



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE INGENIERÍA

CARRERA DE COMPUTACIÓN

TEMA:

**Diseño e implementación de un sistema de acceso vía móvil,
web y escritorio para el control de reservas y aforos usando
Bots de Telegram y códigos QR**

AUTOR:

Roldán Carrillo, Jofre Ronny

**Trabajo de titulación previo a la obtención del título de
INGENIERO EN CIENCIAS DE LA COMPUTACIÓN**

TUTOR:

Ing. Castro Aguilar, Gilberto Fernando

Guayaquil, Ecuador

4 de marzo de 2022



UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL

**FACULTAD DE INGENIERÍA
CARRERA DE COMPUTACIÓN**

CERTIFICACIÓN

Certificamos que el presente trabajo de integración curricular fue realizado en su totalidad por **Roldán Carrillo Jofre Ronny** como requerimiento para la obtención del título de **Ingeniero en Ciencias de la Computación**.

TUTOR

f.



Firmado electrónicamente por:
**GILBERTO
FERNANDO CASTRO
AGUILAR**

Ing. Castro Aguilar, Gilberto Fernando

Guayaquil, a los 4 días del mes de marzo del año 2022



UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL

**FACULTAD DE INGENIERÍA
CARRERA DE COMPUTACIÓN**

DECLARACIÓN DE RESPONSABILIDAD

Yo, **Roldán Carrillo Jofre Ronny**

DECLARO QUE:

El Trabajo de integración curricular: **Diseño e implementación de un sistema de acceso vía móvil, web y escritorio para el control de reservas y aforos usando Bots de Telegram y códigos QR**, previo a la obtención del título de **Ingeniero en Ciencias de la Computación**, ha sido desarrollado respetando derechos intelectuales de terceros conforme las citas que constan en el documento, cuyas fuentes se incorporan en las referencias o bibliografías. Consecuentemente este trabajo es de mi total autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance del Trabajo de Titulación referido.

Guayaquil, a los 4 días del mes de marzo del año 2022

EL AUTOR

f. _____

Roldán Carrillo Jofre Ronny



UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL
FACULTAD DE INGENIERÍA
CARRERA DE COMPUTACIÓN


AUTORIZACIÓN

Yo, Roldán Carrillo Jofre Ronny

Autorizo a la Universidad Católica de Santiago de Guayaquil a la **publicación** en la biblioteca de la institución del Trabajo de Titulación: **Diseño e implementación de un sistema de acceso vía móvil, web y escritorio para el control de reservas y aforos usando Bots de Telegram y códigos QR**, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y total autoría.

Guayaquil, a los 4 días del mes de marzo del año 2022

EL AUTOR:

f. 

Roldán Carrillo Jofre Ronny



UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL
FACULTAD DE INGENIERIA
CARRERA DE COMPUTACIÓN

REPORTE URKUND

URKUND

Documento [Titulación Jofre Roldán P1.docx](#) (D125325194)
Presentado 2022-01-17 15:12 (-05:00)
Presentado por jofre.roldan@cu.ucsg.edu.ec
Recibido gilberto.castro.ucsg@analysis.orkund.com
Mensaje Titulación Jofre Roldán [Mostrar el mensaje completo](#)
0% de estas 41 páginas, se componen de texto presente en 0 fuentes.

TUTOR



Firmado electrónicamente por:
**GILBERTO
FERNANDO CASTRO
AGUILAR**

f. _____

Ing. Castro Aguilar Gilberto Fernando

AGRADECIMIENTO

Agradezco en primer lugar a Dios, por permitirme concluir esta carrera. Fue un sueño desde niño y por fin pude llegar a la meta, cumpliendo ese sueño.

Agradezco a mi madre Silvia Carrillo, quien siempre estuvo conmigo dándome ánimos para seguir adelante, recibéndome con los brazos abiertos cuando más lo necesité y estuve a punto de tirar la toalla. Este título no es solo mío, este título se lo merece ella por ser la mejor madre del mundo y por sobre todas las cosas permitirme volar alto y siempre demostrándome que puedo dar más de mí y que no debo bajar la cabeza ante nadie ni rendirme ante nada sin agotar las posibilidades.

Agradezco a mi hermana Zmarys Roldán por su inmensa ayuda durante la carrera, muchos trabajos míos no hubieran tenido el nivel de detalle y calidad que tienen si no hubiera sido gracias a ella y a sus conocimientos sobre ingeniería comercial.

Agradezco a mi tío Telmo Carrillo, ya que gracias a él pude encontrar mi vocación y descubrir esta hermosa carrera. Esas pequeñas tardes de enseñanza pesaron mucho en quien soy hoy y esos recuerdos los llevaré conmigo para siempre.

Agradezco a Hector Campaña, alguien que, aunque no es directamente de mi familia significa mucho para mí. Gracias a él pude entender el valor del estudio y como siempre puedes ser el “wingman” de alguien cuando menos se lo esperé y de manera desinteresada. Queda muchas anécdotas, que serán recuerdos que siempre tendré en mi mente y en mi corazón.

Agradezco a mi tutor Fernando Castro, por acompañarme durante todo este proceso, quien compartió sus conocimientos y me brindó un excelente asesoramiento para el desarrollo de este proyecto.

Agradezco a mis abuelitos Telmo y Cecilia, ellos fueron una parte fundamental en mi carrera, siempre siendo una mano para ayudar en los días en todo se complicaba por situaciones adversas. Los quiero mucho.

Agradezco a Migajas, mi mascota, por acompañarme en los arduos días y noches de trabajo, siempre cuidando de mí y haciendo más llevadero todo este proceso.

Jofre Roldán Carrillo

DEDICATORIA

Dedico este proyecto a Dios y a la virgen María, por acompañarme en este último paso para lograr la meta de ser ingeniero.

A mi madre, Silvia Carrillo, por siempre acompañarme durante esta aventura tan complicada en la vida de cualquier persona y por todo el gran esfuerzo que realizó para que pueda estar aquí hoy.

A mi hermana, Zmarys Roldán, por siempre aportar su granito de arena hasta las últimas instancias y por qué significa mucho para mí.

A Hector Campaña, quien me enseñó que el estudio es importante y puede cambiar vidas.

A Telmo Carrillo, por esas tardes siendo infiltrado en sus proyectos y aprendiendo de todo lo que él hacía.

Muchas veces nos vamos a encontrar solos contra el mundo y con ganas de tirar la toalla, pero como dice Porter Weston Robinson: Siempre debemos escuchar a esa voz interna que habita dentro de nosotros y te dice “Ya estas cerca, tú puedes, tú puedes, sigue adelante, no te rindas, todo estará bien al final” y darnos ánimos a nosotros mismos para continuar.

Nunca debemos olvidar quienes somos, de donde vinimos, hacia donde vamos, quienes nos apoyaron, todo lo que pasamos y, sobre todo, celebrar lo que logramos.

Jofre Roldán Carrillo

ÍNDICE

ÍNDICE.....	VIII
RESUMEN.....	XIV
ABSTRACT.....	XV
INTRODUCCIÓN.....	2
CAPÍTULO I.....	4
EL PROBLEMA.....	4
1.1 Descripción del problema.....	4
1.2 Formulación del problema.....	5
1.3 Hipótesis de investigación.....	6
1.4 Objetivos de la investigación.....	6
1.4.1 Objetivo general:.....	6
1.4.2 Objetivos específicos de la investigación:.....	6
1.5 Alcance.....	7
1.6 Justificación e importancia del problema.....	7
CAPÍTULO II.....	9
MARCO TEÓRICO.....	9
2.1 Los chatbots y su aporte en los procesos de control de reservas y aforos.....	9
2.1.1 Transformación digital de las sociedades.....	10
2.1.1.1 Transformación digital en el Ecuador.....	11
2.2 Conceptos Varios.....	12
2.2.1 Lenguajes de programación.....	12

2.2.2 Python.....	12
2.2.3 JavaScript.....	12
2.2.4 Ruby	13
2.2.5 C#.....	14
2.2.6 Java	14
2.2.7 GO	15
2.2.8 API.....	16
2.2.9 Telegram.....	16
2.2.10 Chatbot	17
2.2.11 Computación en la nube	17
2.2.12 Heroku	17
2.2.13 AWS (Amazon Web Services)	18
2.2.14 Microsoft Azure.....	18
2.2.15 Google Cloud Platform (GCP)	19
2.2.16 Infraestructura como servicio (IaaS)	19
2.2.17 Plataforma como servicio (PaaS).....	20
2.2.18 Software como servicio (SaaS).....	20
2.2.19 Raspberry Pi	21
2.2.20 Pi Camera.....	21
2.2.21 Raspbian.....	21
2.2.22 DietPi	21
2.2.23 Módulos de Python	22
2.2.24 PIP	22

2.2.25 Paquetes de C#	22
2.2.26 NuGet	22
2.2.27 Telegram Bot API.....	23
2.2.28 pyTelegramBotAPI.....	23
2.2.29 Abstract API.....	23
2.2.30 OpenCV	23
2.2.31 Pyzbar.....	23
2.2.32 PIL (Pillow).....	24
2.2.33 PyMsgBox.....	24
2.2.34 Requests.....	24
2.2.35 Códigos QR	24
2.2.35.1 Códigos QR estáticos.....	25
2.2.35.2 Códigos QR dinámicos.....	25
2.2.36 MySQL Connector/Python	25
2.2.37 Bases de datos	25
2.2.37.1 Bases de datos relacionales	26
2.2.37.2 Bases de datos no relacionales.....	26
2.2.38 MySQL.....	26
2.2.39 ClearDB	27
2.3 Fundamentos legales del proyecto.....	27
CAPÍTULO III METODOLOGÍA DE LA INVESTIGACIÓN	29
3.1 Definición de la investigación.....	29
3.1.1 Investigación cualitativa	29

3.1.2 Investigación cuantitativa.....	30
3.1.3 Técnica del cuestionario	31
3.1.4 Análisis del cuestionario	32
CAPÍTULO IV.....	37
PROPUESTA TECNOLÓGICA.....	37
4.1 Herramientas y recursos para el desarrollo del sistema.....	39
4.2 Requerimientos de software.....	45
4.3 Requerimientos de hardware	46
4.4 Base de datos	46
4.5 Modelo entidad relación	48
4.6 Diagrama de arquitectura de software	51
CONCLUSIONES	52
RECOMENDACIONES.....	54
Referencias.....	55
ANEXOS.....	65

Índice de tablas

Tabla 1 Comparativa entre los principales lenguajes de programación para desarrollo de Bots	13
Tabla 2 Comparativa entre los principales lenguajes de programación compilados.....	15
Tabla 3 Comparación entre bases de datos	46

Índice de ilustraciones

Figura 1 Adopción de nubes públicas por parte de empresas	19
Figura 2 ¿Qué datos le interesa recabar por parte de los asistentes a los encuestados?.....	32
Figura 3 ¿Con que frecuencia se realizan eventos en la Facultad de ingeniería de la Universidad Católica Santiago de Guayaquil?	33
Figura 4 ¿Qué medios utiliza la Facultad de ingeniería de la Universidad Católica Santiago de Guayaquil para realizar las convocatorias a sus eventos?	33
Figura 5 Porcentaje de encuestados a los que le gustaría contar con información electrónica almacenada en una base de datos de todos los asistentes o personas que han mostrado interés sobre sus actividades, eventos y servicios.....	34
Figura 6 Porcentaje de encuestados a los que le gustaría contar con una herramienta informática que gestione los aforos de las distintas actividades, eventos y servicios que oferta por medio de reservas y verificando las mismas por código de barras, códigos QR	35
Figura 7 Porcentaje de encuestados a los que le gustaría contar con una herramienta informática que gestione, verifique y analice la información personal de los asistentes confirmando que efectivamente se trata de quien dice ser	35
Figura 8 Diagrama de arquitectura de software	51

RESUMEN

El presente trabajo de investigación tiene como objetivo principal resolver la necesidad de automatización y modernización de los procesos de obtención de datos, gestión reservas y control de aforos de la facultad de Ingeniería de la Universidad Católica Santiago de Guayaquil mediante el desarrollo de un sistema tecnológico completo, el cual estará conformado por un chatbot desarrollado alrededor del API oficial de Telegram para atender las distintas peticiones y cancelaciones de reservas por parte de los usuarios mediante el uso de códigos QR, además de usarse como herramienta de obtención de datos personales del mismo, adicionalmente estará conformado de un software desarrollado en el lenguaje de programación Python que se ejecutara en una raspberry pi conectada a una pi camera, el cual tiene el objetivo de leer los códigos QR generados por el Bot con la finalidad de reconocer las reservas activas y obtener rápidamente información sobre la validez de las mismas, también contará con un panel de administración donde se podrá revisar de manera manual el estado de las reservas, gestionar los usuarios y obtener reportes de información, entre otras funcionalidades. La implementación de estas tecnologías otorgará a la facultad de Ingeniería de la Universidad Católica Santiago de Guayaquil una mejor administración de los eventos y servicios ofertados además de permitir la creación de un banco de datos que posteriormente podrá ser usado para la creación de nuevas campañas u ofertas que este ente desee ofrecer.

Palabras Clave: *Chatbot, Telegram, Nube, Automatización, Python, C#,*

ABSTRACT

The main objective of this research work is to solve the need for automation and modernization of data collection processes, reservation management and capacity control of the Facultad de Ingeniería de la Universidad Católica Santiago de Guayaquil through the development of a complete technological system. Which will be formed of a chatbot developed around the official Telegram API to attend to the different requests and cancellations of reservations by users through the use of QR codes, in addition to being used as a tool to obtain personal data from it., additionally, it will contain a software developed in the Python programming language that will run on a raspberry pi connected to a pi camera, which has the objective of reading the QR codes generated by the Bot in order to recognize active reservations and quickly obtain information about their validity, it will also have an administration panel where the administrator can manually review the status of reservations, manage users and obtain information reports, among other features. The implementation of these technologies will give the Facultad de Ingeniería de la Universidad Católica Santiago de Guayaquil a better administration of the events and services offered, in addition to allowing the creation of a database that can later be used for the creation of new campaigns or offers that this entity wishes to offer.

Key words: *Chatbot, Telegram, Cloud, Automation, Python, C#*

INTRODUCCIÓN

Hoy, la tecnología se ha convertido en una parte fundamental tanto en negocios, ocio u actividades de la vida cotidiana, algo que ha permitido que esta se siga desarrollando a pasos agigantados en búsqueda de nuevos campos u horizontes para explorar e innovar. Actualmente, se puede ver poco a poco como muchas instituciones educativas de todos los niveles independientemente de la rama, comienzan a ofertar materias de programación, diseño de bases de datos, entre otras similares.

Todos estos elementos forman y abren paso a un futuro conectado y moderno además de a las tecnologías de nueva generación, las mismas que facilitan el diario vivir y aumentan la productividad de las empresas obteniendo mayores beneficios a corto y mediano plazo, una atención al cliente muchos más efectiva y todo esto sin tener costos excesivos que repercutan en un precio mayor para el usuario final.

Existen factores adicionales que han impulsado mucho más el desarrollo de la tecnología, el principal a tomar en cuenta es la pandemia global ocasionada por el SARS-CoV-2 (Coronavirus), según (Budimirovic & Filipova, 2020) indican “Las respuestas globales contra el COVID-19 han acelerado la adopción de las tecnologías digitales por muchos años y muchos de estos cambios podría quedarse a largo plazo”.

El mundo cambió en muchos aspectos, limitando ciertas libertades personales debido a esta pandemia. Las distintas autoridades responsables en los diferentes países del mundo, con el fin del controlar y reducir el contagio del virus, tomaron la decisión de usar aforos en los lugares concurridos, limitando la cantidad de personas que pueden ocupar una locación simultáneamente.

Este límite trajo consigo un nuevo problema, la falta de control por parte las personas responsables de las distintas locaciones a las cuales las autoridades competentes marcaron como potenciales zonas de contagio. En muchas ocasiones, el flujo de gente era demasiado extenso y rápido que no

era humanamente posible el control de los asistentes sin afectar la eficiencia de atención hacia estos.

En el capítulo I se describe el problema de investigación, donde se describirá el contexto, causas y consecuencias del proyecto, la población a la que abarcará, los objetivos principales y generales, su alcance y su debida justificación que aporte la importancia al mismo.

En el capítulo II se examinará el marco teórico del proyecto, este descompone a detalle lo importante de la transformación digital por parte de las empresas en el mundo y en el Ecuador, adicional a esto se describirán múltiples herramientas de hardware y software que serán investigadas con el fin de encontrar las que más se adecuen a los requerimientos brindados por parte de la facultad de Ingeniería en la Universidad Católica Santiago de Guayaquil y sean usadas en el desarrollo e implementación del presente proyecto.

El capítulo III abarca toda la información con respecto a las técnicas de recolección de información usadas en el presente proyecto, además de su análisis con el fin de poder tener una mayor orientación mucho más real de los objetivos que espera alcanzar la facultad al final del proyecto.

En el capítulo IV se presentan las herramientas seleccionadas para el desarrollo del sistema además de uso y funcionalidad dentro del mismo, el modelo entidad-relación de la base de datos usado en este y las plataformas de administración del sistema junto a sus diferentes puertos de accesos por parte de los usuarios.

Para concluir, se añaden conclusiones que contrasten con los objetivos específicos planteados al inicio de la investigación para conocer si estos fueron cumplidos de manera exitosa y también se añaden recomendaciones orientadas hacia la facultad de ingeniería de la Universidad Católica Santiago de Guayaquil que se han observado al llegar a la terminación del presente proyecto.

CAPÍTULO I

EL PROBLEMA

1.1 Descripción del problema

La actividad general de la Universidad Católica Santiago de Guayaquil es la enseñanza cuyo objetivo es lograr profesionalizar al estudiante. Para esto, se utilizan tres funciones sustantivas que son: docencia, investigación y vinculación. Las universidades deben contar con la infraestructura necesaria para que se cumplan estas tres funciones.

La función sustantiva docencia es la que ocupa la mayor parte del tiempo de los estudiantes y donde se comparte conjuntamente entre los docentes y estudiantes dentro de una clase, donde se realizan talleres, trabajos, lecciones, entre otros que enriquezcan el conocimiento. La función sustantiva investigación, se refiere a nuevas metodologías que permiten que el estudiante haciendo uso de internet, bibliotecas y trabajos en campo, descubra por sí solo, un tema de interés para la investigación. La función sustantiva vinculación es el apoyo que la universidad brinda para que el estudiante sea un excelente profesional antes de ingresar al mundo laboral, a través de prácticas de vinculación laborales y sociales.

Las vinculaciones en la academia son aspectos importantísimos para la profesionalización del estudiante, que se llevan a cabo a través de charlas informativas, charlas de conocimientos adicionales, charlas de manejos de ciertos componentes especiales, foros de discusión, talleres prácticos, ferias de trabajo, visitas a sitios de interés, entre otros.

Además de las actividades específicas que apoyan a los conocimientos de los estudiantes, también la universidad busca el bienestar a través de otras actividades vinculativas como son eventos relacionados a música, cine, deporte, entre otros. A este tipo de eventos asisten alumnos de toda las carreras y facultades. Inclusive en estos casos se incluyen a exalumnos como parte del seguimiento post profesional.

Para que estos eventos se realicen de manera correcta, se debe tener un control de asistencia y registro de datos. Muchas veces ocurren errores por no haber realizado una correcta gestión con la inscripción de los asistentes. Esto se deriva en problemas de espacio físico, falta de material, falta de equipos tecnológicos, entre otros. O también ocurre lo contrario, que sobra espacio y materiales, lo cual se puede convertir en desperdicio.

La Facultad de Ingeniería de la Universidad Católica Santiago de Guayaquil, tanto para las carreras de Ingeniería Civil, Ingeniería en Ciencias de la Computación e Ingeniería en Sistemas Computacionales; dentro de sus labores educativas, realiza eventos como charlas, foros y ferias para sus estudiantes. Dichos eventos se encuentran limitados por el espacio físico, los expositores, los equipos tecnológicos, entre otros factores.

En la actualidad, la pandemia originada en el 2020 trastocó los procesos académicos en el mundo y sin ser la excepción, también se vieron afectados las clases regulares de la Universidad Católica Santiago de Guayaquil y, por ende, todos los eventos académicos que se realizaban. En el proceso de vuelta a la normalidad, es necesario un control de aforo.

El control de aforos es esencial en la actualidad para evitar aglomeraciones de personas en zonas de alta concurrencia debido a la pandemia de Covid-19, en caso de no realizar un control adecuado, esto podría desembocar en un mayor riesgo de contagio y aumento de casos de infección que pondría en riesgo a la población adyacente, sin contar que podría ocasionar sanciones económicas o cierres temporales por parte de las autoridades competentes a los distintos establecimientos donde esto llegase a ocurrir.

