

Resumen Ejecutivo de Tesina:

GESTIÓN DE LA MIGRACIÓN DE INSTALACIONES DE GAS USANDO TECNOLOGÍAS SIG

Máster de Sistemas de Información Geográfica



UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH

Jaime Vasquez Milla

Índice

I.	Introducción	2
II.	Planteamiento del problema	2
III.	Objetivos	3
a.	Objetivo principal	3
b.	Objetivos secundarios	3
IV.	Instalaciones de gas	3
A.	Que es una instalación de gas	3
B.	Elementos de la instalación de gas	4
C.	Archivos base para el dibujo	4
V.	Metodología	5
A.	Creación de herramientas en Model Builder	5
B.	Creación de botones y herramientas con Add-in.....	6
C.	Reporte de observaciones.....	7
VI.	Conclusiones.....	10
VII.	Recomendaciones	11
VIII.	Bibliografía	11

Tabla de Ilustraciones

Ilustración 1 - Diseño de una instalación de gas	4
Ilustración 2 - Agrupación de tareas por Modelos.....	5
Ilustración 3 - Interfaz de la herramienta creada.....	6

I. Introducció

La migració de les instal·lacions de gas domiciliaries consisteix en convertir la instal·lació des d'un format paper fins a un Feature Class d'una geodatabase. El procés involucra la preparació dels dades, el dibuix de la instal·lació, el control de qualitat, la comunicació dels errors i la resposta als errors.

El projecte es desenvolupa per demanda de l'empresa administradora del gas natural a Lima. Basant-se en els seus requeriments i limitacions pròpies, es dissenyà una plataforma de gestió basada en tecnologies SIG.

Los actors involucrats en el projecte són l'empresa administradora de gas, les empreses subcontractades que realitzen les instal·lacions físiques de gas, els editors que dibuixen les instal·lacions. El procés de preparació dels dades, dibuix i control de qualitat es desenvolupa des de l'entorn de ArcMap, necessitant 225 tasques per realitzar el procés. En aquest document, es descriu detalladament totes les tasques amb el fi de mostrar el grau d'automatització que es pot aconseguir usant Model Builder i Python.

Per accelerar més el procés, es dissenyaren eines personalitzades, a la interfície de ArcMap, que cobreixen les principals necessitats del projecte. Les eines es desenvoluparen amb l'Assistent d'Add-In de Python, usant la llibreria ArcPy i altres complementàries.

Per a la comunicació i resposta als errors, es implementà una base de dades espacial allotjada a la núvol, usant la tecnologia de PostgreSQL i PostGIS. La base de dades treballa a través de funcions i disparadors que al detectar un esdeveniment realitzen una acció determinada. Els diferents actors involucrats poden accedir a la base de dades mitjançant QGIS.

II. Plantejament del problema

Es necessita crear una plataforma per al tractament de les instal·lacions de gas que permeti l'automatització de processos tant a nivell de control de qualitat com de informe d'errors trobats. El procés és el següent:

Es comença amb la recepció de la instal·lació de gas en format pdf associada a un registre Excel. Es reben en promig 600 instal·lacions diàries i han de ser entregades al final del dia.

Com a primer pas es procedeix a revisar la integritat dels dades, amb el fi d'evitar dibuixar instal·lacions repetides entre elles o amb instal·lacions ja dibuixades prèviament. Seguidament es ha de heredar la geometria als registres Excel; la geometria que es hereda és la del predi on es va a dibuixar. Després de tenir la capa associada a un predi, es ha de revisar si té una xarxa de gas al davant. Finalment es han de guardar còpies de respaldos comprimides.

Després de rebre els elements dibuixats, es han de fer controls de qualitat sobre la geometria i els dades alfanumèrics. En quant a geometria es ha de revisar els següents aspectes: que no hi hagi superposició amb altres instal·lacions, que estiguin dibuixades al predi correcte, que estiguin connectades a la xarxa i als predis, que no tallin el flux de la xarxa, que no es superposin amb instal·lacions ja entregades. En

cuanto a los datos alfanuméricos, se debe revisar que no existan errores de duplicidad en los códigos identificadores.

