

SISTEMA WEB DE INFORMACIÓN GEOGRÁFICA PARA LA GESTIÓN DE INDICADORES AMBIENTALES DEL OBSERVATORIO DE SOSTENIBILIDAD AMBIENTAL COSTATENAS

Ing. Adrián Marcos Quintana¹, Ing. Manuel Alejandro Naranjo Rey²,

Ing. Eduardo Berrio Turiño³

1, 2, 3. Universidad de Matanzas, adrian.marcos@umcc.cu

Resumen

El desarrollo de una herramienta informática que permita a los usuarios georreferenciar los distintos indicadores ambientales y acceder a la información de dichos indicadores sirve de gran ayuda para controlar y divulgar las acciones que se realizan con respecto al medio ambiente. Dedicada a la mejora de la naturaleza, la tecnología puede ser, además de rentable, una fuente de riqueza, innovación y de desarrollo industrial. Para el caso de las empresas matanceras, la gestión ambiental se realiza de manera ineficiente; existe demora en el proceso de obtención, caracterización y almacenamiento de la información. Siendo el objetivo de esta investigación, crear un Sistema de Información Geográfica que constituya un sistema de monitoreo y facilite la toma de decisiones. El resultado esperado es una herramienta que permitirá la representación georreferencial de los indicadores ambientales mediante diferentes capas de mapas, en las cuales se mostrará el estado actual de estos indicadores.

Palabras claves: Georreferenciar; indicadores ambientales; Sistema de Información Geográfica; tecnología; monitoreo.

Introducción

El rápido deterioro del medio ambiente es, actualmente de los problemas más graves que enfrenta la humanidad. Este problema afecta con más fuerza a los países subdesarrollados, debido a las ínfimas condiciones de vidas de la mayoría de sus habitantes. (Berrio Turiño, 2018)

Los avances tecnológicos ayudan a minimizar el impacto de la industria sobre el clima o la naturaleza, por eso estos deben extenderse al mayor número de empresas para que su efecto negativo sobre el Medio Ambiente sea el menor posible. Dedicada a la mejora de la naturaleza, la tecnología puede ser, además de rentable, una fuente de riqueza, innovación y de desarrollo industrial.

La información geográfica aporta un elevado nivel de detalle y de precisión a la hora de generar conocimiento implícito. Es, en este sentido que surgen los Geoportales, como sitios web donde se puede encontrar contenido geográfico.

Para el caso específico de las empresas del territorio matancero, la gestión ambiental se realiza de manera ineficiente; existe demora en el proceso de obtención, caracterización y almacenamiento de la información. Lo anterior deriva, en múltiples ocasiones, en malas interpretaciones de los datos procesados y, por ende, en erróneo proceso de toma de decisiones.

A lo dicho hasta el momento, se suma que actualmente la información generada en las empresas (entiéndase por indicadores ambientales) no es georreferenciada, ni representada

especialmente a través de ninguna de las herramientas automatizadas existentes con este fin. En el mejor de los escenarios, este proceso se realiza de forma manual con mapas físicos, lo que conlleva a pérdida de tiempo, de información y de recursos.

Como un intento de solución a los problemas de gestión ambiental empresarial antes mencionados, surge el Observatorio de la Sostenibilidad Ambiental Costatenas (OBSAMCostatenas). Una propuesta de la Universidad de Matanzas para centralizar y gestionar la información ambiental que se maneja en la provincia, desde una mirada científica. Es una herramienta tecnológica concebida modularmente en un entorno web, y que entre sus módulos comprende la implementación de un Geoportal que permita la representación espacial de los indicadores ambientales. Dicho en otras palabras, un sistema web de información geográfica para la gestión de indicadores ambientales.

El objetivo general que regirá la investigación es: desarrollar un sistema web para la gestión de la información geográfica de los indicadores ambientales del OBSAMCostatenas.

Caracterización de los Sistemas de Información Geográfica

Hoy en día, los seres humanos estamos creando y almacenando información constantemente. Pero no solamente somos los seres humanos quienes contribuimos a este crecimiento enorme de información, existe también la comunicación denominada máquina a máquina, cuyo valor en la creación de grandes cantidades de datos también es muy importante. (S/a, 2019)

Esa información podrá ser muy variada, pero toda comparte una cualidad: ocurre en algún lugar. El análisis de la especialización de toda esa información nos ayuda a ver qué, dónde y por qué está sucediendo en nuestro entorno. Es aquí donde juega un papel fundamental la geografía, la ciencia que estudia y describe las características de la superficie de la Tierra. (S/a, 2019)

Al final, nos encontramos con que tenemos una cantidad de datos para analizar, y una ciencia que nos describe el mundo en el que habitamos. Uniendo estas dos variables se obtiene el análisis de la información. Aquí es donde entran en juego los Sistemas de Información Geográfica (SIG o GIS, por sus siglas en inglés), que no son más que la conjunción de datos relacionados con el espacio físico con herramientas informáticas, es decir, con programas informáticos o *software*. (S/a, 2019)

Así pues, un Sistemas de Información Geográfica (SIG) es un conjunto de componentes específicos que permiten a los usuarios finales crear consultas, integrar, analizar y representar de una forma eficiente cualquier tipo de información georreferenciada (S/a, 2019), que no es más que la ubicación de información en el espacio (Pérez Porto, y otros, 2019). La información geográfica va a ser aquella información que tiene algún componente espacial, es decir, una ubicación, y, además, una información atributiva que nos detalle más sobre ese

elemento en cuestión. Esa ubicación se podrá definir con un nombre de una calle, por ejemplo, o con coordenadas espaciales. (S/a, 2019)