1.2 Formulación del problema

En la actualidad el control de aforos se realiza de manera manual por medio de una persona que se encuentra en la entrada del establecimiento en cuestión, que en muchas ocasiones debido a la velocidad y a la cantidad de gente que está constantemente entrando y saliendo del lugar, se convierte en una tarea muy complicada; y sobre todo hoy con la presencia del virus Covid-

19, se ha evidenciado la necesidad del manejo del distanciamiento y restricciones de aforos. En este sentido, es necesario plantear una solución para el proceso de control de aforos, mediante la implementación de una propuesta tecnológica que permita gestionar eficazmente el flujo de personas, por medio de reservas cronometradas y previamente asignadas según las necesidades del usuario y disponibilidad del establecimiento.

1.3 Hipótesis de investigación

El diseño e implementación de un sistema informático usando Bots de Telegram y códigos QR permitirá el control de reservas, gestión de aforos y accesos a eventos en la facultad de ingeniería de la Universidad Católica Santiago de Guayaquil.

1.4 Objetivos de la investigación

1.4.1 Objetivo general:

Diseñar e implementar un sistema de acceso vía móvil, web y escritorio para el control de reservas, gestión de aforos y accesos a eventos en la facultad de Ingeniería en la Universidad Católica Santiago de Guayaquil, usando Bots de Telegram apoyada con la tecnología de códigos QR.

1.4.2 Objetivos específicos de la investigación:

- Identificar y conocer los requerimientos y funcionalidades que se implementarán en el sistema de acceso vía móvil, web y escritorio usando Bots de Telegram y logrando la ergonomía de la solución.
- Desarrollar el sistema de acceso vía móvil, web y escritorio usando Bots de Telegram con tecnología de códigos QR, para la gestión de aforos y accesos en la facultad de Ingeniería en la Universidad Católica Santiago de Guayaquil.

- Evaluar el sistema de acceso vía móvil, web y escritorio usando Bots de Telegram con tecnología de códigos QR, para la gestión de aforos y accesos en la facultad de Ingeniería en la Universidad Católica Santiago de Guayaquil.

1.5 Alcance

Se ha evidenciado que la facultad de Ingeniería de la Universidad Católica Santiago de Guayaquil no cuenta con un proceso automático de control de aforos y accesos para los eventos que allí se realizan, por tal motivo es importante el desarrollo de una herramienta informática (chatbot) que interactúe virtualmente con las personas para realizar reservas desde el ambiente web, móvil y escritorio. Las características de resultados y funcionamientos serían las siguientes:

- Presentación de una lista de las actividades o zonas disponibles para reserva.
- Conexión por medio de SQL a una base de datos en la nube para llevar el registro de reservas desde cualquier lugar.
- Emisión de un código QR por parte del chatbot para permitir identificar rápidamente la reserva del usuario.

1.6 Justificación e importancia del problema

Como se indicó anteriormente, la automatización del proceso de control de aforos y accesos puede ser utilizada para evitar cuantiosas multas que podrían aplicarse al establecimiento en cuestión, así como también indicar la demanda de este, ya que al analizar las reservas por día y horas se podría tener una visión real de cuan demandado es un comercio, almacén, tienda, etc., detectar horas pico y, además, se disminuiría el riesgo de contagio de Covid-19 entre los visitantes.

Actualmente la gestión de aforos y accesos de las personas a los establecimientos es realizada manualmente, el principal problema de hacerlo de esta manera es la lentitud y el gran índice de error humano que existe de por medio, además que contratar un empleado que se encargue de realizar esto resulta costoso en tiempo y dinero, tanto para los clientes que no pueden

realizar sus actividades como para la empresa. El automatizar esta actividad permitiría acelerar los procesos de control sin afectar la calidad de estos, además de ahorrarle dinero a la empresa a corto plazo.

Algo adicional a ser destacado, es la importancia de la tecnología en las empresas hoy, usando estos recursos se puede proporcionar servicios de calidad y ágiles requeridos por el mercado actual que está cambiando constantemente y debe adaptarse a las necesidades del usuario. El uso de estos recursos son el primer paso hacia la automatización de procesos, lo cual abre la puerta hacia infinitas posibilidades en cualquier empresa u organización.

CAPÍTULO II

MARCO TEÓRICO

Para los procesos de implementación de un sistema informático, es necesario identificar los diferentes aportes teóricos y conceptos vinculados, para el presente proyecto es importante detallar estos aportes desde un ámbito general hasta la puntualización con las herramientas utilizadas.

2.1 Los chatbots y su aporte en los procesos de control de reservas y aforos

Se conoce como chatbot a un programa de computadora que simula y procesa las conversaciones humanas (sean estas escritas o habladas), permitiendo a los humanos interactuar con servicios digitales tal como si se estuvieran comunicando con una persona (Oracle Corporation, 2020).

Los lenguajes de programación en los cuales puede ser desarrollado un chatbot en la actualidad son múltiples, ya que estos manejan algoritmos que analizan las entradas del usuario y responden según un algoritmo preprogramado, sin embargo, según (Ghanchi, 2019) “los principales lenguajes de programación a tomar en cuenta a la hora de desarrollar un chatbot debido a su versatilidad, velocidad de desarrollo y compatibilidad son: Java, Clojure, Python, C++, PHP, Ruby y Lisp”.

Los usuarios valoran mucho los chatbots porque son rápidos, intuitivos y convenientes. Para las empresas, los chatbots ofrecen una manera de construir una experiencia del cliente más atractiva y personalizada, que a cambio ofrece una gran cantidad de información valiosa que permite comprender mejor al cliente y hacer crecer el negocio (Three Piece Marketing, 2019).

Según (Artificial Solutions, 2020) algunas de las ventajas claves que aportan los chatbots a las empresas son:

- Respuestas inmediatas

Los chatbots empresariales permiten los negocios satisfacer la demanda otorgando respuestas inmediatas a inquietudes o problemas.

- Reducción de costos

Los chatbots ayudan a reducir los costos permitiendo a las empresas atender a más clientes sin tener la necesidad de contratar más personal. Los asistentes virtuales pueden ayudar a reducir las consultas entrantes hasta un 40% con respecto a los agentes en vivo.

- Incremento de la lealtad de los clientes

Las empresas que usan chatbots en sus operaciones y brindan una experiencia rápida a sus clientes presentan hasta un 2% de aumento en la retención de clientes.

- Disponibilidad 24/7

Los clientes desean servicio al momento, 24/7,365. Ellos desean comunicarse con la empresa y realizar alguna consulta mientras esperan por un café o mientras conducen hacia el trabajo, y desean hacerlo usando los mismos dispositivos que usan día a día.

- Aumento del compromiso de los clientes

Los clientes comprometidos con la marca compran 90% más frecuentemente y en promedio gastan 60% más por compra.

2.1.1 Transformación digital de las sociedades

La transformación digital está revolucionando a la sociedad. Ya sea en términos de modelos económicos, trabajos, cultura corporativa, o la misma estructura laboral. En la actualidad, no hay otra opción más que adaptarse a esta transformación para poder mantenerse competitivos, ya que esta transformación es sinónimo de agilidad y sobre todo resiliencia ante las crisis, tales como la que ha enfrentado el mundo en los últimos años (Dang Tran, 2021).

Parte de esta transformación digital es entender el rol crucial y el enorme impacto de las nuevas tecnologías en las sociedades. Según (Regenesys Business School, 2020) “Las personas de hoy son eruditas en la tecnología digital y están conectadas 24 horas del día, los 7 días de la semana a las redes sociales por medio de sus dispositivos móviles, por tanto, esperan que sus proveedores estén disponibles las 24 horas del día. Esto es

importante, porque una persona siempre conectada puede comparar precios y pasar de una marca a otra con un esfuerzo mínimo”.

Para las organizaciones es importante no rehusarse a la transformación digital, ellos encontrarán que su negocio o institución es completamente obsoleta y no relevante para la sociedad digital del futuro (Gartner, 2020).

Los procesos vinculados a las actividades académicas tampoco están apartados de las transformaciones y avances digitales, consecuentemente los mayores crecimientos se han evidenciado en los últimos 2 años, con el surgimiento de la pandemia provocada por el Covid-19.

2.1.1.1 Transformación digital en el Ecuador

En el Ecuador se han llevado adelante varias medidas en el marco de la agenda digital y las iniciativas del MINTEL (Ministerio de Telecomunicaciones). En su reporte 2021-2022, de reciente publicación, consideran a la agenda digital como “el instrumento para la recuperación y reactivación integral, a través de la transformación digital, productiva, social y de política pública en el país (Diario Los Andes, 2021).

Los grandes ejes de la Agenda Digital del Ecuador están definidos a través de la mejora de la conectividad, la eficiencia de la administración pública con el fomento del gobierno digital, el desarrollo de una cultura digital que impulse la economía basada en tecnologías digitales e innovación para el uso de tecnologías emergentes (Ministerio de Telecomunicaciones y de la Sociedad de la información del Ecuador, 2021)

Según (Deloitte, 2021) “Las nuevas capacidades digitales (plataformas en la nube, automatización de procesos, inteligencia artificial) brindan grandes posibilidades. La automatización de procesos robóticos acelera los esfuerzos para hacer frente a los crecientes volúmenes de transacciones. Los chatbots ayudan con el aumento de las demandas de preguntas de clientes. Ahora, más que nunca, las organizaciones tienen que llevar la tecnología a la sala ejecutiva a la hora de desarrollar sus estrategias sobre “donde competir” y “como competir””.

2.2 Conceptos Varios

2.2.1 Lenguajes de programación

Los lenguajes de programación permiten dar instrucciones a una computadora en un lenguaje que la computadora entiende. Así como existen muchos lenguajes basados en humanos, existe una variedad de lenguajes de programación de computadoras que los programadores pueden usar para comunicarse con esta. La parte del lenguaje que una computadora puede entender se llama “binario”. La traducción del lenguaje de programación al lenguaje binario es lo que se conoce como “compilación” (ComputerScience.org Staff, 2021).

2.2.2 Python

Python es un lenguaje de programación interpretado, orientado a objetos y de alto nivel con semántica dinámica. Sus estructuras de datos integradas de alto nivel, combinadas con tipado y enlace dinámicos, lo hacen muy atractivo para el desarrollo rápido de aplicaciones, así como para su uso como lenguaje de scripts o pegamento para conectar componentes existentes. La sintaxis simple y fácil de aprender de Python enfatiza la legibilidad y, por lo tanto, reduce el costo de mantenimiento del programa. Python admite módulos y paquetes, lo que fomenta la modularidad del programa y la reutilización del código. El intérprete de Python y la extensa biblioteca estándar están disponibles en formato fuente o binario sin cargo para todas las plataformas principales y se pueden distribuir libremente (Python Software Foundation, 2021).

2.2.3 JavaScript

JavaScript es un lenguaje de programación basado en texto que se usa tanto en el lado del cliente como en el lado del servidor y que le permite hacer que las páginas web sean interactivas. Mientras que HTML y CSS son lenguajes que dan estructura y estilo a las páginas web, JavaScript proporciona a las páginas web elementos interactivos que atraen al usuario. Los ejemplos comunes de JavaScript que puede usar todos los días incluyen el cuadro de búsqueda en Amazon, un video de resumen de noticias

incrustado en The New York Times o actualizar los posts de su cuenta de Twitter (Hack Reactor, 2021).

2.2.4 Ruby

Ruby es un lenguaje de secuencias de comandos orientado a objetos de código abierto inventado a mediados de los 90 por Yukihiro Matsumoto. A diferencia de lenguajes como C y C++, un lenguaje de secuencias de comandos no se comunica directamente con el hardware. Se escribe en un archivo de texto y luego un intérprete lo analiza y lo convierte en código. Estos programas son generalmente de naturaleza procesal, lo que significa que se leen de arriba abajo (ACG Technical Editors Team, 2021).

Tabla 1 Comparativa entre los principales lenguajes de programación para desarrollo de Bots

Criterios	Python	JavaScript	Ruby
Uso sencillo de APIs	10	8	7
Desarrollo constante por parte de la comunidad	9	9	7
Documentación	9	8	5
Uso de librerías	10	9	8
Total	38	34	27

Tal como se evidencia en la tabla 1 podemos notar como Python y JavaScript son candidatos importantes a la hora de tomar la decisión de desarrollar un Bot de cualquier índole. Ambos lenguajes son muy buenos a la hora de interactuar con APIs desarrolladas por terceros, un beneficio que facilita en gran cantidad el trabajo del desarrollador y acorta considerablemente el periodo de tiempo hacia el entregable final. Sin embargo, Python cuenta con un wrapper oficial del API de Telegram, que está

en constante desarrollo y que cuenta con todas las herramientas necesarias para una comunicación efectiva con la aplicación, en su defecto, JavaScript cuenta con un wrapper no oficial desarrollado por la comunidad y que depende de su framework Node JS. Con respecto a la documentación del lenguaje, Python al ser “el lenguaje de programación de la comunidad” es muy amplia y perfectamente estructurada, además de contar con controles de calidad que eviten el desarrollo de librerías o módulos basura o de poca relevancia, elementos importantes para cualquier trabajo investigativo, tal como es el caso.

2.2.5 C#

C# (pronunciado "Si Sharp") es un lenguaje de programación moderno, orientado a objetos y tipado. C # permite a los desarrolladores crear muchos tipos de aplicaciones seguras y sólidas que se ejecutan en .NET. C# tiene sus raíces en la familia de lenguajes C y será inmediatamente familiar para los programadores de C, C++, Java y JavaScript. C# es un lenguaje de programación orientado a objetos y orientado a componentes. C# proporciona constructores de lenguaje para admitir directamente estos conceptos, lo que convierte a C# en un lenguaje natural en el que crear y utilizar componentes de software. Desde su origen, C# ha agregado funciones para admitir nuevas cargas de trabajo y prácticas emergentes de diseño de software. En esencia, C# es un lenguaje orientado a objetos. El usuario define tipos y su comportamiento (Microsoft, 2021).

2.2.6 Java

Java es un lenguaje de programación orientado a objetos y una plataforma de software ampliamente utilizada, que se ejecuta en miles de millones de dispositivos, que incluyen computadoras portátiles, dispositivos móviles, consolas de juegos, dispositivos médicos y muchos otros. Las reglas y la sintaxis de Java se basan en los lenguajes C y C++. Una de las principales ventajas de desarrollar software con Java es su portabilidad. Una vez que haya escrito el código para un programa Java en una computadora portátil, es muy fácil mover el código a un dispositivo móvil. Cuando el lenguaje fue inventado en 1991 por James Gosling de Sun Microsystems (luego adquirido

por Oracle), el objetivo principal era poder "escribir una vez, ejecutar en cualquier lugar" (IBM Cloud Education, 2019).

2.2.7 GO

Go, también conocido como Golang, es un lenguaje de programación de código abierto, compilado y tipado estáticamente diseñado por Google. Está diseñado para ser simple, de alto rendimiento, legible y eficiente (Kolade, 2021).

Tabla 2 Comparativa entre los principales lenguajes de programación compilados

Criterios	C#	Java	GO
Diseño de interfaces gráficas	9	7	5
Uso de controles nativos en el sistema operativo Windows	10	7	6
Velocidad de ejecución	9	8	8
Portabilidad y generación de instaladores	10	9	6
Total	38	31	26

Tal como describe la Tabla 2, vemos algunas diferencias y principales ventajas entre 3 lenguajes compilados. Dado que la solución tecnológica será orientada a usuarios no técnicos se espera que esta sea de fácil uso, rápido y con interfaces atractivas y fáciles de comprender. Esta última afirmación cabe

con exactitud para el lenguaje de programación C#, esto dado que al usar controles nativos del sistema estos serán adaptados a la interfaz del sistema donde se ejecute, asegurando así una estética impecable y atractiva a la vista. Con respecto a la velocidad de ejecución, este apartado se lo lleva C# dado que se ejecuta nativamente como un archivo .exe totalmente compilado, característica que Java y Go cumplen a medias, dado que requieren una herramienta adicional como es la JVM y el Go Environment para su ejecución. En el apartado de portabilidad Go se queda muy atrás dado que requiere la instalación de su paquete de desarrollo para poder ser ejecutado, Java normalmente se encuentra instalado en los ordenadores, por lo cual no es un limitante directo, sin embargo, C# empaqueta todo de manera automática en un instalador multilenguaje que puede ser fácilmente transportado de un equipo a otro además de contar con firmas de seguridad y herramientas adicionales sin requerir librerías de terceros.

2.2.8 API

Una API es un conjunto de definiciones y protocolos que se utiliza para desarrollar e integrar el software de las aplicaciones. API significa interfaz de programación de aplicaciones. Las API permiten que sus productos y servicios se comuniquen con otros, sin necesidad de saber cómo están implementados. Esto simplifica el desarrollo de las aplicaciones y permite ahorrar tiempo y dinero. Las API le otorgan flexibilidad; simplifican el diseño, la administración y el uso de las aplicaciones, y proporcionan oportunidades de innovación, lo cual es ideal al momento de diseñar herramientas y productos nuevos (o de gestionar los actuales) (Red Hat, 2021).

2.2.9 Telegram

Telegram es una popular aplicación de mensajería multiplataforma que es usada ampliamente porque ofrece características mejoradas de privacidad y encriptación, así también como soporte para funciones de chat de grupos grandes. Además, no tiene vínculos con otras plataformas de redes sociales (tanto Facebook Messenger como WhatsApp son propiedad de Facebook, por ejemplo), lo que hace que el servicio sea más atractivo para algunos usuarios a los que la privacidad les es atractiva. La aplicación cuenta con versiones

disponibles para iOS, Android, Windows, Mac y Linux. Adicional a estas también se puede acceder a Telegram desde un navegador web (Johnson, 2021).

2.2.10 Chatbot

Un chatbot (derivado de "chat robot") es un programa informático que simula una conversación humana, ya sea a través de la comunicación por medio de voz o de texto. Estos programas se pueden personalizar y utilizar de distintas maneras según se desee implementar. La mayoría de nosotros estamos familiarizados con los Bots para el servicio al cliente en nuestra vida diaria como consumidores, y también con plataformas populares de chat y mensajería como SMS, Facebook Messenger, WhatsApp y WeChat, en los cuales estos chatbots son integrados (Higgins, 2021).

2.2.11 Computación en la nube

La computación en la nube es el uso de recursos informáticos a través de Internet bajo la modalidad de pago por uso. En lugar de comprar, poseer y mantener centros de datos y servidores físicos, se puede acceder a servicios tecnológicos, como la potencia de cálculo, el almacenamiento y las bases de datos, en función de las necesidades del usuario por medio de proveedores como Amazon Web Services, Azure o Google cloud Platform (Amazon Web Services (AWS), 2022).

2.2.12 Heroku

Heroku es conocido por ejecutar aplicaciones en dynos, que en realidad son solo computadoras virtuales ubicadas en la nube que se pueden encender o apagar según el tamaño de su aplicación. Piense en los dynos como bloques de construcción maleables para ejecutar su aplicación. Si desea procesar más datos o ejecutar tareas más complejas, necesitará agregar más bloques (lo que se denomina escalado horizontal) o aumentar el tamaño de los bloques (lo que se denomina escalado vertical). Luego, Heroku le cobra una tarifa mensual basada en la cantidad de dynos que tiene y el tamaño de cada uno de ellos. Heroku provee su infraestructura usando AWS (Rostad, 2018).

Según (Vennam, 2020) en comparación con la TI tradicional en las instalaciones, y dependiendo de los servicios en la nube que seleccione, la computación en la nube ayuda a hacer lo siguiente

- **Reducir los costes de TI:** La nube le permite disminuir parte o la mayor parte de los costes y el esfuerzo de comprar, instalar, configurar y gestionar su propia infraestructura local.
- **Mejorar la agilidad y el tiempo de obtención de valor:** Con la nube, su organización puede empezar a utilizar las aplicaciones empresariales en cuestión de minutos, en lugar de esperar semanas o meses a que el departamento de TI responda a una solicitud, compre y configure el hardware de apoyo e instale el software. La nube también le permite facultar a determinados usuarios, en concreto a los desarrolladores y científicos de datos, para que se ayuden a sí mismos con el software y la infraestructura de soporte.
- **Es más fácil de escalar y rentable:** La nube proporciona elasticidad: en lugar de comprar un exceso de capacidad que no se utiliza durante los periodos de baja actividad, se puede aumentar y reducir la capacidad en respuesta a los picos y las caídas del tráfico. También puede aprovechar la red global de su proveedor de nube para acercar sus aplicaciones a los usuarios de todo el mundo.

