

Interfaz de ubicación de unidades de recolección de basura

MARTÍNEZ-AGUILAR, Walter Fernando*†, PENICHE-RUIZ, Larissa Jeanette, GAMBOA-BORGES, José Raymundo, TURRIZA-PEÑA, José Armando

*Tecnológico Nacional de México
Instituto Tecnológico de Mérida*

Recibido Julio 6, 2017; Aceptado Septiembre 20, 2017

Resumen

La recolección de basura es un servicio importante en las ciudades de México; se estima que por ciudad se generan 270 toneladas de basura a la semana. El objetivo es diseñar y desarrollar una aplicación móvil en el sistema operativo Android capaz de ubicar las unidades de recolección por medio de coordenadas de ubicación, mandar alertas en tiempo real y recibir notificaciones push. El desarrollo se realizó en Android Studio con el uso de herramientas como XAMPP y Gennymotion, aplicando la metodología de Desarrollo de Prototipos junto con la de iWeb para la aplicación móvil de usuario. La aplicación permite la visualización en tiempo real de la ubicación del camión de basura con respecto a su ruta y empresa; también permite hacer pagos en línea y la opción de reportes. A la fecha se cuenta con la interfaz de usuario y el módulo de administración montado en un servidor. Se han logrado obtener, enviar y recibir ubicaciones correctas en tiempo real.

Aplicación móvil, ubicación, notificaciones push

Abstract

Urban trash collection is a very important service in all cities in México; 270 tons are estimated to be produced in each city every week. The aim of this project is to design and develop an Android application that is able to locate a trash recollection unit through location coordinates, send the user an alert in real-time and get push notifications. The app was developed in Android Studio with the use of tools like XAMPP and Gennymotion, using the Prototypes Development methodology along with iWeb methodology for the user app. This application allows real time display of the trash truck location according to its route and associated company; it also allows to do the necessary payments on line and it provides an option for reports. Up to date, it counts with the user interface module and the management module mounted on a server. It has been successfully achieved both sending and receiving correct locations with display on real time.

Mobile app, location, push notifications

Citación: MARTINEZ-AGUILAR, Walter Fernando, PENICHE-RUIZ, Larissa Jeanette, GAMBOA-BORGES, José Raymundo, TURRIZA-PEÑA, José Armando. Interfaz de ubicación de unidades de recolección de basura. Revista de Cómputo Aplicado 2017, 1-3: 1-14

* Correspondencia al Autor (Correo Electrónico: aguilar_fer5@hotmail.com)

† Investigador contribuyendo como primer autor.

Introducción

La contaminación en los principales puntos de México es una situación preocupante para toda la población mexicana. En México se recolectan diariamente 86 mil 343 toneladas de basura, pero más de la mitad de basura que se genera en todo el país se recolecta en siete entidades. [1]

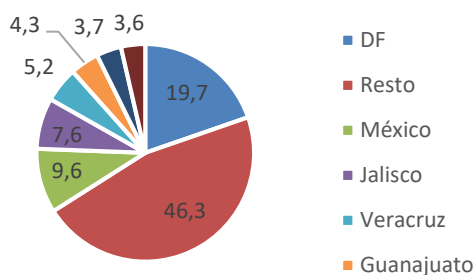


Gráfico 1 Producción de basura en México. Elaboración propia.

En Yucatán se recolectan en promedio 1,361 toneladas diarias de residuos sólidos urbanos o desechos generados en las viviendas, parques, jardines y edificios públicos, principalmente, y representan 2% de la recolección nacional.

El informe estadístico básico sobre el medio ambiente de Yucatán revela que, en cinco municipios, donde reside más de la mitad de la población estatal (52%) se recoge 75% de todos los residuos; Mérida registra la mayor proporción con 832 toneladas (61%), seguido de Tekas, 62 toneladas (4%); Tizimín, 43 toneladas; Hunucmá, 41 toneladas y Umán con 40 toneladas. Según los datos publicados por el Instituto Nacional de Estadística Geografía e Informática (INEGI), la recolección promedio diario por habitante a nivel estatal es alrededor de 1.70 kg. [2]

La aplicación móvil en un principio fue diseñada para el estado de Yucatán, sin embargo a medida que se avanzó en la investigación es claro que puede ser implementada en otros estados del país dado que sus condiciones de aplicación son generales.

Planteamiento del problema.

El sistema de recolección de basura en la ciudad de Mérida, Yucatán cuenta con diferentes días de recolección y cobranza del servicio. En la actualidad existen quejas de los usuarios respecto al servicio de recolección de basura, por diversas causas: falta de servicio al no pasar el camión, recolección incompleta, cambio de rutas sin previo aviso a los usuarios; por parte del usuario y no imputable a las compañías de servicio de recolección está por supuesto que al momento de sacar la basura los animales callejeros destruyen las bolsas y la basura se esparce por todas partes; también está el simple olvido de sacar la basura.

De igual forma cuando alguien de la empresa pasa a cobrar la basura y no encuentran al dueño de la casa, el servicio se suspende. Esto enoja al usuario que lo percibe como suspensión del servicio, aunque sea, no por no poder o querer pagar, sino porque no hay a quien cobrarle.