El uso de este tipo de sistemas facilita la visualización de los datos obtenidos en un mapa con el fin de reflejar y relacionar fenómenos geográficos de cualquier tipo, como es el caso de los distintos indicadores ambientales, los que se pueden describir como valores derivados de parámetros que describe el estado de un fenómeno, medio ambiente o un área, con un significado que se extiende más allá de que estén directamente vinculados con el valor de un parámetro. (Loné Pérez, 2016)

Los indicadores ambientales ofrecen una visión de las condiciones ambientales, son sencillos, fáciles de interpretar y capaces de mostrar las tendencias temporales, responden a cambios en el ambiente, proporcionan una base para las comparaciones internacionales y son aplicables a escala nacional o regional. Todas estas características hacen que sean útiles para monitorear las condiciones en las que se encuentran los distintos ecosistemas. (Loné Pérez, 2016)

Metodología de desarrollo de software

eXtreme Programming (XP) es una metodología ágil de desarrollo de software con el objetivo de conseguir un código de calidad y flexible y mejorar la productividad. Se trata de un método muy útil para abordar proyectos con requisitos cambiantes o poco definidos. La clave de esta metodología es que es adaptativa, es decir, hace hincapié en la adaptabilidad, más que en la previsibilidad, un criterio que deja de ser válido cuando entramos en el terreno de la incertidumbre y los cambios imprevistos. Para alcanzar el éxito en un proyecto de desarrollo, la metodología XP considera imprescindible potenciar las relaciones interpersonales, promover el trabajo en equipo y favorecer la comunicación entre sus miembros. (Rocha Garrido, 2019)

La metodología XP define cuatro variables para cualquier proyecto de software: costo, tiempo, calidad y alcance. Además, se especifica que, de estas cuatro variables, sólo tres de ellas podrán ser fijadas arbitrariamente por actores externos al grupo de desarrolladores (clientes y jefes de proyecto). El valor de la variable restante podrá ser establecido por el equipo de desarrollo, en función de los valores de las otras tres. Este mecanismo indica que, por ejemplo, si el cliente establece el alcance y la calidad, y el jefe de proyecto el precio, el grupo de desarrollo tendrá libertad para determinar el tiempo que durará el proyecto. (Joskowicz José, 2008)

Considero que XP es la metodología correcta para desarrollar el proyecto debido a que proporciona una gran garantía a la hora de implementar el sistema, esta se centra en realizar un producto con calidad y lo más rápido posible, también se puede realizar cambios durante el proceso de desarrollo ya que el cliente forma parte indiscutible en el equipo de desarrollo.

Además, al sustentarse en los valores antes mencionados, promueve el trabajo en equipo y la comunicación constante no solo con el equipo de desarrollo, sino también con el cliente.

Herramientas y Tecnologías

- *Pre Hypertext – Processor* (PHP): es un lenguaje de programación de uso general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico. Fue uno de los primeros lenguajes de programación del lado del servidor que se podían incorporar directamente en el documento HTML en lugar de llamar a un archivo externo que procese los datos. El código es interpretado por un servidor web con un módulo de procesador de PHP que genera la página Web resultante. PHP ha evolucionado por lo que ahora incluye también una interfaz de línea de comandos que puede ser usada en aplicaciones gráficas independientes. Puede ser usado en la mayoría de los servidores web al igual que en casi todos los sistemas operativos y plataformas sin ningún costo. (Q-Success, 2015)
- *Hyper Text Markup Language* (HTML): Es la pieza más básica para la construcción de la web y se usa para definir el sentido y estructura del contenido en una página web. "Hipertexto" se refiere a los enlaces que conectan las páginas web entre sí, ya sea dentro de un mismo sitio web o entre diferentes sitios web. los vínculos son un aspecto fundamental de la web. HTML usa "marcado" para anotar textos, imágenes y otro contenido para ser mostrado en un navegador web. (Mozilla, 2020)
- *Cascading Stylesheets* (CSS): Es el lenguaje utilizado para describir la presentación de documentos HTML. CSS describe como debe ser renderizado el elemento estructurado en pantalla, en papel, hablado o en otros medios. Es uno de los lenguajes base de la *Open Web*. Desarrollado en niveles, CSS1 es ahora obsoleto, CSS2.1 es una recomendación y CSS3, ahora dividido en módulos más pequeños, está progresando en camino al estándar. (Mozilla, 2020)

Al crear una página web, se utiliza en primer lugar el lenguaje HTML para marcar los contenidos, es decir, para designar la función de cada elemento dentro de la página: párrafo, titular, texto destacado, tabla, lista de elementos, etc. Una vez creados los contenidos, se utiliza el lenguaje CSS para definir el aspecto de cada elemento: color, tamaño y tipo de letra del texto, separación horizontal y vertical entre elementos, posición de cada elemento dentro de la página. (Eguiluz Javier, 2020)

- TypeScript: Es lo que se conoce como un superset de Javascript, es Javascript con nuevas utilidades que lo convierten en un lenguaje más completo. Entre otras muchas cosas, lo más particular de TypeScript es que incluye tipos en el lenguaje Javascript, para convertirlo en un lenguaje fuertemente tipado o de tipado estático. Es un lenguaje de programación de alto nivel que implementa muchos de los mecanismos más habituales de la programación orientada a objetos, pudiendo extraer grandes beneficios que serán

especialmente deseables en aplicaciones grandes, capaces de escalar correctamente durante todo su tiempo de mantenimiento. La característica fundamental de TypeScript es que compila en Javascript nativo, por lo que se puede usar en todo proyecto donde se esté usando Javascript. Dicho con otras palabras, cuando se usa TypeScript en algún momento se realiza su compilación, convirtiendo su código a Javascript común. El navegador, o cualquier otra plataforma donde se ejecuta Javascript, nunca llegará a enterarse que el código original estaba escrito en TypeScript, porque lo único que llegará a ejecutar es el Javascript resultante de la compilación. (S/a, 2016)

