| Client | Health provider institution |
| --- | --- |
| User | Receptionist |
| Functional Requirements | * R.1 Allow the patients to entry * Permitir el Ingreso de pacientes   + R.1.1 Register new patients   + Registro de Pacientes Nuevos   + R.1.2 Search patients that were previously added in the system   + Buscar Pacientes previamente registrados en el sistema * R.2 The program should be able to save the registered patient in a text file * El programa debe poder almacenar los pacientes registrados en un archivo de texto * R.3. The program should offer an option to attend the head of each queue * R.4 The program should be able to read a text file in order to load the patients into the application.   El programa debe poder leer un archivo de texto y a partir de este cargar los pacientes   * R.5 The program should have an undo option that allows the user to go back before the action was executed * El programa debe contar con una opción deshacer que permite regresar al momento antes de llevar a cabo una acción * R.6. The program should be able to print the row of each unit in order, and the patients available for check out   El programa debe de imprimir el orden de prioridad en las distintas filas en cada unidad, los presentes en el laboratorio y los pendientes a egreso  **Bonus:**   * **R.7** The program should automatically take out the patient located at the head of each queue every 1 to 2 minutes * El programa debe sacar automáticamente a la persona ubicada de primera en cada una de filas pasados entre 1 a 2 minutos * **R.8** The program should offer a manual checkout option * El programa debe ofrecer una opción de egreso manual |
| Context of the problem | Create preliminary version of a software for a laboratory clinic that allows to manage the check in and check out of the patients |

| Name | **R.1.1:** Register new patients | | |
| --- | --- | --- | --- |
| Summary | The system allows to register a new patient in the database with all the personal information and the correspondent unit | | |
| Inputs | **Input name** | **Data type** | **Selection or repetition condition** |
| name | String |  |
| id | long |  |
| unit | int | The value must be 1 or 2 |
| aggravation | int | The value must be a number between 1-4 |
| dateOfBirth | String | The input must be in the format year/month/day |
| admissionReason | String |  |
| General activities needed to obtain the results | The system read and transform the dateOfBirth input into a calendar variable  The system read and transform aggravation input input into a AggravationEnum  With the AggravationEnum the program calculate the priority of the patient | | |
| Result or Postcondition | Create the patient and register the patient in the system and in the clinical laboratory | | |
| Output | **Output name** | **Data type** | **Selection or repetition condition** |
| void | | |

| Name | **R.1.2:** Search patients that were previously added in the system | | |
| --- | --- | --- | --- |
| Summary | The system search in the database if the patient exist | | |
| Inputs | **Input name** | **Data type** | **Selection or repetition condition** |
| id | long |  |
| General activities needed to obtain the results | The system use the id to created a hashCode to search the patient in the dataBase | | |
| Result or Postcondition | Return a value that the system interpret to know if the patient it´s inside of the laboratory or if the patient exists in the dataBaser or don’t | | |
| Output | **Output name** | **Data type** | **Selection or repetition condition** |
| void | | |

| Name | **R.2:** Save the registered patient in a text file | | |
| --- | --- | --- | --- |
| Summary | The system creates a txt file in Json format that will contains the values to create every patient that was previously registered | | |
| Inputs | **There’s no input because the ArrayList that contains all patients is a global variable** | | |
|
| General activities needed to obtain the results | The system set the status of all the patients in the database how OUT\_OF\_HOSPITAL  The system get the AllPatient ArrayList (the dataBase) and transform it into a json format  The system looks for the path where it is going to save the data and writes the information to a file. | | |
| Result or Postcondition | A txt file that contains the patients data | | |
| Output | **Output name** | **Data type** | **Selection or repetition condition** |
| Void | | |

| Name | **R.3:** Manual Attendant | | |
| --- | --- | --- | --- |
| Summary | The system allows the user to manually attend the head of the queue that it selects | | |
| Inputs | **Input name** | **Data Type** | **Selection or repetition condition** |
| unit | int | The value must be 1 or 2 |
| General activities needed to obtain the results | Call the methods that extract the max of each priority queue and changes the statusPatient to TO\_CHECKOUT | | |
| Result or Postcondition | Print a message that notifies the user that the queue has advanced and sends the patient to wait for checkout | | |
| Output | **Output name** | **Data type** | **Selection or repetition condition** |
| Void | | |

| Name | **R.4.**Load patients | | |
| --- | --- | --- | --- |
| Summary | The system is able to read a text file in Json format in order to load the patients and all their information into the application . | | |
| Inputs | **Input name** | **Data type** | **Selection or repetition condition** |
| **There’s no input because the path used to save the file is already defined** | | |
| General activities needed to obtain the results | Read the data between the curly brackets and creates a Patient with the information inside of it, it will repeat until every curly bracket has been read | | |
| Result or Postcondition | An ArrayList with all the patients created | | |
| Output | **Output name** | **Data type** | **Selection or repetition condition** |
| sent | ArrayList<Patients> |  |

| Name | **R.5.**Undo option | | |
| --- | --- | --- | --- |
| Summary | The system is able to undo the last action executed by the user (register, attention and check out) | | |
| Inputs | **There is no input because the undo action checks the automatic updated stack of previously done actions** | | |
| General activities needed to obtain the results | There is at least one action executed by the user that has not been undone | | |
| Result or Postcondition | Patient is return to the “ available for check out” group, to the row he was in before or out of the lab (depending on the last action) | | |
| Output | **Output name** | **Data type** | **Selection or repetition condition** |
| Undo confirmation | String |  |

| Name | **R.6.**Show patients in the lab | | |
| --- | --- | --- | --- |
| Summary | The system is able to print the row of each unit in order, and the patients available for check out. | | |
| Inputs | **There is no input because this action check the data structures where patients are saved** | | |
| General activities needed to obtain the results | None | | |
| Result or Postcondition | the list of patients in Hematology and general unit in order and patients ready for check out are gathered in the variable “out” | | |
| Output | **Output name** | **Data type** | **Selection or repetition condition** |
| out | String |  |

| Name | **R.7.** Automatically advance the queue | | |
| --- | --- | --- | --- |
| Summary | The program should automatically take out the patients located at the head of each queue every 1 to 2 minutes | | |
| Inputs | **Input name** | **Data type** | **Selection or repetition condition** |
| None because it’s a scheduled runnable in periods | | |
| General activities needed to obtain the results | Call the methods that extract the max of each priority queue and changes the statusPatient to TO\_CHECKOUT | | |
| Result or Postcondition | Print a message that notifies the user that the queue has advanced and sends the patient to wait for checkout | | |
| Output | **Output name** | **Data type** | **Selection or repetition condition** |
| Void | | |

| Name | **R.8.** Manual Checkout | | |
| --- | --- | --- | --- |
| Summary | The program should offer an option that allows the user to manually check out a patient | | |
| Inputs | **Input name** | **Data type** | **Selection or repetition condition** |
| id | long |  |
| General activities needed to obtain the results | Call the methods that extract the max of each priority queue | | |
| Result or Postcondition | Change the statusPatient to OUT\_OF\_HOSPITAL | | |
| Output | **Output name** | **Data type** | **Selection or repetition condition** |
| out (confirmation message) | String |  |