REQUISITOS:

* RF\_01 - Buscar por apellidos: Se realizará una búsqueda por apellidos y se mostrará por pantalla los datos de los usuarios. En caso de no encontrar ningún usuario que corresponda con los apellidos establecidos, se procederá a la activación del RF\_02.1.(Peroles) → 1
* RF\_02 - Gestión del usuario: Se dará de alta a un nuevo usuario ( RF\_02.1 ), modificar los datos de un usuario ya existente previa búsqueda ( RF\_02.2) o borrar un usuario existente ( RF\_02.3 ).
  + RF\_02.1 - Insertar usuario (Laura) → 2
  + RF\_02.2 - Modificar usuario ( Laura ) → 3
  + RF\_02.3 - Borrar usuario ( Laura ) → 4
* RF\_03 - Ordenar por apellidos: Se ordenará la agenda y se activará RF\_04. (Francisco) → 5
* RF\_04 - Mostrar por pantalla: Se imprimirá por pantalla el usuario o usuarios en texto plano o en HTML. (Francisco) → 6
* RF\_05 - Copia de seguridad: Se copiará a un fichero los datos para que no se pierdan. (Antonio) → 7
* RNF\_01 - Escrito en C++: Esta agenda será desarrollada en lenguaje de alto nivel C++.
* RNF\_02 - Se ejecuta en un ordenador con Linux: Utilizamos Linux porque es un SO que nos aporta herramientas para el desarrollo y compilación del programa y es de software libre.
* RNF\_03 - Nivel de seguridad cero: No necesitamos ningún sistema de seguridad para nuestro programa
* RNF\_04 - Integridad de los datos: Se debe mantener una copia de los datos en un fichero.
* RNF\_05 - Volumen máximo de datos de 200 entradas: Para evitar la saturación de datos, el programa mantendrá un límite de 200 entradas.
* RNF\_06 - Almacenamiento en texto plano/binario: Se almacenará en un texto plano/binario.

→ casos de uso

→ historia de usuario

Casos de uso:

|  |
| --- |
| Caso de uso: BuscarApellido |
| ID: 1 |
| Breve descripción: Se realizará una búsqueda por apellidos y se mostrará por pantalla los datos de los usuarios. En caso de no encontrar ningún usuario que corresponda con los apellidos establecidos, se procederá a insertar el usuario. |
| Actores principales: Administrativo |
| Actores secundarios: Dentista |
| Precondiciones: Que el usuario esté insertado en la base de datos del programa. |
| Flujo principal:   1. El caso de uso empieza cuando el Administrativo selecciona “Buscar apellido” 2. El sistema pregunta al administrativo el apellido 3. El administrativo escribe el apellido correspondiente. 4. El sistema busca en la base de datos el apellido que le ha facilitado el administrativo 5. Si el sistema encuentra el apellido:    1. El sistema muestra todos los datos que se correspondan con el apellido facilitado por el administrativo. 6. Sino:    1. El sistema no hace nada. |
| Postcondiciones: Ninguna |
| Flujo alternativo: Ninguno |

|  |
| --- |
| Caso de uso: InsertarUsuario |
| ID: 2 |
| Breve descripción: Se dará de alta a un nuevo usuario |
| Actores principales: Administrativo |
| Actores secundarios: Ninguno |
| Precondiciones: Ninguno |
| Flujo principal:   1. El caso de uso empieza cuando el Administrativo selecciona “Insertar usuario” 2. El sistema pregunta al administrativo los datos 3. El administrativo escribe los datos correspondiente. 4. El sistema guarda en la base de datos al nuevo usuario |
| Postcondiciones: Ninguna |
| Flujo alternativo: Ninguno |

|  |
| --- |
| Caso de uso: ModificarUsuario |
| ID: 3 |
| Breve descripción: Modificar los datos de un usuario ya existente previa búsqueda |
| Actores principales: Administrativo |
| Actores secundarios: Ninguno |
| Precondiciones: Que el usuario este en la base de datos |
| Flujo principal:   1. El caso de uso empieza cuando el Administrativo selecciona “Modificar usuario” 2. El sistema pregunta al administrativo el apellido del usuario a modificar 3. El administrativo le aporta el apellido cuyos datos se modificaran. 4. El sistema modifica los datos que se deseen modificar de la base de datos |
| Postcondiciones: Ninguna |
| Flujo alternativo: Ninguno |

|  |
| --- |
| Caso de uso: BorrarUsuario |
| ID: 4 |
| Breve descripción: Borrar un usuario existente |
| Actores principales: Administrativo |
| Actores secundarios: Ninguno |
| Precondiciones: Que el usuario este en la base de datos |
| Flujo principal:   1. El caso de uso empieza cuando el Administrativo selecciona “Borrar usuario” 2. El sistema pregunta al administrativo el apellido del usuario a borrar 3. El administrativo le aporta el apellido del usuario que se borrara. 4. El sistema borra el usuario con el apellido aportado. |
| Postcondiciones: Ninguna |
| Flujo alternativo: Ninguno |

