**Identificación del problema y análisis de requerimientos**

|  |  |
| --- | --- |
| Cliente | Ministerio de Ambiente y Desarrollo Sostenible |
| Usuario | Colaboradores y empleados del ministerio y de la COP16 |
| Contexto del problema | *La sede de la COP16 ha sido elegida en Cali, un evento internacional que congrega representantes de más de 190 países para abordar los desafíos ambientales y la conservación de la biodiversidad. Para este evento, el alcalde ha solicitado el desarrollo de una aplicación que permita conocer las comunidades que cuidan lugares con diversidad biológica. Esta aplicación debe manejar información detallada sobre los lugares, incluyendo su ubicación, tipo, especies presentes y recursos necesarios, así como datos sobre las comunidades, como sus problemas principales y los productos que ofrecen. Además, la aplicación debe incluir un menú administrativo para ingresar y modificar datos, y un menú de consultas para acceder a información detallada sobre lugares y comunidades, así como datos específicos, como el lugar con mayor diversidad de especies y los más extensos por kilómetro cuadrado.* |
| Requerimientos funcionales | *RF1 - Registrar comunidades*  *RF2 - Registrar un producto a una comunidad*  *RF3 - Eliminar un producto a una comunidad*  *RF4 - Registrar lugares*  *RF5 - Registrar una especie a un lugar*  *RF6 - Modificar los datos registrados de una especie*  *RF7 - Consultar la información registrada de un lugar*  *RF8 - Consultar la información de las comunidades registradas*  *RF9 - Consultar la información de las comunidades donde falte un hospital o donde falte una escuela*  *RF10 - Consultar el nombre del lugar con mayor cantidad de especies*  *RF11 - Consultar los tres lugares con mayor área* |
| Requerimientos no funcionales | ***Usabilidad*** *-La aplicación debe ser intuitiva y fácil de usar para que los representantes de las comunidades y los administradores puedan ingresar y acceder a la información sin dificultad, incluso considerando la premura del evento.*  ***Eficiencia*** *- Dado que la aplicación debe manejar grandes volúmenes de datos sobre los lugares con diversidad biológica, comunidades y productos, es importante que el sistema pueda manejar eficientemente esta información para garantizar tiempos de respuesta aceptables.*  ***Seguridad*** *- Se debe implementar un sistema de autenticación y autorización para proteger la información sensible almacenada en la aplicación, ya que esta contendrá datos sobre comunidades, lugares con biodiversidad y productos que podrían ser sensibles y requerir protección.*  ***Escalabilidad*** *- Dado que la aplicación debe crecer y adaptarse para incluir información sobre más lugares, comunidades y productos, es esencial que el sistema sea capaz de escalar sin comprometer su funcionamiento.*  ***Mantenibilidad*** *- El código de la aplicación debe estar bien estructurado y documentado para facilitar su mantenimiento y futuras actualizaciones, considerando que el evento de la COP de biodiversidad es recurrente y es probable que se requieran modificaciones y mejoras en la aplicación.*  ***Portabilidad*** *- La aplicación debe poder ejecutarse en diferentes plataformas y dispositivos para garantizar su accesibilidad a una amplia gama de usuarios, incluidos los representantes de comunidades que pueden tener acceso a dispositivos móviles diversos.* |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Identificador y nombre | *RF1 - Registrar comunidades* | | | |
| Resumen | *Permite registrar información detallada sobre una comunidad, incluyendo su nombre, tipo étnico, representante, tamaño de la población, problemas principales enfrentados y una lista de productos que ofrece. A este requerimiento solo se tendrá acceso desde el menú administrativo.* | | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | | **Condición valores válidos** |
| community\_name | String | | *Máximo 40 caracteres* |
| ethnicity | String | | *Se tiene que escoger entre Afrocolombiana, Indigena y Raizal* |
| representative\_name | String | | *Máximo 40 caracteres* |
| representative\_phone | String | | Mínimo 5 dígitos máximo 15 |
| population | int | | Tiene que ser un número positivo entero |
| major\_problems | String | | *Se tienen que escoger los problemas mayores que se tengan en la comunidad. Se pueden escoger varios de las siguientes opciones: falta de hospital, falta de escuela, falta de agua potable, falta de acceso a alimentación básica* |
