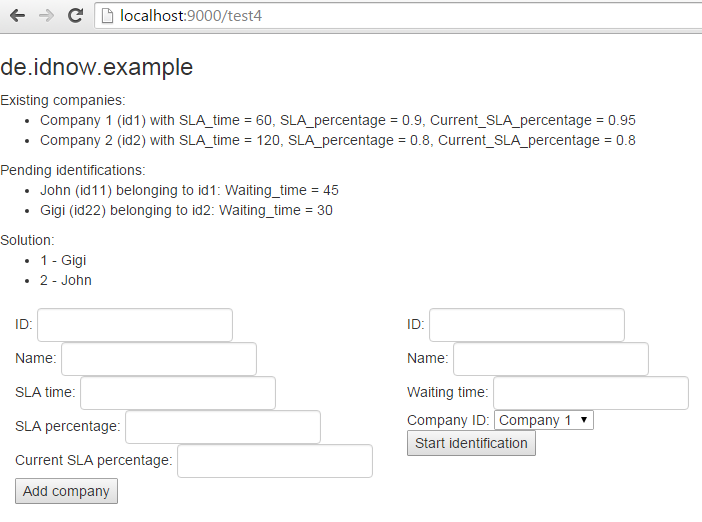
<http://localhost:9000/test3>

<http://localhost:9000/test4>

depending on which example you want to run in the browser. Also new data can be added to the existing tests. After the data is inserted, the browser refresh and the displays the new solution that includes the new data added.

* For running other examples, access link <http://localhost:9000> and you will have the possibility to add the data you want.

For example, this is the display of <http://localhost:9000/test4>



Things that I didn’t do:

I couldn’t use the on memory database, because of configurations problems, so I have created a DBEmulator, that acts like an application cache and keeps the companies and identifications. It also has an init method that can initialize the test cases. Anyway I would have preferred to use the persistence instead of this.

Also the imports are not optimized ☹ (also some local configuration issues)