2.2.13 AWS (Amazon Web Services)

AWS (Amazon Web Services) es una plataforma de computación en la nube integral y en constante evolución proporcionada por Amazon que incluye una combinación de ofertas de infraestructura como servicio (IaaS), plataforma como servicio (PaaS) y software como servicio (SaaS). Los servicios de AWS pueden ofrecer a una organización herramientas como potencia informática, almacenamiento de bases de datos y servicios de entrega de contenido (Gillis, 2020).

2.2.14 Microsoft Azure

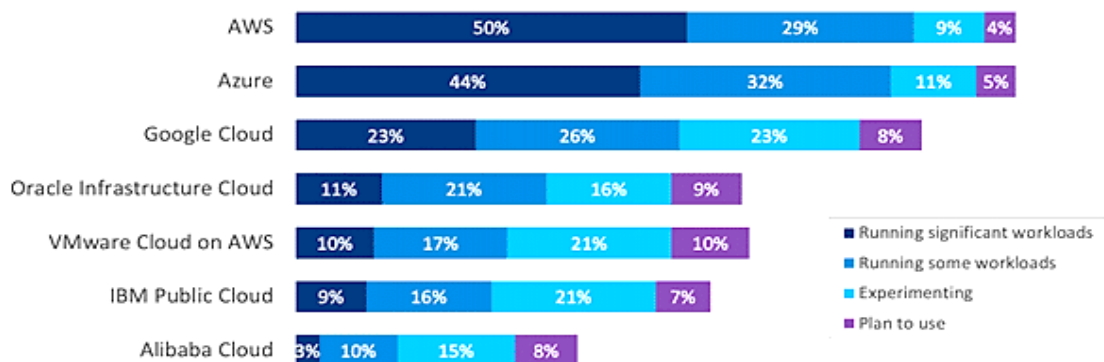
Microsoft Azure, anteriormente conocido como Windows Azure, es la plataforma informática en la nube pública de Microsoft. Proporciona una variedad de servicios en la nube, que incluyen cómputo, análisis,

almacenamiento y redes. Los usuarios pueden elegir entre estos servicios para desarrollar y escalar nuevas aplicaciones o ejecutar aplicaciones existentes en la nube pública (Bigelow, 2020).

2.2.15 Google Cloud Platform (GCP)

GCP es un proveedor de servicios en la nube pública, como sus competidores Amazon Web Services (AWS) y Microsoft Azure. Con GCP y otros proveedores de la nube, los clientes pueden acceder a los recursos informáticos alojados en los centros de datos de Google en todo el mundo de forma gratuita o mediante pago por uso. GCP ofrece un conjunto de servicios informáticos para hacer de todo, desde la administración de costos de GCP hasta la administración de datos, la entrega de web y video a través de la web, la IA y las herramientas de aprendizaje automático (Knox, 2021).

Figura 1 Adopción de nubes públicas por parte de empresas (Luxner, 2021)



En la Figura 1 se observa como Luxner menciona algunos de los proveedores de servicios en la nube más populares, además de describirlos en 4 categorías que son: Porcentaje de ejecución de cargas de trabajo significativas, porcentaje de ejecución de carga de trabajo normales, porcentaje de empresas experimentando, porcentaje de empresas que planean empezar a usarlas. Se puede notar como AWS, Azure y GCP son las principales y más usadas a nivel mundial.

2.2.16 Infraestructura como servicio (IaaS)

La infraestructura como servicio (IaaS) es un servicio de computación en la nube donde las empresas alquilan servidores para computación y

almacenamiento en la nube. Los usuarios pueden ejecutar cualquier sistema operativo o aplicaciones en los servidores alquilados sin el mantenimiento y los costos operativos de esos servidores. Otras ventajas de la infraestructura como servicio incluyen brindar a los clientes acceso a servidores en ubicaciones geográficas cercanas a sus usuarios finales. IaaS se escala automáticamente, tanto hacia arriba como hacia abajo, según la demanda y proporciona un acuerdo de nivel de servicio (SLA) garantizado tanto en términos de tiempo de actividad como de rendimiento. Una de sus características más importantes es que elimina la necesidad de aprovisionar y administrar manualmente servidores físicos en centros de datos (AVI Networks, 2021).

2.2.17 Plataforma como servicio (PaaS)

PaaS, o Platform-as-a-Service, es un modelo de computación en la nube que brinda a los clientes una plataforma en la nube completa (hardware, software e infraestructura) para desarrollar, ejecutar y administrar aplicaciones sin el costo, la complejidad y la inflexibilidad que a menudo se presenta con la construcción y el mantenimiento de esa plataforma en las instalaciones. El proveedor de PaaS aloja todo (servidores, redes, almacenamiento, software del sistema operativo, bases de datos, herramientas de desarrollo) en su centro de datos. Por lo general, los clientes pueden pagar una tarifa fija para proporcionar una cantidad específica de recursos para un número específico de usuarios, o pueden elegir el precio de "pago por uso" para pagar solo por los recursos que utilizan (IBM Cloud Education, 2021).

2.2.18 Software como servicio (SaaS)

El software como servicio (o SaaS) es una forma de entregar aplicaciones a través de Internet, como servicio. En lugar de instalar y mantener el software, simplemente acceda a él a través de Internet, liberándose de la compleja gestión de software y hardware. Las aplicaciones SaaS a veces se denominan software basado en web, software bajo demanda o software alojado. Cualquiera sea el nombre, las aplicaciones SaaS se ejecutan en los servidores de un proveedor de SaaS. El proveedor administra

el acceso a la aplicación, incluida la seguridad, la disponibilidad y el rendimiento (Salesforce, 2021).

2.2.19 Raspberry Pi

Raspberry Pi es el nombre de una serie de computadoras de placa única fabricadas por la Fundación Raspberry Pi, una organización benéfica del Reino Unido que tiene como objetivo educar a las personas en informática y crear un acceso más fácil a la educación informática. La Raspberry Pi es una computadora muy barata y de tamaño reducido que ejecuta Linux, pero también proporciona un conjunto de pines GPIO (entrada / salida de propósito general), lo que le permite controlar componentes electrónicos para su uso en computación física y también para explorar el Internet de las cosas (IoT) (Opensource community by RedHat, 2021).

2.2.20 Pi Camera

El módulo Pi Camera es una cámara portátil y liviana que admite la Raspberry Pi. Se comunica con la raspberry Pi mediante el protocolo de interfaz en serie de la cámara MIPI. Normalmente se utiliza en procesamiento de imágenes, aprendizaje automático o en proyectos de vigilancia. Además de estos módulos, la raspberry Pi también puede usar cámaras web USB normales que se usan normalmente junto con la computadora (Components101, 2018).

2.2.21 Raspbian

Raspbian es un sistema operativo gratuito lanzado en julio de 2012 que se ejecuta en la computadora de placa única Raspberry Pi. Se deriva de Debian Linux y utiliza el entorno de escritorio LXDE de forma predeterminada (Computer Hope, 2020).

2.2.22 DietPi

DietPi es una distribución de Linux basada en Debian, desarrollada principalmente para computadoras de placa única como Raspberry Pi, Orange Pi u Odroid caracterizada por su bajo consumo de recursos. DietPi también suministra compilaciones para computadoras personales x86 de 64 bits y

máquinas virtuales. DietPi se envía con una serie de herramientas de configuración basadas en menús que se pueden ejecutar desde un terminal (Atea Atarua Limited, 2020).

2.2.23 Módulos de Python

En Python, los módulos son archivos con la extensión “.py” que contienen código Python creado por nosotros o terceros que pueden ser importados dentro de otro programa del mismo lenguaje. En términos simples, podemos considerar que un módulo es lo mismo que una biblioteca de código o un archivo que contienen un conjunto de funciones que el programador desea incluir en la aplicación que se encuentra desarrollando (Royal, 2021).

2.2.24 PIP

PIP es un gestor de paquetes y módulos para Python. Eso significa que es una herramienta que le permite al usuario instalar y administrar librerías y dependencias adicionales que no son distribuidas como parte de la librería estándar de Python (Rodríguez, 2021).

2.2.25 Paquetes de C#

Son un mecanismo por el cual los desarrolladores pueden crear, compartir y consumir código útil. La forma de compartir este código con otros desarrolladores interesados es por medio de “paquetes”, estos contienen código compilado (como DLLs) junto con otros contenidos adicionales necesarios en los proyectos que consuman estos paquetes (Microsoft, 2021).

2.2.26 NuGet

NuGet es el gestor de paquetes oficiales para .NET, básicamente un paquete NuGet contiene un archivo zip único con la extensión .nupkg que consiste en código compilado (DLLs), e información como el número de versión del paquete que se va a proceder a instalar. NuGet gestiona todo el proceso de manera automática y rápida para el usuario, solo con realizar la búsqueda del paquete deseado en el buscador que este complemento incluye (GeeksforGeeks, 2020).

2.2.27 Telegram Bot API

El Bot API de Telegram es una interfaz basada en HTTP creada para desarrolladores interesados en crear Bots para Telegram de manera rápida y sencilla usando código y métodos que faciliten el desarrollo del mismo y su puesta en marcha por medio de la app de Telegram (Telegram Core Team, 2021).

2.2.28 pyTelegramBotAPI

Es una implementación de código abierto en el lenguaje de programación Python de la Telegram Bot API, está en constante desarrollo y a día de hoy soporta la última versión del API oficial. Es desarrollada y mantenida por el usuario “eternnoir” en su repositorio oficial de GitHub (Wang, 2021).

2.2.29 Abstract API

Es una empresa que se encarga de proveer poderosas APIs de todo tipo que permiten enriquecer la experiencia del usuario o automatizar el flujo de trabajo. Es usada por más de 10000 desarrolladores a nivel mundial. Cuenta con planes gratuitos renovables mensualmente y con planes de pago para desarrollos más ambiciosos y que requieran mayores recursos (Abstract, 2021).

2.2.30 OpenCV

OpenCV es una enorme biblioteca de código abierto que se usa para el desarrollo de visión por computadora, aprendizaje automático y procesamiento de imágenes. Open CV permite su uso en una gran variedad de lenguajes de programación, tales como Python, C++, Java, etc. Puede procesar imágenes y videos para identificar objetos, rostros e incluso escritura a mano de un humano (GeekforGeeks, 2021).

2.2.31 Pyzbar

Pyzbar es un módulo del lenguaje de programación Python, es responsable de leer y decodificar códigos de barra de una dimensión o

códigos QR de manera sencilla, además, requiere el módulo PIL (Pillow) para funcionar de manera adecuada (AskPython, 2021).

2.2.32 PIL (Pillow)

La biblioteca de imágenes de Python (PIL por sus siglas en inglés) agrega capacidades de procesamiento de imágenes al intérprete de Python por medio de su módulo PIL. Esta biblioteca proporciona un amplio soporte de formatos de archivos, una representación interna eficiente y capacidades de procesamiento de imágenes bastante rápidas y poderosas (Clark, 2021).

2.2.33 PyMsgBox

Es un módulo sencillo, multiplataforma y de Python puro que implementa cajas de texto. Este módulo usa el módulo TKInter ya incluido en la librería estándar de Python. Funciona en Python 2 y 3. Es un fork derivado del módulo EasyGui de Stephen Raymond (Sweigart, 2021).

2.2.34 Requests

Requests es un módulo de Python que se puede usar para enviar todo tipo de solicitudes HTTP. Es una biblioteca fácil de usar con muchas características que van desde pasar parámetros en URL hasta enviar encabezados personalizados y verificación SSL. Permite recibir las respuestas de estas peticiones y convertirlas a variables del tipo JSON además de evitar la creación de amplias cadenas usando Urllib3 (Rao, 2021).

2.2.35 Códigos QR

El código de respuesta rápida (QR por sus siglas en inglés), es una versión bidimensional del código de barras, este es capaz de transmitir una gran cantidad de información al instante con solo ser escaneado por un dispositivo móvil o un lector habilitado para este fin. Los códigos QR pueden almacenar hasta 7089 dígitos o 4296 caracteres, incluidos signos de puntuación y caracteres especiales. Los códigos QR funcionan inclusive si han sufrido daños dada las repeticiones que incluye en su estructura, gracias a esto permite que hasta un 30% del mismo sea ilegible y este pueda seguir funcionando (QR Code Generator Marketing Team, 2020).

2.2.35.1 Códigos QR estáticos

Un código QR estático contiene información fija y esta no puede editada una vez que se ha generado el código. Son excelentes para uso personal y para generar grandes lotes de códigos para identificaciones de empleados, insignias de eventos, documentación de productos y mucho más (QR Code Generator Marketing Team, 2020).

2.2.35.2 Códigos QR dinámicos

Un código QR dinámico es aquel que contiene una url de redirección corta codificada. La información que intenta comunicar no esta codificada en el código en sí, la información que intenta comunicar esta en un sitio web y el código QR cumple la función de redireccionar a ese sitio web. Por tanto, se considera dinámica, ya que este redireccionamiento puede cambiar con el tiempo sin necesidad de un nuevo código (SproutQR, 2020).

2.2.36 MySQL Connector/Python

MySQL Connector/Python es un módulo que permite que los programas desarrollados en el lenguaje de programación Python tengan acceso a bases de datos del tipo MySQL, usando un API diseñada según la especificación de bases de datos en Python en su versión 2.0 documento identificador PEP 249 (Oracle Corporation - MySQL Team, 2021).

2.2.37 Bases de datos

Una base de datos es una colección organizada de información estructurada, o datos, que normalmente se almacenan electrónicamente en un sistema informático. Una base de datos suele estar controlada por un sistema de gestión de bases de datos (DBMS). Juntos, los datos y el DBMS, junto con las aplicaciones que están asociadas con ellos, se conocen como un sistema de base de datos, a menudo abreviado como base de datos. Los datos dentro de los tipos más comunes de bases de datos en funcionamiento en la actualidad se modelan típicamente en filas y columnas en una serie de tablas para que el procesamiento y la consulta de datos sean eficientes. La

mayoría de las bases de datos utilizan un lenguaje de consulta estructurado (SQL) para escribir y consultar datos (Oracle Corporation, 2021).

2.2.37.1 Bases de datos relacionales

Una base de datos relacional es una colección de datos con relaciones predefinidas entre ellos. Estos elementos están organizados como un conjunto de tablas con columnas y filas. Las tablas se utilizan para almacenar información sobre los objetos que se van a representar en la base de datos. Cada columna de una tabla contiene un cierto tipo de datos y un campo almacena el valor del dato. Las filas de la tabla representan una colección de valores relacionados de un objeto o entidad. Cada fila de una tabla se puede marcar con un identificador único llamado llave primaria, y las filas entre varias tablas se pueden relacionar mediante llaves foráneas. Se puede acceder a estos datos de muchas formas diferentes sin reorganizar las tablas de la base de datos usando el lenguaje SQL (Amazon Web Services, 2021).

2.2.37.2 Bases de datos no relacionales

Una base de datos no relacional es cualquier base de datos que no utiliza el esquema tabular de filas y columnas como en las bases de datos relacionales. Más bien, su modelo de almacenamiento está optimizado para el tipo de datos que almacena. Las bases de datos no relacionales también se conocen como bases de datos NoSQL, que significa "No solo SQL". Cuando las bases de datos relacionales solo usan SQL, las bases de datos no relacionales pueden usar otros tipos de lenguaje de consulta (Logi Analytics, 2021).

2.2.38 MySQL

MySQL es un sistema de gestión de bases de datos relacionales de código abierto. Al igual que con otras bases de datos relacionales, MySQL almacena datos en tablas formadas por filas y columnas. Los usuarios pueden definir, manipular, controlar y consultar datos mediante el lenguaje de consulta estructurado, más comúnmente conocido como SQL. El nombre de MySQL

es una combinación de "My", el nombre de la hija del creador de MySQL, Michael Widenius, y "SQL" (Drake, 2020).

2.2.39 ClearDB

ClearDB es una base de datos MySQL de alta disponibilidad que se ejecuta en múltiples instancias en la nube, así como en entornos híbridos de nube / centro de datos. Está diseñado para eliminar la necesidad de administrar su propia base de datos MySQL. ClearDB proporciona clústeres de bases de datos MySQL especializados y de alta disponibilidad que se basan en la premisa contra fallas para garantizar que la base de datos esté siempre en línea y completamente disponible, las 24 horas del día, los 7 días de la semana. Usa tecnología MySQL estándar, con una pila de software en capas "envuelta" para brindar soporte automático e inteligente para que pueda el desarrollador pueda concentrarse en su aplicación, y no en administrar su base de datos (Navisite, 2021)

2.3 Fundamentos legales del proyecto

El proyecto planteado está sujeto a ciertas regulaciones establecidas en Ecuador. Estas se encuentran en la Ley de Comercio Electrónico, Firmas Electrónicas y Mensajes de Datos, vigente desde el año 2002. Los artículos que en ella se mencionan incluyen:

Artículo 48 "Consentimiento para recibir mensajes de datos" - Antes de que un consumidor o usuario acepte recibir grabaciones o mensajes electrónicos, debe tener un conocimiento claro, preciso y satisfactorio del equipo y los procedimientos necesarios para acceder a dichos registros o mensajes.

Los Usuarios/Consumidores deben razonablemente demostrar que tiene acceso a la información que consiente al dar o confirmar por medio de tales métodos, electrónicamente de su consentimiento los datos que ha aceptado deben proporcionarse de manera clara, precisa y satisfactoria también modificará la información solicitada, y se le notificará su derecho a retirar su consentimiento previo otorgado sin ninguna condición, tarifa o consecuencia.

En caso de que estos cambios afecten a los intereses de los consumidores o usuarios, deberán adoptarse las medidas necesarias para evitar tales efectos hasta la resolución del contrato o acuerdo que requiera el consentimiento previo." (Lexis Ecuador, 2012).

Artículo 50 "Información puesta a disposición de los consumidores" - Los consumidores deberán estar plenamente informados de sus derechos y obligaciones al momento de prestar servicios electrónicos en el Ecuador, de conformidad con lo dispuesto en la Ley Orgánica de Protección al Consumidor y lo dispuesto en la presente Ley Uso o utilización de bienes o servicios , los contratistas deberán notificar los requisitos, condiciones y restricciones para que los consumidores puedan obtener y utilizar los bienes y servicios anunciados, promocionados y utilizar los servicios por medios electrónicos a través de las redes electrónicas de información, incluida Internet, sancionado de acuerdo al ordenamiento jurídico vigente en el Ecuador.

Las actividades de publicidad y promoción a través de las redes electrónicas de información, incluida Internet, proporcionarán a los consumidores acceso ilimitado a toda la información disponible sobre bienes o servicios, en las mismas condiciones y utilizarán la información disponible para promocionar bienes o servicios.

En el envío de mensajes de datos que contengan cualquier tipo de información de forma periódica, de forma individual o mediante listas de correo, directamente o mediante cadenas de mensajes, el mismo remitente deberá proporcionar un código hash para que el destinatario pueda confirmar la suscripción, registrarse o solicitar el registro en cualquier momento. Excluya de la lista, el hilo de mensajes o la base de datos a la que está suscrito y haga que se envíen notificaciones con datos arriba. Una solicitud de renuncia es vinculante para el emisor desde el momento en que se recibe. El envío de mensajes periódicos adicionales no solicitados de cualquier tipo estará sancionado por lo dispuesto en la cláusula anterior que establece que el Usuario de la Red Electrónica podrá aceptar recibir mensajes periódicos de datos para informar sobre toda clase de productos y servicios, pero no estará obligado a ello (Lexis Ecuador, 2012).

CAPÍTULO III

METODOLOGÍA DE LA INVESTIGACIÓN

En este capítulo se describe las diferentes metodologías que se aplicaron en el desarrollo del proyecto, identificando las razones de uso para las mismas, además de las distintas técnicas que se requirieron para la obtención de datos y su uso en la misma.

3.1 Definición de la investigación

3.1.1 Investigación cualitativa

La selección de esta metodología para la obtención de datos se debe principalmente a la facilidad que ofrece al usuario de poder valorar las distintas funcionalidades y la forma de presentación de las mismas, que se desarrollaran en el aplicativo de acuerdo con los comentarios otorgada por parte de estos.

La investigación cualitativa será llevada a cabo mediante la elaboración de un cuestionario, esto, con la finalidad de obtener la información necesaria acerca de los gustos del usuario con respecto al trato y la manera en la que se presentaran las opciones en el software a ser desarrollado.

