**ANÁLISIS DEL PROBLEMA – TAREA INTEGRADORA II**

**Juan Jose Diaz Parra - A00381098**

**Requerimientos del problema:**

* **R1:** El usuario debe poder crear un humedal
* **R2:** El usuario debe poder registrar una nueva especie en el humedal
* **R3:** El usuario debe poder registrar un evento en el humedal
* **R4:** Se debe informar para cada humedal, la cantidad de mantenimientos en un año dado por el usuario.
* **R5:** Se debe poder desplegar el nombre del humedal con menos especies de flora.
* **R6:** El sistema debe poder desplegar los humedales donde se encuentre una especie, dado su nombre.
* **R7:** El sistema debe poder desplegar la información de todos nuestros humedales, incluyendo el total de especies por tipo (no incluya información de eventos).
* **R8:** El sistema debe poder desplegar el nombre del humedal con mayor cantidad de animales (aves, mamíferos y acuáticos)

**Análisis de requerimientos:**

**Requerimiento 1:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre o identificador** | **R1:** El usuario debe poder crear un humedal | | |
| **Resumen** | El usuario debe poder ingresar los datos necesarios para poder agregar un humedal al sistema. Estos datos son el nombre, la zona de ubicación (urbana, rural), tipo (público o privado), cantidad de km2, url de la foto y si se ha declarado área protegida. Si es rural se tendrá el nombre del corregimiento y si es urbano tendrá el nombre del barrio. Finalmente se debe mostrar un mensaje de confirmación si el humedal se registró adecuadamente. | | |
| **Entradas** | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| name | String |  |
| numLocation | int |  |
| numType | Int |  |
| área | double |  |
| urlOfThePicture | String |  |
| numProtectedArea | int |  |
| locationName | String |  |
| **Actividades generales necesarias para obtener los resultados** | Se debe ejecutar el metodo addWetland() en el objeto de la clase control, e insertar como parámetros los valores que se leyeron en la clase main. Finalmente la cadena que retorna ese metodo se debe imprimir en pantalla para informar al usuario si este proceso se hizo correctamente | | |
| **Resultado o postcondición** | Se agrega un nuevo humedal al arreglo de humedales que esta en la clase control | | |
| **Salidas** | Nombre salida | Tipo de dato | Condición de selección o repetición |
| stateOfTheProcess | String |  |

**Requerimiento 2:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre o identificador** | **R2:** El usuario debe poder registrar una nueva especie en el humedal | | |
| **Resumen** | El usuario debe poder ingresar los datos necesarios para poder agregar un evento a un humedal en sistema. Los datos que se necesitan son: El día, el mes, el año en que se realiza el evento, quien realiza el evento, el valor, una descripción y el tipo (mantenimientos, visitas de colegio, actividades de mejoramiento y celebraciones). Finalmente se debe mostrar un mensaje de confirmación si la especie se registró adecuadamente. | | |
| **Entradas** | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| nameWetland | String |  |
| nameSpecie | String |  |
| scientificName | String |  |
| optionMigratorySpecie | int |  |
| numType | int |  |
| **Actividades generales necesarias para obtener los resultados** | Se debe ejecutar el metodo addSpecie() en el objeto de la clase control, e insertar como parámetros los valores que se leyeron en la clase main. Finalmente, la cadena que retorna ese metodo se debe imprimir en pantalla para informar al usuario si este proceso se hizo correctamente | | |
| **Resultado o postcondición** | Se agrega una nueva especie al arreglo de especies que hay en el humedal que se seleccionó que esta en la clase control | | |
| **Salidas** | Nombre salida | Tipo de dato | Condición de selección o repetición |
| stateOfTheProcess | String |  |

**Requerimiento 3:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre o identificador** | **R3:** El usuario debe poder registrar un evento en el humedal | | |
| **Resumen** | El usuario debe poder ingresar los datos necesarios para poder agregar una especie a un humedal en sistema. Los datos que se necesitan son: nombre del humedal al que se va a agregar, nombre de la especie, nombre científico, si la especie es migratoria, tipo (flora terrestre, flora acuática, ave, mamífero, acuático). Finalmente se debe mostrar un mensaje de confirmación si el evento se registró adecuadamente. | | |
| **Entradas** | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| nameWetland | String |  |
| clientName | String |  |
| price | double |  |
| description | String |  |
| day | int |  |
| month | Int |  |
| year | Int |  |
| optionTypeEvent | int |  |
| **Actividades generales necesarias para obtener los resultados** | Se debe ejecutar el metodo addEvent() en el objeto de la clase control, e insertar como parámetros los valores que se leyeron en la clase main. Finalmente, la cadena que retorna ese metodo se debe imprimir en pantalla para informar al usuario si este proceso se hizo correctamente | | |
| **Resultado o postcondición** | Se agrega un nuevo evento al arreglo de eventos que hay en el humedal que se seleccionó que está en la clase control | | |
| **Salidas** | Nombre salida | Tipo de dato | Condición de selección o repetición |
| stateOfTheProcess | String |  |