En tal situación, todo se presta a que las personas digan que existen muchos inconvenientes en el servicio de recolección de basura que se les está prestando. Solo en el mes de octubre del año 2015 se recibieron 7500 quejas sobre el servicio de recolección de basura en la ciudad de Mérida, teniendo un promedio de 8000 quejas mensuales durante ese año. [3]

Ante esta situación se propone desarrollar una aplicación móvil que ayude a los ciudadanos a tener facilidades para pagar el servicio y tener aviso oportuno de la ubicación de los camiones dentro de su ruta de recolección permitiendo tiempo suficiente para sacar su basura.

Interfases con Google Material Design.

Esta nueva interfaz para aplicaciones, la más reciente usada por Google, provee novedosos materiales de animación y estilo además de actividades de transición.

Este nuevo topic, Material Design, tiene nuevos widgets para vistas complejas con sombras y animaciones personalizadas. Algunos de estos elementos son los siguientes: Material Design permite configurar la paleta de colores y animaciones predeterminadas para información táctil y transiciones de actividades.

Material Design permite desplegar listas y tarjetas con estilos y animaciones predefinidas.

Lo más interesante es quizás que, aparte de las clásicas propiedades horizontal X y vertical Y, las vistas tienen una propiedad Z nueva, que representa la elevación o sobre posición de un elemento sobre la vista actual y en consecuencia genera una vista sombreada para permitir que el nuevo elemento en Z destaque sobre la interfaz. El elemento en Z activo debe ser resuelto antes de continuar.

Por último, las nuevas API de animaciones pueden responder a eventos táctiles de cada acceso mediante animaciones de información táctil; permiten ocultar y desplegar circularmente las vistas con animaciones y hasta pueden alternar entre actividades con animaciones personalizadas de transición de actividades.

Estado del arte

Existen proyectos similares que cumplen parcialmente con las funcionalidades antes mencionadas:

Mérida Móvil: Aplicación desarrollada para dispositivos móviles que muestra las rutas de recolección de basura en Mérida [4]. Ver Figura 1.



Figura 1 Mapa de la aplicación Mérida Móvil. Elaboración propia

Uber: Uber es una empresa de tecnología. Usando su aplicación, los usuarios que necesitan transporte consiguen fácilmente encontrar socios conductores que ofrecen este servicio. Uber ofrece una opción más para moverse por una ciudad, con más estilo, seguridad y comodidad. Está presente en más de 310 ciudades alrededor del mundo [5]. Ver Figura 2.

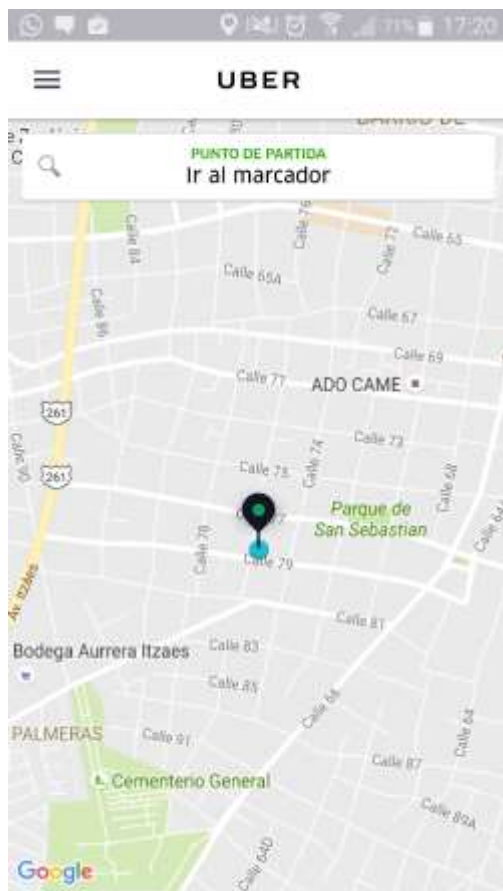


Figura 2 Vista principal aplicación Uber. Elaboración propia

i-City: Este proyecto “Sistema de gestión integral de reciclaje y recolección de residuos” obtuvo el segundo lugar en el Concurso Regional de Proyectos Innovadores 2016, organizado la Asociación Nacional de Universidades e Instituciones de Educación Superior (ANUIES) [6] . El objetivo de aprovechar los recursos tecnológicos con base en el internet de las cosas en búsqueda de una ciudad inteligente, reluciente y sostenible que solucione los problemas de programación en las rutas de recolección y reciclaje. Con la aplicación móvil, el usuario podrá saber la ubicación de los puntos limpios, centros de reciclaje y la forma correcta de separar los residuos, siendo el medio de enlace al mercado de valoración de sus residuos.

Metodología por prototipos.

1.-Investigación preliminar.

En esta etapa lo esencial es determinar el problema y su ámbito, la importancia y los efectos potenciales que tendrán sobre la organización, identificar una idea general de la solución para realizar un estudio que determine la factibilidad de una solución software.

2.-Definición de los requerimientos del sistema.

Esta es la fase más importante de todo el ciclo de vida del método de prototipos, el objetivo en esta fase es determinar todos los requerimientos y deseos que los usuarios tienen en relación al proyecto que se está deseando implementar.