La investigación cualitativa implica recopilar y analizar datos no numéricos (por ejemplo, texto, video o audio) para comprender conceptos, opiniones o experiencias. Se puede utilizar para recopilar información detallada sobre un problema o generar nuevas ideas para la investigación. La investigación cualitativa es lo opuesto a la investigación cuantitativa, que implica recopilar y analizar datos numéricos para el análisis estadístico. La investigación cualitativa se utiliza comúnmente en las ciencias humanas y sociales, en materias como antropología, sociología, educación, ciencias de la salud, historia, entre otras (Bhandari, 2020).

Los autores (Busseto, Wick, & Gumbinger, 2020), señalan que:

La investigación cualitativa se define como “el estudio de la naturaleza de los fenómenos”, incluyendo “su calidad, diferentes manifestaciones, el contexto en el que aparecen o las perspectivas desde las que pueden ser percibidos”, pero excluyendo “su alcance, frecuencia y lugar en una cadena de causa y efecto determinada objetivamente”. Esta definición formal se puede complementar con una regla empírica más pragmática: la investigación cualitativa generalmente incluye datos en forma de palabras en lugar de números.

3.1.2 Investigación cuantitativa

Adicional al uso de la investigación cualitativa, será sumado el uso de la investigación cuantitativa, esto con el fin de poder conocer por medio de cálculos estadísticos la intención o tendencia por parte las personas encuestadas hacia un objetivo tecnológico.

El uso de este tipo de investigación será por medio de un cuestionario con preguntas objetivas y cerradas que permitan a los encuestados optar por varias alternativas previamente seleccionadas según el criterio y metas que pretende alcanzar el presente trabajo de investigación.

Según (Qualtrics XM, 2021) La investigación cuantitativa es un método de recogida de datos en un contexto de estudios principalmente científicos. En base a los datos recogidos, se pueden probar hipótesis predefinidas. Además de los métodos cuantitativos, también existen métodos cualitativos. Dependiendo del objetivo de la investigación, conviene realizar una investigación cualitativa o cuantitativa o una combinación de ambos métodos.

La investigación cuantitativa consiste en recolectar y analizar datos numéricos. Este método es ideal para identificar tendencias y promedios, realizar predicciones, comprobar relaciones y obtener resultados generales de poblaciones grandes. Este método se utiliza

ampliamente en las ciencias naturales y sociales: biología, química, psicología, economía, sociología, marketing, etc (Ortega, 2022).

3.1.3 Técnica del cuestionario

Un cuestionario es un instrumento de investigación que consta de un conjunto de preguntas u otros tipos de indicaciones que tienen como objetivo recopilar información de un encuestado. Un cuestionario de investigación suele ser una combinación de preguntas cerradas y preguntas abiertas. Las preguntas abiertas y de formato largo ofrecen al encuestado la capacidad de desarrollar sus pensamientos. Los datos recopilados a partir de un cuestionario de datos pueden ser de naturaleza tanto cualitativa como cuantitativa. Un cuestionario puede entregarse o no en forma de encuesta, pero una encuesta siempre consta de un cuestionario (Question Pro, 2021).

Las encuestas cualitativas utilizan preguntas abiertas para producir respuestas escritas de forma extensa. Las preguntas tendrán como objetivo revelar opiniones, experiencias, narrativas o relatos. A menudo, es un precursor muy útil de las entrevistas o los grupos focales, ya que ayudan a identificar los temas o problemas iniciales para luego indagar más en la investigación. Las encuestas se pueden utilizar de forma iterativa, cambiándose y modificando a lo largo de la investigación para obtener nueva información (Deakin University, 2021).

El usar cuestionarios como técnica de recolección de datos tiene algunas ventajas importantes, según (Debois, 2019) estas son:

- Tiene un bajo costo
- Es práctico
- Otorga resultados rápidos
- Es escalable
- Es comparable
- Es sencillo de analizar
- Es válido y confiable

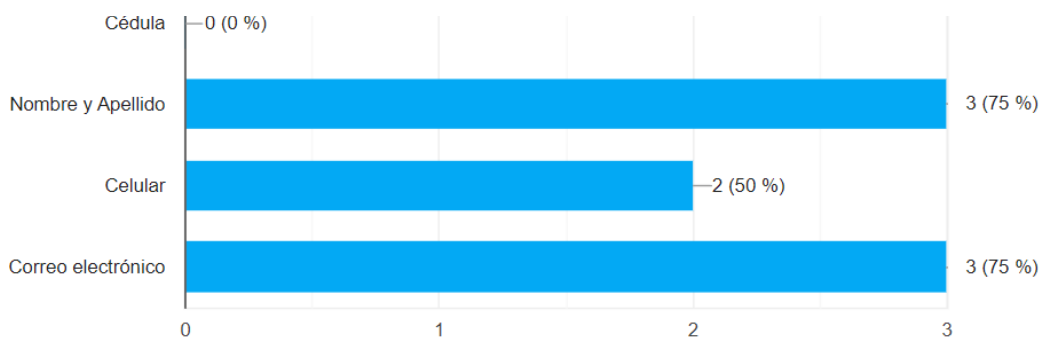
- Es estandarizado
- No pone presión sobre la persona que responde
- Permite respuestas anónimas

3.1.4 Análisis del cuestionario

En el modelo de cuestionario (ver Anexo 1), realizado a las principales personas encargadas de la gestión de eventos dentro de la facultad de ingeniería en la Universidad Católica Santiago de Guayaquil se incluyen preguntas acerca de la periodicidad de los eventos, la información solicitada a sus asistentes y las principales formas de comunicación y publicidad de estos.

Con los datos obtenidos, se puede notar el interés por parte de la facultad con respecto a la automatización y modernización de procesos de reserva y la obtención de datos de los asistentes de manera organizada e inequívoca para futuras campañas relacionadas a este ente.

Figura 2 ¿Qué datos le interesa recabar por parte de los asistentes a los encuestados?

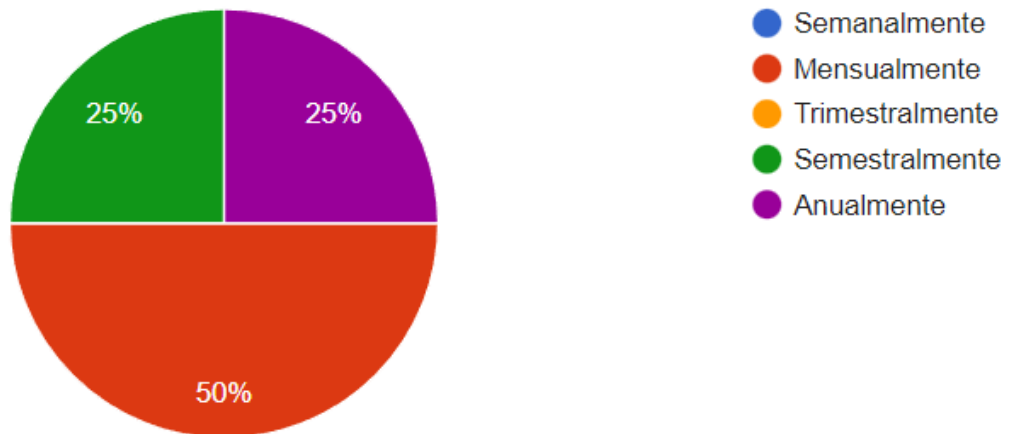


Tal como se puede visualizar en la Figura 2, se aprecia que el 75% de los encuestados está de acuerdo en dos puntos en común, como es la obtención de los nombres y apellidos de los asistentes además de su correo electrónico. Sin embargo, un detalle que no debe ser ignorado es

que la mitad de los encuestados están de acuerdo en que también es importante la obtención del número de celular de los asistentes.

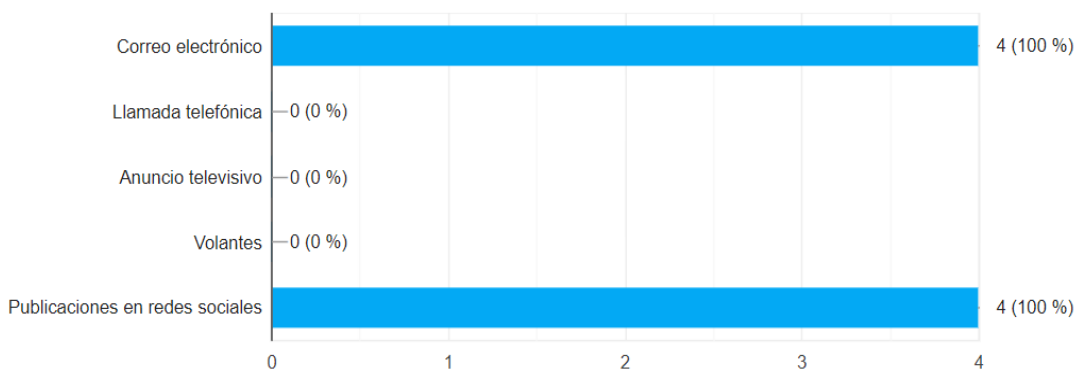
La Figura 3 muestra según el porcentaje de encuestados, la frecuencia

Figura 3 ¿Con que frecuencia se realizan eventos en la Facultad de ingeniería de la Universidad Católica Santiago de Guayaquil?



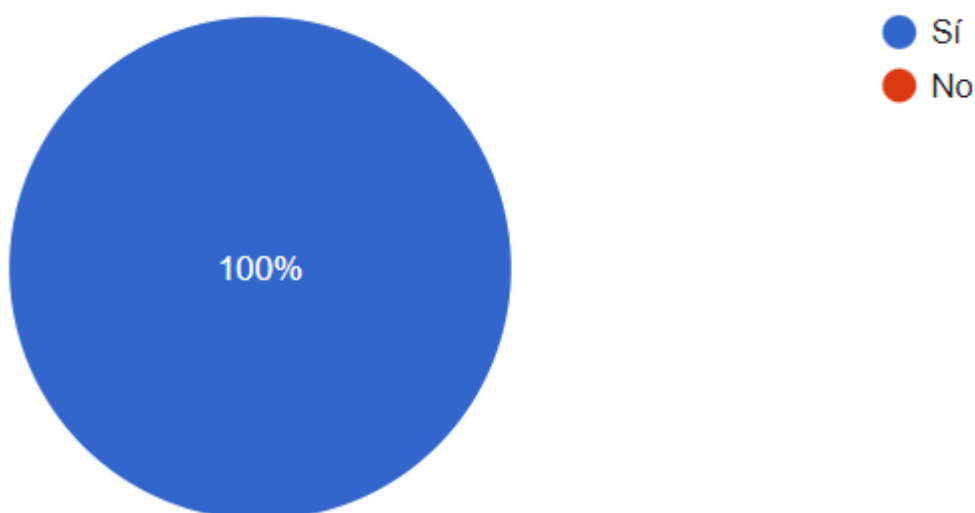
con la que se suelen realizar eventos en la facultad de ingeniería de la Universidad Católica Santiago de Guayaquil. Como se puede observar en el gráfico la mitad de encuestados concuerda en que estos se realizan mensualmente. Si se toma en cuenta que muchos de estos eventos suelen estar divididos en múltiples charlas, exposiciones, etc., al realizarse mensualmente existe la oportunidad de obtener mucha información por parte de los asistentes y adicionalmente ofertar métodos de comunicación tecnológicos que faciliten la conexión entre la facultad y los asistentes.

Figura 4 ¿Qué medios utiliza la Facultad de ingeniería de la Universidad Católica Santiago de Guayaquil para realizar las convocatorias a sus eventos?



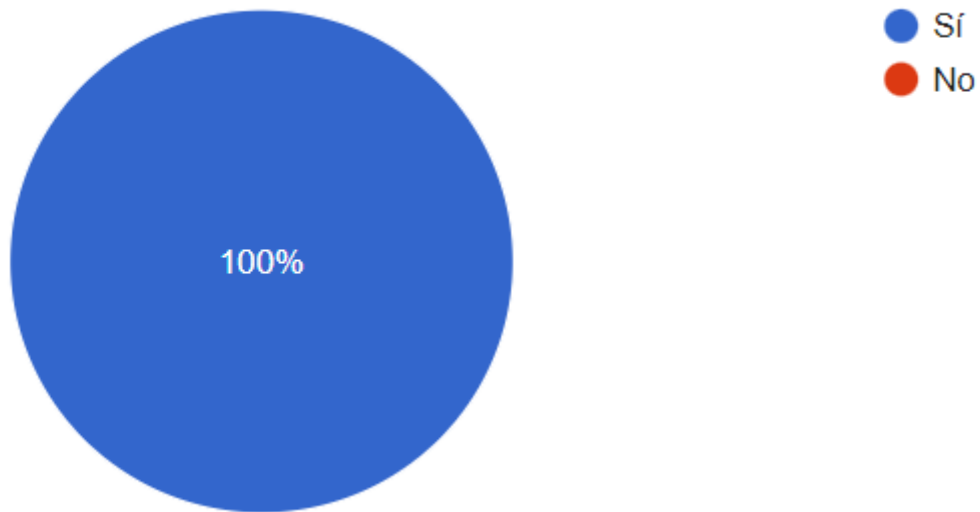
La Figura 4 muestra el porcentaje de medios de comunicación usados por la facultad para realizar la convocatoria de los distintos eventos. Todos los encuestados estuvieron de acuerdo que los principales medios son las publicaciones en redes sociales y vía correo electrónico, esto denota un interés que tiende hacia la parte tecnológica, ya que como se puede notar en el gráfico nadie votó por medios tradicionales tales como volantes, llamada telefónica o anuncio televisivo. Esto presenta una ventana de oportunidad para los proyectos de modernización y automatización de procesos.

Figura 5 Porcentaje de encuestados a los que le gustaría contar con información electrónica almacenada en una base de datos de todos los asistentes o personas que han mostrado interés sobre sus actividades, eventos y servicios



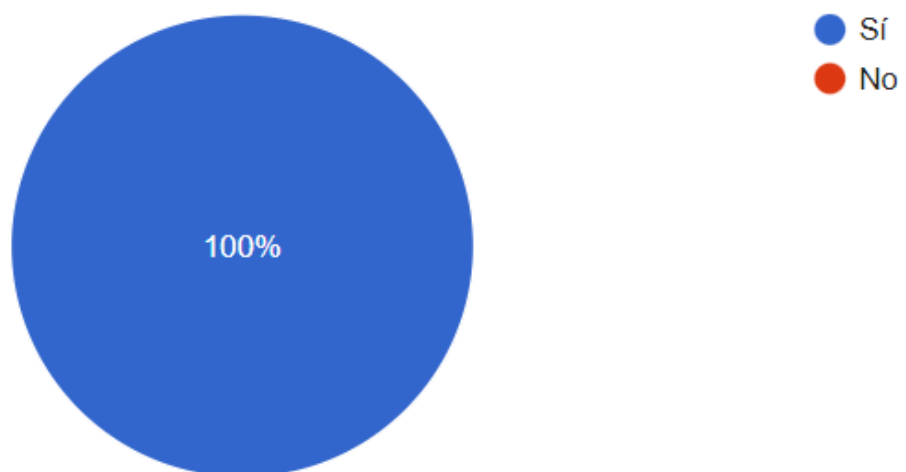
La Figura 5 identifica y analiza el porcentaje de encuestados interesados en manejar toda la información de los asistentes o personas interesadas en los eventos, actividades, etc., ofertados por la facultad de Ingeniería de la Universidad Católica Santiago de Guayaquil por medio de un sistema electrónico, dado que el 100% de los encuestados otorgó una respuesta positiva, esto confirma lo previamente dicho con respecto a la Figura 4, que las tendencias de la facultad se inclinan hacia un ambiente tecnológico y modernizado.

Figura 6 Porcentaje de encuestados a los que le gustaría contar con una herramienta informática que gestione, verifique y analice la información personal de los asistentes confirmando que efectivamente se trata de quien dice ser



La Figura 6 continua el patrón de respuestas positivas con respecto a la tecnología observado en la pregunta 5. En esta pregunta se cuestiona a los encuestados acerca de su interés sobre contar con una herramienta tecnológica que gestione, verifique y analice la información personal de las personas interesadas en los eventos, confirmando que sea quien dice ser y así automatizando este proceso que suele hacerse manualmente, además de ofrecer mayores garantías a los asistentes de estos eventos.

Figura 7 Porcentaje de encuestados a los que le gustaría contar con una herramienta informática que gestione los aforos de las distintas actividades, eventos y servicios que oferta por medio de reservas y verificando las mismas por código de barras, códigos QR



La Figura 7 representa la opinión de todos los encuestados con respecto a contar con un sistema tecnológico que gestione los aforos y las reservas de los distintos eventos y actividades ofertados por parte la facultad de ingeniería en la Universidad Católica Santiago de Guayaquil además de contar con un sistema verificador de las mismas por medio de nuevas tecnologías como son los códigos QR, códigos de barras, entre otros. El gráfico nuevamente refleja una intención positiva por parte de todos los encuestados, nuevamente evidenciando la postura tecnológica de sus organizadores.

El hecho de que actualmente la facultad de ingeniería de la Universidad Católica Santiago de Guayaquil no cuente con las herramientas informáticas para administrar los procesos de reserva, control accesos y gestión de datos de los asistentes e interesados es una ventana de oportunidad sobre un problema que debe ser solucionado con el sistema adecuado, además que, según se pudo notar en el análisis de los cuestionarios realizados, existe la postura por parte de los interesados hacia una propuesta tecnológica que automatice estos procesos.

CAPÍTULO IV

PROPUESTA TECNOLÓGICA

En este capítulo se describirá la fase de desarrollo de la solución tecnológica para el diseño e implementación de un sistema de acceso vía móvil, web y escritorio para el control de reservas y aforos usando Bots de Telegram y códigos QR para la facultad de ingeniería de la Universidad Católica Santiago de Guayaquil.

Entre los distintos tópicos que aquí se detallaran, se encuentran: el proceso de desarrollo, las distintas funciones que cumplirá el Bot a la hora de atender solicitudes de los usuarios, los requerimientos del sistema, su implementación y distintas pruebas sobre el mismo.

El modelo de desarrollo que usaremos como base para el desarrollo del aplicativo, como se mencionó anteriormente, será el modelo iterativo y su respectiva metodología: análisis, diseño, programación e implementación. El sistema estará conformado por 4 programas que se interconectan para trabajar en conjunto dentro del sistema. Estos son chatbot, lector de códigos QR, gestor de horarios para reservas y panel de administración.

El chatbot contará con las siguientes funciones:

- Registro del usuario.
- Búsqueda de usuario previamente registrado (para acelerar el proceso de reserva en próximas ocasiones).
- Mostrar al usuario los diferentes eventos disponibles y la cantidad de cupos disponibles para reserva.
- Generación de código QR único asociado a la reserva del usuario.
- Envío de la reserva al correo electrónico del usuario registrado.
- Validación de reserva activa asociada al número de cédula del usuario.
- Cancelación de reserva activa.
- Envío de código de verificación de 4 dígitos para confirmar cancelación de reserva al correo del usuario registrado.

El lector de códigos QR contara con las siguientes funciones:

- Lectura del código QR que el chatbot otorgo al usuario a la hora de que este realizo la reserva por medio de una Pi camera y una raspberry pi.
- Validación de reserva comprobando si esta es válida o no.
- Mostrar por pantalla la información de la reserva.

El gestor de horarios para reservas contara con las siguientes funciones:

- Monitorear constantemente las reservas activas para detectar si alguna de estas caducó.
- Monitorear constantemente los eventos activos para detectar si alguno de estos caducó y ya no presentarlo como disponible al usuario al momento de reservar.

El panel de administración contara con las siguientes funciones:

- Obtener lista de las reservas activas y sus datos.
- Convertir a un archivo Excel la lista de reservas activas y sus datos.
- Imprimir lista de reservas activas y sus datos.
- Obtener lista de las personas registradas y sus datos.
- Convertir a un archivo Excel la lista de personas registradas y sus datos.
- Imprimir lista de personas registradas y sus datos.
- Editar los datos de las personas registradas
- Obtener la lista de las actividades disponibles para reserva y sus datos.
- Convertir a un archivo Excel la lista de las actividades disponibles para reserva y sus datos.
- Imprimir lista de las actividades disponibles para reserva y sus datos.
- Agregar nuevas actividades que podrán ser reservadas por los usuarios.