Herramientas de desarrollo

- PostgreSQL: Es un potente sistema de base de datos relacional de objetos de código abierto con más de 30 años de desarrollo activo que le ha valido una sólida reputación de fiabilidad, solidez de características y rendimiento. (The PostgreSQL Global Development Group, 2020)

Es gratuito y libre, además de que hoy nos ofrece una gran cantidad de opciones avanzadas. De hecho, es considerado el motor de base de datos más avanzado en la actualidad. Una característica interesante de PostgreSQL es el control de concurrencias multiversión. Este método agrega una imagen del estado de la base de datos a cada transacción. Esto nos permite hacer transacciones eventualmente consistentes, ofreciéndonos grandes ventajas en el rendimiento. No se requiere usar bloqueos de lectura al realizar una transacción lo que nos brinda una mayor escalabilidad. También PostgreSQL tiene *Hot-Standby*. Este permite que los clientes hagan búsquedas (sólo de lectura) en los servidores mientras están en modo de recuperación o espera. Así podemos hacer tareas de mantenimiento o recuperación sin bloquear completamente el sistema. PostgreSQL aporta mucha flexibilidad a nuestros proyectos. (The PostgreSQL Global Development Group, 2020)

- PhpStorm: Es un Entorno de Desarrollo Integrado (IDE por sus siglas en inglés) de uso generalizado en el desarrollo PHP. Por defecto trae un diccionario de inglés, que se encarga de validar la corrección ortográfica del texto. Es ideal para trabajar con Symfony, Laravel, Drupal, WordPress, Zend Framework, Magento, ¡Joomla!, CakePHP, Yii y otros marcos de trabajo. El editor "capta" su código y comprende su estructura en profundidad, además de ser compatible con todas las funcionalidades del lenguaje PHP para proyectos tanto nuevos como heredados. Proporciona la mejor finalización de código, refactorizaciones, prevención de errores sobre la marcha y más. (JetBrains s.r.o, 2020)
- Visual Studio Code: Es un editor de código fuente desarrollado por Microsoft para Windows, Linux y macOS. Incluye soporte para la depuración, control integrado de Git, resaltado de sintaxis, finalización inteligente de código, fragmentos y refactorización de código. También es personalizable, por lo que los usuarios pueden cambiar el tema del

editor, los atajos de teclado y las preferencias. Es gratuito y de código abierto, aunque la descarga oficial está bajo software propietario requiriendo tus datos de uso del programa legalmente. Va más allá del resaltado de sintaxis y autocompleta con IntelliSense, lo que proporciona terminaciones inteligentes basadas en tipos de variables, definiciones de funciones y módulos importados. Tiene extensiones que se ejecutan en procesos separados, lo que garantiza que no ralentizarán su editor. Con Microsoft Azure puede implementar y alojar sus sitios React, Angular, Vue, Node, Python, y más. Almacena y consulta datos basados en documentos y relacionales, y escalar con computación sin servidor, todo con facilidad. (Microsoft, 2020)

Frameworks

- **Laravel:** Es un *framework* PHP. Es uno de los más utilizados y de mayor comunidad en el mundo de Internet. Resulta bastante moderno y ofrece muchas utilidades potentes a los desarrolladores, que permiten agilizar el desarrollo de las aplicaciones web. Laravel pone énfasis en la calidad del código, la facilidad de mantenimiento y escalabilidad, lo que permite realizar proyectos desde pequeños a grandes o muy grandes. Además, permite y facilita el trabajo en equipo y promueve las mejores prácticas. Trabaja con una arquitectura de carpetas avanzada, de modo que promueve la separación de los archivos con un orden correcto y definido, que guiará a todos los integrantes del equipo de trabajo y será un estándar a lo largo de los distintos proyectos. Por supuesto, dispone también de una arquitectura de clases también muy adecuada, que promueve la separación del código por responsabilidades. Su estilo arquitectónico es MVC.
- **Angular:** Es un *framework* Javascript potente, muy adecuado para el desarrollo de aplicaciones *frontend* modernas, de complejidad media o elevada. El tipo de aplicación Javascript que se desarrolla con Angular es del estilo SPA (*Single Page Application*) o también las denominadas PWA (*Progressive Web App*). Ofrece una base para el desarrollo de aplicaciones robustas, escalables y optimizadas, que promueve además las mejores prácticas y un estilo de codificación homogéneo y de gran modularidad. Aunque brinda principalmente una base para el desarrollo de la parte frontal, la programación Javascript del lado del cliente, también aborda técnicas de desarrollo de la parte del *backend*, para la implementación del *Server Side Rendering*. A esta parte se le llama Angular Universal.
- **Leaflet:** Es la biblioteca de JavaScript de código abierto líder para mapas interactivos aptos para dispositivos móviles. Con un peso de aproximadamente 38 KB de JS, tiene todas las características de mapeo que la mayoría de los desarrolladores necesitan. Está diseñado teniendo en cuenta la *simplicidad*, el *rendimiento* y la *usabilidad*. Funciona de manera eficiente en todas las principales plataformas de escritorio y móviles, se puede ampliar con muchos complementos, tiene una API hermosa, fácil de usar y bien documentada y un código fuente simple y legible. (Agafonkin Vladimir, 2010)

