|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Clase** | **Método** | **Escenario** | **Valores entrada** | **Resultado** |
| TechniqueTest | compareTo(Technique); | setupStage(); | ------------------- | Es verdadero,  Puesto que retorna que es mayor |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Clase** | **Método** | **Escenario** | **Valores entrada** | **Resultado** |
| Character | deleteTechnique(String); | setupStage(); | ------------------- | Es verdadero,  Puesto que elimino |
| Character | findTechnique(String); | setupStage(); | ---------------------- | Es verdadero, puesto que retorno el toString() correspondiente |
| Character | ordeningFactor (); | setupStage(); | ---------------------- | Es verdadero, puesto que ordeno correctamente |
| Character | compareTo(Character); | setupStage(); | ------------------- | Es verdadero,  Puesto que retorna que es menor |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Clase** | **Método** | **Escenario** | **Valores entrada** | **Resultado** |
| Clan | ordeningAll(); | setupStage(); | ------------------- | Es verdadero,  Puesto que ordeno correctamente |
| Clan | findCharacter (String); | setupStage(); | ---------------------- | Es verdadero, puesto que retorno el toString() correspondiente |
| Clan | compare(Clan, Clan); | setupStage(); | ------------------- | Es verdadero,  Puesto que retorna que es mayor |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Clase** | **Método** | **Escenario** | **Valores entrada** | **Resultado** |
| Controller | ordeningClan(); | setupStage(); | ------------------- | Es verdadero,  Puesto que ordeno correctamente |

**REQUERIMIENTOS FUNCIONALES**

|  |  |
| --- | --- |
| RF1 |  |
| RESUMEN | Este requerimiento permite registrar un clan en la base de datos |
| ENTRADA | * Nombre |
| SALIDA | El clan fue registrado correctamente |

|  |  |
| --- | --- |
| RF2 |  |
| RESUMEN | Este requerimiento permite registrar un personaje en la base de datos |
| ENTRADA | * Nombre * Personalidad * Fecha creación personaje * Poder * Técnica usada |
| SALIDA | El personaje fue registrado correctamente |

|  |  |
| --- | --- |
| RF3 |  |
| RESUMEN | Este requerimiento permite registrar una técnica en la base de datos |
| ENTRADA | * Nombre de la técnica * Factor influyente |
| SALIDA | La técnica fue registrada correctamente |

|  |  |
| --- | --- |
| RF4 |  |
| RESUMEN | Este requerimiento permite eliminar un clan de la base de datos |
| ENTRADA | * Nombre del clan |
| SALIDA | El clan fue eliminado correctamente |

|  |  |
| --- | --- |
| RF5 |  |
| RESUMEN | Este requerimiento permite eliminar un personaje de la base de datos |
| ENTRADA | * Nombre del personaje |
| SALIDA | El personaje fue eliminado correctamente |

|  |  |
| --- | --- |
| RF6 |  |
| RESUMEN | Este requerimiento permite eliminar una técnica de la base de datos |
| ENTRADA | * Nombre de la técnica |
| SALIDA | La técnica fue eliminada correctamente |

|  |  |
| --- | --- |
| RF7 |  |
| RESUMEN | Este requerimiento permite ordenar ascendentemente todas las técnicas que tiene un personaje. |
| ENTRADA |  |
| SALIDA | Las técnicas del personaje ordenadas ascendentemente |

|  |  |
| --- | --- |
| RF8 |  |
| RESUMEN | Este requerimiento permite buscar un personaje por alguna de las dos estrategias de búsqueda (binaria o secuencial) |
| ENTRADA |  |
| SALIDA | Búsqueda realizada correctamente |

|  |  |
| --- | --- |
| RF9 |  |
| RESUMEN | Este requerimiento permite buscar una técnica por alguna de las dos estrategias de búsqueda (binaria o secuencial) |
| ENTRADA |  |
| SALIDA | Búsqueda realizada correctamente |

|  |  |
| --- | --- |
| RF10 | Verificar la existencia de un clan |
| RESUMEN | Este requerimiento no funcional permite verificar si existe el clan que el usuario va a registrar. |
| ENTRADA |  |
| SALIDA | Existe o no el clan |

|  |  |
| --- | --- |
| RF11 |  |
| RESUMEN | Este requerimiento funcional permite verificar si existe el personaje que el usuario va a registrar |
| ENTRADA |  |
| SALIDA | Existe o no el personaje |

|  |  |
| --- | --- |
| RF12 |  |
| RESUMEN | Este requerimiento no funcional permite verificar si existe la técnica que el usuario va a registrar. |
| ENTRADA |  |
| SALIDA | Existe o no la técnica |

|  |  |
| --- | --- |
| RF13 |  |
| RESUMEN | Este requerimiento funcional permite implementar una estrategia de búsqueda para utilizarlas en las funcionalidades del programa. |
| ENTRADA |  |
| SALIDA | Retorna el clan, personaje o técnica deseada. |

No funcionales

|  |  |
| --- | --- |
| RNF1 |  |
| RESUMEN | Este requerimiento no funcional permite implementar 3 métodos de ordenamiento en el funcionamiento del programa (burbuja, selección y inserción) |

|  |  |
| --- | --- |
| RNF2 |  |
| RESUMEN | implementar y utilizar las dos interfaces Comparable y Comparator en las funcionalidades de programa |