**NOTAS**

* Herramientas usadas: Visual Studio Community 2019
* Repositorio GitHub: <https://github.com/marcetorterolo/MTR-SYSTwo>

Dicho repositorio es “público” por lo cual se puede acceder y ver los cambios realizados.

* En cuanto a las **PRUEBAS UNITARIAS** es la primera vez que me tocó hacerlas utilizando la herramienta de Visual Studio, pues anteriormente estaba acostumbrado a hacer pruebas caseras. Fue un desafío que me gustó.
* Si bien en la tarea se especificaba crear 3 computadoras, he creado un set de las mismas. Al momento de ejecutarse el controlador HomeController, se carga el juego de datos compuesto por: discos, procesadores y memorias. Con los componentes mencionados se crean distintas computadoras en el sistema.
* El nombre/identificador de una computadora está compuesto por los identificadores de sus componentes (disco-procesador-memoria)

DISCOS

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **CAPACIDAD** | **MARCA** | **TIPO** |
| 1 | 128 | Toshiba | SDD |
| 2 | 254 | Samsung | SDD |
| 3 | 1024 | Sony | HDD |

PROCESADORES

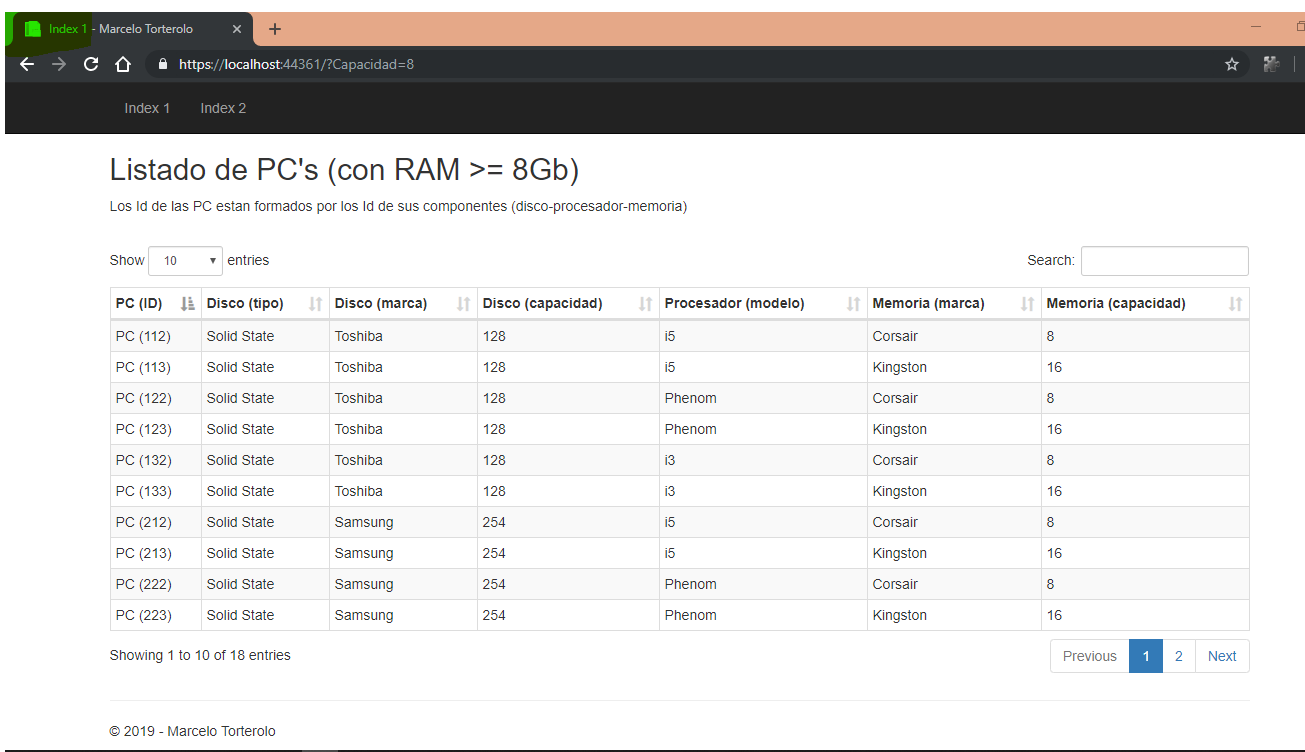
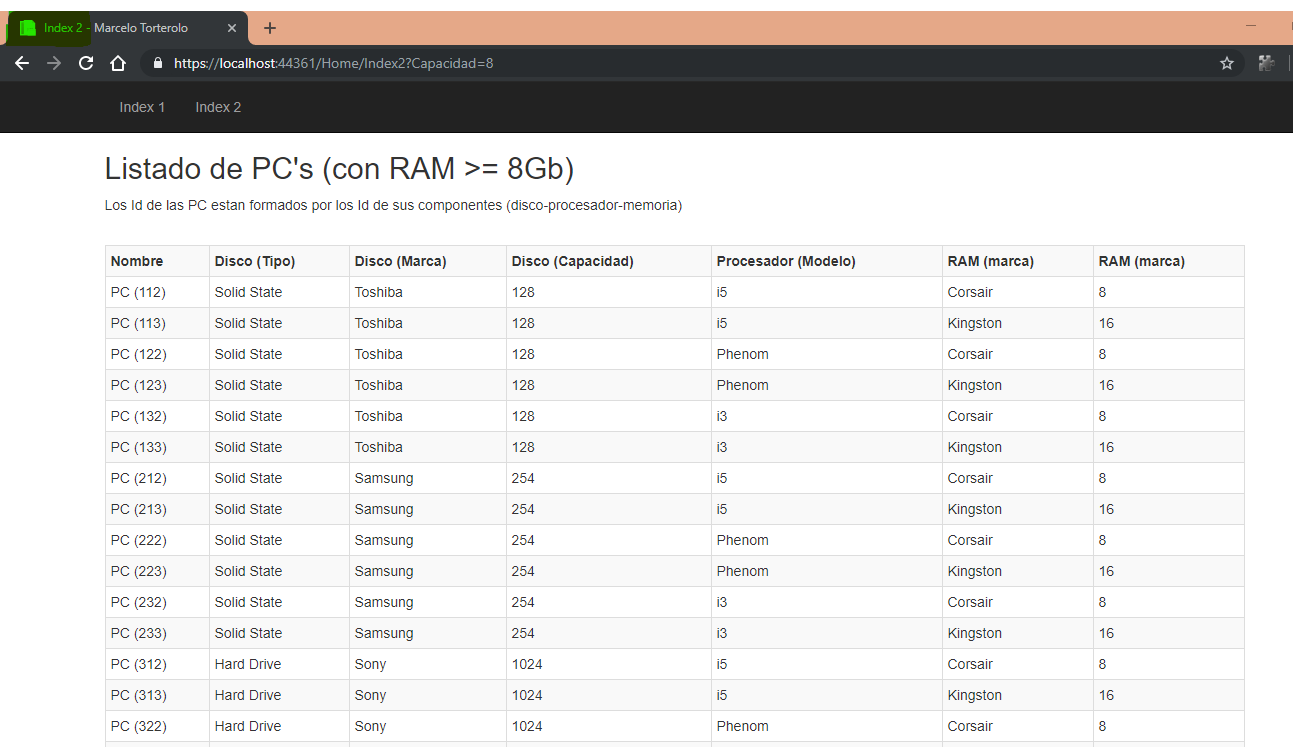
|  |  |  |
| --- | --- | --- |
| **ID** | **MARCA** | **MODELO** |
| 1 | Intel | i5 |
| 2 | AMD | Phenom |
| 3 | Intel | i3 |

MEMORIAS

|  |  |  |
| --- | --- | --- |
| **ID** | **CAPACIDAD** | **MARCA** |
| 1 | 4 | OCZ |
| 2 | 8 | Corsair |
| 3 | 16 | Kingston |

* Con el juego anterior, se crean un total de 27 computadoras en el sistema, pero en las grillas (y en el resultado jSON) se visualizan solamente aquellas con RAM >= 8 GB.
* En la aplicación se verán dos opciones(links) **Index 1** e **Index 2.**

La vista Index 1, se arma utilizando HTML, JavaScript, jQuery y Bootstrap (DataTable), mientras que la vista Index 2 es realizada con el motor de vistas RAZOR

* Finalmente el servicio /api/homeservice/8 retorna jSON con todas las computadoras en el sistema que cuya memoria ram sea mayor a 8 GB.