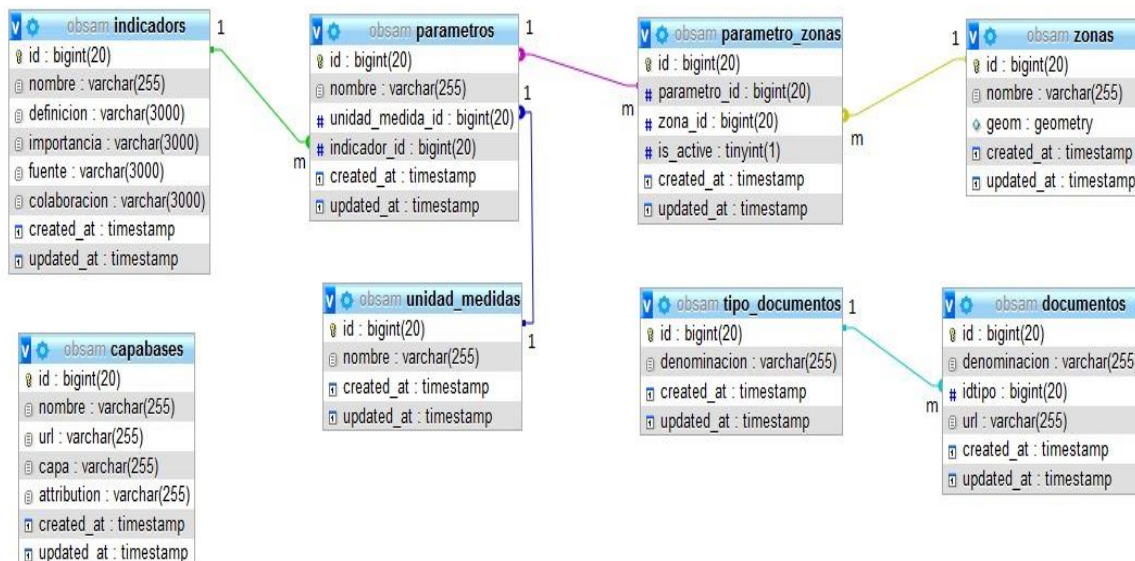
- Highcharts: Es una biblioteca de gráficos de JavaScript basada en SVG. Este paquete también contiene Highstock, el paquete de gráficos financieros y Highmaps para mapas geográficos. Este paquete está diseñado para admitir gráficos JavaScript del lado del cliente a través de paquetes como Parcel o Webpack y entornos como Babel o TypeScript. Si tiene la intención de generar gráficos estáticos en el lado del servidor. (Highsoft, 2009)

Es usado por decenas de miles de desarrolladores y más del 80% de las 500 compañías más grandes del mundo. Highcharts es la API de gráficos más simple pero más flexible del mercado. Presenta navegación sofisticada, anotaciones y análisis del lado del cliente con más de 40 indicadores técnicos incorporados. Permite la creación de diagramas interactivos e impactantes para su sitio de noticias o blog, o para compartir con sus amigos en las redes sociales. (Highsoft, 2009)

Base de Datos

El Sistema de Información Geográfica, para el modulo web Geoportal del OBSAMCostatenas presenta una base de datos con un total de 9 tablas hasta el alcance de este trabajo. Esta base de datos está sujeta a cambios, esto se debe a la complejidad del proceso de georreferenciación de los indicadores ambientales. Las relaciones predominantes en la base de datos son del tipo “1 a muchos”. Para su diseño y construcción se tuvo en cuenta que esta sea lo más flexible posible con respecto a las exigencias de los clientes. A continuación, se muestra el diseño de base de datos del proyecto.

Figura 2.2.1. Diseño de la base de datos



Fuente: elaboración propia

Resumen de Historias de Usuario

La siguiente tabla muestra un resumen con las Historias de Usuario (HU) que fueron planificadas inicialmente. Se definieron 4 HU. El proceso de planificación está sujeto a cambios que pueden ocurrir durante el desarrollo del *software*.

Tabla 2.4.1. Resumen de Historias de Usuario

| No | Nombre | Prioridad | Riesgo | Iteración | Entrega | Duración (semanas) |
|----|----------------------|-----------|--------|-----------|-----------|--------------------|
| 1 | Gestionar mapa | alta | alto | 1 | 22/1/2020 | 2 |
| 2 | Gestionar zonas | alta | alto | 2 | 5/2/2020 | 2 |
| 3 | Gestionar mediciones | alta | alto | 3 | 19/2/2020 | 2 |
| 4 | Gestionar capas base | media | bajo | 5 | 4/3/2020 | 2 |

Fuente: elaboración propia

Historias de Usuario iniciales

A continuación, se presenta un ejemplo de las HU elaboradas para el desarrollo del sistema.