Después del control de calidad, se deben heredar datos tanto del Excel asociado como de otras capas de información de la cartografía. Finalmente se deben adecuar los datos para la entrega del producto en una geodatabase.

Un proceso paralelo es el reporte de errores a las empresas responsables de las instalaciones. El reporte debe ser por cada instalación indicando el tipo de error; las empresas cuentan con un día hábil para responder con la subsanación.

El proceso está caracterizado por requerimientos específicos de la empresa y por algunas limitaciones de parte del equipo.

Requerimientos:

- Se debe trabajar con archivos Feature Class, usando la versión de ArcMAP 10.4 en adelante.
- El proceso se debe realizar diariamente.
- No se puede tener contacto directo con las empresas responsables de las instalaciones.
- No se puede realizar las labores dentro de la empresa.
- Se debe tener discreción con los datos.

Limitaciones:

- El equipo de trabajo esta espacialmente disperso.
- Solo se puede acceder a una licencia de ArcGIS.
- No se cuenta con servidor en la nube propia.

III. Objetivos

a. Objetivo principal

- Desarrollar una plataforma basado en tecnologías de los sistemas de información geográfica, para la gestión de las instalaciones de gas.

b. Objetivos secundarios

- Crear herramientas que automaticen tareas usando el Model Builder de ArcGIS.
- Crear herramientas usando el asistente Add-In de ArcGIS que permitan un desarrollo ágil en el entorno de ArcMAP.
- Crear una base de datos alojada en la nube para la comunicación y respuesta de los errores detectados en el dibujo de las instalaciones.

IV. Instalaciones de gas

A. Que es una instalación de gas.

La instalación de gas es la unión, por medio de una tubería, entre la red general de gas y la vivienda. El tipo de gas que transporta es natural y se hace generalmente por tuberías de polietileno. El tipo de instalación es doméstica, las instalaciones comerciales e industriales tienen otro tratamiento. Solo se toma en cuenta la conexión externa de la vivienda y no la interna.

La instalación de gas, principalmente, se compone de 3 elementos; el primero es la tubería de conexión, luego el empalme entre la tubería y la red troncal, y el gabinete de regulación.

B. Elementos de la instalación de gas

Los tres elementos descritos antes, se representan cartográficamente mediante vectores. Para el caso de la tubería de conexión se representa mediante una línea, para el empalme y el gabinete de regulación se representa mediante puntos.

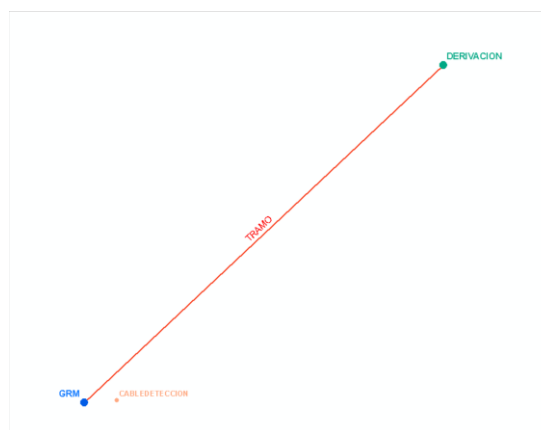
TRAMO: Es un elemento de tipo línea que almacena datos de la conexión como fecha puesta de servicio, diámetros de la red y de la tubería.

GRM: Es un elemento de tipo punto que se encuentra al final de la tubería de conexión, es el elemento que almacena todos los códigos que identifican a la instalación. Existen varios tipos de gabinete.

DERIVACION: Es un elemento de tipo punto que se encuentra al inicio de la tubería de conexión. Almacena características de la instalación

CABLEDETECCION: Es un elemento de tipo puntual. No tiene una función conocida, pero es un elemento requerido.