Aquí el desarrollador interactúa con el usuario y sus necesidades mediante la construcción, demostración y retroalimentaciones del prototipo.

3.-Análisis de los requerimientos y prototipo.

Esta etapa es un proceso que busca aproximar las visiones del usuario y del desarrollador mediante sucesivas iteraciones.

Para la definición de los requerimientos tenemos cinco etapas entre dos de las cuales se establece un ciclo interactivo

- Análisis grueso y especificación

En esta fase se busca desarrollar un diseño básico para el prototipo inicial.

- Diseño y construcción

Lo que se consigue en esta fase es obtener un prototipo inicial, aquí el desarrollador debe concentrarse en construir un sistema con la máxima funcionalidad, poniendo énfasis en la interfaz del usuario.

– Evaluación

Los objetivos de esta etapa son obtener por parte de los usuarios la especificación de los requerimientos adicionales del sistema y verificar que el prototipo desarrollado lo haya sido en concordancia con la definición de requerimientos del sistema.

En el caso de que los usuarios identifiquen fallas en el prototipo el desarrollador corrige dichas fallas antes de continuar con la siguiente evaluación.

Se modifica y se evalúa cuantas veces sea necesario hasta que los requerimientos del sistema sean satisfechos.

En el proceso de evaluación se efectúan cuatro pasos separados:

- Preparación.
- Demostración.
- Uso del prototipo.
- Discusión de comentarios.

Esta es la fase en donde se decide si el prototipo es aceptado o modificado.

– Modificación

Se da cuando la definición de requerimientos del sistema es alterada en la etapa de evaluación. El desarrollador entonces debe modificar el prototipo de acuerdo a los comentarios hechos por los usuarios.

– Término.

Una vez que se ha desarrollado un prototipo estable y completo, es necesario ponerse de acuerdo en relación a aspectos de calidad y de representación del sistema.

4.- Diseño técnico.

En esta etapa el sistema debe ser rediseñado y tener la respectiva documentación guiándose en los estándares que tiene la organización la cual servirá como ayuda en mantenciones futuras del mismo.

En este punto existen dos etapas:

- Producción de una documentación de diseño la cual especifica y describe la estructura del software, interfaces de usuario, funciones y el control de flujo.
- Producción de todo lo requerido para promover cualquier mantención futura del software.

5.- Programación y prueba.

En esta etapa es donde los cambios identificados en el diseño técnico son implementados y probados para asegurar la corrección y completitud de los mismos con respecto a los requerimientos.

Las pruebas serán de realizarse tantas veces sea necesarias para verificar cualquier tipo de anomalía en el sistema.

6.- Operación y mantención

En esta fase se realiza ya la instalación y mantenimiento del software, la complejidad en este caso resulta menor ya que en las etapas anteriores los usuarios han trabajado con el sistema al momento de hacer las pruebas de prototipos, además la mantención también debería ser una fase menos importante, ya que se supone que el refinamiento del prototipo permitiría una mejor claridad en los requerimientos, mediante lo cual las mantenciones perfectivas se reducirían.

Si existiese el caso en el cual se requiera una manutención entonces el proceso de prototipado es repetido y se definirá un nuevo conjunto de requerimientos[7].

Implementación

- La aplicación móvil Clean City Camión desarrollada en Android, envía una notificación push con la ubicación de la unidad de recolección de basura a un servidor para luego ser consultada por un usuario. Es capaz de enviar la ubicación de las unidades de recolección de basura en tiempo real. Otra notificación push permite que el usuario pueda ser avisado de la cercanía a su domicilio de una unidad de recolección y así darle tiempo al usuario, ya sea de sacar la basura o hacer arreglos para que esto se realice en el domicilio.

La aplicación Android se ha desarrollado siguiendo las fases de la metodología de prototipos.

Fase 1 – Planeación y análisis.

Se llevó a cabo la ingeniería de requerimientos donde se obtuvieron los requerimientos funcionales y no funcionales:

- Desplegar al usuario la ubicación de la unidad de recolección de basura.

- Crear alarmas personales.
- Recibir notificaciones en tiempo real de cuando el camión se encuentre cerca del domicilio.
- Realizar pagos en línea.
- Permitir al usuario la generación de reportes en caso de alguna anomalía o queja.
- La interfaz deberá ser adaptable al cliente.
- Requiere conexión a internet.

El objetivo fijado es diseñar y desarrollar una aplicación Android que permita obtener en tiempo real la ubicación de una unidad de recolección de basura y desplegarla en la interfaz del usuario, además de facilitar las alarmas personalizadas, el pago en línea y el envío de quejas o información relativa al servicio.

Fase 2 – Diseño rápido.

Una vez identificados los requerimientos funcionales y no funcionales, se elabora el diagrama de la arquitectura del servidor y la aplicación, como se muestra en la Figura 3.