Tabla 2.5.1 HU 1: Gestionar mapa

| Historia de Usuario | |
|------------------------------|----------------------------|
| Número: 1 | Usuario: Cualquiera |
| Nombre de HU: Gestionar mapa | |
| Prioridad en negocio: alta | Riesgo en desarrollo: alto |
| Puntos estimados: 2 | Iteración asignada: 1 |

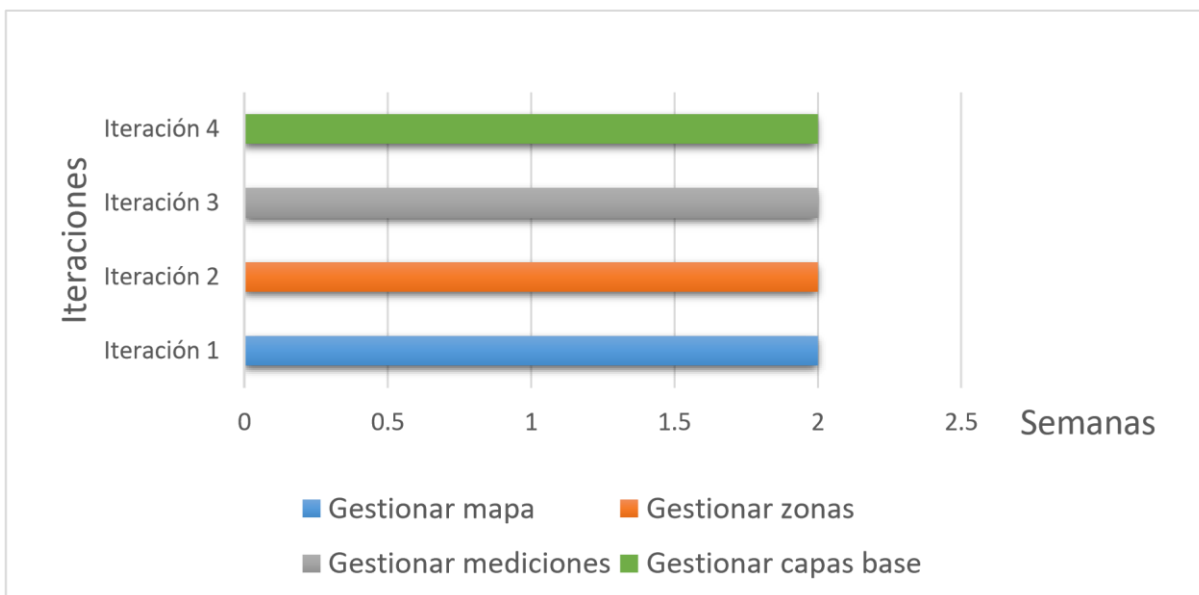
| |
|---|
| Programador responsable: Adrián Marcos Quintana |
| Descripción: Contempla las funciones de mostrar zonas y capas base en el mapa |
| Observaciones: |

Fuente: Elaboración propia

Planificación de iteraciones

Las iteraciones se planificaron para un total de 8 semanas de trabajo como se muestra a continuación.

Figura 2.6.1 Planificación de Iteraciones



Planificación de Iteraciones

Fuente: Elaboración propia

Tareas iniciales a desarrollar

A continuación, se presentan las tareas a desarrollar por cada historia de usuario.

Tabla 2.7.1 Distribución de las Tareas iniciales a desarrollar

| H U | No. Tarea | Nombre de la tarea | Tiempo estimado (días) |
|--|----------------------|--|---------------------------------------|
| Iteración 1: Gestionar mapa | | | |
| 1 | 1 | Diseño e implementación de Mostrar zonas en el mapa | 6 |
| 1 | 2 | Diseño e implementación de Mostrar capas base en el mapa | 6 |
| Iteración 2: Gestionar zonas | | | |
| 2 | 1 | Diseño e implementación de Crear zonas | 4 |
| 2 | 2 | Diseño e implementación de Modificar zonas | 4 |
| 2 | 3 | Diseño e implementación de Eliminar zonas | 3 |
| Iteración 3: Gestionar mediciones | | | |
| 3 | 1 | Diseño e implementación de Listar mediciones | 3 |
| 3 | 2 | Diseño e implementación de Crear mediciones | 3 |
| 3 | 3 | Diseño e implementación de Modificar mediciones | 3 |
| 3 | 4 | Diseño e implementación de Graficar mediciones | 2 |

| Iteración 4: Gestionar capas base | | | |
|--|---|---|---|
| 4 | 1 | Diseño e implementación de Listar capas base | 3 |
| 4 | 2 | Diseño e implementación de Crear capas base | 3 |
| 4 | 3 | Diseño e implementación de Modificar capas base | 3 |
| 4 | 4 | Diseño e implementación de Eliminar capas base | 2 |

Fuente: Elaboración propia Descripción de tareas a desarrollar por HU

A continuación, se muestran las descripciones de las tareas correspondientes a las HU número 1. La descripción de todas las tareas de cada HU aparece en la Documentación del Sistema, entregada como complemento de esta investigación.