Ilustración 1 - Diseño de una instalación de gas



C. Archivos base para el dibujo

Como insumos para la digitalización se recibe dos tipos de archivos, el primero es una ficha escaneada con la información levantada en campo y un archivo Excel que tiene los detalles asociadas a la instalación.

Ficha escaneada: Es una ficha que se usa para registrar todas las características que se pueden recopilar al momento de la instalación. Los datos que se registran son el tipo de gabinete, el diámetro de la tubería de conexión, el tipo de y diámetro de la derivación, también se registran los datos de identificación de la instalación, como el número de instalación, código equipo de la tubería de conexión, número de contrato, y finalmente la dirección local, el distrito y la zona a la que pertenece.

Excel: Es un registro Excel que almacena los atributos y especificaciones de las instalaciones.

V. Metodología

A. Creación de herramientas en Model Builder

Son 225 tareas que se necesitan realizar para cumplir con el proceso, las tareas se agrupan para convertirse en un subproceso. Las tareas se agrupan porque forman parte de una secuencia para obtener un resultado, en las secuencias pueden incluir una o varias herramientas.

A continuación, se muestra en un gráfico las tareas que agrupas cada herramienta, esto con el fin de evidenciar todos los procesos que se pueden automatizar.

Ilustración 2 - Agrupación de tareas por Modelos

1	Nombre de la herramienta	Numero de herramientas								
		1	2	3	4	5	6	7	8	9
1	Espacio de trabajo									
2	Exportar Cartografía	10	11	12	13	14	15	16	17	
3	Crear Tablas	19	20	21	22	23	24	25	26	
4	Validación de Códigos	27	28	29	30	31	32	33	34	35
		36	37	38	39	40	41	42	43	44
		45	46	47	48	49	50	51	52	53
		54								
5	Extraer Predios	55	56	57	58					
6	validación de Red	61	62	63	64	65	66	67		
7	Editores Almacenar	68	69	70	71	72	73	74	75	
8	Distancia GRM-TRAMO	76	77	78						
9	Distancia DERIVACION-TRAMO	79	80	81						
10	Sentido del TRAMO	82	83	84	85	86	87			
11	GRM Punto Final	88	89							
12	DERIVACION Punto Final	90	91							
13	Superposición Espacial	92	93	94	95	96	97			
14	Distancia Predios	98	99	100	101	102				
15	Conectividad	103	104	105						
16	Cruce de red	106								
17	Duplicidad de Códigos	107	108	109						
18	Heredar Códigos	110	111	112	113	114	115	116	117	
19	Comprobar Predio	119	120	121	122	123	124	125		
20	Distancia a TRAMO GIS	126	127	128						
21	Heredar Valores	129	130	131	132	133	134	135	136	137
		138	139	140	141	142	143	144	145	146
		147	148	149	150					
22	Heredar Códigos SAP	151	152	153	154	155	156	157	158	159
		160	161	162	163	164	165	166	167	168
		170	171	172	173	174	175	176	177	178
		179	180							
23	Heredar Código Padre	181	182	183	184	185	186			
24	Distribuir Códigos Padre	187	188	189	190	191	192	193	194	195
25	Heredar CABLEDETECCION	196	197	198	199	200				
26	Cambiar GEODATABASE	201	202	203	204	205				

27	Heredar Códigos Finales	206	207	208	209	210	211			
28		212	213	214	215	216	217	218	219	220
	Distribuir Códigos Finales	221	222							
29	Comprobar Códigos Heredados	228	229	230	231	232	233	234	235	

El diseño de los modelos y de las tareas se encuentra en el documento completo.

B. Creación de botones y herramientas con Add-in

El proceso incluye tareas pequeñas, como el apagado de todas las capas visibles del proyecto de ArcMAP, eliminar la base de datos temporal, o consultar en Street View las fotografías del predio.