4.1 Herramientas y recursos para el desarrollo del sistema

Las herramientas y recursos que se usan para el desarrollo del sistema de acceso vía móvil, web y escritorio para el control de reservas y aforos usando Bots de Telegram y códigos QR para la facultad de ingeniería de la Universidad Católica Santiago de Guayaquil son las siguientes:

- Python (Lenguaje de programación interpretado): Permite crear funciones y objetos que definan el comportamiento de un chatbot.
 - Por medio de este lenguaje de programación se implementó el módulo Telebot en su wrapper oficial llamado PyTelegramBot. Este módulo se lo usa dentro del Bot para gestionar todas las conexiones hacia la nube de Telegram, controlando así las comunicaciones con los distintos usuarios y las sesiones de los mismos. El Bot constantemente está controlando nuevas peticiones, las cuales las recibe por medio de un JSON enviado por la nube de Telegram con toda la información necesaria para crear una nueva sesión e instancia de servicio.
 - Se usa el módulo PyTelegramBot para el envío de respuestas del Bot hacia los usuarios. El uso de este módulo permite adicionalmente el uso de respuestas enriquecidas tales como botones, marcado de texto, emojis entre otros.
 - Se usa Python para la creación de clases que permitan gestionar adecuadamente el conjunto de datos que se usan para ser almacenados y usados de cada transacción.
 - Se implementaron APIs de verificación de correo electrónico y de número de teléfono usando el módulo Requests de Python para gestionar las respectivas peticiones “GET” y sus parámetros.
 - Se usa la librería JSON de Python para convertir las respuestas de las APIs en un formato de fácil manejo para el flujo del programa.
 - Se definieron funciones dentro del lenguaje de programación encargadas de gestionar cada uno de los pasos tanto de solicitud de datos como de creación de reservas.

- Se usaron códigos del tipo Unicode para permitir al Bot el envío de emojis y que este sea más amigable con el usuario.
- Se realizó las validaciones para el ingreso de datos usando las herramientas propias del lenguaje sin requerir a módulos o librerías externas, de esta manera se acelera el tiempo de respuesta y se evita sobrecargas en tiempos de ejecución (esto dado a la naturaleza interpretada de Python)
- Se definió una función para la creación de códigos QR dentro de Python usando el módulo qrcode y su función Make, donde esta función recibe como parámetro el token de reserva único generado por el sistema para codificarlos dentro de este QR a ser generado. La versión de QR usada por el Bot es la versión 3 con un tamaño de 10 puntos y un borde de seguridad de 4 puntos. La imagen es generada en formato PNG que es el estándar general para toda imagen a ser usada en protocolos web de cualquier tipo.
- Se realizó la creación del software encargado de gestionar la lectura de códigos QR por medio de la PI Camera. Para la programación de este se usó las librerías CV2 para visión por computadora, la librería Pyzbar para la decodificación de códigos de barras y QRs y la librería Time para controlar las pausas del sistema. Usando CV2 el sistema se encuentra constantemente leyendo la información presente en el flujo de video de la cámara cuadro a cuadro y analizándola por medio de Pyzbar. En caso de que Pyzbar detecte un código QR en el fotograma, este decodificará el código y realizará una petición a la base de datos para confirmar la validez de este. Una vez se realice la comprobación se mostrará una caja de texto en pantalla, generada por la librería pymsgbox, que, en caso de ser afirmativa, mostrará la información de la reserva, caso contrario mostrará un error indicando que el código QR no es válido. Todos los códigos usados recientemente se agregarán a una lista, donde en caso de tratar de ser usados nuevamente serán ignoradas por la cámara, esto con el fin de no saturar el sistema con reservas ya revisadas y tener así mayor cantidad de recursos disponibles.

- Por medio de Python se implementó todo un archivo .py donde se encuentran declarada la conexión a la base de datos usando el módulo MySQL Connector, este permite realizar conexiones rápidas además de contar con gestión de excepciones incorporado. Únicamente este módulo debe recibir como parámetros el host, el user, el password y el nombre de la base MySQL a la que se desea conectar. Adicionalmente en este archivo se encuentran definidas todas las funciones que escriben, consultar, actualizan y eliminan datos de la base de datos. Estas funciones son llamadas desde las distintas funciones del Bot según sea necesario para realizar la gestión de la base de datos en la nube.
 - Usando Python también se realizó el desarrollo de un programa que, usando las funciones definidas en el archivo de Base de datos, está constantemente revisando y actualizando el estado de las reservas activas, las reservas canceladas y los eventos caducados.
- C# (Lenguaje de programación compilado): Permite el diseño de interfaces graficas amigables de manera sencilla además de definir funciones y objetos que permitan administrar el Bot.
 - Por medio de este lenguaje de programación se realizó la creación del apartado visual del panel de administración del sistema, esto usando únicamente controles propios del lenguaje, de manera que se tenga una interfaz nativa y que pueda ser fácilmente transformable a un instalador sin mayores complicaciones para su posterior envío y uso.
 - Todas las validaciones de datos y eventos son realizadas únicamente con controles propios del lenguaje, de esta manera se obtiene una mayor compatibilidad y coherencia de desarrollo, además de que al no requerir el uso de librerías externas se obtiene una mayor velocidad de carga al momento de que el usuario realice una acción.

- Al ser un lenguaje compilado, este tiene una mayor velocidad a la hora de ejecutarse, además de que se tiene mayor control sobre el código que se ejecuta dado que debido a la naturaleza del lenguaje este no debe ser visible.
 - Muchas de las funciones del programa cuentan con la capacidad de convertir tablas de datos a archivos Excel o de imprimir estas tablas. Al ser un lenguaje desarrollado por Microsoft permite la fácil integración de estas capacidades usando librerías tales como Microsoft Office Interop que se comunican directamente con la distribución de Office instalada en el sistema sin tener que usar intermediarios para la copia de datos o el acceso a la impresora usando la misma comunicación del sistema operativo (centro de impresión) del sistema operativo.
 - Se definió una clase de control de base de datos donde se definieron todas las consultas o queries que pueden realizar las otras partes del programa, esto con el fin de centralizar en una sola ubicación todos estos datos y que, en caso de ser modificados, este cambio se realice en todas las partes del programa. Adicionalmente, siguiendo esta idea, en esta clase se definió también la cadena de conexión a la base.
 - Para realizar la conexión a la base de datos en la nube y poder ejecutar las funciones de consulta, inserción, modificación y eliminación en esta se usa la librería MySQL Data Client, la cual proporciona todos los métodos necesarios desde la creación de la conexión, pasando por la generación de Querys, hasta el cierre de la misma.⁷
- MySQL (Sistema de base de datos relacional): Permite almacenar y consultar datos de manera rápida y ordenada por medio de comandos estructurados llamados Querys.
 - Se usa MySQL para administrar todo el modelo entidad relación previamente mencionado y para controlar toda la información que debe ser registrada y consultada para el correcto funcionamiento del sistema de reservas.

- ClearDB (Implementación de MySQL en la nube): Permite usar todas las ventajas que ofrece MySQL con respecto al manejo de datos desde cualquier lugar del mundo ya que la base de datos estará cargada en internet.
 - Al ser una implementación de MySQL en la nube y al estar montada en Heroku, que ofrece una disponibilidad de 99.9%, se afirma que el Bot tendrá alta disponibilidad, además de que podrá acceder a los datos sin requerir hardware adicional in situ y que también se podrá acceder desde cualquier lugar del mundo conectado a internet.
- Amazon Web Services (Plataforma en la nube como servicio): Permite montar una máquina virtual en la nube la cual esté disponible 24/7/365 ejecutando todos los procesos que sea necesario dentro de las capacidades contratadas dado que ejecuta un sistema operativo común como cualquier ordenador o servidor local.
 - Se lo uso para crear una máquina virtual en la nube del tipo EC2 corriendo el sistema operativo Ubuntu sin interfaz gráfica para ahorrar recursos. Dentro de este sistema se instaló las librerías necesarias para la ejecución además de Python 3. Esta máquina virtual se tuvo que configurar para permitir conexiones entrantes y salientes para poder consultar a la base de datos de Heroku y para permitir el ingreso y salida de mensajes del Bot. Dentro de esta máquina virtual corren dos de los aplicativos de Python necesarios para el funcionamiento del sistema (Bot de Telegram y el control de Tiempos y Spam) ofreciendo así una disponibilidad 24/7.
- Telegram Bot API (API oficial de Telegram para el desarrollo de Bots): Permite crear Bots en múltiples lenguajes de programación siguiendo todos los lineamientos de Telegram además de facilitar herramientas que puede ocupar el desarrollador para agilizar el proceso de desarrollo.
 - El Bot consulta constantemente el API de Telegram para verificar si se encuentran nuevas peticiones que deban ser

atendidas, para posteriormente descargar los datos en un archivo JSON e iniciar una nueva transacción.

- El Bot utiliza el API para poder enviar las respuestas a los usuarios por medio del chatid previamente informado por parte del API mismo cuando se realizó la petición “GET” inicial.
- Abstract API (API para verificar validez de un correo electrónico o número de celular): Permite verificar por múltiples parámetros la validez de un correo electrónico o de un celular, recibiendo como parámetro único el correo electrónico o el número de celular en cuestión.
 - El Bot la usa para poder consultar la validez de un correo electrónico y de un numero de celular. Asegurando la veracidad de los mismos y que no se trata de correos desechables o elementos similares. De esta manera se controla que la calidad de los datos en la base sea optima y no sean datos basura.
- DietPi (Distribución de Linux): Sistema operativo diseñado para ejecutarse en el miniordenador Raspberry Pi consumiendo bajos recursos.
 - Se la usa como sistema operativo base para la ejecución del sistema encargado de leer los códigos QR, al ser un sistema operativo de bajos recursos permite orientar la capacidad de procesamiento hacia los procesos ocupados por el Bot, volviéndolo así más eficiente.
- PiCamera (Accesorio de cámara para la Raspberry Pi): Minicámara diseñada para usarse en la Raspberry Pi por medio del puerto de serie diseñado con este fin.
 - Se la usa para la lectura de códigos QR generados del Bot junto a un script de Python que analiza constantemente el flujo de video para revisar si se encuentra algún código QR en la imagen y posteriormente revisar la validez del mismo en el sistema.

- Raspberry Pi (Miniordenador de bajo costo): Miniordenador de bajo costo que permite realizar desarrollos de todo tipo, sobre todo orientado al campo de IoT o internet de las cosas.
 - Es el hardware necesario donde corre el sistema operativo DietPi y el script para lectura de códigos QR. Es una placa pequeña, económica y potente.
- PyCharm (IDE de desarrollo para el lenguaje de programación Python): Permite crear aplicaciones usando el lenguaje de programación Python de manera rápida y sencilla ofreciéndole al usuario distintas herramientas para la escritura, revisión, pruebas y desarrollo ágil de código.
 - Se uso para el desarrollo de todos los programas realizados en el lenguaje de programación Python. Al ser un IDE especializado para este lenguaje, se puede realizar un desarrollo rápido y eficaz sin arriesgar a fallos usando toda la amplia gama de herramientas que este programa ofrece.
- Visual Studio 2022 (IDE de desarrollo para el lenguaje de programación C# entre otros): Permite crear aplicaciones usando el lenguaje de programación C# de manera rápida y sencilla ofreciéndole al usuario distintas herramientas para la escritura, revisión, pruebas y desarrollo ágil de código.
 - Se uso para el desarrollo del panel de administración del sistema. Es la herramienta base que ofrece Microsoft para desarrollar su lenguaje de programación propietario C#. Cuenta con una gama de capacidades diferentes que permitieron un desarrollo rápido y eficaz.

Para mayor información acerca del software revisar el anexo 3 al final del documento.

4.2 Requerimientos de software

En relación a los requisitos mínimos de software requeridos para el uso del chatbot, se obtiene la siguiente información:

- Sistema operativo (Indiferente de si es móvil o de escritorio, mientras cuente con el aplicativo oficial de Telegram).
- Navegador Web (Indiferente, por medio de Telegram Web).

El chatbot no exhibe ninguna limitación con los elementos previamente mencionados.

4.3 Requerimientos de hardware

Con respecto a los requisitos mínimos de hardware para el uso del chatbot, se presenta la siguiente información:

- Dispositivo electrónico con conexión a internet (indiferente)

El chatbot no exhibe ninguna limitación con los elementos previamente mencionados dado que se encuentra implementado en la nube y es accesible por medio de dispositivos de escritorio, móviles o con acceso a un navegador web.

4.4 Base de datos

Para el desarrollo del sistema se concluyó que la mejor base de datos que se adapta a las necesidades del proyecto es la de MySQL en su implementación de ClearDB.

A continuación, se muestra una tabla comparativa donde se definen las diferentes ventajas de usar MySQL sobre otro tipo de bases de datos:

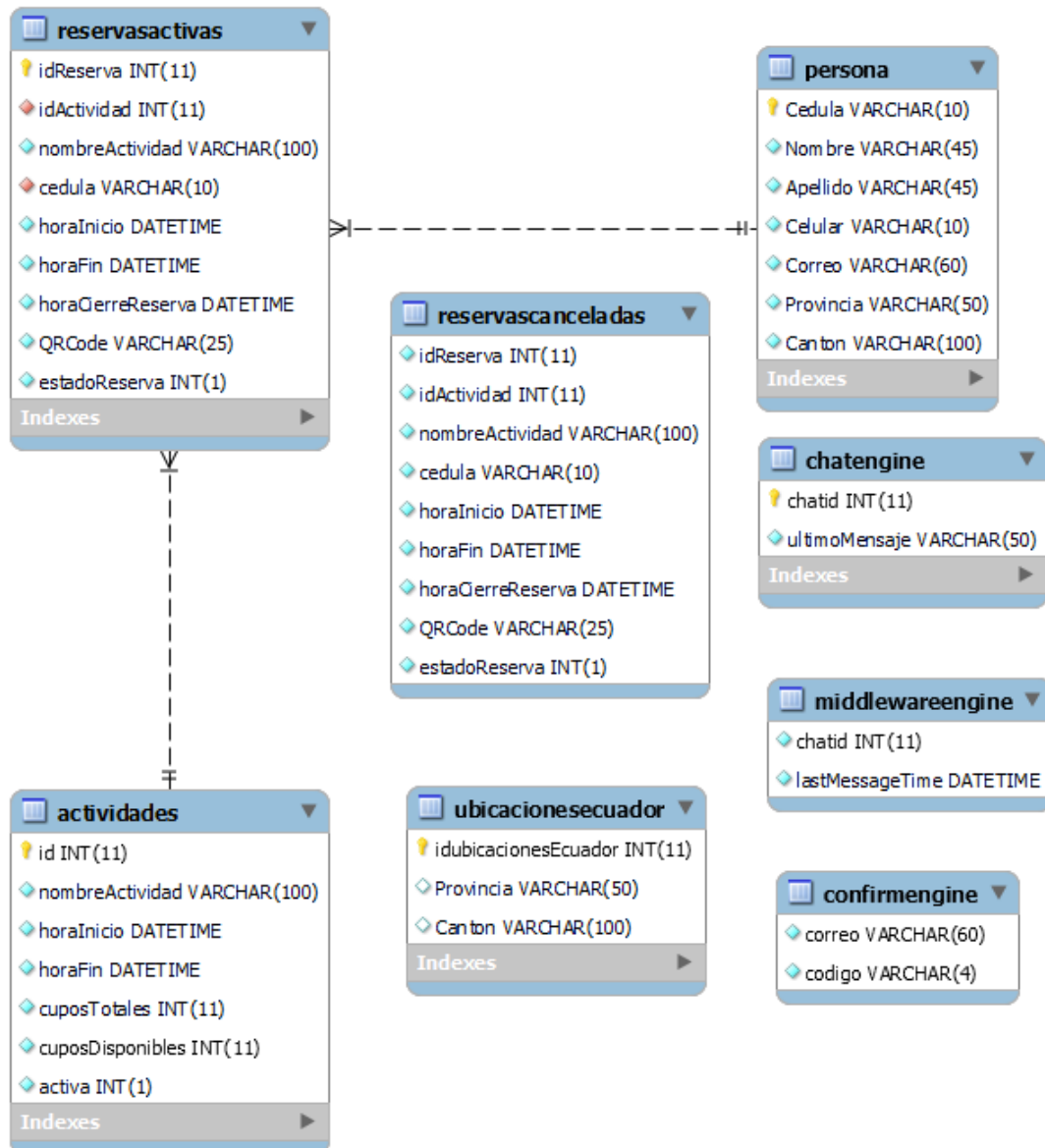
Tabla 3 Comparación entre bases de datos (Murga Campuzano, 2019)

Característica	MySQL	Oracle	MariaDB
Velocidad transaccional	✓	✓	✓
Rápida y sencilla de instalar	✓		✓

Poca demanda de recursos de software	✓		✓
Baja probabilidad de corrupción de datos	✓	✓	✓
Open Source	✓		✓
Es escalable	✓	✓	✓
Posee un entorno gráfico para su administración	✓	✓	
Es compatible con aplicaciones web	✓	✓	✓
Es compatible con varios SO	✓	✓	✓
Calificación	9/9	6/9	8/9

4.5 Modelo entidad relación

El modelo entidad relación usado para el desarrollo del sistema se encuentra conformado y definido de la siguiente manera:



En el objeto “actividades” se almacena toda la información relacionada a las actividades que se encuentran disponibles o que se encontraban disponibles. Este control se realiza por medio de un campo de estado el cual indicara si la actividad se encuentra activa o inactiva. Este objeto se relaciona con la tabla “reservasactivas” por medio del campo “id” definido como llave primaria y autoincremento por cada nueva actividad agregada, permitiendo comprobar la existencia de una actividad al momento de realizar una reserva. Todos los campos del objeto están definidos como “No vacíos”.

En el objeto “chatengine” se almacena el identificador único del chat con un usuario en específico, además de su último mensaje o interacción con el Bot. El Bot por cada nueva interacción realiza una revisión a esta tabla verificando así bajo ciertas condiciones que no se trate de un intento de spam hacia el mismo. Todos los campos del objeto están definidos como “No vacíos”.

En el objeto “confirmengine” se almacenan los distintos códigos de verificación generados por el Bot para confirmar transacciones de registro y eliminación por parte del usuario. Estos códigos de 4 dígitos al ser alfanuméricos tienen asignado un campo del tipo Varchar. Todos los campos del objeto están definidos como “No vacíos”.

En el objeto “middlewareengine” se almacena el identificador único del chat con un usuario en específico además de la estampa de tiempo del último mensaje enviado por el usuario dentro de un campo de tipo Datetime para su fácil comprensión por el Bot. Esto con el fin de comprobar la diferencia en minutos del último mensaje con respecto al mensaje actual, si esta es mayor que 5 minutos da por terminada la sesión con ese usuario, de manera que el Bot no se sature y pueda atender a más personas. Todos los campos del objeto están definidos como “No vacíos”.

En el objeto “persona” se almacena toda la información relacionada los usuarios que interactúan con el Bot. Este objeto se relaciona con la tabla “reservasactivas” por medio del campo “cedula” definido como llave primaria, ya que, al ser un valor único otorgado a las personas dentro de Ecuador, permite identificar fácilmente a un usuario dentro del sistema y evita el riesgo de repetición dentro de la base. Este campo se encuentra definido como Varchar dentro de la base, dado que debido a que, a pesar de contener únicamente números, estos pueden empezar por 0, ocasionando que al ser definido como otro tipo de dato ese dígito sea omitido. Todos los campos del objeto están definidos como “No vacíos”.

En el objeto “reservasactivas” se almacena la información de todas las reservas activas realizadas por los usuarios. Esta tabla cuenta con dos llaves foráneas que hacen referencia al campo “id” de la tabla actividades y al campo

“cedula” de la tabla persona respectivamente. En esta tabla también se encuentra el campo QRCode de tipo Varchar el cual almacena la cadena alfanumérica de 25 caracteres que permite identificar las reservas. Todos los campos del objeto están definidos como “No vacíos”.

En el objeto “reservascanceladas” se encuentran todas las reservas canceladas por el usuario o por el administrador del sistema por medio del panel de administración. Esta tabla contiene los mismos campos de la tabla “reservasactivas” dado que cuando una reserva activa se cancela, el Bot ejecuta un query que se encarga de copiar esa reserva a la tabla de reservas canceladas y posterior un query para eliminar los datos copiados de la tabla “reservasactivas”. Todos los campos del objeto están definidos como “No vacíos”.