Tabla 2.7.2 Diseño e implementación de Mostrar zonas en el mapa

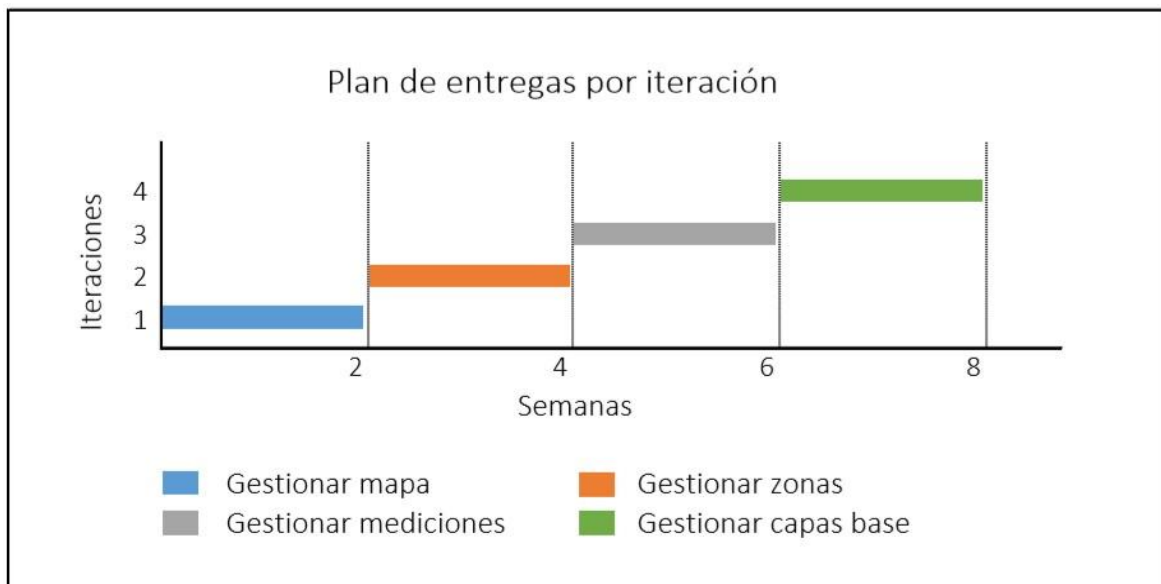
| Tarea | |
|---|----------------------------------|
| Número de Tarea: 1 | Número de HU: 1 |
| Nombre de la Tarea: Diseño e implementación de Mostrar zonas en el mapa | |
| Tipo de Tarea: Desarrollo | Horas: 42 |
| Fecha de Inicio: 8/1/2020, 9:00 am | Fecha de Fin: 14/1/2020, 5:00 pm |
| Programador responsable: Adrián Marcos Quintana | |
| Descripción: | |

Plan de entregas iniciales

El cronograma de entregas establece que historias de usuario serán agrupadas para conformar una entrega, y el orden de las mismas. Este cronograma será el resultado de una reunión entre todos los actores del proyecto. Se realiza en base a las estimaciones de tiempos de desarrollo realizadas por los desarrolladores. (Joskowicz José, 2008)

El plan de entrega de los módulos de la aplicación corresponde con el fin de cada iteración. Cuando termina una iteración, se presentan los resultados al cliente, quien los valida mediante pruebas funcionales. En caso de que no se cumpla con lo negociado en el plan de entrega, se ajustará el mismo para evitar futuros incumplimientos. La siguiente gráfica muestra el plan de entrega de cada segmento del sistema:

Figura 2.8.1 Plan de entregas por iteración



Incidencias

Es muy complicado cumplir con cada detalle requerido por el sistema que se desea desarrollar una vez realizado la planificación inicial del proceso de desarrollo. Por tanto, el cumplimiento del plan de entregas está sujeto a modificaciones constantemente.

A medida que el proceso de desarrollo del sistema fue avanzando surgieron nuevos cambios en la base de datos, u esta creció con respecto al número de tablas iniciales con las que se contaba, inicialmente fueron creadas 5 tablas, y actualmente el sistema cuenta con 9 tablas. El diseño e implementación del sistema cuenta con un nivel de complejidad bastante alto, debido a esto en el transcurso del proceso de desarrollo van aumentando los requisitos del sistema.

Utilizando la metodología XP, el cliente es una pieza fundamental en el proceso de desarrollo de *software*, pues el mismo es quien valida el resultado de cada entrega y aporta nuevas ideas, las cuales pueden llegar a ser soluciones. Así queda demostrado la valía del cliente como miembro del equipo de desarrollo.

Estimación de costo

Al utilizar XP como metodología de desarrollo de *software*, se decide emplear la fórmula de Bohem para calcular el costo del proyecto porque con dicha metodología se estima el tiempo de desarrollo del proyecto. Esta fórmula plantea que:

Costo = Cantidad de Hombres * Salario Medio * Tiempo de Desarrollo

Cantidad de hombres = 1 desarrollador

Salario Medio = \$467.5

Tiempo de desarrollo = 2 meses

Se realizan los cálculos pertinentes con los datos anteriores y se obtiene un costo de \$935

Análisis de costos y beneficios

El Sistema tiene como objetivo satisfacer las necesidades del CITMA con respecto a la georreferenciación de los indicadores ambientales que allí se estudian y el análisis del comportamiento en el tiempo de dichos indicadores, debido a que este proceso se realiza de forma manual con mapas físicos. La herramienta informática desarrollada cumple con una serie de requisitos planteados por el proceso antes mencionado, resultando ser un proyecto complejo. Resulta factible la obtención de este sistema por lo antes mencionado, por tanto,

realizando un análisis entre los beneficios y costos de la aplicación, se llegaron a las conclusiones siguientes.

- El proceso de georreferenciación de los indicadores ambientales se realiza de forma más eficiente y organizada
- La herramienta informática desarrollada resulta atractiva y tiene gran aceptación por quienes la utilizan
- Facilita el proceso de toma de decisiones al poder analizar gráficamente el comportamiento histórico de cada indicador ambiental
- Los especialistas pueden entregar las mediciones realizadas de cada indicador ambiental en tiempo real a través de la aplicación, lo cual aumenta su motivación.

Al analizar los beneficios que brinda el sistema que puede establecer que el desarrollo del proyecto presenta una correcta relación y correspondencia entre el costo y los beneficios obtenidos.

Pruebas de caja blanca

Las pruebas de caja blanca se basan en el diseño de casos de prueba que usa la estructura de control del diseño procedimental para derivarlos.