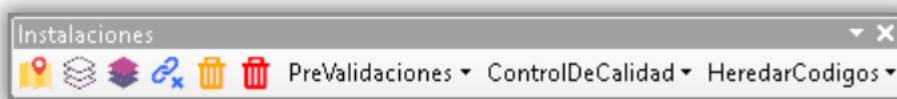
Estas tareas se pueden simplificar con el Asistente Add-in que a través del lenguaje de programación Python se le pueden asignar realizar ciertas tareas al oprimir un botón o realizar una acción con el cursor.

Se crearon las siguientes funcionalidades:

- Crear una barra de herramientas que agrupe todas las herramientas creadas.
- Poder visualizar fotografías del predio, para poder contextualizarse en caso de dudas. Street View de Google Maps ofrece esta posibilidad.
- Poder apagar y prender todas las capas prender todas las capas visibles del ArcMap.
- Poder eliminar todas las capas visibles.
- Poder quitar todas las uniones ejecutadas.
- Acceder a los modelos desde una barra desde la barra de herramientas

El resultado es una herramienta totalmente funcionalidad:

Ilustración 3 - Interfaz de la herramienta creada



Las funciones de cada una de ellas son:

Street View Lima: Primer botón

Tipo: Tool

Función: Extrae las coordenadas del punto a elegir, la convierte a coordenadas geográficas, y las inserta en una dirección web de Street View de Google Maps.

Apagar las capas: Segundo botón

Tipo: botón

Nombre: Apagar las capas

Función: Apaga o desactiva todas las capas que se encuentran activas.

Prender las capas: Tercer botón

Tipo: botón

Nombre: Encender las capas

Función: Enciende o activa todas las capas que se encuentran desactivadas.

Bloque de código:

Capas que llaman a Modelos y funciones: Del cuarto botón en adelante

Tipo: botón

Nombre: Adopta el nombre de cada capa

Función: Ejecutar el modelo o script almacenado en la caja de herramientas de los modelos

C. Reporte de observaciones

Para el reporte de errores, debido a las restricciones de comunicación, se creó una plataforma de interacción basado en PostgreSQL, que sirve para almacenar, publicar y responder a las incidencias de las instalaciones. Las instalaciones son mostradas a través de tablas que heredan los atributos del shapefile, también hereda la geometría, por eso se usa la extensión de PostGIS, que sirve para poder visualizar la geometría. La base de datos se encuentra alojado en la nube con el fin de poder acceder a la base de datos desde una aplicación o desde una web.

La base de datos se desarrolló bajo funciones y disparadores que permiten la automatización de tareas. Los disparadores al detectar ciertos cambios en los registros hacen determinadas tareas automáticas, como copiar registros entre tablas, capturar datos y eliminar registros.

Las funciones son bloques de código en lenguaje 'PL/PgSQL' y su fin es realizar determinada tarea cuando se le invoque, en este caso todas las funciones creadas son invocadas por disparadores.

Se crearon las siguientes funciones:

- SP_UPDATE_INSERT_CARGADO: Al detectar la palabra 'Cargado' inserta la fila a la tabla *cargados*.
- SP_UPDATE_INSERT_ED_ERRORES: Al detectar la palabra 'Observado' inserta la fila a la tabla *ediciondiariaerrores*.
- SP_UPDATE_INSERT_ERRORESRED: Al detectar la palabra 'Observado por red' inserta la fila a la tabla *erroresred*.
- SP_UPDATE_DELETE_ZLOG_ED: Inserta los registros en la tabla *logediciondiaria*.
- SP_UPDATE_INSERT_LOG: Inserta los registros en la tabla *logediciondiaria*.
- SP_INSERT_INSERT_CAPTURAFECHA_GENERAL: Captura la fecha de ingreso del registro
- SP_UPDATE_DELETE_CARGADO_ER(): Elimina los registros que se marcaron como Cargado.
- SP_UPDATE_CAPTURARFECHA_RESPUESTA_IO(): Captura la fecha de respuesta.
- SP_UPDATE_INSERT_CORREGIDO_PENDIENTEDIBUJO_IO: Cada vez que detecta una actualización en la columna estado con la palabra 'Corregido', envía los registro a la tabla *pendientedibujo*.
- UPDATE_DELETE_CORREGIDOS_IO: Elimina los registros después de trasladar a *pendientesdibujo*

- UPDTE_DELETE_CARGADO_PD: Después de dibujar las instalaciones de *pendientesdibujo*, se cambia el estado a Cargado. El disparador los elimina de la tabla.