| offered\_products | String | | *Máximo 1000 caracteres* |
| Resultado o Postcondición | Los datos son registrados correctamente en el arreglo. Se muestra por pantalla el mensaje “Los datos han sido guardados”. | | | |
| Salidas | **Nombre salida** | | **Tipo de dato** | **Formato** |
| community\_message | | String | Se muestra un mensaje de éxito o de error al guardar la información de la comunidad |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Identificador y nombre | *RF2 – Registrar un producto a una comunidad* | | | |
| Resumen | *Permite agregar nuevos productos a la lista de productos ofrecidos por una comunidad. Solo se tendrá acceso a este requerimiento desde el menú administrativo.* | | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | | **Condición valores válidos** |
| community | String | | *Tiene que escoger una de las comunidades disponibles de la lista de comunidades registradas* |
| product\_name | String | | *Máximo 200 caracteres* |
| natural\_products\_percentage | double | | *El porcentaje tiene que estar en formato decimal “0.92”* |
| product\_type | String | | *Tiene que escoger entre Alimento o Artesanía* |
| handmade | String | | Tiene que escoger entre Si y No |
| Resultado o Postcondición | Los datos son registrados correctamente en el arreglo. Se muestra por pantalla el mensaje “Los datos han sido guardados”. | | | |
| Salidas | **Nombre salida** | | **Tipo de dato** | **Formato** |
| community\_product\_message | | String | Se muestra un mensaje de éxito o de error al guardar la información del producto que ofrece la comunidad |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Identificador y nombre | *RF3– Eliminar un producto a una comunidad* | | | |
| Resumen | *Permite eliminar productos existentes de la lista de productos ofrecidos por una comunidad. Se requiere especificar el producto que se desea eliminar, seleccionando de una lista de productos disponibles en la comunidad. Este requerimiento hace parte del menú administrativo, pero no tiene entradas, solo se mostrará por pantalla que el producto ha sido eliminado.* | | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | | **Condición valores válidos** |
| community\_name | String | | *Se selecciona una comunidad de la lista de comunidades registradas* |
| product\_name | String | | *Teniendo en cuenta la comunidad que seleccionó el usuario, se le muestra una lista de los productos que ofrece esta comunidad para que seleccione el que va a eliminar* |
| Resultado o Postcondición | Se mostrará por pantalla el mensaje “El producto ha sido eliminado”. | | | |
| Salidas | **Nombre salida** | | **Tipo de dato** | **Formato** |
| elimination\_message | | String | Se muestra un mensaje de éxito o de error al eliminar el producto que ofrece la comunidad |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Identificador y nombre | *RF4 - Registrar lugares* | | | |
| Resumen | *Permite registrar información sobre un lugar específico, como su nombre, ubicación geográfica (departamento), tamaño, tipo de área, fecha de inauguración, una imagen representativa, la comunidad encargada de su cuidado y recursos económicos necesarios.* | | | |
| Entradas | **Nombre entrada** | | **Tipo de dato** | **Condición valores válidos** |
| place\_name | | String | *Máximo 200 caracteres* |
| place\_department | | String | *Tiene que escoger entre Chocó, Valle del Cauca, Cauca o Nariño* |
| area | | double | *Tiene que ser el área en Km^2* |
| place\_type | | String | *Tiene que escoger entre Área Protegida, Parque Nacional o Área Privada* |
| opening\_date | | Date | *Tiene que ser en formato DD/MM/AAAA* |
| place\_photo | | image | *Tiene que ser una foto* |
| protective\_community | | String | *Tiene que ser el nombre de la comunidad encargada de cuidar el lugar* |
| maintenance\_budget | | double | *Tiene que ser el presupuesto que se tiene que tener para un buen cuidado del lugar.* |
|  | specie\_per\_habitad | | int | *máx 50 especies* |
| Resultado o Postcondición | Los datos son registrados correctamente en el arreglo. Se muestra por pantalla el mensaje “Los datos han sido guardados”. | | | |
| Salidas | **Nombre salida** | **Tipo de dato** | | **Formato** |