Aunque las pruebas de caja blanca son aplicables a varios niveles: unidad, integración y sistema. Habitualmente se aplican a las unidades de software. Su cometido es comprobar los flujos de ejecución dentro de cada unidad (función, clase, módulo, etc.) pero también pueden testear los flujos entre unidades durante la integración, e incluso entre subsistemas, durante las pruebas de integración, e incluso entre subsistemas, durante las pruebas de sistema.

Existen varias técnicas para aplicar las pruebas de caja blanca, entre ellas se encuentra el Camino Básico la cual es válida para encontrar posibles errores en nuestros ciclos funcionales.

La prueba del camino básico es una técnica de prueba de la Caja Blanca propuesta por Tom McCabe. Esta técnica permite obtener una medida de la complejidad lógica de un diseño y usar esta medida como guía para la definición de un conjunto básico. La idea es derivar casos de prueba a partir de un conjunto dado de caminos independientes por los cuales puede circular el flujo de control. Para obtener dicho conjunto de caminos independientes se construye el Grafo de Flujo asociado y se calcula su complejidad ciclomática.

A continuación, se muestra un ejemplo que ilustra la aplicación de esta técnica sobre el código del programa. La figura 3.1 muestra un fragmento de código sobre el cual se realizó la prueba de Camino Básico.

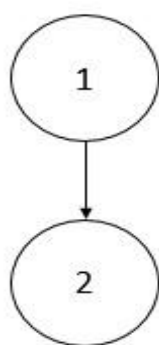
Figura 3.1. Código original de capas_bases.service.ts que se encarga de actualizar las capas base

```
61 updateCapaBase(capaBase:CapaBase){
62     const cb:CapaBase=capaBase;
63     1
64     this.http.put<{mensaje:string; capaBase:CapaBase}>>(Backend_CapasBases_Url+'/'+capaBase.id, cb).subscribe(response=>{
65         const updatedCapasBases = [...this.capasBases];
66         const oldCapasBasesIndex = updatedCapasBases.findIndex(p=> p.id ===cb.id);
67         updatedCapasBases[oldCapasBasesIndex]= cb;
68         this.capasBases=updatedCapasBases;
69         2
70         this.capasBasesActualizadas.next({
71             capasBases: [...this.capasBases]
72         });
73         this.toastr.success(response.mensaje,'Operación exitosa');
74     });
75 }
76
```

Fuente: elaboración propia

A partir de la numeración dada a cada paso del método anterior se elaboró el diagrama de flujos del proceso que se presenta a continuación.

Figura 3.2. Diagrama de flujo del método updateCapaBase



Fuente: elaboración propia

Se procede a calcular la complejidad ciclomática a partir del diagrama anterior, teniendo en cuenta la fórmula: $V(G) = A - N + 2$ donde:

- A: Cantidad de aristas del grafo, en este caso es un árbol el resultado del método pues no tiene ciclos.
- N: Cantidad de Nodos

Sustituyendo en la formula tenemos que:

- A=1 aristas
- N=2 nodos

$$V(G) = 1 - 2 + 2 = 1$$

Caminos que surgen como resultado de la aplicación del diagrama anterior. Camino

1: 1, 2

Tabla 3.1. Caso de Prueba para el Camino 1

| Caso de Prueba para el camino 1 |
|---|
| <p>Entrada:</p> <pre>{ id: 1, nombre: "Mapa Temático de Matanzas", url: "http://localhost:8080/geoserver/obsam/wms", capa: "obsam:Mapa Temático de Matanzas", attribution: "CITMA 2020" }</pre> |

Resultado Esperado: Éxito de la Operación. Se envían los datos de la capa base a actualizar al *backend* para que la capa base sea actualizada en la base de datos

Evaluación: Satisfactoria

Pruebas de Aceptación

Las pruebas de aceptación son definidas por el usuario del sistema y preparadas por el equipo de desarrollo, aunque la ejecución y aprobación final corresponden al usuario. (Cillero Manuel, 2017)

Estas pruebas van dirigidas a comprobar que el sistema cumple los requisitos de funcionamiento esperado, recogidos en el catálogo de requisitos y en los criterios de aceptación del sistema de información, y conseguir así la aceptación final del sistema por parte del usuario. (Cillero Manuel, 2017)

Casos de Prueba para la Iteración 1

En la Iteración 1 se cumplió la Historia de Usuario 1, Gestionar Mapa. Las siguientes tablas representan algunos de los casos de prueba que se realizaron en esta iteración.

Tabla 3.2. Caso de Prueba 1. Corresponde a la Tarea Diseño e implementación de Mostrar zonas en el mapa

| |
|---|
| Caso de Prueba: 1 |
| Historia de Usuario: 1 |
| Nombre del Caso de Prueba: Mostrar zonas |
| Descripción: En el sistema, sin necesidad de autenticarse está la sección de Geoportal, con la opción Mapa. En esta opción se muestra todas las zonas georreferenciadas, cada una con su respectiva información. |

| |
|---|
| <p>Condiciones de Ejecución: El sistema debe estar funcionando. Esto se refiere a todo el entorno que permite que la aplicación esté en funcionamiento. En este caso que el servidor en que se encuentre el sistema esté activo.</p> |
| <p>Entrada: Refrescar la página para ver si devuelve algún error o cambia los datos sin alterar la base de datos. Buscando errores.</p> |
| <p>Resultado Esperado: Éxito en la Operación (no se encontraron errores).Las zonas se muestran en el visor de mapas al ser cargadas de la base de datos.</p> |
| <p>Evaluación: Satisfactoria</p> |