Los disparadores son funciones que están alertas a eventos en la tabla al cual esta asociada, al detectar un cambio, previamente definido, ejecuta las instrucciones que tienen programadas. Los eventos que activan los disparadores son el ingreso de un nuevo registro, la actualización de un registro y la eliminación de un registro.

En base a esas funciones se creo los siguientes disparadores.

Tablas *ediciondiaria*:

- tr_update_insert_ca: Después de realizar una actualización para cada fila, ejecutar la función sp_update_insert_ca().
- tr_update_insert_erroresred: Después de realizar una actualización para cada fila, ejecutar la función sp_update_insert_erroresred().
- tr_update_insert_errores: Después de realizar una actualización para cada fila, ejecutar la función sp_update_insert_erroresred().
- tr_update_delete_zlog_ed: Después de eliminar un registro, se ejecuta la función sp_update_zlog_ed() para cada fila.
- tr_update_insert_log: Después de modificar un registro, se ejecuta la función sp_update_insert_log() para cada fila.

Tabla *cargados*:

- tr_insert_insert_capturafecha_ca: Antes de ingresar un nuevo registro, se ejecuta la función sp_insert_insert_capturafecha_general() para cada fila.

Tabla *erroresred*:

- tr_insert_insert_capturafecha_er: Antes de ingresar un nuevo registro en la tabla, se ejecuta la función sp_insert_insert_capturafecha_general() para cada fila.
- tr_update_insert_cargado_er: Después de actualizar un registro en la tabla, se ejecuta la función sp_update_insert_cargado_er() para cada fila.
- tr_update_delete_cargado_er: Después de actualizar un registro en la tabla, se ejecuta la función sp_update_delete_cargado_er() para cada fila.

Tabla *ediciondiariaerrores*:

- tr_insert_insert_capturafecha_ede: Antes de ingresar un nuevo registro, se ejecuta la función sp_insert_insert_capturafecha_general() para cada fila.
- tr_update_capturafecha_respuesta_io: Antes de actualizarse la tabla debe ejecutar la función sp_update_capturafecha_respuesta_io() para cada registro.
- tr_update_insert_correcto_pendientesdibujo_io: Antes de actualizarse la tabla debe ejecutar la función sp_update_insert_corregido_pendientesdibujo_io() para cada registro.
- tr_update_delete_corregidos_io: Después de actualizar la tabla debe ejecutar la función sp_update_delete_corregidos_io () para cada registro.

Tabla *pendientesdibujo*:

- tr_insert_insert_capturafecha_pd: Antes de ingresar un nuevo registro se debe ejecutar la función sp_insert_insert_capturafecha_general() para cada registro.

- `tr_update_insert_cargado_ca`: Después de actualizar los registros de la tabla debe ejecutarse la función `sp_update_insert_cargado_ca()` para cada registro.

Tabla *basegeneral*:

- `tr_insert_insert_capturafecha_bg`: Después de ingresar una nueva fila, se debe ejecutar la función `sp_insert_insert_capturafecha_general()`.

Tabla *loghistorico*:

- `tr_insert_insert_capturafecha_lh`: Después de ingresar una nueva fila, se debe ejecutar la función `sp_insert_insert_capturafecha_general()`.

Tabla *Zlog.logediciondiaria*

- `tr_insert_insert_capturafecha_pd`: Después de ingresar una nueva fila, se debe ejecutar la función `sp_insert_insert_capturafecha_general()`.

Funcionamiento de la base de datos.