Figura 3 Arquitectura de la aplicación. Elaboración propia

La arquitectura de la aplicación en la Figura 3 muestra todos los módulos que la componen: En el lado izquierdo los 4 módulos del usuario final Ubicación, Noticias, Reportes, Pagos, y Noticias se enlazan al servidor Clean Server, quien a su vez se enlaza a los módulos Camión y Alertas asociados a cada unidad recolectora de la empresa como se muestra en el lado derecho de la figura 3. El proceso de ubicación de unidades de recolección se detalla más adelante. De la mano de este módulo se pueden configurar las alarmas personalizadas de cada usuario para tener avisos oportunos de sacar su basura a tiempo de ser recolectada; esto de nuevo se implementa mediante notificaciones push del sistema. Por otro lado, el módulo Camión se encarga de enviar vía GPS e Internet en lapsos desde 3 seg. hasta 15 seg. sus coordenadas exactas de ubicación. Además, otras transacciones generadas por la aplicación se guardan en el servidor llamado Clean Server y pueden ser consultadas por la aplicación de usuario mediante el uso de un Web Service desarrollado exclusivamente para la arquitectura del Sistema. La empresa encargada del servicio de recolección puede usar el módulo de Noticias para comunicar a sus usuarios afiliados sobre calendario de recolección, modificaciones y eventos especiales como campañas de descacharrización, por ejemplo. El módulo Pagos estará encargado de facilitar el cumplimiento del pago en línea de este servicio a los usuarios quienes podrán programar un recordatorio de fecha de pago y mediante notificaciones push la app genera el aviso. De este modo para el usuario será cómodo realizar sus pagos y la empresa tendrá una mejora en su recaudación. Finalmente, el módulo de Reportes permite una comunicación permanente entre empresa y usuarios de modo que se pueda informar de sucesos y poner quejas dado el caso.

El módulo de Ubicación funciona principalmente mediante un algoritmo creado para la búsqueda de ubicaciones de las unidades de recolección de residuos. Este algoritmo sufrió varias mejoras a lo largo de su creación e implementación hasta lograr satisfacer las necesidades del sistema, sin que el usuario pierda la sensación de que todo funciona en tiempo real.

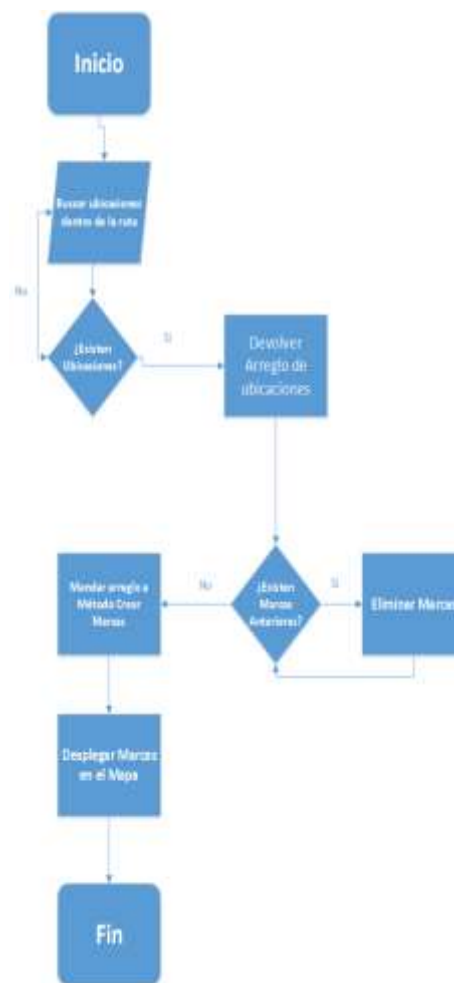


Figura 4 Diagrama de flujo del algoritmo Búsqueda de Ubicaciones. Elaboración propia

La Figura 4 muestra el diagrama de flujo para el algoritmo de búsqueda de ubicaciones. Inicialmente, se debe tener la ubicación del domicilio alrededor de cuya ruta asignada es necesario encontrar las unidades de recolección, las cuales mandan una notificación push en tiempo real con su ubicación al servidor.

Al momento de encontrar unidades, el Web Service implementado devolverá un arreglo de ubicaciones, que se marcan por la aplicación para su despliegue en el mapa. Este arreglo es clave en este proceso pues la aplicación lo valora para eliminar marcas antiguas y actualizar las ubicaciones cada vez que sea necesario para mantener actualizadas las ubicaciones de los camiones recolectores activos.

En la Figura 5 se muestra el diagrama de caso de uso general de la aplicación Clean City App desarrollada en Android, en donde se puede observar la comunicación entre las aplicaciones y el servidor.

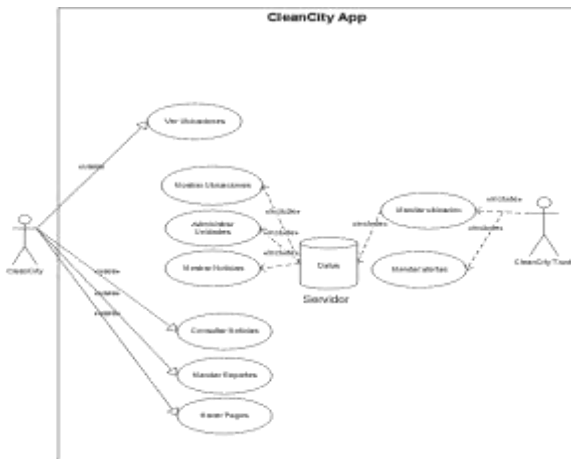


Figura 5 Diagrama de caso de uso de la aplicación Clean City App. Elaboración propia.

