**Requirement Specification Document**

|  |  |  |  |
| --- | --- | --- | --- |
| Number | Type | Requirement Description | Implication |
| 1 | Functional | The ability to add a new Pokémon record to the storage. | User interface to enter Pokémon details |
| 2 | Functional | The ability to hold the data and information of Pokémon. | Use an array to store Pokémon data. Allow it to be dynamic. |
| 3 | Functional | The ability to search through stored data by National Dex Number, ID number or by Pokémon Types (Primary/Secondary). | Implement an appropriate searching algorithm based on a given field. |
| 4 | Functional | The ability to sort records by National Dex Number, ID Number or by Type. | Implement an appropriate sorting algorithm given a field to sort on. |
| 5 | Performance | Chosen sorting algorithm is most efficient for the selected data type(s). | Algorithm chosen must suit a dynamic array of classes. |
| 6 | Performance | Chosen data type(s) store the data in the most efficient way possible. | Limit the amount of unused space in data types by selecting the best data type. |
| 7 | Physical and Operational | A suitable user interface is chosen to allow for the user to easily navigate the Pokedex. | Create an interface that is easy and instinctive. |
| 8 | Functional | Allow Pokémon registered to have multiple typing. | Use of string within class and Boolean to show whether there is or is not a second type with string to show second if it is. |
| 9 | Functional | Allow for all Pokémon to be viewed at once, in order of ID Number. | Implement a sort and display the results in a 10 per page window |
| 10 | Functional | Allow the ability to view Pokémon that evolve and Pokémon that do not in separate lists. | Allow a search of Pokémon and only show ones with an evolution or prevolution |
| 11 | Functional | Allow the viewing of evolutions lines of Pokémon that do evolve. | Allow selection of Pokémon with evolutions and display that evolution tree |
| 12 | Operational | The date that a Pokémon is registered is recorded. | When a Pokémon is entered get system date and store it with the Pokémon |
| 13 | Operational | Keep records of Pokémon registered by the user, to be viewed whenever needed. | Allow entered Pokémon to be stored in data structure. |
| 14 | Operational | Allow Pokémon to be deleted from the system. | Allow the user the choice of deleting a Pokémon record. Then remove this record from our system. |
| 15 | Operational | Allow for the modification or updating of an existing record | Allow the user to make changes to each record then store this changes in the system |
| 16 | Functional | Stop duplication of previously entered Pokémon | If a Pokémon entered is already in the system don’t allow it to be entered twice. Ask if it needs to be edited |