**Requerimiento 4:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre o identificador** | **R4:** Se debe informar para cada humedal, la cantidad de mantenimientos en un año dado por el usuario | | |
| **Resumen** | El usuario debe poder ingresar el año en que desea hacer la búsqueda, y el sistema debe mostrar en pantalla la cantidad de mantenimientos que hubo en ese año en cada humedal. | | |
| **Entradas** | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| yearToSearch | int |  |
| **Actividades generales necesarias para obtener los resultados** | Se debe ejecutar el metodo numberMantenanceWetlands() en el objeto de la clase control, poniendo como parámetro el año que se ingresó en el metodo main. Finalmente, la cadena que retorna ese metodo se debe imprimir en pantalla para informar al usuario si este proceso se hizo correctamente | | |
| **Resultado o postcondición** | Se muestra en pantalla el numero de mantenimientos por cada humedal pero no se genera algún cambio dentro de los objetos. | | |
| **Salidas** | Nombre salida | Tipo de dato | Condición de selección o repetición |
| numberOfMantenance | String |  |

**Requerimiento 5:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre o identificador** | **R5:** Se debe poder desplegar el nombre del humedal con menos especies de flora. | | |
| **Resumen** | El sistema debe mostrar en pantalla el nombre del humedal que tiene menos especies de flora | | |
| **Actividades generales necesarias para obtener los resultados** | Se debe ejecutar el metodo nameWetlandWithLessFlora() en el objeto de la clase controladora. Lo que hace este metodo es iterar la lista de humedales buscando cual es el que tiene menos especies de flora. Finalmente se imprime en pantalla el mensaje que retorna este metodo. | | |
| **Resultado o postcondición** | Se muestra en pantalla el mensaje, pero no hay cambios en los objetos del sistema | | |
| **Salidas** | Nombre salida | Tipo de dato | Condición de selección o repetición |
| wetlandWithLessFlora | String |  |

**Requerimiento 6:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre o identificador** | **R6:** El sistema debe poder desplegar los humedales donde se encuentre una especie, dado su nombre. | | |
| **Resumen** | El usuario debe poder ingresar el nombre de una especie. El sistema deberá mostrar en pantalla todos los nombre de los humedales que tengan esa especie | | |
| **Entradas** | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| nameSpecie | String |  |
| **Actividades generales necesarias para obtener los resultados** | Se debe ejecutar el metodo wetlandsWithSameSpecie() en el objeto de la clase control, poniendo como parámetro el nombre de la especie que se ingresó en el metodo main. Finalmente, la cadena que retorna ese metodo se debe imprimir en pantalla, mostrando cuales son los humedales que tienen esa especie | | |
| **Resultado o postcondición** | Se muestra en pantalla los nombres de los humedales que tienen la especie que se ingresó en consola, pero no se genera algún cambio dentro de los objetos. | | |
| **Salidas** | Nombre salida | Tipo de dato | Condición de selección o repetición |
| namesWetlandsWithSameSpecie | String |  |

**Requerimiento 7:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre o identificador** | **R7:** El sistema debe poder desplegar la información de todos nuestros humedales, incluyendo el total de especies por tipo (no incluya información de eventos). | | |
| **Resumen** | El sistema deberá mostrar en pantalla la información de cada humedal, mostrando el valor de los atributos que tiene y mostrando la cantidad de especies que hay por los tipos (Flora acuática, flora terrestre, ave, mamífero, acuático). Sin embargo, no se deben mostrar la información de los eventos dentro del humedal | | |
| **Actividades generales necesarias para obtener los resultados** | Se debe ejecutar el metodo toString() en el objeto de la clase control, este itera el arreglo de humedales que tiene y ejecuta el metodo toString() de cada uno de ellos. Finalmente, la cadena que retorna ese metodo se debe imprimir en pantalla, mostrando toda la información requerida | | |
| **Resultado o postcondición** | Se muestra en pantalla la información de todos los humedales, sin embargo, no se genera algún cambio en los objetos del sistema | | |
| **Salidas** | Nombre salida | Tipo de dato | Condición de selección o repetición |
| informationWetlands | String |  |

**Requerimiento 8:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre o identificador** | **R8:** El sistema debe poder desplegar el nombre del humedal con mayor cantidad de animales | | |
| **Resumen** | El sistema debe mostrar en pantalla el nombre del humedal que tenga mayor cantidad de especies de tipo ave, mamífero, y acuático, es decir los animales. | | |
| **Actividades generales necesarias para obtener los resultados** | Se debe ejecutar el metodo nameWetlandWithMoreAnimals() en el objeto de la clase control. Este metodo lo que hace es iterar el arreglo de humedales que hay en la clase control y ejecutar su metodo numberOfAnimals() para comparar quien tiene el mayor número de humedales .Finalmente, la cadena que retorna ese metodo se debe imprimir en pantalla, mostrando cual es el nombre del humedal con la mayor cantidad de especies. | | |
| **Resultado o postcondición** | Se muestra en pantalla el nombre del humedal que tiene más animales, pero no se genera algún cambio dentro de los objetos. | | |
| **Salidas** | Nombre salida | Tipo de dato | Condición de selección o repetición |
| nameWetlandWithMoreSpecies | String |  |