En función al objetivo y los requerimientos obtenidos se diseñan [8], los diagramas de caso de uso para la función Ingresar del módulo Ubicación y para el módulo Camión, ilustradas en las Figuras 6 y 7 respectivamente.

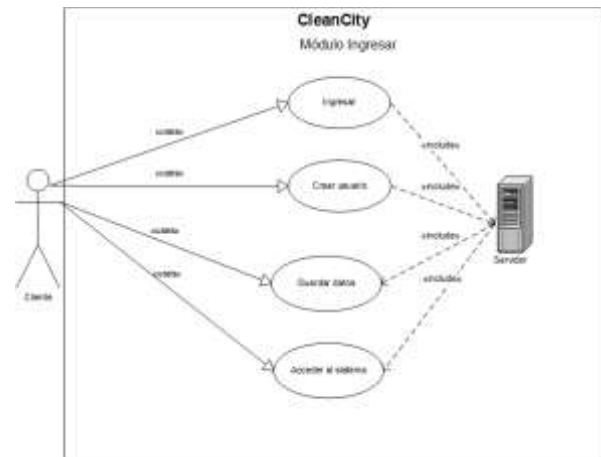


Figura 6 Diagrama de caso de uso de ingresar. Elaboración propia

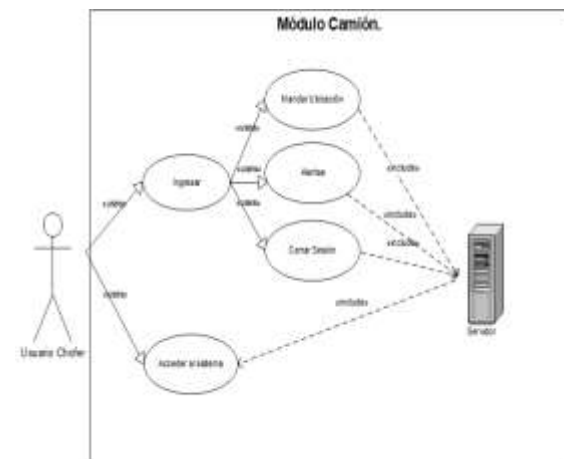


Figura 7 Diagrama de caso de uso del módulo Camión. Elaboración propia

Dados los requerimientos de la aplicación propuesta, se diseña el diagrama del modelo relacional, que consta de 10 tablas entre las que se encuentran las de Ubicación, Domicilio, Rutas y Unidades como se muestra en la Figura 8. Estas tablas permiten articular cada módulo para el correcto funcionamiento de la aplicación. La base de datos es elemento clave para el cálculo, envío, despliegue y recálculo de las ubicaciones de cada unidad de recolección de basura.

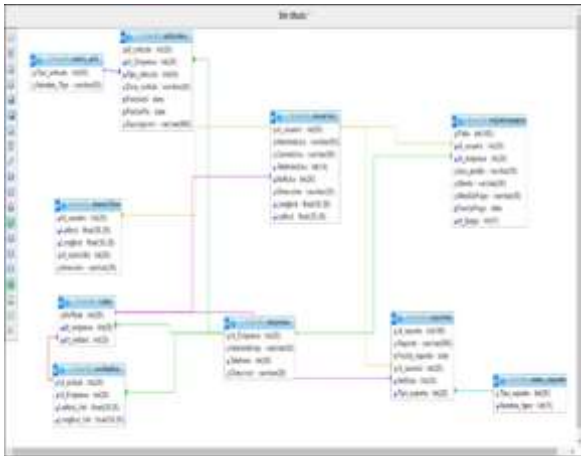


Figura 8 Diagrama relacional de Clean City App. Elaboración propia

Diseño de la interfaz

En las Figuras 9 y 10 se pueden observar

Los primeros diseños de las distintas interfaces de la aplicación Clean City App.