| place\_message | | String | Se muestra un mensaje de éxito o de error al guardar la información del lugar |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Identificador y nombre | *RF5 – Registrar una especie a un lugar* | | | |
| Resumen | *Permite registrar nuevas especies que habitan en un lugar específico, incluyendo su nombre, tipo (flora o fauna) y cantidad de ejemplares presentes. Este requerimiento hace parte del menú administrativo.* | | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | | **Condición valores válidos** |
| place\_name | String | | *Se muestra una lista de los lugares registrados para que el usuario seleccione donde se encuentra la especie que está registrando* |
| specie\_name | String | | *Máximo 40 caracteres* |
| specie\_type | String | | Flora o fauna |
| specie\_photo | image | | Imagen de la especie |
| amountByRegion | Int | | Número entero |
| Resultado o Postcondición | Los datos son registrados correctamente en el arreglo. Se muestra por pantalla el mensaje “Los datos han sido guardados”. Además, por cada especie que se guarda exitosamente se agrega 1 al número de especies que alberga ese lugar para luego mostrarlo en la consulta. | | | |
| Salidas | **Nombre salida** | | **Tipo de dato** | **Formato** |
| species\_message | | String | Se muestra un mensaje de éxito o de error al registrar una especie al lugar |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Identificador y nombre | *RF6 – Modificar los datos registrados de una especie* | | | |
| Resumen | *Este requerimiento se encarga de modificar la información almacenada sobre una especie en un lugar con diversidad biológica permitiendo al usuario la edición de esta información dentro de la aplicación* | | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | | **Condición valores válidos** |
| specie\_name | String | | *Máximo 40 caracteres* |
| specie\_type | String | | Flora o fauna |
| specie\_photo | image | | Imagen de la especie |
| amountByRegion | Int | | Número entero |
| Resultado o Postcondición | La información de la especie seleccionada se actualiza con los nuevos datos proporcionados por el usuario | | | |
| Salidas | **Nombre salida** | | **Tipo de dato** | **Formato** |
| modification\_message | | String | *Un mensaje de éxito o error en el proceso de modificación de los datos* |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Identificador y nombre | *RF7 – Consultar la información registrada de un lugar* | | | |
| Resumen | *En este requerimiento se permite la consulta de uno de los lugares de la lista de lugares registrados, y se muestran todos los datos relacionados a este* | | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | | **Condición valores válidos** |
| place\_name | String | | *Se muestra una lista de opciones de los lugares guardados, para escoger uno y mostrar la información de este.* |
| Resultado o Postcondición | El sistema muestra la información completa del lugar seleccionado, incluyendo su nombre, departamento, tamaño, tipo, fecha de inauguración, foto, comunidad que lo cuida, recursos económicos necesarios y las especies que habitan en él. | | | |
| Salidas | **Nombre salida** | | **Tipo de dato** | **Formato** |
| place\_name | String | | *Máximo 200 caracteres* |
| place\_department | String | | *Tiene que escoger entre Chocó, Valle del Cauca, Cauca o Nariño* |
| area | double | | *Es el área en Km^2* |
| place\_type | String | | *Muestra si el lugar es un Área Protegida, Parque Nacional o Área Privada* |
| opening\_date | Date | | *Fecha de inauguración del lugar* |
| place\_photo | image | | *Tiene que ser una foto* |
| protective\_community | String | | *Es la comunidad encargada de cuidarlo* |
| maintenance\_budget | double | | *Es el presupuesto que se tiene que tener para un buen cuidado del lugar.* |
| Species\_quantity | int | | *Es el número de especies que alberga el lugar* |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Identificador y nombre | *RF8 – Consultar la información de las comunidades registradas* | | | |
| Resumen | *Este requerimiento se encarga de mostrar la información de las comunidades de un departamento seleccionando, asumiendo que la información de las comunidades que va a mostrar son las comunidades que se encargan de cuidar los lugares que se encuentran en el departamento escogido* | | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | | **Condición valores válidos** |
