|  |  |
| --- | --- |
| Requerimiento | RF1: Crear un clan |
| Resumen | Permite crear una cantidad indefinida de clanes al cual se le podrán agregar personajes |
| Entrada | Nombre al cual se le quiere asignar al Clan |
| Salida | El clan es creado a la lista |

|  |  |
| --- | --- |
| Requerimiento | RF2: Mostrar clanes |
| Resumen | Permite mostrar la información de los clanes que se han agregado al programa |
| Entrada | N/A |
| Salida | Los clanes son mostrados |

|  |  |
| --- | --- |
| Requerimiento | RF3: Actualizar información a un clan ya creado |
| Resumen | Permite cambiar los atributos de un clan como el nombre a otro ingresado por el usuario |
| Entrada | El nuevo nombre que el usuario le quiere asignar al clan |
| Salida | El nombre del clan es cambiado |

|  |  |
| --- | --- |
| Requerimiento | RF4: Borrar un clan creado anteriormente |
| Resumen | Permite borrar un clan que se ha creado previamente, lo buscara con su nombre para identificar que clan se le esta solicitando que se borre |
| Entrada | El nombre del clan que se quiere eliminar |
| Salida | El clan es eliminado / Información del clan eliminado al usuario |

|  |  |
| --- | --- |
| Requerimiento | RF5: Crear un personaje y añadirlo a un clan |
| Resumen | Permite crear un personaje con los atributos deseados por el usuario y añadirlos a un clan que se haya creado previamente |
| Entrada | El nombre, personalidad, fecha de creación y habilidad de poder del personaje y nombre del Clan. |
| Salida | El Clan es agregado |

|  |  |
| --- | --- |
| Requerimiento | RF6: Mostrar personajes de un clan |
| Resumen | Permite mostrar los personajes que hay en un clan especificado por el usuario por el nombre |
| Entrada | Nombre del clan del cual se quieren mostrar los personajes |
| Salida | Los clanes son mostrados |

|  |  |
| --- | --- |
| Requerimiento | RF7: Modificar atributos a un personaje |
| Resumen | Permite actualizar los atributos de un personaje de un clan por los nuevos deseados por el usuario |
| Entrada | El nuevo nombre, personalidad, fecha de creación y habilidad de poder del personaje y nombre del Clan. |
| Salida | Los atributos del personaje son actualizados / Información del personaje antes de actualizarse |

|  |  |
| --- | --- |
| Requerimiento | RF8: Eliminar un personaje |
| Resumen | Permite eliminar un personaje agregado a un clan previamente, buscando primero el clan solicitado por el usuario donde se encuentra el personaje a eliminar. |
| Entrada | El nombre del clan donde se encuentra el personaje, el nombre del personaje a eliminar. |
| Salida | El personaje es eliminado / Nombre de personaje eliminado |

|  |  |
| --- | --- |
| Requerimiento | RF9: Crear una técnica y añadirla a un personaje |
| Resumen | Permite crear una técnica y agregarla a un personaje que se encuentra en un clan especificado por el usuario |
| Entrada | Clan del personaje, nombre del personaje, nombre de la técnica, factor de influencia |
| Salida | La técnica es agregada / Personaje al que se le agrego la técnica |

|  |  |
| --- | --- |
| Requerimiento | RF10: Mostrar técnicas |
| Resumen | Permite mostrar todas las técnicas agregadas previamente a un personaje |
| Entrada | Clan donde pertenece el personaje, nombre del personaje |
| Salida | Muestra las técnicas de un personaje |

|  |  |
| --- | --- |
| Requerimiento | RF11: Eliminar una técnica |
| Resumen | Permite eliminar una técnica agregada a un personaje previamente |
| Entrada | Nombre del clan, nombre del personaje, nombre de la técnica |
| Salida | La técnica es eliminada / Muestra la técnica eliminada |

|  |  |
| --- | --- |
| Requerimiento | RNF 1: Utilizar el método de ordenamiento clásico burbuja |
| Resumen | Permite ordenar la lista doblemente enlazada de personajes mediante el ordenamiento de burbuja |
| Entrada | N/A |
| Salida | Lista ordenada de manera ascendente por burbuja |

|  |  |
| --- | --- |
| Requerimiento | RNF 2: Utilizar el método de ordenamiento clásico inserción |
| Resumen | Permite ordenar listas simplemente enlazadas de técnicas mediante el ordenamiento de inserción |
| Entrada | N/A |
| Salida | Lista ordenada de manera ascendente por inserción |

|  |  |
| --- | --- |
| Requerimiento | RNF 3: Utilizar el método de ordenamiento clásico selección |
| Resumen | Permite ordenar el ArrayList de clanes mediante el ordenamiento clásico de selección |
| Entrada | N/A |
| Salida | Lista ordenada de manera ascendente por selección |

|  |  |
| --- | --- |
| Requerimiento | RNF 4: Buscar datos de manera secuencial |
| Resumen | Permite hacer la busca de información mediante búsqueda secuencial |
| Entrada | 0bjeto o dato que se requiera buscar |
| Salida | Objeto o dato buscado |

|  |  |
| --- | --- |
| Requerimiento | RNF 5: Implementar la interfaz Comparable |
| Resumen | Permite implementar la interfaz comparable para definir que objeto es mayor o menor según el orden natural definido por el problema |
| Entrada | N/A |
| Salida | N/A |

|  |  |
| --- | --- |
| Requerimiento | RNF 6: Implementar la interfaz Comparator |
| Resumen | Permite implementar la interfaz comparator para definir que objeto es mayor o menor según dos objetos dados |
| Entrada | Dos objetos para comparar |
| Salida | Objeto mayor |

|  |  |
| --- | --- |
| Requerimiento | RNF 7: Ordenar las técnicas ascendentemente |
| Resumen | Ordena todas las técnicas de un personaje de manera ascendente por el factor de influencia |
| Entrada | N/A |
| Salida | Las técnicas son ordenadas de manera ascendente |

|  |  |
| --- | --- |
| Requerimiento | RNF 8: El nombre del personaje no se debe repetir |
| Resumen | Permite validar que los nombres de personajes no se repitan a la hora de agregar nuevas técnicas o personajes |
| Entrada | Nombre del personaje |
| Salida | Valida |

|  |  |
| --- | --- |
| Requerimiento | RNF 9 El nombre de la técnica no se debe repetir |
| Resumen | Permite validar que los nombres de las técnicas no se repiten en ningún momento de ejecución del programa |
| Entrada | Nombre de la técnica |
| Salida | Valida |

|  |  |
| --- | --- |
| Requerimiento | RNF 10: Los personajes deben ser implementados usando una lista doblemente enlazada |
| Resumen | Permite almacenar a los personajes en una lista doblemente enlazada |
| Entrada | Personajes que se quieren a agregar |
| Salida | N/A |

|  |  |
| --- | --- |
| Requerimiento | RNF 11: Los tecnicas deben ser implementadas usando una lista enlazada sencilla |
| Resumen | Permite almacenar las técnicas de los personajes en una lista simplemente enlazada |
| Entrada | Técnicas que se quieren agregar al personaje |
| Salida | N/A |