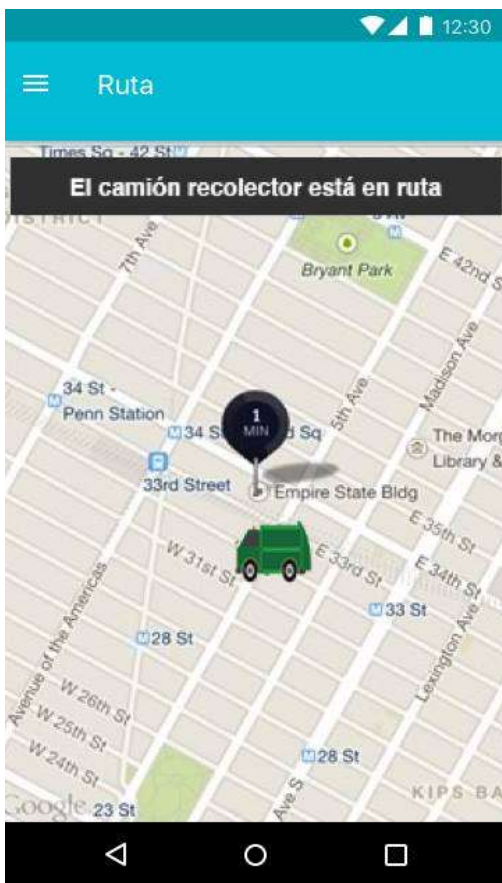


Figura 9 Diseño módulo Ubicación Clean City App. Elaboración propia

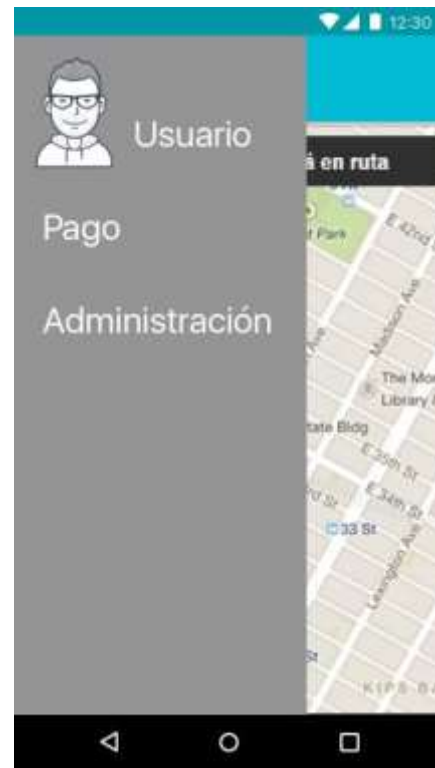


Figura 10 Menú desplegable Clean City App. Elaboración propia

Fase 3 – Construcción del prototipo.

Para acceder a la aplicación Clean City App es necesario registrarse en el sistema con un usuario y contraseña que es el único medio para autenticar al usuario válido en el sistema. Esta ventana de Inicio se muestra en la Figura 11.

La aplicación móvil se encuentra dividida en módulos distintos los cuales fueron programados de manera separada para luego ser integrarlos en una sola aplicación y puedan funcionar en sinergia. Hay 3 módulos que son imprescindibles:



Figura 11 Ventana “Inicio”. Elaboración propia

El primer módulo llamado Ubicación se encarga de enviar al servidor las coordenadas geográficas del domicilio del usuario y recibir las ubicaciones de los camiones que pertenezcan a su ruta, siempre y cuando éstos se encuentren dentro del rango de la ruta del domicilio. Ver Figura 12. También se encarga de desplegar en tiempo real la ubicación de cada unidad de recolección. La interfaz que se despliega al cliente se adapta según su propio logo y paleta de colores asociadas a la empresa.

En este módulo se implementaron los elementos en Z que permiten que un menú desplegable se encuentre flotando sobre la vista sombreada del mapa, haciendo que el mapa presente un tema más opaco en una vista en Z.

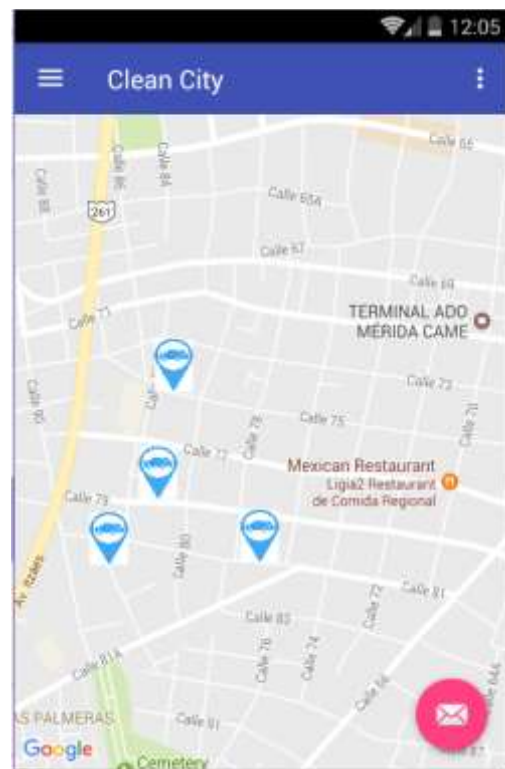


Figura 12 Ventana “Ubicación”. Elaboración propia.

El segundo módulo importante es el de Reportes que permite al usuario realizar reportes de incidencias que pudieran presentarse durante sus días de recolección de residuos. Los usuarios pueden clasificar cada reporte en diferentes tipos de situaciones; esto permite al administrador de la compañía visualizar los reportes en un ambiente más amigable para su manejo, permitiendo una mejor respuesta a cada tipo de incidencia. Como se observa en la Figura 13, la interfaz despliega un área de texto de hasta 300 caracteres para teclear el aviso o mensaje que se desea transmitir. Al terminar el mensaje, el usuario puede simplemente pulsar el botón de envío de correo sin preocuparse de saber la dirección de correo ya que se tiene pre cargada.



Figura 13 Ventana “Reportes”. Elaboración propia

El tercer módulo es el de Camión, que permite a las unidades de recolección mandar su ubicación en tiempo real en lapsos de 3 seg hasta 15 seg, según las necesidades de la empresa. Estas ubicaciones son almacenadas en Clean Server, esto para que el módulo Ubicación pueda desplegarlas para el usuario.

