**Functional requirements.**

**Req1.** Add a new user with its type of document, document number, names, last names, pone and address. The program will not allow to add a new user if there’s a user registered with the same information. Also, the program will not allow to add a user if the information “type of document, document number, names and last names” is empty.

**Req2.** Search a user by its document number. The program will search a user by its document number. If it find it, it will return the information about the user. Otherwise, the program will indicates that the user is not registered on the system.

**Req3.** Assign a shift to one user. The program will search a user by its document number and then it will assign the shift to this one. If the user is not found, the program will allow to register the user and it will assign the respective shift. If there’s no shifts on the system, the program will automatically assign a shift to the user.

**Req4.** Generates a new shift. The program will generates a new shift to assign. The format of every shift is a letter and a number (for example, the first shift will be A00 and the next one A01). Every time the number of the shift changes to 99 the shift will change the letter. If the shifts changed to the letter and number Z99, the shift will start again on the letter A.

**Req5.** Consult the next shift to attend. The program will show the next shift to attend assigned to a user. If there’s no shift to attend or shifts generated on the system, the program will notify the situation.

**Req6.** Attend a shift assigned to a user. The program will attend a user’s shift, searching the user and verifiying if both of them exist. If the user or the shift doesn’t exist the program will notify it.

**Test case design.**

**Stages.**

|  |  |  |
| --- | --- | --- |
| **Name** | **Class** | **Stage** |
| **setupStage1** | **ControlSystem** | **An object of the class ControlSystem.** |
| **setupStage2** | **ControlSystem** | **An object of the class ControlSystem with an user added with the attributes:**  **typeOfDocument = “Identity card”**  **documentNumber = “654789”**  **names = “Andres”**  **lastNames = “Perez”**  **phone = “”**  **address = “”** |
| **setupStage3** | **ControlSystem** | **An object of the class ControlSystem with 3 users added with the attributes:**  **typeOfDocument = “Identity card”**  **documentNumber = “654789”**  **names = “Andres”**  **lastNames = “Perez”**  **phone = “”**  **address = “”**  **typeOfDocument = “Foreigner Id”**  **documentNumber = “135791”**  **names = “Josh”**  **lastNames = “Stuart”**  **phone = “”**  **address = “”**  **typeOfDocument = “Identity card”**  **documentNumber = “246810”**  **names = “Antonio”**  **lastNames = “Uchiha”**  **phone = “”**  **address = “”** |
| **setupStage4** | **ControlSystem** | **An object of the class ControlSystem with some shifts generated.** |
| **setupStage5** | **ControlSystem** | **An object of the class ControlSystem with some shifts generated.** |
| **setupStage6** | **ControlSystem** | **setupStage2.**  **An object of the class ControlSystem with one shift generated and assigned to the user added previously.** |
| **setupStage7** | **ControlSystem** | **setupStage3.**  **An object of the class ControlSystem with some shifts generated and one shift assigned to every user added previously.** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test objective: Verify that the method addUser works adding a new user.** | | | | |
| Class | Method | Stage | Input values | Result |
| ControlSystem | addUser | setupStage1 | typeOfDocument = “Citizenship card”  documentNumber = “123456”  names = “Pepe”  lastNames = “Morales”  phone = “3137886655”  address = “Valle del lili” | The user is correctly added. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test objective: Verify that the method addUser add an user that is already registered.** | | | | |
| Class | Method | Stage | Input values | Result |
| ControlSystem | addUser | setupStage2 | typeOfDocument = “Identity card”  documentNumber = “654789”  names = “Andres”  lastNames = “Perez”  phone = “”  address = “” | The user is not added because is already registered on the system. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test objective: Verify that the method addUser add a new user if there are more users added.** | | | | |
| Class | Method | Stage | Input values | Result |
| ControlSystem | addUser | setupStage3 | typeOfDocument = “Citizenship card”  documentNumber = “1006048”  names = “Luis”  lastNames = “Hurtado”  phone = “”  address = “” | The user is correctly added. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test objective: Verify that the method searchUser returns an user that already exists on the program.** | | | | |
| Class | Method | Stage | Input values | Result |
| ControlSystem | searchUser | setupStage2 | documentNumber = “654789” | The method returns the user searched. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test objective: Verify that the method searchUser returns an user if the arrayList of users is empty.** | | | | |
| Class | Method | Stage | Input values | Result |
| ControlSystem | searchUser | setupStage1 | documentNumber = “654789” | The method returns nullPointerException because there were not users added. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test objective: Verify that the method searchUser returns an user with more users added.** | | | | |
| Class | Method | Stage | Input values | Result |
| ControlSystem | searchUser | setupStage3 | documentNumber = “246810” | The user is correctly found. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test objective: Verify that the method addShift generates the first shift A00.** | | | | |
| Class | Method | Stage | Input values | Result |
| ControlSystem | addShift | setupStage1 | None. | The first shift generated is the A00 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test objective: Verify that the method addShift generates the shift E00 if the shifts added finish in the shift D99** | | | | |
| Class | Method | Stage | Input values | Result |
| ControlSystem | addShift | setupStage4 | None. | The method correctly generates the shift E00. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test objective: Verify that the method addShift generates the shift A00 if the last shift generated was the Z99.** | | | | |
| Class | Method | Stage | Input values | Result |
| ControlSystem | addShift | setupStage5 | None. | The method correctly generates the shift A00. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test objective: Verify that the method addShift generates the shift A01 with one shift previously generated.** | | | | |
| Class | Method | Stage | Input values | Result |
| ControlSystem | addShift | setupStage6 | None. | The method correctly generates the shift A01. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test objective: Verify that the method consultShiftToAttend returns the next shift to be attended.** | | | | |
| Class | Method | Stage | Input values | Result |
| ControlSystem | consultShiftToAttend | setupStage6 | None. | The method correctly returns the shift to be attended. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test objective: Verify that the method attendUserShift attends the shift assigned to an user.** | | | | |
| Class | Method | Stage | Input values | Result |
| ControlSystem | consultShiftToAttend | setupStage7 | option = 2  shift = “A49” | The method attended the user’s shift and deleted the shifts that were attended. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test objective: Verify that the method assignShiftToUser assigns a shift to an user.** | | | | |
| Class | Method | Stage | Input values | Result |
| ControlSystem | assignShiftToUser | setupStage6 | None. | The method successfully assigned the shift to the user. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test objective: Verify that the method assignShiftToUser assigns a shift to an user with more users added.** | | | | |
| Class | Method | Stage | Input values | Result |
| ControlSystem | assignShiftToUser | setupStage2 | documentNumber = “654789”  shift = “A01” | The method successfully assigned the shift to the user. |