|  |  |  |  |
| --- | --- | --- | --- |
|  | | Diseñadores | Camilo Vargas Cabrera |
| **Análisis y Diseño Detallado** | | | |
| **Requerimientos funcionales** | | | |
| **RF 1:**  **Cargar el clima de todas las capitales del mundo** | **Resumen: Consulta al servicio de geolocalización de yahoo para obtener el woeid de cada capital, posteriormente consulta el clima de dicha capital.** | | |
| **Entradas: Nombres de las capitales** | | |
| **Resultados: Se ha inicializado la base de datos con la información de las capitales** | | |
| **RF 2: Consultar el clima de una ciudad** | **Resumen: Consulta el clima de una en base a su WOEID. Si ha pasado más de una hora desde la última consulta vuelve a solicitar el pronóstico al servicio de Yahoo!** | | |
| **Entradas: WOEID de la ciudad** | | |
| **Resultados: Arreglo que contiene las condiciones de clima para la ciudad dada** | | |
| **RF 3: Consultar el clima de una ciudad no registrada** | **Resumen: Utiliza el servicio de geolocalización para determinar el WOEID de la ciudad dado su nombre, posteriormente consulta el pronóstico para dicha ciudad utilizando en WOEID** | | |
| **Entradas: Nombre de la ciudad** | | |
| **Resultados: Arreglo que contiene las condiciones de clima para la ciudad dada** | | |
| **RF 4: Acceder al portal de administración** | **Resumen: Despliega el portal de administración si el nombre de usuario y la contraseña son correctos** | | |
| **Entradas: Nombre de usuario y contraseña** | | |
| **Resultados: Se despliega el panel de administración si el usuario y contraseña coincide. En caso contrario se despliega la página principal.** | | |
| **RF 5: Registrar un anuncio** | **Resumen: Registra un anuncio para una ciudad específica** | | |
| **Entradas: WOEID de la ciudad, Descripción, Link y Banner del anuncio** | | |
| **Resultados: Se ha registrado un nuevo anuncio** | | |
| **RF 6: Ver listado de anuncios** | **Resumen: Muestra el resumen de los anuncios registrados** | | |
| **Entradas: N/A** | | |
| **Resultados: Se muestra una tabla con la información de los anuncios registrados** | | |
| **RF 7: Ver historial de impresiones** | **Resumen: Se despliega junto con el resumen de los anuncios su número de impresiones** | | |
| **Entradas: N/A** | | |
| **Resultados: Se muestra una tabla con el número de impresiones de cada anuncio.** | | |
| **RF 8: Mostrar ciudades calidas** | **Resumen: Muestra las ciudades que tienen más de 20 grados** | | |
| **Entradas:N/A** | | |
| **Resultados:Una lista que contiene las ciudades que cumplen el criterio** | | |
| RF 9:  Dar Pronostico | **Resumen: Entrega la temperatura del día siguiente** | | |
| **Entradas: WOEID de la ciudad** | | |
| **Resultados: Arreglo que contiene las condiciones y pronóstico de clima para la ciudad dada** | | |

|  |
| --- |
| **Análisis del modelo del mundo** |
|  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Diseño de Operaciones (métodos): Para las interfaces relevantes en el diseño** | | | |
| ICupiClima | **public** String[][] buscarCiudad(String criterioBusqueda); | | Precondición: Se ha recuperado la instancia del mundo |
| Postcondición: Se genera un arreglo que contiene los resultados de la busqueda. |
| Excepciones: N/A |
| ICupiClima | **public** String[] darCondiciones(String woeid); | | Precondición: Se ha recuperado la instancia del mundo, se conoce un WOEID válido. |
| Postcondición: Se genera un arreglo que contiene las condiciones del WOEID pasado como parámetro |
| Excepciones: N/A |
| ICupiClima | **public** String[][] darFotos(String nombreCiudad); | | Precondición: Se ha recuperado la instancia del mundo |
| Postcondición: Se genera un arreglo que contiene la descripción y url de las imágenes coincidentes |
| Excepciones: N/A |
| ICupiClima | **public** String[][] darFotosWoeid(String woeid) **throws** Exception; | | Precondición: Se ha recuperado la instancia del mundo |
| Postcondición: Se genera un arreglo que contiene la descripción y url de las imágenes coincidentes |
| Excepciones: En caso de no encontrar ninguna imagen para el WOEID enviado como parámetro |
| ICupiClima | **public** IListaEncadenada<String> darCiudades(); | | Precondición: Se ha recuperado la instancia del mundo |
| Postcondición: Se genera una lista con los nombres y WOEID de las ciudades en la base de datos |
| Excepciones: N/A |
| ICupiClima | **public** **void** registrarAnuncio(String woeid, String texto, String link, String rutaImagen); | | Precondición: Se ha recuperado la instancia del mundo |
| Postcondición: Se registra un nuevo Anuncio relacionado con el WOEID pasado como parámetro |
| Excepciones: N/A |
| ICupiClima | **public** IListaEncadenada<String[]> darAnuncios(); | | Precondición: Se ha recuperado la instancia del mundo |
| Postcondición: Se genera una lista con la información relevante de todos los anuncios registrados |
| Excepciones: |
| ICupiClima | **public** IListaEncadenada<String[]> darCiudadesTemperatura(); | | Precondición: Se ha recuperado la instancia del mundo |
| Postcondición: Se genera una lista con las ciudades que cumplen la condición |
| Excepciones: |
|  |  | |  |
| **Diseño de Estructuras de Datos: Utilice un esquema de representación para ilustrar** | | | |
|  | | | |
| **Análisis de Complejidad : Para cada operación descrita en el diseño de operaciones, calcule y justifique la complejidad** | | | |
| buscarCiudad | | Solo se recuperan 10 resultados del servidor de yahoo | |
| O(1) | |
| darCondiciones | | Cualquier ciudad se ubica en máximo 37 pasos | |
| O(1) | |
| darFotos | | Solo se recuperan 15 fotos de Flirck | |
| O(1) | |
| darFotosWoeid | | Solo se recuperan 15 fotos de Flirck | |
| O(1) | |
| darCiudades | | Donde n es el número de ciudades registradas en la base de datos | |
| O(n) | |
| registrarAnuncio | | Cualquier ciudad se ubica en máximo 37 pasos | |
| O(1) | |
| darAnuncios | | Donde n es el número de anuncios registrados para dicha ciudad | |
| O(n) | |
| darCiudadesTemperatura | | Donde n es el número de ciudades registradas en la base de datos, en el peor de los casos… solo se recorre hasta que se encuentran 5 | |
| O(n) | |
|  | |  | |
|  | |
|  | |  | |
|  | |
|  | |  | |
|  | |

|  |
| --- |
| **Diagrama UML Final** |
|  |