Fuente: elaboración propia

Tabla 3.3. Caso de Prueba 2. Corresponde a la Tarea Diseño e implementación de Mostrar capas base en el mapa

| |
|---|
| <p>Caso de Prueba: 2</p> |
| <p>Historia de Usuario: 1</p> |
| <p>Nombre del Caso de Prueba: Mostrar capas base</p> |
| <p>Descripción: En el sistema, sin necesidad de autenticarse está la sección de Geoportal, con la opción Mapa. En esta opción se muestra todas las capas base del visor de mapas disponibles para el usuario.</p> |
| <p>Condiciones de Ejecución: El sistema debe estar funcionando. Esto se refiere a todo el entorno que permite que la aplicación esté en funcionamiento. En este caso que el servidor en que se encuentre el sistema esté activo.</p> |
| <p>Entrada: Refrescar la página para ver si devuelve algún error o cambia los datos sin alterar la base de datos. Buscando errores.</p> |

Resultado Esperado: Éxito en la Operación (no se encontraron errores).Las capas base se muestran en el visor de mapas y en el control de capas del visor al ser cargadas de la base de datos.

Evaluación: Satisfactoria

Fuente: Elaboración propia

Conclusiones

Como resultado de la investigación realizada se cumplieron satisfactoriamente los objetivos trazados, permitiendo concluir que las tecnologías empleadas en el desarrollo del sistema permitieron obtener un producto escalable y flexible ante actualizaciones tecnológicas, el sistema ofrece servicios que pueden ser consumidos por terceras aplicaciones, dada la arquitectura y técnicas de programación utilizadas, como la implementación de una API REST. Además, se obtuvo un sistema agradable, intuitivo, objetivo y fácil de utilizar puesto que se emplearon tendencias de diseño basadas en la experiencia de los usuarios. Se desarrolló un Sistema de Información Geográfica que permite georreferenciar los distintos indicadores ambientales que se estudian en el observatorio, contribuye a una correcta toma de decisiones ya que facilita el acceso a la información puesto que esta se encuentra centralizada, además se trabaja con mapas digitales, lo que agiliza el trabajo del usuario. Al realizar las pruebas de software se corrigieron los errores detectados por las mismas, lo que permitió que la aplicación desarrollada sea más funcional y robusta ante errores de los usuarios.

Bibliografía

Agafonkin Vladimir. 2010. Leaflet. [En línea] 2010. [Citado el: 20 de Abril de 2020.] <https://leafletjs.com/index.html>.

Berrio Turiño, Eduardo Javier. 2018. *Sistema Automatizado para la Gestión Ambiental Empresarial*. Matanzas : s.n., 2018.

Cillero Manuel. 2017. manuel.cillero.es. [En línea] 12 de Marzo de 2017. [Citado el: 16 de 2020 de 2020.] <https://www.manuel.cillero.es/doc/metrica-3/tecnicas/pruebas/acceptacion/>.

Eguiluz Javier. 2020. Uniwebsidad. [En línea] 2020. [Citado el: 20 de Abril de 2020.] <https://uniwebsidad.com/libros/css/>.

Highsoft. 2009. Highcharts. [En línea] 2009. <https://www.highcharts.com/>.

JetBrains s.r.o. 2020. Jet Brains. [En línea] 2020. [Citado el: 15 de Abril de 2020.] <https://www.jetbrains.com/es-es/phpstorm/>.

Joskowicz José. 2008. *Reglas y Practicás en eXtreme Programming*. Universidad de Vigo, España : s.n., 2008.

Loné Pérez, Pedro Pablo. 2016. Comunidad ism. [En línea] 26 de Septiembre de 2016. [Citado el: 15 de Abril de 2020.] <http://www.comunidadism.es/blogs/%C2%BFque-es-un-indicador-ambiental>.

Microsoft. 2020. Visual Studio Code. [En línea] 2020. [Citado el: 15 de Abril de 2020.] <https://code.visualstudio.com/>.

Mozilla. 2020. MDN Web Docs. [En línea] 2020. [Citado el: 20 de Abril de 2020.] <https://developer.mozilla.org/es/docs/Web/HTML>.

—. **2020.** MDN Web Docs. [En línea] 2020. [Citado el: 20 de Abril de 2020.] <https://developer.mozilla.org/es/docs/Web/CSS>.

Pérez Porto, Julián y Merino María. 2019. Definición de. [En línea] 2019. [Citado el: 15 de Abril de 2020.] <https://www.definicion.de/georeferenciacion>.

Q-Success. 2015. Historical trends in the usage of server-side programming languages for websites. [En línea] 2015. [Citado el: 20 de Abril de 2020.] http://w3techs.com/technologies/history_overview/programming_language.

Rocha Garrido, Guillermo. 2019. Profile.es. [En línea] 7 de Marzo de 2019. [Citado el: 15 de Abril de 2020.] <https://www.profile.es/blog/programador-extremo-extreme-programming>.

S/a. 2016. Desarrolloweb.com. [En línea] 2 de Junio de 2016. [Citado el: 20 de Abril de 2020.] <https://desarrolloweb.com/articulos/introduccion-a-typescript.html>.

—. **2019.** Geoinnova. [En línea] 2019. [Citado el: 15 de Abril de 2020.] <https://www.geoinnova.org/cursos/que-son-los-sistemas-informacion-geografica-sig/>.

The PostgreSQL Global Development Group. 2020. PostgreSQL. [En línea] 2020. [Citado el: 15 de Abril de 2020.] <https://www.postgresql.org/>.