Fase 4 – Evaluación del prototipo por usuarios

La aplicación fue puesta a prueba en un ambiente controlado en los parámetros siguientes: 20 personas elegidas al azar, pero que pertenecieran a la misma ruta de recolección de residuos; 4 unidades de recolección; con 6 seg de intervalo de tiempo entre envíos de ubicación.

El módulo Camión encargado de enviar las ubicaciones en tiempo real, funcionó correctamente en las 4 unidades en movimiento.

Mientras estas unidades se movían por la zona, todas las personas podían contar con la aplicación de Clean City App para visualizar el movimiento de las unidades. En este módulo se incluyó un modo de identificar a qué compañía pertenece la ruta, dado que los colores que presentan los camiones, se asocian al color distintivo de cada una de las empresas. Las personas que fungieron como unidades de recolección instalaron la aplicación Clean City App en un 100% y proporcionaron los siguientes comentarios:

- Interfaz amigable para el chofer.
- Sin necesidad de monitoriar la aplicación.
- La aplicación es fácil de usar.
- Las alertas son enviadas en un formato sencillo y agradable.

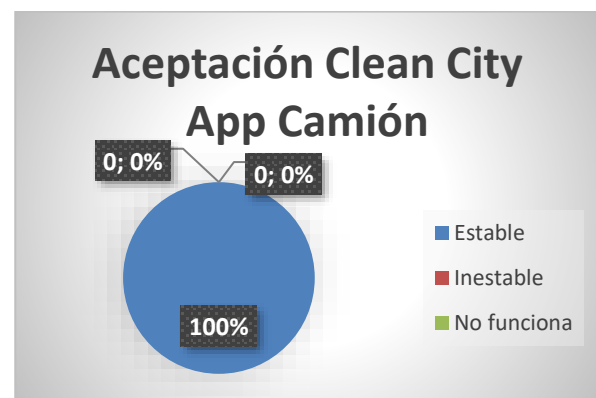


Gráfico 2 Gráfica de aceptación Clean City Camión. Elaboración propia

De las 20 personas que utilizaron la aplicación Clean City App los resultados fueron los siguientes:

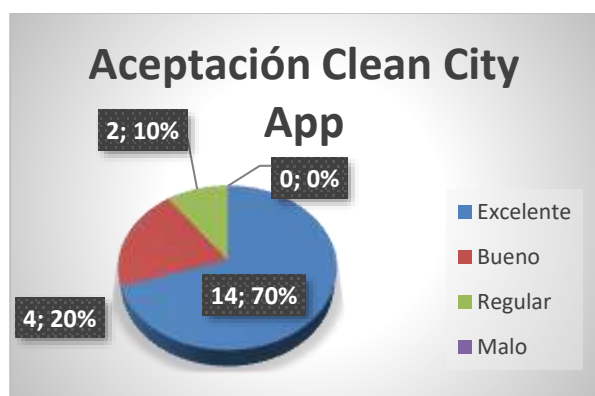


Gráfico 3 Gráfica de aceptación Clean City App. Elaboración propia

Los usuarios manifiestan amplio gusto por la aplicación, y sugieren lo siguiente:

- La aplicación es fácil de utilizar, pero les gustaría más temas de personalización.
- El apartado de reportes es claro aunque pueden incluir más clasificaciones.
- Que la aplicación pueda utilizarse en diferentes plataformas móviles, no solo Android.
- Que la aplicación permita agregar más domicilios.

Resultados y discusión

La aplicación es realmente de mucha ayuda para los usuarios, dado que les permite tener un mejor manejo de los residuos en los domicilios; muchos usuarios manifestaron contento con esta idea, que ayudan a la conservación del medio ambiente, dado que apoyan a evitar la contaminación. La intención es que esta nueva aplicación pueda ser implementada en las ciudades más grandes de México y eventualmente sea una aplicación utilizada en todos los rincones del territorio mexicano.

Las consideraciones que fueron propuestas por los usuarios serán tomadas en cuenta para una próxima iteración del sistema, pudiendo mejorar la aplicación hasta lograr una aplicación móvil robusta, de alto rendimiento y amigable para el usuario.

Conclusiones

La aplicación móvil cumple con las principales características planteadas para su desarrollo. Con base en Material Design y las pruebas de aceptación realizadas la interfaz es amigable y agradable al usuario.

Durante el desarrollo de la base de datos se encontraron diferentes problemáticas, dado que cada vez que el sistema crecía en requerimientos era necesario rediseñar la base de datos para un mejor funcionamiento. Se sabe que el sistema necesita crecer aún más para poder tener una base de datos sólida que ayude a un mejor funcionamiento del sistema, pero es necesario ser cuidadosos con el sistema.

Una problemática surgió mientras se creaba el algoritmo para la búsqueda de ubicaciones de las unidades de recolección. El sistema Android tiene ciertas limitantes en la realización de procesos, por lo cual fue necesario, crear nuevos algoritmos que logran satisfacer las necesidades de la aplicación, así como adaptarse a las características y requerimientos del sistema operativo Android.

En la siguiente etapa, se trabajará el prototipo para hacerlo robusto y seguro mediante algoritmos de seguridad que pueda coexistir en el internet sin temor a perder datos sensibles de los usuarios, pero maximizando eficiencia.