| place\_department | String | | *Tiene que escoger entre Chocó, Valle del Cauca, Cauca o Nariño* |
| Resultado o Postcondición | El sistema muestra la información completa de las comunidades registradas en el departamento seleccionado, incluyendo su nombre, tipo, representante, cantidad de habitantes, problemas principales y productos ofrecidos. | | | |
| Salidas | **Nombre salida** | | **Tipo de dato** | **Formato** |
| name | String | | *El nombre de la comunidad* |
| ethnicity | String | | *La etnia de la comunidad* |
| representative\_name | String | | *Nombre del representante de la comunidad* |
| representative\_phone | String | | Numero de eléfono del representante de la comunidad |
| population | int | | Número de habitantes de la comunidad |
| major\_problems | String | | *Principales problemas de la comunidad* |
| offered\_products | String | | *Productos que ofrece la comunidad* |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Identificador y nombre | *RF9 – Consultar la información de las comunidades donde falte un hospital o donde falte una escuela* | | | |
| Resumen | *Permite identificar y mostrar información sobre las comunidades que enfrentan problemas específicos, como la falta de hospital o escuela. Este requerimiento funcional hará parte del menú de consulta, por lo tanto no tiene entradas, solo una salida, una lista de comunidades donde falte un hospital o escuela.* | | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | | **Condición valores válidos** |
| N.A | N.A | | *N.A* |
| Resultado o Postcondición | A partir de los datos guardados desde el menú administrativo, se mostrará por pantalla las comunidades donde falte un hospital o donde falte una escuela. | | | |
| Salidas | **Nombre salida** | | **Tipo de dato** | **Formato** |
| Comunidades donde falte hospital o escuela | | String | Lista de comunidades donde falte un hospital o escuela |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Identificador y nombre | *RF10 – Consultar el nombre del lugar con mayor cantidad de especies* | | | |
| Resumen | *Permite identificar y mostrar el nombre del lugar que alberga la mayor diversidad de especies. Este requerimiento funcional hará parte del menú de consulta, por lo tanto no tiene entradas, solo una salida, el lugar con mayor cantidad de especies.* | | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | | **Condición valores válidos** |
| N.A | N.A | | *N.A* |
| Resultado o Postcondición | A partir de los datos guardados desde el menú administrativo, se mostrará por pantalla el lugar con mayor cantidad de especies. | | | |
| Salidas | **Nombre salida** | | **Tipo de dato** | **Formato** |
| Lugar con la mayor cantidad de especies. | | String | *Se muestra por pantalla igual que como se guardó* |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Identificador y nombre | *RF11 – Consultar los tres lugares con mayor área* | | | |
| Resumen | *Permite identificar y mostrar los tres lugares con mayor extensión territorial en términos de kilómetros cuadrados. Este requerimiento funcional hará parte del menú de consulta, por lo tanto no tiene entradas, solo una salida, los tres lugares con mayor área.* | | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | | **Condición valores válidos** |
| N.A | N.A | | *N.A* |
| Resultado o Postcondición | A partir de los datos guardados desde el menú administrativo, se mostrará por pantalla los tres lugares con mayor área. | | | |
| Salidas | **Nombre salida** | | **Tipo de dato** | **Formato** |
| Tres lugares con la mayor área. | | String | *Se muestra una lista de los tres lugares con mayor área* |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| ReqFunc001  *Registrar comunidades* | Class Comunity | +addComunity(community\_name : String, type\_ehnicity : String, representant\_name : String, representant\_phone : String, population : int, mayor\_problems : String, offerd\_product : String) |
| Controller | +addComunity(community\_name : String, type\_ehnicity : String, representant\_name : String, representant\_phone : String, population : int, mayor\_problems : String, offerd\_product : String) |
|  |  |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| ReqFunc002  *Registrar un producto a una comunidad* | Comunity | +addProduct():Product |
| Controller | +Product(product\_name : String, percentage\_natural\_product : double, type\_product : string, handmade : String) |