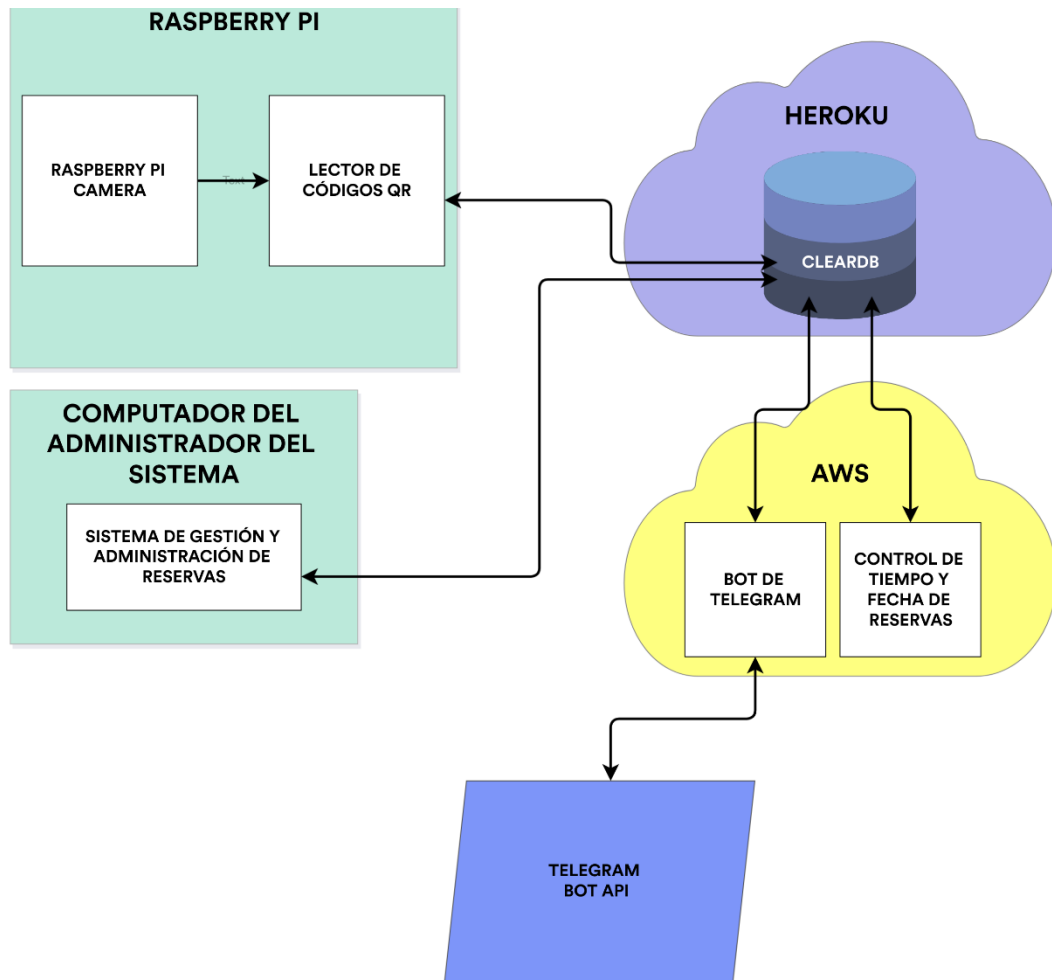
En el objeto “ubicacionesecuador” se encuentran todas las provincias y cantones del Ecuador. En esta tabla se repiten los valores de provincias según sea necesario dada la asociación de cada una con sus respectivos cantones. Todos los campos del objeto están definidos como “No vacíos”.

El modelo de datos, el tipo de cada campo, las referencias entre tablas y la descripción detallada de la base de datos se encuentran en el Anexo 2 al final del documento.

4.6 Diagrama de arquitectura de software

El siguiente diagrama detalla de manera grafica la estructura del sistema de acceso vía móvil, web y escritorio para el control de reservas y aforos usando Bots de Telegram y códigos QR.

Figura 6 Diagrama de arquitectura de software



CONCLUSIONES

A medida que el tiempo avanza, podemos notar como la tecnología continuamente sigue adquiriendo mayor importancia y relevancia en todo tipo de actividades, desde las más cotidianas hasta los procesos más complejos. Siendo así, se vuelve menester el modernizar labores manuales, con el fin de lograr mayor eficiencia, eficacia y sobre todo un acercamiento más real y tecnológico con el mundo que nos rodea.

Por medio del instrumento de recolección de datos, en este caso un cuestionario, se pudo identificar y conocer los requerimientos y funcionalidades que requería la facultad de ingeniería de la Universidad Católica Santiago de Guayaquil, entre los cuales se encuentra un sistema tecnológico que permitiese la atención concurrente de múltiples usuarios interesados en la reservación y cancelación de actividades, eventos, etc., por medio de múltiples vías siendo las principales aplicativos móviles, web y ordenadores de escritorio, otro requerimiento es la creación de una base de datos de los asistentes e interesados que únicamente deba ser llenada con datos validados y sin información errónea, además de permitir la generación de reportes por usuarios, eventos y reservas a medios tradicionales como son archivos xlsx (Microsoft Excel) para su posterior análisis, también se contempló la creación de un panel de administración que agrupe múltiples herramientas que faciliten el trabajo por parte de la persona encargada de la gestión y administración del sistema donde pueda crear, eliminar y diligencias usuarios y eventos, entre otros.

A través de la investigación realizada, se pudo notar como existen múltiples herramientas que permiten el desarrollo de chatbots, sin embargo, esta misma búsqueda permitió la elección de las mejores herramientas en base a los criterios y requerimientos previos al desarrollo, entre estas fueron elegidas Python como lenguaje de programación principal para la mayor parte del proyecto por su velocidad y potencia a la hora del desarrollo además de tener una compatibilidad total con el API oficial de Telegram y ser el lenguaje número uno para el desarrollo en placas como la raspberry pi, se seleccionó C# como lenguaje de programación para el panel de administración por su

apartado visual de fácil comprensión y su rapidez en ejecución, como base de datos se optó por Clear DB que no es nada más que una implementación en la nube de una de las bases de datos más populares como lo es MySQL y para la ejecución en la nube se decidió usar AWS (Amazon Web Services) por ser el mejor proveedor de servicios en cloud como se mostró en la Figura 1 logrando así, el desarrollo de un sistema de chatbot completamente automático, con tiempos de respuesta inmediatos, con control de accesos, con compatibilidad en múltiples dispositivos y sobre todo alta disponibilidad orientada hacia los usuarios. Además, de contar con tecnologías de nueva generación, asegurando así la mantenibilidad y validez del sistema hacia el futuro, y haciendo uso de APIs que no modifiquen la estructura del sistema pero que añadan funcionalidades según Telegram siga desarrollando su ecosistema.

Finalmente se evaluó el sistema, con el fin de que este cumpla todos los requerimientos y lineamientos señalados por la universidad. Entre estas pruebas se realizaron múltiples peticiones simultáneas al chatbot a las cuales este respondió de manera adecuada y rápidamente sin perder el hilo de la conversación con el usuario, además se realizaron pruebas en tiempo real desde el panel de administración del sistema donde se editó y consulto las reservas activas y canceladas, donde se usó la opción de generación de reportes, donde estos se generaron sin problemas ni incongruencias. Todas las pruebas realizadas sobre el sistema en general resultaron positivas, desde su uso y disponibilidad en múltiples dispositivos, abarcando su alta disponibilidad, pasando por la generación y filtración de información válida, la creación de reservas, el trato amable hacia el usuario entre otros aspectos que convierten a este proceso en un éxito.

RECOMENDACIONES

Dado que el resultado del presente proyecto quedara bajo la administración de la facultad de Ingeniería de la Universidad Católica Santiago de Guayaquil se adjuntan las siguientes recomendaciones que podrían tomarse en cuenta para el correcto funcionamiento y la constante actualización y mejor del software.

- Dado que todos los servicios y aplicativos están ejecutándose en la nube, considerar la posibilidad de adquirir planes de paga que permitan tener un mayor soporte al cliente, además de permitir laborar con cargas de trabajo mucho mayores.
- Gestionar el respaldo manual o automático de la base de datos, con el fin de evitar la pérdida de información valiosa por cualquier causa.
- Dado que el sistema recopila información por parte de las personas, gestionar la compra o desarrollo de algún software que pueda tomar esa información, la analice y por medio de los resultados de ese análisis idear nuevas campañas y servicios por parte de la universidad.
- Idear un plan de mantenimiento y revisión cada cierto tiempo al sistema, esto con el fin de mitigar cualquier posible falla o para realizar mejoras en el sistema.
- Dado que el lector de códigos QR necesita de internet para revisar las reservas en la base de datos ubicada en la nube, considerar la adquisición de una fuente auxiliar de internet en caso de fallo de la conexión principal, como puede ser un modem móvil.

Referencias

Abstract. (15 de diciembre de 2021). *Automate routine dev work*. Obtenido de

Abstract: <https://www.abstractapi.com/>

ACG Technical Editors Team. (5 de noviembre de 2021). *What is the Ruby*

programming language? Obtenido de A cloud guru:

<https://acloudguru.com/blog/engineering/what-is-the-ruby-programming-language>

Amazon Web Services (AWS). (2022). *What is cloud computing?* Obtenido de

Amazon Web Services: <https://aws.amazon.com/what-is-cloud-computing/>

Amazon Web Services. (8 de octubre de 2021). *What is a Relational*

Database? Obtenido de Amazon Web Services:

<https://aws.amazon.com/relational-database/>

Artificial Solutions. (22 de enero de 2020). *The Value of AI Chatbots for*

Business. Obtenido de Medium:

[https://medium.com/@artificialsolutions/the-value-of-ai-chatbots-for-business-](https://medium.com/@artificialsolutions/the-value-of-ai-chatbots-for-business-c34ccc222884#:~:text=Chatbots%20help%20to%20reduce%20costs,in%20excess%20of%20live%20agents)

[c34ccc222884#:~:text=Chatbots%20help%20to%20reduce%20costs,in%20excess%20of%20live%20agents](https://medium.com/@artificialsolutions/the-value-of-ai-chatbots-for-business-c34ccc222884#:~:text=Chatbots%20help%20to%20reduce%20costs,in%20excess%20of%20live%20agents).

AskPython. (20 de julio de 2021). *The pyzbar module: Decoding Barcodes in*

Python. Obtenido de AskPython: [https://www.askpython.com/python-](https://www.askpython.com/python-modules/pyzbar-module)

[modules/pyzbar-module](https://www.askpython.com/python-modules/pyzbar-module)

Atea Ataroa Limited. (18 de febrero de 2020). *DietPi*. Obtenido de Distro Watch: <https://distrowatch.com/table.php?distribution=dietpi>

AVI Networks. (19 de marzo de 2021). *Infrastructure As A Service (IaaS)*. Obtenido de Avi Networks: <https://avinetworks.com/glossary/infrastructure-as-a-service-iaas/>

Bhandari, P. (30 de julio de 2020). *An introduction to qualitative research*. Obtenido de Scribbr: <https://www.scribbr.com/methodology/qualitative-research/#:~:text=Qualitative%20research%20involves%20collecting%20and,concepts%2C%20opinions%2C%20or%20experiences.&text=Qualitative%20research%20is%20commonly%20used,health%20sciences%2C%20history%2C%20etc>

Bigelow, S. J. (6 de abril de 2020). *Microsoft Azure*. Obtenido de Tech Target: <https://searchcloudcomputing.techtarget.com/definition/Windows-Azure>

Budimirovic, M., & Filipova, V. (13 de noviembre de 2020). *Technology in times of pandemic*. Obtenido de WCO News: <https://mag.wcoomd.org/magazine/wco-news-94/technology-in-times-of-pandemic/>

Busseto, L., Wick, W., & Gumbinger, C. (27 de mayo de 2020). *How to use and assess qualitative research methods*. doi:<https://doi.org/10.1186/s42466-020-00059-z>

Clark, A. (24 de octubre de 2021). *Pillow*. Obtenido de Pillow Documentation: <https://pillow.readthedocs.io/en/stable/>

Components101. (9 de diciembre de 2018). *Pi Camera Module – 5MP*.

Obtenido de Components101: <https://components101.com/misc/pi-camera-module#:~:text=The%20Pi%20camera%20module%20is,learning%20or%20in%20surveillance%20projects>.

Computer Hope. (8 de febrero de 2020). *Raspbian*. Obtenido de Computer

Hope: <https://www.computerhope.com/jargon/r/raspbian.htm>

ComputerScience.org Staff. (2 de noviembre de 2021). *Computer*

Programming Languages. Obtenido de ComputerScience.org: <https://www.computerscience.org/resources/computer-programming-languages/>

Dang Tran, J. (12 de octubre de 2021). *What are the future challenges of the*

digital transformation of companies? Obtenido de Manutan: <https://www.manutan.com/blog/en/news/what-are-the-future-challenges-of-the-digital-transformation-of-companies>

Deakin University. (4 de junio de 2021). *Qualitative Study Design*. Obtenido de

Deakin University Library: <https://deakin.libguides.com/qualitative-study-designs>

Debois, S. (8 de marzo de 2019). *10 Advantages And Disadvantages Of*

Questionnaires. Obtenido de Survey Anyplace Blog: <https://surveyanyplace.com/blog/questionnaire-pros-and-cons/>

Deloitte. (21 de septiembre de 2021). *La transformación digital en el centro de*

la estrategia organizacional. Obtenido de Deloitte Ecuador:

<https://www2.deloitte.com/ec/es/pages/strategy/articles/La-transformacion-digital-en-el-centro-de-la-estrategia-organizacional.html>

Diario Los Andes. (30 de julio de 2021). *Transformación digital en Latinoamérica: la situación en Ecuador*. Obtenido de Diario Los Andes: <https://www.diariolosandes.com.ec/transformacion-digital-en-latinoamerica-la-situacion-en-ecuador/>

Drake, M. (14 de diciembre de 2020). *What is MySQL?* Obtenido de Digital Ocean: <https://www.digitalocean.com/community/tutorials/what-is-mysql>

Gartner. (24 de diciembre de 2020). *Digital Business Transformation*. Obtenido de Gartner IT: <https://www.gartner.com/en/conferences/emea/symposium-south-africa/featured-topics/digital-transformation>

GeekforGeeks. (28 de marzo de 2021). *OpenCV Python Tutorial*. Obtenido de GeekforGeeks: <https://www.geeksforgeeks.org/opencv-python-tutorial/#:~:text=OpenCV%20is%20a%20huge%20open,C%2B%2B%2C%20Java%2C%20etc.>

GeeksforGeeks. (22 de jun de 2020). *What is NuGet?* Obtenido de GeeksforGeeks: <https://www.geeksforgeeks.org/what-is-nuget/#:~:text=NuGet%20is%20the%20official%20package,library%20with%20some%20descriptive%20metadata.&text=NuGet%20package%20contains%20a%20single,like%20the%20package's%20version%20number.>

- Ghanchi, J. A. (6 de junio de 2019). *Top 7 Programming Languages to Develop an AI-Powered Chatbot*. Obtenido de Chatbots Life: <https://chatbotslife.com/top-7-programming-languages-to-develop-an-ai-powered-chatbot-f253e728845c>
- Gillis, A. S. (7 de abril de 2020). *Amazon Web Services (AWS)*. Obtenido de Techtarget - AWS Section: <https://searchaws.techtarget.com/definition/Amazon-Web-Services>
- Hack Reactor. (26 de agosto de 2021). *What is JavaScript used for?* Obtenido de HackReactor: <https://www.hackreactor.com/blog/what-is-javascript-used-for>
- Higgins, A. (4 de agosto de 2021). *What Is a Chatbot? How Is it Changing Customer Experience?* Obtenido de Salesforce: <https://www.salesforce.com/blog/what-is-a-chatbot/>
- IBM Cloud Education. (8 de mayo de 2019). *What is Java?* Obtenido de IBM Cloud Learn Hub: <https://www.ibm.com/cloud/learn/java-explained>
- IBM Cloud Education. (14 de julio de 2021). *PaaS (Platform-as-a-Service)*. Obtenido de IBM Cloud Learn Hub: <https://www.ibm.com/cloud/learn/paas>
- Johnson, D. (24 de marzo de 2021). *What is Telegram? A quick guide to the fast and secure messaging platform*. Obtenido de Business Insider: <https://www.businessinsider.com/what-is-telegram>
- Knox, K. (17 de mayo de 2021). *What is Google Cloud Platform (GCP)?* Obtenido de A cloud guru:

<https://acloudguru.com/blog/engineering/what-is-google-cloud-platform-gcp>

Kolade, C. (7 de octubre de 2021). *What is Go? Golang Programming Language Meaning Explained*. Obtenido de FreeCodeCamp: <https://www.freecodecamp.org/news/what-is-go-programming-language/>

Lexis Ecuador. (noviembre de 2012). *Ley de comercio electrónico, firmas y mensaje de datos*. Obtenido de Telecomunicaciones Gobierno Ecuador: <https://www.telecomunicaciones.gob.ec/wp-content/uploads/downloads/2012/11/Ley-de-Comercio-Electronico-Firmas-y-Mensajes-de-Datos.pdf>

Logi Analytics. (1 de junio de 2021). *What's the Difference? Relational vs Non-Relational Databases*. Obtenido de Logi Analytics: <https://www.logianalytics.com/relational-vs-non-relational-databases/>

Luxner, T. (15 de marzo de 2021). *Cloud Computing Trends: 2021 State of the Cloud Report*. Obtenido de Flexera: <https://www.flexera.com/blog/cloud/cloud-computing-trends-2021-state-of-the-cloud-report/>

Microsoft. (1 de octubre de 2021). *A tour of the C# language*. Obtenido de Microsoft Docs: <https://docs.microsoft.com/en-us/dotnet/csharp/tour-of-csharp/>

Microsoft. (25 de enero de 2021). *An introduction to NuGet*. Obtenido de Microsoft Docs: <https://docs.microsoft.com/en-us/nuget/what-is->

Oracle Corporation. (07 de octubre de 2020). *What is a Chatbot?* Obtenido de Oracle: <https://www.oracle.com/chatbots/what-is-a-chatbot/>

Oracle Corporation. (27 de septiembre de 2021). *What is a database?* Obtenido de Oracle Cloud: <https://www.oracle.com/database/what-is-database/>

Ortega, C. (24 de febrero de 2022). *Investigación cuantitativa. Qué es y cómo realizarla.* Obtenido de QuestionPro: <https://www.questionpro.com/blog/es/que-es-la-investigacion-cuantitativa/>

Python Software Foundation. (2021). *What is Python? Executive Summary.* Obtenido de Python: <https://www.python.org/doc/essays/blurb/>

QR Code Generator Marketing Team. (22 de junio de 2020). *QR Codes 101: A Beginner's Guide.* Obtenido de QR Code Generator: <https://www.qr-code-generator.com/qr-code-marketing/qr-codes-basics/>

Qualtrics XM. (17 de febrero de 2021). *Investigación cuantitativa.* Obtenido de Qualtrics - Gestión de experiencia: <https://www.qualtrics.com/es/gestion-de-la-experiencia/investigacion/investigacion-cuantitativa/>

Question Pro. (4 de noviembre de 2021). *Questionnaires: The ultimate guide, samples & examples.* Obtenido de Question Pro Blog: <https://www.questionpro.com/blog/what-is-a-questionnaire/>

Rao, A. (15 de julio de 2021). *Python Requests Tutorial: GET and POST Requests in Python*. Obtenido de Edureka! Python Programming: <https://www.edureka.co/blog/python-requests-tutorial/#z1>

Red Hat. (2021). *Qué son las API y para qué sirven*. Obtenido de Red Hat: <https://www.redhat.com/es/topics/api/what-are-application-programming-interfaces>

Regenesys Business School. (21 de agosto de 2020). *What is Digital Transformation in business?* Obtenido de RegInsights: <https://insights.regenesys.net/what-is-digital-transformation-in-business/>

Rodriguez, I. (16 de enero de 2021). *What Is Pip? A Guide for New Pythonistas*. Obtenido de Real Python: <https://realpython.com/what-is-pip/>

Rostad, S. (28 de febrero de 2018). *What is Heroku? A Simple Explanation for Non-Techies*. Obtenido de Trifin: <https://trifinlabs.com/what-is-heroku/>

Royal, C. (8 de julio de 2021). *Working with Modules in Python: Must Known Fundamentals for Data Scientists*. Obtenido de Analytics Vidhya: <https://www.analyticsvidhya.com/blog/2021/07/working-with-modules-in-python-must-known-fundamentals-for-data-scientists/#:~:text=In%20Python%2C%20Modules%20are%20simply,t o%20include%20in%20your%20application.>

Salesforce. (2021). *What is SaaS?* Obtenido de Salesforce: <https://www.salesforce.com/in/saas/>

SproutQR. (10 de noviembre de 2020). *Static QR Codes vs. Dynamic QR Codes | The Difference*. Obtenido de SproutQR blog: <https://www.sproutqr.com/blog/static-vs-dynamic-qr-codes#:~:text=other%20are%20big,-,What%20Is%20a%20Static%20QR%20Code%3F,all%20aligned%20in%20various%20ways>.