La tabla *ediciondiaria* contiene todos los registros del grupo que se está trabajando, pero con la particularidad que esta tabla se publica y pueden acceder los editores para registrar los errores encontrados en el proceso de digitalización. Los editores pueden registrar los errores encontrados a través del número de instalación y se registra usando la calculadora de campos de QGIS.

Al culminar la digitalización del grupo y posterior a la carga de las instalaciones dibujadas, se procede a distribuir los registros de *ediciondiaria* en 3 tablas diferentes, según el estado de cada instalación.

La columna *estado* puede almacenar 3 opciones:

- Cargado: En caso no haya presentado errores y se haya cargado sin problemas.
- Observado por red: En caso la instalación no tenga red de gas al frente del predio.
- Observado: En caso presente alguna observación distancia a la falta de red.

A las instalaciones que han sido cargadas, se digita en la columna estado la palabra 'Cargado', lo que activa un primer disparador, trasladando esos registros a la tabla *cargados*.

Las instalaciones que no tuvieron red de gas al frente del predio tienen registrado el valor 'Sin red', a ellos en estado se digita 'Observado por red', lo que activa un segundo disparador, trasladando los registros a la tabla *erroresred*.

Y las que presentan otro tipo de observación en *obs_temp* se digita la palabra 'Observado', lo que activa un tercer disparador, trasladando los registros a *ediciondiariaerrores*.

Finalmente se elimina todos los valores de *ediciondiaria* con el fin de que este vacía para el próximo grupo. Al eliminar los registros, se activa un disparador que antes de eliminar traslada los registros a tablas de tipo log. Paralelo a todas las acciones, un disparador está pendiente de cualquier cambio, de suceder, lo registra en la tabla *loghistorico*.

Los registros que se trasladaron a *erroresred* permanecerán hasta que se diseñe la red de gas por el predio. Cuando se haya diseñado la red, se digitalizará la instalación y en la columna estado se registrará como 'Cargado' lo que activara dos disparadores, el primero traslada los registros a la tabla *cargados* y el segundo elimina los registros de la tabla *erroresred*.

La tabla *ediciondiariaerrores* se publica para que los responsables puedan responder con la subsanación en la columna respuesta, al responder se activa un disparador que registra la hora y fecha de la respuesta. Si la respuesta es satisfactoria, se registra en la columna estado con la palabra 'Corregido', lo que activa dos disparadores, el primero traslada los registros a la tabla *pendientedibujo* y el segundo elimina los registros.

La tabla *pendientedibujo* almacena todas las instalaciones que fueron corregidas para la digitalización. Después de digitalizar se registra en la columna estado la palabra 'Cargado' lo que activa dos disparadores, el primero traslada los registros a la tabla 'Cargado' y el segundo elimina los registros de la tabla.

Las demás tablas almacenan los datos y para todos ellos se activa un disparador al ingresar un nuevo dato, que captura la hora y fecha.

VI. Conclusiones

Se implementó la plataforma para la migración de instalaciones de gas, se elaboró modelos que encadenan tareas permitiendo reducir el número de procesos a ejecutar. Para facilitar la interacción con las capas de trabajo en la interfaz de ArcMap, se diseñó herramientas personalizadas y para gestionar los errores se elaboró una base de datos en PostgreSQL.

El Model Builder o constructor de modelos de ArcMAP, permitió reducir el número de procesos ejecutados de 225 procesos a solo 32. Pero también permitió simplificar procesos que normalmente no se tienen en cuenta, como eliminar una unión, abrir una capa, unir capas, agregar como campo predeterminado las capas necesarias para la ejecución de herramientas. Como resultado se redujo las horas destinadas a ejecutar todo el proceso, pasando de 7 horas a una hora y media. Además, al usar los modelos creados permitió mantener siempre una estructura de trabajo para todos los grupos, esto trae beneficios estructurales.