Agradecimientos

Se agradece al Tecnológico Nacional de México y al Instituto Tecnológico de Mérida por el valioso apoyo brindado para la realización de este proyecto.

Referencias

Cuentame.inegi.org.mx. (2016). Medio ambiente. Cuéntame de México. [online] Available at: <http://cuentame.inegi.org.mx/territorio/ambiente/basura.aspx?tema=T> [Accessed 16 Sep. 2016].

Ana Hernández . SIPSE, G. (2013). Manejo de basura es preocupante. [online] SIPSE.com. Available at: <http://sipse.com/milenio/manejo-de-basura-es-preocupante-25711.html> [Accessed 18 Sep. 2016].

Ana Hernández . SIPSE, G. (2015). Basureros se reparten el negocio, pero no las quejas. [online] SIPSE.com. Available at: <http://sipse.com/milenio/merida-recibe-comuna-quejas-por-recoleccion-basura-173368.html> [Accessed 18 Sep. 2016].

Reporteros Hoy - Noticias de Yucatán. (2016). APP para recolección de basura en Mérida Reporteros Hoy - Noticias de Yucatán. [online] Available at: <http://reporteroshoy.mx/wp/app-recoleccion-basura-merida.html> [Accessed 13 Nov. 2016].

Uber Blog. (2015). Pero al fin de cuentas, ¿Qué es Uber y cómo se usa? | Uber Blog. [online] Available at: <https://www.uber.com/es-CL/blog/que-es-uber/> [Accessed 14 Oct. 2016].

Gaceta.udg.mx. (2016). La gaceta UdeG | Una app para la recolección de basura. [online] Available at: http://www.gaceta.udg.mx/G_notas1.php?id=20662 [Accessed 13 Nov. 2016].

TELECOMUNICACIONES, G., TELECOMUNICACIONES, G. and perfil, V. (2011). MODELO DE PROTOTIPO. [online] Gestionrrhhusm.blogspot.mx. Available at: <http://gestionrrhhusm.blogspot.mx/2011/05/modelo-de-prototipo.html> [Accessed 6 Aug. 2016].

Barquintero, J. (2013). Tipos de relaciones en diagramas de casos de uso. UML. | Blog SEAS. [online] Seas.es. Available at: <http://www.seas.es/blog/informatica/tipos-de-relaciones-en-diagramas-de-casos-de-uso-uml/> [Accessed 13 Nov. 2016].

Benbourahla, N. (2015). Android 5. [online] Google Books. Available at: <https://books.google.com.mx/books?hl=es&lr=&id=XIcXXFIO1FgC&oi=fnd&pg=PA13&dq=Android+5&ots=FCUpPjUyls&sig=cSzZbLXkQdpUIRz2x6EJnml9Zi0#v=onepage&q=Android%205&f=false> [Accessed 14 Dec. 2016].

Loor Villamar, A. D., & Morocho Ramos, M. A. (2016). Desarrollo e implementación de un aplicativo para dispositivos móviles con sistema operativo android y geolocalización que permita ofrecer y solicitar servicios de alquileres de casas con ubicación exacta ya la vez sugiera las más cercanas a la ubicación actual en cualquier parte de guayaquil (Doctoral dissertation, Universidad de Guayaquil. Facultad de Ciencias Matemáticas y Físicas. Carrera de Ingeniería en Sistemas Computacionales).<http://repositorio.ug.edu.ec/handle/redug/16296>

Acar, M., Bormann, C., Bray, T., Davies, J., Dürst, M. J., Höhrmann, B., ... & Reschke, J. JSON Pointer draft-ietf-appsawg-json-pointer-02. <https://tools.ietf.org/id/draft-ietf-appsawg-json-pointer-01.html>

Nottingham, M. (2012). JSON Pointer draft-ietf-appsawg-json-pointer-04.<https://tools.ietf.org/id/draft-ietf-appsawg-json-pointer-04.html>

Soriano, J. E. A. (2011). Android: Programación de dispositivos móviles a través de ejemplos. Marcombo.<https://books.google.com.mx/books?hl=es&lr=&id=Akha4zgQUzIC&oi=fnd&pg=PA11&dq=como+hacer+una+aplicacion+android+con+gps&ots=28LzlrY90M&sig=gzvv1y97yixk5MWfX1KuOjP-HDY#v=onepage&q=como%20hacer%20una%20aplicacion%20android%20con%20gps&f=false>

Urrutia, A. G., & GONZALEZ, R. C. (2013). Cabify.<http://repositoriodigital.corfo.cl/handle/11373/8445>

Jarquín González, M. L., Jaime, B., Margine, D., Barrios, A., & Pavel, Y. (2015). Aplicación para el envío masivo de correo electrónico utilizando PhpList para la Empresa COMVARSA en Ciudad de Managua, año 2015 (Doctoral dissertation, Universidad Nacional Autónoma de Nicaragua, Managua). <http://repositorio.unan.edu.ni/1362/>

Oh, J. H., Hanson, D., Kim, W. S., Han, Y., Kim, J. Y., & Park, I. W. (2006, October). Design of android type humanoid robot Albert HUBO. In Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on (pp. 1428-1433). IEEE.<http://ieeexplore.ieee.org/abstract/document/4058572/>