Sweigart, A. (1 de septiembre de 2021). *PyMsgBox Basics*. Obtenido de Github: <https://github.com/asweigart/PyMsgBox/blob/master/docs/basics.rst>

Telegram Core Team. (5 de noviembre de 2021). *Telegram Bot API*. Obtenido de Telegram Core: <https://core.telegram.org/bots/api>

Three Piece Marketing. (11 de diciembre de 2019). *How Chatbots Can Add Value to Your Business*. Obtenido de Three Piece Marketing: <https://threepiece.marketing/blogs/how-chatbots-can-add-value-to-your-business/>

Valarezo, E. (15 de mayo de 2018). *Cédula de identidad ecuatoriana y ruc: La guía definitiva*. Obtenido de Jybaro: <https://www.jybaro.com/blog/cedula-de-identidad-ecuatoriana/>

Vennam, S. (18 de agosto de 2020). *Cloud Computing*. Obtenido de IBM Cloud Learning: <https://www.ibm.com/cloud/learn/cloud-computing>

Wang, F. (11 de noviembre de 2021). *pyTelegramBotAPI*. Obtenido de Github: <https://github.com/eternnoir/pyTelegramBotAPI>

ANEXOS

Anexo 1: Cuestionario

El siguiente cuestionario que se presenta a continuación se realizó a las distintas autoridades encargadas de la gestión y publicidad de eventos dentro de la facultad de ingeniería en la Universidad Católica Santiago de Guayaquil.

1. Al momento de realizar un evento en la Facultad de ingeniería de la Universidad Católica Santiago de Guayaquil, ¿Qué datos le interesa recabar por parte de los asistentes? Puede seleccionar más de uno.

- Cédula
- Nombre y Apellido
- Celular
- Correo electrónico
- Otros (Indicar)

2. ¿Con que frecuencia se realizan eventos en la Facultad de ingeniería de la Universidad Católica Santiago de Guayaquil?

- Semanalmente
- Mensualmente
- Trimestralmente
- Semestralmente
- Anualmente

3. ¿Qué medios utiliza la Facultad de ingeniería de la Universidad Católica Santiago de Guayaquil para realizar las convocatorias a sus eventos? Puede seleccionar más de uno.

- Correo electrónico
- Llamada telefónica
- Anuncio televisivo
- Volantes
- Publicaciones en redes sociales
- Otros (Indicar)

4. ¿A usted le gustaría contar con información electrónica almacenada en una base de datos de todos los asistentes o personas que han mostrado interés sobre sus actividades, eventos y servicios?
- Sí
 - No
5. ¿A usted le gustaría contar con una herramienta informática que gestione, verifique y analice la información personal de los asistentes confirmando que efectivamente se trata de quien dice ser?
- Sí
 - No
6. ¿A usted le gustaría contar con una herramienta informática que gestione los aforos de las distintas actividades, eventos y servicios que oferta por medio de reservas y verificando las mismas por código de barras, códigos QR, entre otros?
- Sí
 - No

Anexo 2: Descripción de las tablas y campos de la base de datos

Nombre de la tabla	actividades
Descripción de la tabla	Contiene toda la información de las distintas actividades que podrán reservar los usuarios además de un histórico los eventos pasados.

Nombre	Tipo de dato	¿Es nulo?	Descripción
Id	Int (Primary key) (11)	No es nulo	Identificador único de cada una de las actividades agregadas al sistema (autoincrementado).
nombreActividad	Varchar (100)	No es nulo	Nombre de la actividad.
horaInicio	Datetime	No es nulo	Fecha y hora de inicio del evento.
horaFin	Datetime	No es nulo	Fecha y hora de finalización del evento.
cuposDisponibles	Int (11)	No es nulo	Cantidad de cupos disponibles seleccionada por el administrador al publicar un nuevo evento.

Activa	Int (1)	No es nulo	Indica si un evento aún no ha pasado y puede reservarse.
--------	---------	------------	--

Nombre de la tabla	chatEngine
Descripción de la tabla	Permite llevar un control de las interacciones generadas por un usuario para evitar que sature al chatbot con muchas peticiones simultaneas protegiéndolo así del spam.

Nombre	Tipo de dato	¿Es nulo?	Descripción
Chatid	Int (Primary key) (11)	No es nulo	Almacena el identificador único generado por un chat al interactuar con el Bot.
ultimoMensaje	Varchar (50)	No es nulo	Almacena los mensajes enviados para el usuario para así detectar si existe o no riesgo de spam y solo atender una solicitud del usuario a la vez.

Nombre de la tabla	middlewareEngine
Descripción de la tabla	Permite llevar un control de las horas con las que el usuario interactuó con el Bot y en caso de que este no responda en un tiempo o no concluya una transacción se dará por terminada su sesión.

Nombre	Tipo de dato	¿Es nulo?	Descripción
Chatid	Int (11)	No es nulo	Almacena el identificador único generado por un chat al interactuar con el Bot.
lastMessageTime	Datetime	No es nulo	Almacena el día y la hora del último mensaje del usuario en formato UTC.

Nombre de la tabla	confirmEngine
Descripción de la tabla	Permite llevar el control de los distintos códigos generados por el chatbot que son enviados por correo a los usuarios para verificar ciertas partes de la transacción como verificación de cancelación de las

	reservas o verificación de correo electrónico y confirmar que estamos tratando con la persona en cuestión y no Bots.
--	--

Nombre	Tipo de dato	¿Es nulo?	Descripción
Correo	Varchar (60)	No es nulo	Correo al cual se envió el código de confirmación.
Código	Varchar (4)	No es nulo	Código alfanumérico de cuatro dígitos generado de manera aleatoria por el Bot

Nombre de la tabla	persona
Descripción de la tabla	Contiene toda la información personal relacionada a los usuarios registrados en el sistema.

Nombre	Tipo de dato	¿Es nulo?	Descripción
Cedula	Varchar (10) (Primary key)	No es nulo	Contiene los diez dígitos de la cedula del usuario.

Nombre	Varchar (45)	No es nulo	Contiene el nombre del usuario.
Apellido	Varchar (45)	No es nulo	Contiene el apellido del usuario.
Celular	Varchar (16)	No es nulo	Contiene los dígitos del número celular del usuario
Provincia	Varchar (50)	No es nulo	Almacena la provincia donde se encuentra el usuario actualmente.
Canton	Varchar (100)	No es nulo	Almacena el cantón donde se encuentra ubicado el usuario actualmente. Este cantón debe pertenecer a la provincia previamente seleccionada.
Correo	Varchar (60)	No es nulo	Contiene el correo del usuario.

Nombre de la tabla	reservasactivas
Descripción de la tabla	Contiene toda la información relacionada a las reservas activas realizadas por los usuarios.

Nombre	Tipo de dato	¿Es nulo?	Descripción
idReserva	Int (11) (Primary key)	No es nulo	Almacena el identificador único de la reserva (autoincrementado).
idActividad	Int (11) (Foreign key -> actividades (id))	No es nulo	Almacena el identificador único de la actividad escogida, este debe existir en la tabla actividades.
nombreActividad	Varchar (100)	No es nulo	Almacena el nombre de la actividad reservada.
Cedula	Varchar (10) (Foreign key -> persona (Cedula))	No es nulo	Almacena los 10 dígitos de la cedula del usuario que realizo la reserva. Esta cedula debe existir en la tabla persona.
horalInicio	Datetime	No es nulo	Almacena la hora de inicio de la reserva.

horaFin	Datetime	No es nulo	Almacena la hora de fin de la reserva.
horaCierreReserva	Datetime	No es nulo	Almacena la hora de vencimiento de la reserva en caso de no ser activada pasado 15 minutos la hora de inicio del evento.
QRCode	Varchar (25)	No es nulo	Almacena los 25 caracteres únicos representados en el código QR enviado al usuario para identificar la reserva.
estadoReserva	Int (1)	No es nulo	Almacena un identificador que permite saber si una reserva aún no está activada (0), si ya se activó (1) o si caducó (-1).

Nombre de la tabla	reservascanceladas
Descripción de la tabla	Contiene toda la información relacionada a las reservas canceladas por parte de los usuarios.

Nombre	Tipo de dato	¿Es nulo?	Descripción
idReserva	Int (11) (Primary key)	No es nulo	Almacena el identificador único de la reserva (autoincrementado).
idActividad	Int (11) (Foreign key -> actividades (id))	No es nulo	Almacena el identificador único de la actividad escogida, este debe existir en la tabla actividades.
nombreActividad	Varchar (100)	No es nulo	Almacena el nombre de la actividad reservada.
Cedula	Varchar (10) (Foreign key -> persona (Cedula))	No es nulo	Almacena los 10 dígitos de la cedula del usuario que realizo la reserva. Esta cedula debe existir en la tabla persona.
horaInicio	Datetime	No es nulo	Almacena la hora de inicio de la reserva.
horaFin	Datetime	No es nulo	Almacena la hora de fin de la reserva.
horaCierreReserva	Datetime	No es nulo	Almacena la hora de vencimiento de la reserva en caso

			de no ser activada pasado 15 minutos la hora de inicio del evento.
QRCode	Varchar (25)	No es nulo	Almacena los 25 caracteres únicos representados en el código QR enviado al usuario para identificar la reserva.
estadoReserva	Int (1)	No es nulo	Almacena un identificador que permite saber si una reserva aún no está activada y fue cancelada (0)

Anexo 3: Manual de usuario (Características del sistema, validaciones y funcionamiento)

A continuación, se adjuntarán las distintas pantallas del sistema y se detallara la función que desempeña cada una de ellas además de sus características y validaciones respectivas en caso de ser necesario.

3.1 Chatbot

El chatbot es aquel que es accesible por el usuario por medio de un computador de escritorio, un dispositivo móvil, o por medio de un navegador usando las aplicaciones oficiales de Telegram, en este las personas realizan su registro en el sistema además de realizar las reservas respectivas.

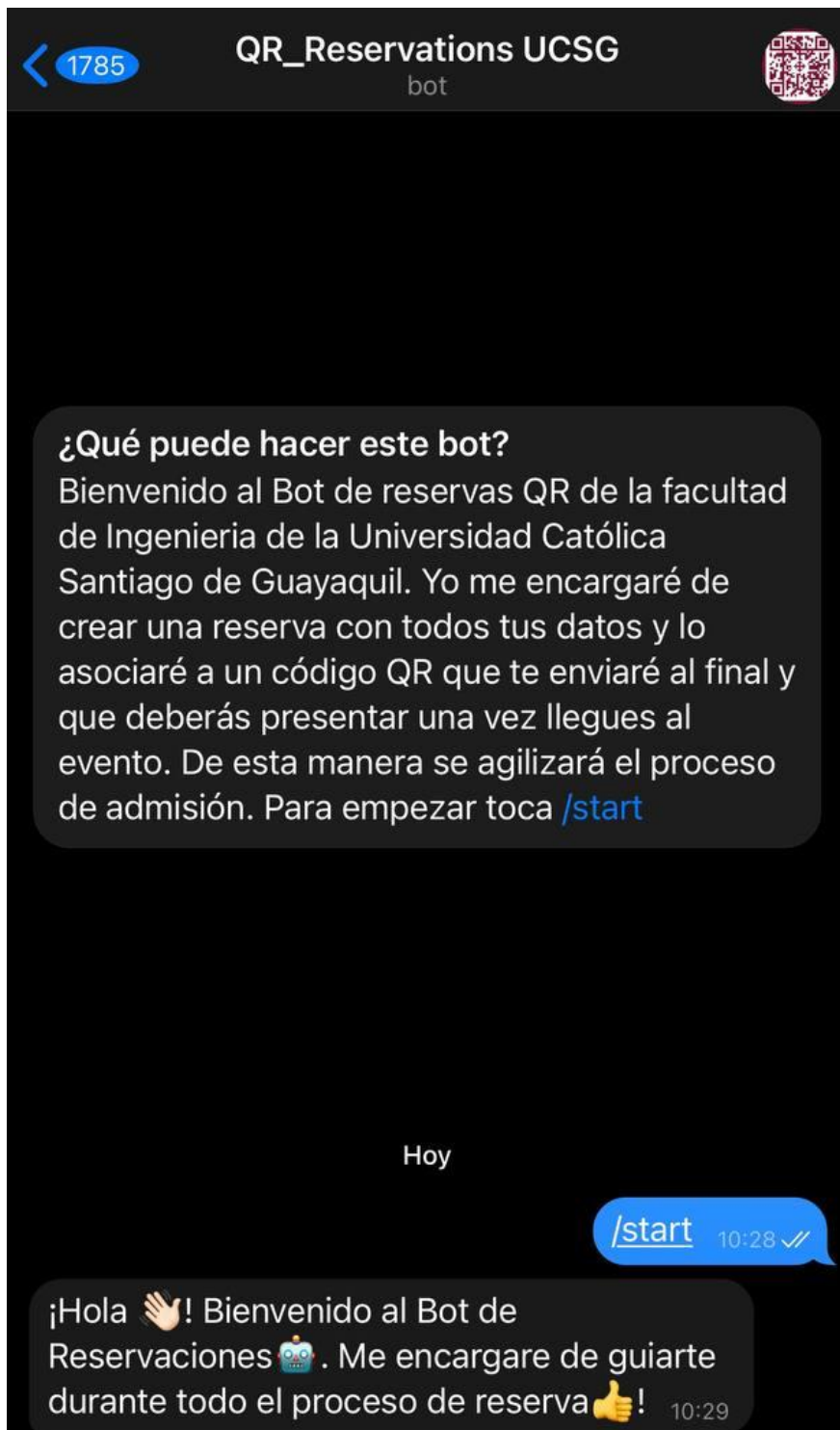
3.1.1 Saludo

Una vez el usuario ingresa al Bot por primera vez se le muestra un párrafo introductorio indicando la funcionalidad del Bot. El Bot al recibir el mensaje de “/start” iniciara la conversación enviando un saludo amistoso al usuario y posteriormente iniciando la solicitud de datos para empezar con el registro de usuario y la reserva de eventos.

3.1.1.1 Validaciones

- El primer comando que recibe el Bot en su primer uso debe ser “/start”, caso contrario el Bot no responderá.

- En los siguientes usos el Bot ya responderá a saludos comunes como “Buenos días”, “Hola” entre otros.



3.1.2 Cédula

El primer dato que solicitará el Bot para iniciar cualquier transacción con el usuario será el número de cédula. Este número es único y permitirá identificar al usuario dentro de la solicitud actual y en futuras reservas permitirá saltarse la parte de obtención de datos personales para ir de inmediato a la reserva ya que el usuario será reconocido con los datos proporcionados en reservas anteriores.

3.1.2.1 Validaciones

- El número de cédula ingresado debe tener una longitud exacta de 10 dígitos.
- El número de cédula ingresado únicamente debe constar de números.
- Debe cumplir con el algoritmo de verificación de cédula para la ciudadanía ecuatoriana.
- Si el número de cédula ya se encuentra en la base de datos, tomar el resto de información de los registros y saltar a selección de evento a reservar.

3.1.2.2 Algoritmo de verificación de cédula para la ciudadanía ecuatoriana

Según (Valarezo, 2018) el algoritmo de verificación es:

El décimo dígito de la cédula es el resultado de un cálculo realizado con los anteriores nueve, de la siguiente manera:

1. Se agrupan los nueve primeros dígitos en dos conjuntos: los que están en posiciones impares y los que están en pares. Por ejemplo, para la cédula de identidad 1713175071, las posiciones impares

serían 1, 1, 1, 5 y 7, mientras que las posiciones pares serían 7, 3, 7 y 0.

2. A las posiciones impares se multiplican por dos (2). Si el resultado de esta multiplicación es mayor a nueve, se le resta nueve. Por ejemplo, si los números en posiciones impares son 1, 1, 1, 5 y 7, la multiplicación por dos de cada número sería 2, 2, 2, 10 y 14. Ya que algunos resultados son mayores a nueve, se procede a restar nueve a esos valores, quedando así: 2, 2, 2, 1 y 5.
3. Se suman los dígitos en posiciones pares, más los resultados de las operaciones en las posiciones impares. Por ejemplo, si los dígitos pares son 7, 3, 7 y 0, la suma de dígitos en posiciones pares sería 17. Por otro lado, si el resultado de las operaciones en las posiciones impares es 2, 2, 2, 1 y 5, su suma sería 12. El total de la suma sería $17 + 12 = 29$
4. Se obtiene el módulo 10 de la suma total anterior, es decir, se toma el dígito más a la derecha del resultado de la suma. Por ejemplo, si suma 29, el módulo 10 sería 9.
5. Si el resultado anterior es cero entonces el dígito verificador es cero, caso contrario al número 10 se le resta el resultado obtenido en el

anterior paso. Por ejemplo, si el resultado del paso anterior es 9, la resta sería $10 - 9 = 1$, siendo 1 el dígito verificador calculado.



3.1.3 Nombre y Apellido

El Bot solicitará el nombre y apellido el usuario para asociarlo a sus registros, además de que una vez el Bot obtenga este dato, podrá llamar al usuario por su nombre.

3.1.3.1 Validaciones

- El usuario deberá ingresar al menos un nombre y un apellido separados por un espacio.

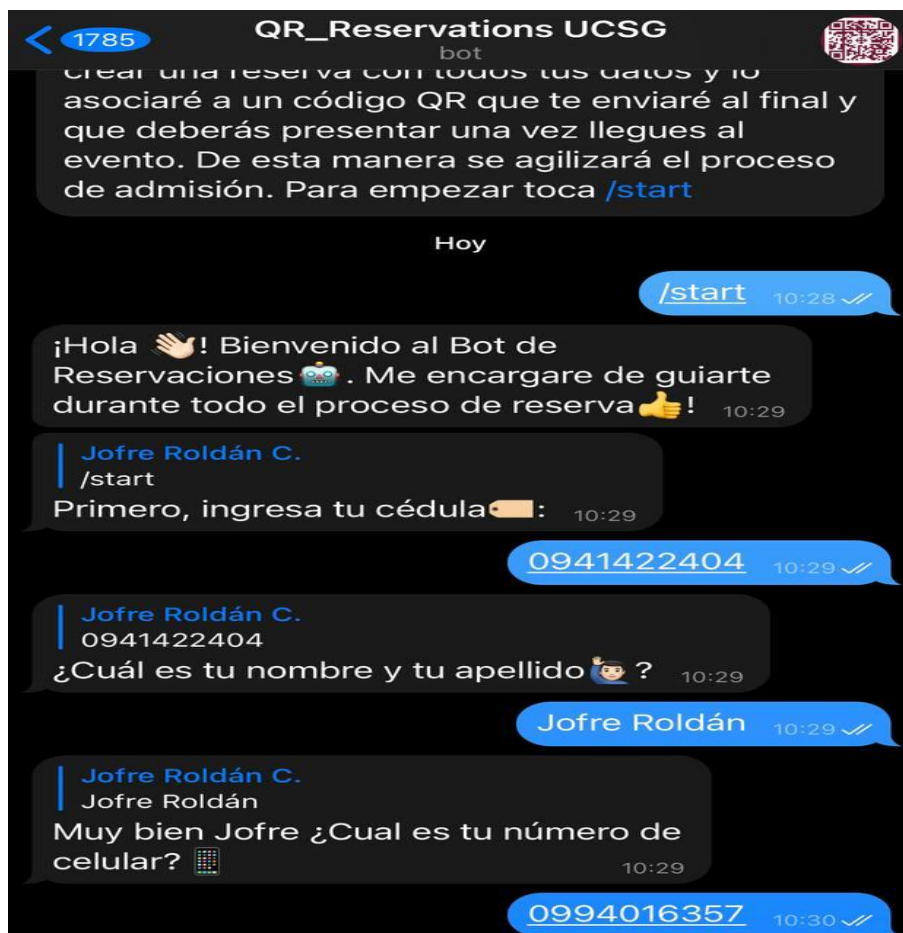


3.1.4 Número de celular

El Bot solicitará el número del celular para asociarlo a la información del usuario de manera que se pueda contar con esta información vital para contactarse con el usuario en caso de eventualidades.

3.1.4.1 Validaciones

- En Ecuador todos los números de celular deben empezar por “0”, en caso de no ser así el Bot solicitará el número nuevamente reportándole al usuario un error.
- El número de celular debe constar de 10 dígitos.
- El número de celular debe estar conformado únicamente por números.
- Usando el API de Abstract se verificará la validez del número de celular, esta deberá devolver un resultado afirmativo, caso contrario se le solicitará el número de celular nuevamente al usuario.