| Product | +Product(product\_name : String, percentage\_natural\_product : double, type\_product : string, handmade : String) |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| ReqFunc003  *Eliminar un producto a una comunidad* | Comunity | +eliminateProduct(product\_name : String, percentage\_natural\_product : double, type\_product : string, handmade : String) |
| Controller | + eliminateProduct(product\_name : String, percentage\_natural\_product : double, type\_product : string, handmade : String) |
| Product | +Product(product\_name : String, percentage\_natural\_product : double, type\_product : string, handmade : String) |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| ReqFunc004  *Registrar lugares* | Place | +Place(place\_name : String, place\_department : String, area : double, type\_place : String, opening\_date : Date, place\_photo : String, maintenance\_budget : double, specie\_per\_habitad : int) |
| Controller | +addPlace(place\_name : String, place\_department : String, area : double, type\_place : String, opening\_date : Date, place\_photo : String, maintenance\_budget : double, specie\_per\_habitad : int) |
|  |  |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| ReqFunc005  *Registrar una especie a un lugar* | Specie | +Specie(specie\_name : String, specie\_type : String, specie\_photo : String, amount\_per\_place : int) |
| Controller | +Specie(specie\_name : String, specie\_type : String, specie\_photo : String, amount\_per\_place : int) |
| Place | +addspecie() : Specie |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| ReqFunc06  *Modificar los datos registrados de una especie* | Specie | +Specie(specie\_name : String, specie\_type : String, specie\_photo : String, amount\_per\_place : int) |
| Controller | +editSpecie(specie\_name : String, specie\_type : String, specie\_photo : String, amount\_per\_place : int) |
| Place | +addspecie() : Specie |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| ReqFunc07  *Consultar la información registrada de un lugar* | Place | +Place(place\_name : String, place\_department : String, area : double, type\_place : String, opening\_date : Date, place\_photo : String, maintenance\_budget : double, specie\_per\_habitad : int) |
| Controller | +consultPlace(place\_name : String, place\_department : String, area : double, type\_place : String, opening\_date : Date, place\_photo : String, maintenance\_budget : double, specie\_per\_habitad : int) |
|

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| ReqFunc08  *Consultar la información de las comunidades registradas* | Comunity | +Comunity(community\_name : String, type\_ehnicity : String, representant\_name : String, representant\_phone : String, population : int, mayor\_problems : String, offerd\_product : String) |
| Controller | +consultComunity(community\_name : String, type\_ehnicity : String, representant\_name : String, representant\_phone : String, population : int, mayor\_problems : String, offerd\_product : String) |
|

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| ReqFunc09  *Consultar la información de las comunidades donde falte un hospital o donde falte una escuela* | Comunity | +consultComunityProblems(tostringmayor\_problems : String) : Comunty List<NOT HOSPITAL, NOT SCHOOL>  +Comunity(community\_name : String, type\_ehnicity : String, representant\_name : String, representant\_phone : String, population : int, mayor\_problems : String, offerd\_product : String) |
| Controller | +consultComunityProblems(tostringmayor\_problems : String) : Comunty List<NOT HOSPITAL, NOT SCHOOL> |
|

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| ReqFunc10  *Consultar el nombre del lugar con mayor cantidad de especies* | Specie | +getAmount\_per\_place() : int |
| Controller | +consultPlaceAmounspecie(tostringsetAmount\_per\_place, tostringtPlace\_name) : Place |
| Place | +PlaceAmounspecie(tostringsetAmount\_per\_place, getPlace\_name) : void |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| ReqFunc11  *Consultar los tres lugares con mayor área* | Place | +places\_mayor\_area(tostringgetPlace\_name, tostringArea) : void  +getArea() : double  +getPlace\_name() : String |
| Controller | +places\_mayor\_area(tostringgetPlace\_name, tostringArea) : Place |
|

**A blue screen with text

Description automatically generated**