**FUNCTIONAL REQUERIMENTS:**

**R#1.REGISTER:** The program must be able to register restaurants, clients, orders, or a product.

**R#2.UPDATE DATA:** program must have the possibility to update the data of a restaurant, client, product, or order. To find the object to update they count with an identificator like code, number, etc.

**R#3.SORT:** The client list must always be ordered alphabetically descending by surname and first name.

**R#4.CHANGE STATE:** The program must allow changing the status of an order between REQUESTED, IN PROCESS, SHIPPED and DELIVERED. You can change the order status forward but not backward.

**R#5.VERIFICATION:** it must be verified that all the products of an order belong to the restaurant that the client chooses to place the order.

**R#6.SAVE:** The program must save all its information through the serialization of its objects in files. Each time information is registered or updated it will be saved.

**R#7.EXPORT:** A csv file of orders must be generated, including for each requested product record, the data of the restaurant, the client, and the requested product. The user exporting this file must specify the separator. The csv file must be sort.

**R#8.IMPORT:** The program must allow importing data from a csv file with information of restaurants, clients, products, or orders.

**R#9:SHOW IN SCREEN:** The program must have an option that allows all restaurants or clients to be listed on the screen.

List of restaurants: they will be in ascending alphabetical order.

List of clients: in order of their descending phone number.

**R#10: FIND CLIENT:** The program must have an option that allows you to efficiently search for a client given a name and indicate the time that the search took.

**JUNIT TEST**

**CONFIGURATION OF THE STAGES:**

|  |  |  |
| --- | --- | --- |
| Name | Class | Stage |
| setupStage1 | ClientTest | Create a client |
| setupStage2 | ClientTest | Create a client with orders and set the state of the order in delivered |
| setupScenary1 | OrderTest | Create an order |
| setupScenary1 | ProductTest | Create a product |
| setupScenary1 | RestaurantAssociationTest | Create a restaurant association with a restaurant |
| setupScenary1 | RestaurantTest | Create a restaurant with a product |

**TEST CASE:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TEST OBJETIVE: verifies if the identification exception works | | | | |
| CLASS | METHOD | SCENARY | INPUT | RESULT |
| Client | Client | setupStage1 | IdentificationNumber:21212  Phone:32343  identificationType:PO  firstName:JUAN  LastName: GOMEZ  Addres: CIUDAD JARDIN | The Identification exception throws |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TEST OBJETIVE: Verifies if the changeStateException works | | | | |
| CLASS | METHOD | SCENARY | INPUT | RESULT |
| Client | chageState | setupStage2 | State: “REQUESTED” | The changeStateException throws |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TEST OBJETIVE:Verifies if the order makes with state of “REQUESTED” | | | | |
| CLASS | METHOD | SCENARY | INPUT | RESULT |
| Order | Order | setupScenary1 | CODE: 2121324  CLIENTCODE:232432  RESTAURANTNIT:23131  LIST<PRODUCT> | The new instance of order is correct |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TEST OBJETIVE:Verifies if the setState method works | | | | |
| CLASS | METHOD | SCENARY | INPUT | RESULT |
| Order | setState | setupScenary1 | State: “IN PROCCES” | The state was change |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TEST OBJETIVE:Verifies the correct update of all data | | | | |
| CLASS | METHOD | SCENARY | INPUT | RESULT |
| Product | updateAllData | setupScenary1 | CODE: 1212  COST:4343  NAME:”NOAJI”  DESCRIPTION:”NOPICA” | All data was update |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TEST OBJETIVE:Verifies de get name method works | | | | |
| CLASS | METHOD | SCENARY | INPUT | RESULT |
| Product | getName | setupScenary1 |  | The name of the product |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TEST OBJETIVE:verifies if the searchRestaurant method works | | | | |
| CLASS | METHOD | SCENARY | INPUT | RESULT |
| RestaurantAssociation | searchRestaurant | setupScenary1 | NIT: 23131 | Restaurant found |
| TEST OBJETIVE:verifies if the restaurant was correct register | | | | |
| CLASS | METHOD | SCENARY | INPUT | RESULT |
| RestaurantAssociation | register Restaurant | setupScenary1 | A restaurant | the restaurant was added |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TEST OBJETIVE:Verifies if a product adds successfully in the restaurant | | | | |
| CLASS | METHOD | SCENARY | INPUT | RESULT |
| Restaurant | addProduct | setupScenary1 | A Product | The product was added successfully |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TEST OBJETIVE: Verifies the search of a product | | | | |
| CLASS | METHOD | SCENARY | INPUT | RESULT |
| Restaurant | searchProduct | setupScenary1 | Code: 23232 | Product found |