|  |
| --- |
| Caso de uso: OrdenarAgenda |
| ID: 5 |
| Breve descripción: La agenda se ordenara por los apellidos de los usuarios. |
| Actores principales: Sistema |
| Actores secundarios: Administrativo |
| Precondiciones: Ninguna |
| Flujo principal:   1. El caso de uso se ejecuta automaticamente cuando insertan un nuevo usuario en el sistema. |
| Postcondiciones: Ninguna |
| Flujo alternativo: Ninguno |

|  |
| --- |
| Caso de uso: MostrarPantalla |
| ID: 6 |
| Breve descripción: Se mostrarán los datos del usuario o usuarios en la terminal o en un archivo de texto plano o HTML. |
| Actores principales: Administrativo |
| Actores secundarios: Ninguno |
| Precondiciones: Haber realizado anteriormente la búsqueda del usuario o usuarios. |
| Flujo principal:   1. El caso de uso se ejecuta al terminar el caso de BuscarApellido. 2. Mostrará por pantalla los datos del usuario/s previamente buscado/s. |
| Postcondiciones: Ninguna |
| Flujo alternativo: Ninguno |

|  |
| --- |
| Caso de Uso: Hacer copia de seguridad |
| ID: 7 |
| Breve descripción: Se realizará una copia de seguridad a un fichero con el objetivo de mejorar la seguridad. |
| Actores principales: Administrativo |
| Actores secundarios: Ninguno |
| Precondiciones: Que haya algún dato en la base de datos. |
| Flujo Principal:  13 El caso de uso empieza cuando el administrativo decide hacer copia de seguridad.  14 El sistema pide al administrativo confirmar la copia.  15 El administrativo confirma la copia de seguridad.  16 El sistema guarda en un fichero los datos de la base de datos. |
| Postcondiciones: Ninguna. |
| Flujo alternativo: Ninguno. |

Historias de usuario:

|  |
| --- |
| ID: 1 BuscarApellido |
| Descripción: Se realizará una búsqueda por apellidos y se mostrará por pantalla los datos de los usuarios. En caso de no encontrar ningún usuario que corresponda con los apellidos establecidos, se procederá a insertar el usuario.  Estimación: 4  Prioridad: 100  Dependencias: 2 |
| Pruebas de aceptación:   * Insertar un apellido erróneo y comprobar que no aparece ninguno de nuestra base de datos. |

|  |
| --- |
| ID: 2 InsertarUsuario |
| Descripción: Se dará de alta a un nuevo usuario.  Estimación: 6  Prioridad: 300  Dependencias: 0 |
| Pruebas de aceptación:   * Insertar un usuario que ya se encuentren en nuestra base de datos y que el sistema nos muestre un mensaje de error “usuario ya insertado”. * Insertar caracteres no apropiados para un apellido y que aparezca un mensaje de error “ no existen apellidos )%$123...” |

|  |
| --- |
| ID: 3 ModificarUsuario |
| Descripción: Modificar los datos de un usuario ya existente previa búsqueda.  Estimación: 5  Prioridad: 200  Dependencias: 1, 2 |
| Pruebas de aceptación:   * Insertar un usuario que no se encuentre en la base de datos y que envie un mensaje de error de “usuario no encontrado”. * Insertar un dato que no se encuentre en los específicos del cliente y que envie un mensaje de error “dato a modificar no encontrado”. |

|  |
| --- |
| ID: 4 BorrarUsuario |
| Descripción: Borrar un usuario existente.  Estimación: 2  Prioridad: 200  Dependencias: 1, 2 |
| Pruebas de aceptación:   * Buscar un usuario para borrar que no se encuentre en la base de datos y que envie un mensaje de error de “usuario no encontrado” |

|  |
| --- |
| ID: 5 OrdenarAgenda |
| Descripción: La agenda se ordenara por los apellidos de los usuarios, esto lo hara automaticamente el sistema al incluir un nuevo usuario.  Estimación: 4  Prioridad: 400  Dependencias: 2 |
| Pruebas de aceptación:   * Mostrar agenda completa y verificar el orden de esta. |

|  |
| --- |
| ID: 6 MostrarPantalla |
| Descripción: Se mostrarán los datos del usuario o usuarios en la terminal o en un archivo de texto plano o HTML.  Estimación: 6  Prioridad: 200  Dependencias: 1 |
| Pruebas de aceptación:   * Buscar un usuario/s por pantalla y sea el correcto. |

|  |
| --- |
| ID: 7 CopiaSeguridad |
| Descripción: Hacer copia de seguridad.  Estimación: 8  Prioridad: 400  Dependencias: 0 |
| Pruebas de aceptación:   * Si el número de registros del fichero no es igual al número de registro en la base de datos, corroboramos que la copia es errónea. |