El Model Builder se puede ejecutar solo en la versión del ArcMap creado, cuando se tuvo que hacer una actualización en la versión del ArcMap, algunas herramientas dejaron de funcionar. Otra desventaja del Model Builder es la incapacidad de deshacer los cambios realizados en caso fallara a la mitad del proceso, es decir, si un proceso falla al final, todos los procesos previos se ejecutaron, y carece de la capacidad de deshacer los cambios, lo que impide que se vuelva a ejecutar la herramienta después de subsanar el error, debido a que algunas tareas la etiqueta como ya realizadas lo que desactiva el modelo.

Se creó funcionalidades que ayudan a que el proceso sea menos tedioso, por ejemplo, la posibilidad de desactivar todas las capas presionando un botón permite agilizar el entorno de ArcMap ya que deja de procesar la visualización de capas. La herramienta creada para poder visualizar las fotos del Street View desde el ArcMap ayuda enormemente en la resolución de dudas al momento de dibujar la instalación.

Los Add-in permiten llamar a modelos desde una herramienta, esto ayuda significativamente en el acceso de los modelos, evitando navegar desde el ArcCatalog cada vez que se quiera usarlo. Al ejecutarse desde la herramienta permitió agrupar junto a los botones creados.

El addin solo llama a las herramientas creadas desde el Model Builder mas no los almacena en su sistema, a diferencia de los botones creados desde la plantilla del Add-in. Esto trae limitantes y ventajas, se limita porque depende de un archivo externo, que

puede dañarse o perderse, y ventaja, porque siempre se puede actualizar el modelo sin necesidad de volver a ejecutar el instalador del Add-In.

La base de datos permitió automatizar el proceso de reporte y respuesta a los errores, por medio de las funciones y disparadores del PostgreSQL, que trasladan los registros entre tablas según el estado detectado. Como ventaja del uso del PostGIS es que la base de datos siempre va asociado a un predio, lo que facilita la consulta espacial en todo el proceso.

La base de datos al estar implementado en la nube se puede acceder desde cualquier parte del mundo, con una conexión de internet. También permite el acceso desde diversas plataformas según las necesidades y capacidades del que consulta los datos. Para los usuarios SIG se implementó un acceso desde QGIS, esto facilito en la usabilidad, porque el tratamiento que le da QGIS a un archivo de PostGIS es como si fuera un shapefile.

La base de datos esta implementada para la publicación, no está elaborada para que funcione como una base de datos relacional, esto puede ocasionar que algunos procesos no funcionen eficientemente, pero al ser una cantidad reducida diaria estos procesos ineficientes pueden pasar desapercibidos.

VII. Recomendaciones

Los modelos están limitados por las capacidades propias del Model Builder, entonces para mejorar las capacidades del modelo, deberán ser optimizadas usando Python. Al usar Python se puede llamar librerías externas lo que permitiría un mejor funcionamiento.

Se debería repensar el proceso creado con el Model Builder con el fin de simplificar los modelos y así conseguir un funcionamiento más rápido.

La Base de datos también debería pasar un proceso de revisión de la eficiencia de las funciones y disparadores creados con el fin de un mejor funcionamiento y ahorro de espacio en el disco.

El uso de la base de datos está limitado a usuarios que tengan conocimientos de SIG o de SQL, entonces se debería crear una plataforma donde también puedan acceder usuarios no relacionadas a los SIG.

VIII. Bibliografía

- Dobesova, Z. (2013, diciembre). Fortalezas y debilidades en los diagramas de flujo de datos en SIG. En *2013 Conferencia internacional sobre ciencias de la computación y aplicaciones* (pp. 803-807). IEEE
- Ramsey, P. y Columbia, VB (2005). Introducción a Postgis. *Refractions Research Inc* , 34-35.
- Márquez, A. (2015). *PostGIS esencial* . Packt Publishing Ltd.
- Blasby, D. (2001). Construyendo una base de datos espacial en postgresql. *Cumbre de base de datos de código abierto* .
- Toms, S. (2015). *ArcPy y ArcGIS: análisis geoespacial con Python* . Packt Publishing Ltd.
- Zandbergen, PA (2015). *Python scripting para ArcGIS* . Esri press.