3.1.5 Provincia

El Bot solicitará la provincia en la que se ubica el usuario, para poder asociarlo con su información y conocerlo un poco más. Esta información resulta vital para poder conocer donde se encuentran agrupados los usuarios y que tipos de eventos son los que más les atraen según su posición geográfica.

3.1.5.1 Validaciones

- La provincia seleccionada debe existir en la lista de provincias que aparecen en el menú mostrado, en caso de ser seleccionada una que no se encuentre en la lista el Bot retornará un error y le solicitará al usuario que lo intente nuevamente.



3.1.6 Cantón

Una vez el usuario seleccione de manera correcta una provincia automáticamente el Bot consultara la base de datos en búsqueda de todos los cantones asociados a esa provincia creando un nuevo menú que presenta una lista de estos cantones. El usuario deber elegir en que cantón se ubica según la provincia previamente seleccionada.

3.1.6.1 Validaciones

- El cantón debe existir en la lista de cantones presentada al usuario, en caso de no ser así, el Bot mostrara un error y le solicitara que reingrese los datos nuevamente al usuario.

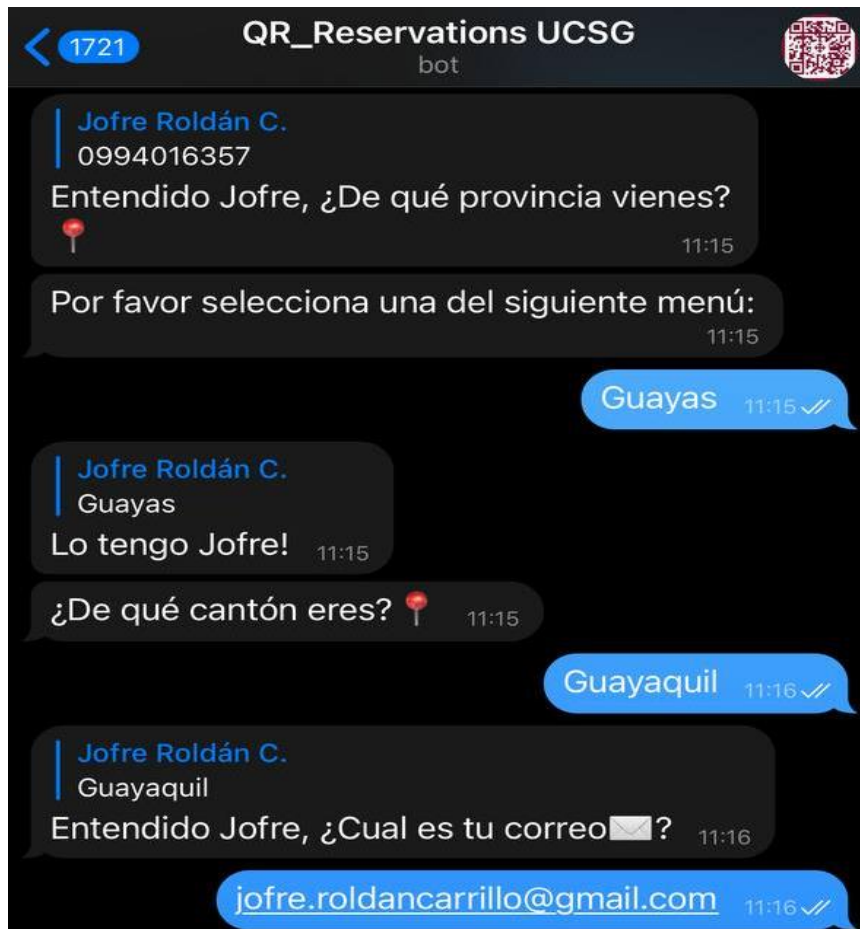


3.1.7 Correo electrónico

El Bot solicitará el correo electrónico de la persona que interactúa para asociarlo a la información del usuario, además este correo se usará para la confirmación de datos y para la eliminación de reservas por medio de códigos de seguridad.

3.1.7.1 Validaciones

- Se verificará que el correo introducido no tenga caracteres inválidos por medio de expresiones regulares de Python.
- Usando el API de Abstract se verificará la validez del correo electrónico, esta deberá devolver confirmación en tres aspectos, los cuales serían que no sea un correo desechable, que la dirección de correo electrónico esté disponible para recibir correos y que el correo electrónico cumpla con una calificación de al menos 0.70/1 en la escala de Abstract.



3.1.8 Confirmación de datos por parte del usuario

Una vez el usuario concluya con la entrada de los datos solicitados anteriormente, el Bot le mostrara de manera ordenada al usuario todos los datos que ingreso para cada variable. Preguntándole por medio de un teclado emergente si estos están correctos o si desea corregirlos.

3.1.8.1 Validaciones

- Si el usuario selecciona la opción “Si, están correctos” se enviará un código de verificación al correo electrónico ingresado por el usuario para verificar la validez de los datos, si el usuario falla en tres ocasiones el código la transacción terminará.
- Si el usuario selecciona la opción “No, deseo corregirlos” el Bot procederá a preguntarle nuevamente sus datos para que realice las correcciones respectivas.



3.1.9 Selección de evento a reservar

Una vez se hayan registrado con éxito los datos del usuario se procederá a la reserva de eventos. La forma en que se presentan los eventos disponibles al usuario es por medio de un teclado emergente donde se encontraran los distintos eventos siguiendo el formato Nombre de evento | Fecha de inicio (año – mes – día) | Fecha de finalización (año – mes – día) | cupos disponibles para reservar.

3.1.9.1 Validaciones

- Se verifica que el usuario haya elegido un evento valido por medio del teclado emergente.
- El sistema internamente verifica no mostrar en el teclado emergente los eventos que no cuenten con cupos disponibles a pesar de ser eventos activos.

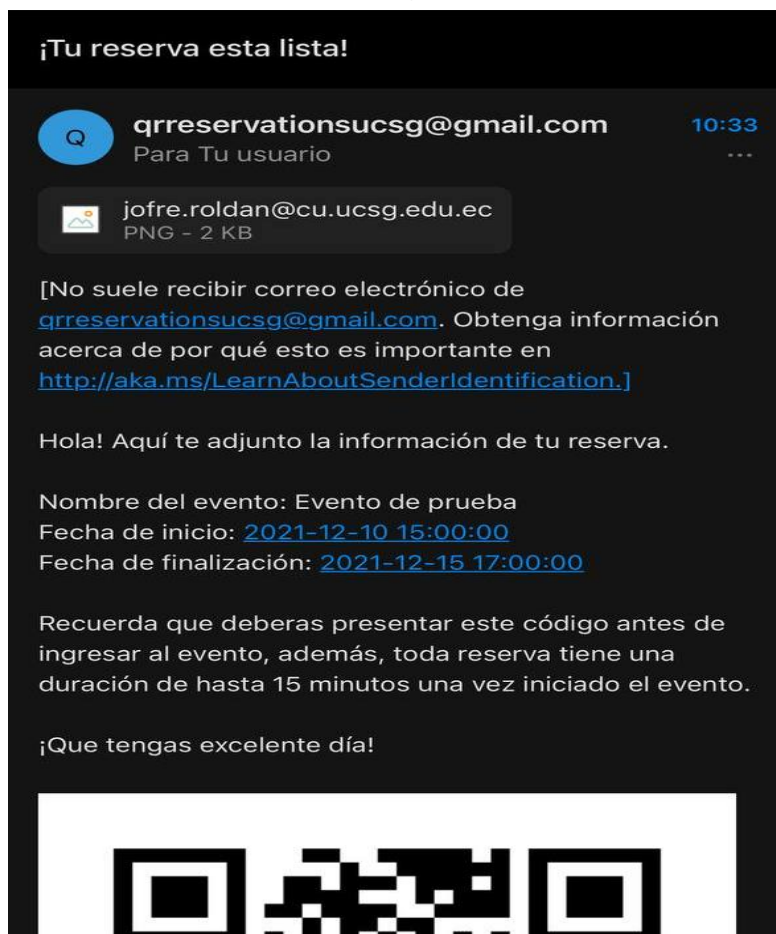


3.1.10 Reserva exitosa

Una vez el usuario seleccione un evento válido, automáticamente el Bot generará un código QR único para el usuario donde estará contenido un código alfanumérico único de 25 caracteres el cual será usado en conjunto con lector de códigos QR del evento para identificar rápidamente la reserva, adicionalmente, en el mismo mensaje se adjuntará la información del evento elegido y también se enviará un correo electrónico al usuario con la información de la reserva como respaldo a la conversación del chatbot. Finalmente, el Bot enviará un mensaje de despedida informando que la transacción concluyó y da la apertura de que en caso de necesitar algo más por parte del usuario, puede escribirle nuevamente.

3.1.10.1 Validaciones

- Se registra la reserva de inmediato en la base de datos para que pueda ser reconocida por el lector QR al momento de ingresar al evento.





Hemos terminado Jofre 🤖, Este es tu código QR para el siguiente evento:

Nombre del evento: Evento de prueba

Fecha de inicio: 2021-12-10
15:00:00

Fecha de finalización: 2021-12-15
17:00:00

Recuerda que deberas presentar este código antes de ingresar al evento, además, toda reserva tiene una duración de hasta 15 minutos una vez iniciado el evento.

10:32



Eso ha sido todo por hoy! Espero haberte ayudado 😊. Recuerda escribirme nuevamente si necesitas algo mas 😊

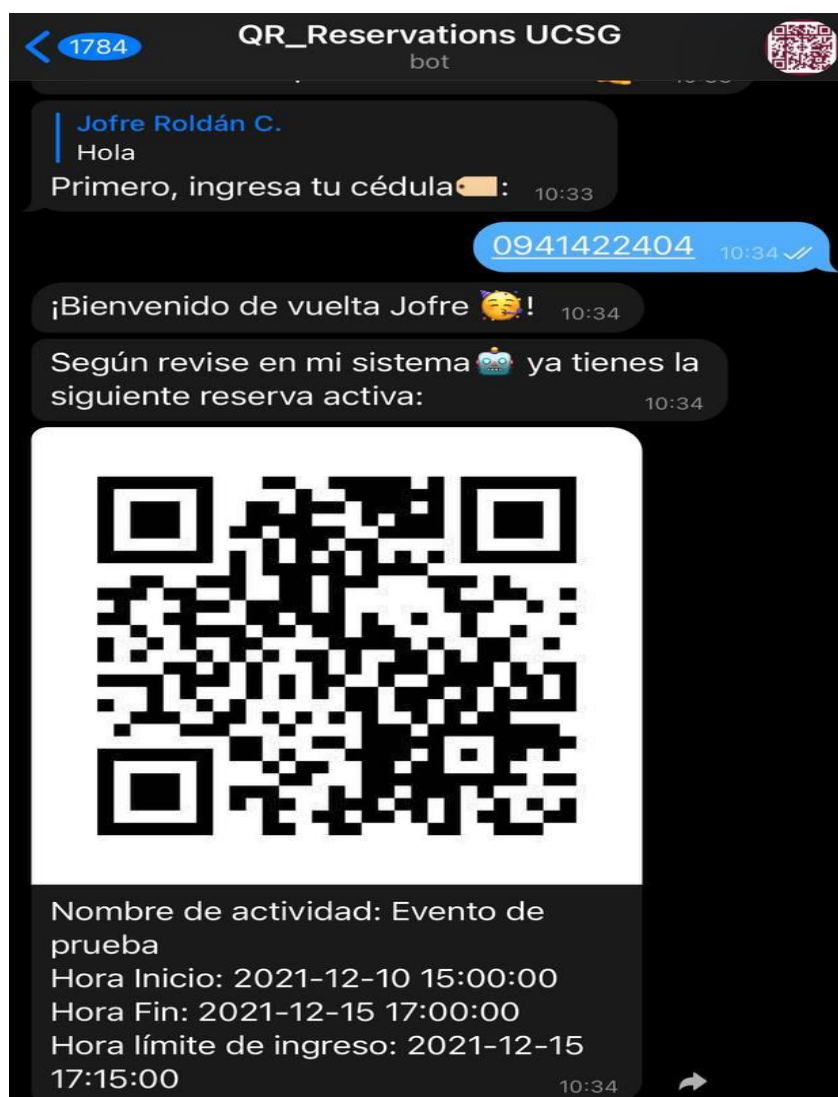
10:32

3.1.11 Consulta de reserva activa

Si el usuario desea gestionar su reserva puede iniciar una nueva transacción con el Bot y enviarle el número de cédula del usuario con reserva activa. El Bot automáticamente verificara si existe una reserva asociada a ese número y enviara un mensaje con la información relacionada a esta.

3.1.11.1 Validaciones

- El número de cédula ingresado debe tener una longitud exacta de 10 dígitos.
- El número de cédula ingresado únicamente debe constar de números.
- El número de cédula ingresado debe constar en la base de datos del Bot asociado a una reserva activa.



3.1.12 Modificación de reserva activa

Si el usuario cuenta con una reserva activa este le dará dos opciones para administrar su reserva. La opción “Nada, todo esta perfecto” donde el Bot se despedirá y cerrará la transacción o la opción “¡Deseo cancelarla!” donde el Bot mostrará los pasos a seguir para la cancelación de la reserva.

3.1.12.1 Validaciones

- El usuario debe escoger entre las opciones del teclado emergente, caso contrario se le enviara un mensaje de error y se le mostrara el teclado emergente nuevamente.



3.1.13 Cancelación de reserva activa

Para proceder a la cancelación de la reserva activa, el Bot enviara un código de verificación al correo electrónico asociado a la reserva. El usuario deberá ingresar este código en el Bot, en caso de ser correcto, el Bot informará sobre la eliminación de la reserva y le preguntará al usuario si desea realizar una nueva.

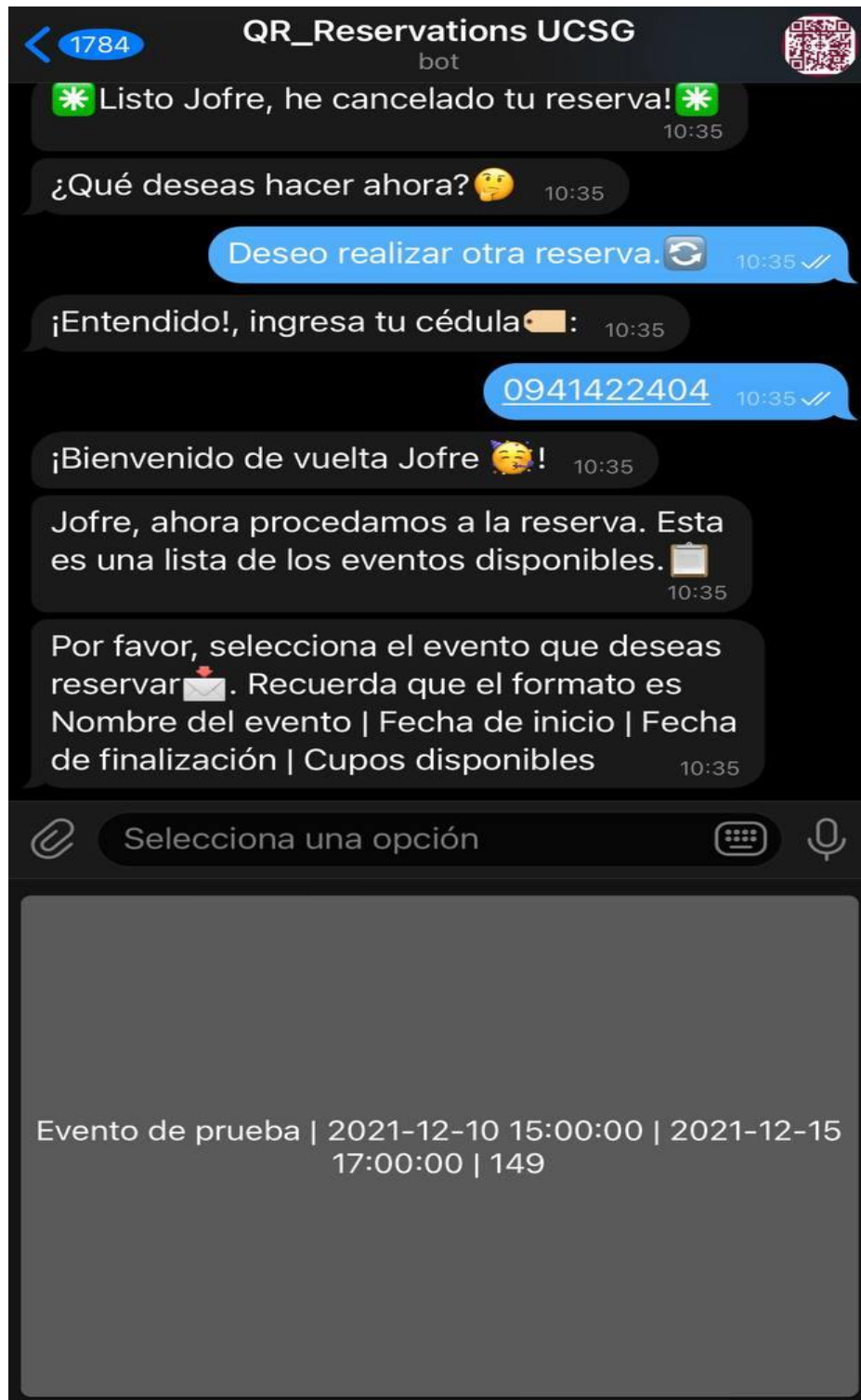
3.1.13.1 Validaciones

- El usuario tendrá 3 intentos para ingresar el código enviado al correo electrónico correctamente, caso contrario se dará por terminada la transacción y no se cancelará la reserva.



3.1.14 Creación de nueva reserva

Una vez cancelada la reserva el Bot le brinda al usuario la posibilidad de generar una nueva reserva. Si el usuario opta por esta opción el Bot preguntara por la cédula de la persona que hará la reserva y todo el proceso arrancara nuevamente siguiendo los lineamientos y validaciones previamente mencionados.



3.2 Panel de administración

El panel de administración del sistema es una herramienta que se le brinda a la persona encargada del control de actividades y datos del sistema. Por medio de este problema se puede generar reportes, editar clientes, verificar reservas activas, agregar actividades, entre otras actividades.

3.2.1 Menú principal

El menú principal del panel de administración es la primera ventana que aparece al iniciar el archivo ejecutable del programa. Cuenta con una lista de botones que redirigen a las distintas funcionalidades del programa.



3.2.2 Reservas activas

El menú de reservas activas muestra una lista de todas las reservas que el Bot ya ha registrado y que les ha generado un QR asociado. Por defecto se muestran únicamente las reservas que aún no han sido activadas usando el lector QR. Sin embargo, la ventana cuenta con una opción que permite ver las reservas que ya fueron activadas en caso de ser necesario.

Se ofrece la opción de filtrar reservas por año de evento por medio de una lista desplegable que reconoce automáticamente los distintos años y basta con seleccionar uno y la lista de reservas activas automáticamente se filtrara.

La ventana cuenta también con opciones útiles como permitir ordenar los valores por cada uno de los parámetros de la tabla, además de permitir transferir los datos a una hoja Excel en caso de ser necesario el manipularlos para usarlos como fuentes de datos.

La ventana también ofrece un botón para imprimir la tabla que se muestra en caso de querer contar con una versión física de la misma.

The screenshot shows a web application window titled 'ReservasActivas'. The main header is 'QR Reservations' with the subtitle 'Facultad de Ingeniería UCSG'. To the right, it says 'LISTA DE LAS RESERVAS ACTIVAS'. Below the header, there are two buttons: 'Convertir a archivo Excel' and 'Imprimir'. To the right of these buttons is a checkbox labeled 'Mostrar reservas pasadas' which is currently unchecked. Next to it is a dropdown menu showing '2023' and 'Todas'. Below this is a table with the following columns: idReserva, idActividad, nombreActividad, cedula, horaInicio, horaFin, horaCierreReserva, and QRCode. The first row of data is highlighted in blue and contains the following values: 5, 385, Prueba 2023, 0941422404, 21/12/2023 10:00, 22/12/2023 10:00, 22/12/2023 10:15, and FFXPBTT30GB2H29V43GXDJAT3. The rest of the table area is greyed out.

idReserva	idActividad	nombreActividad	cedula	horaInicio	horaFin	horaCierreReserva	QRCode
5	385	Prueba 2023	0941422404	21/12/2023 10:00	22/12/2023 10:00	22/12/2023 10:15	FFXPBTT30GB2H29V43GXDJAT3

3.2.3 Reservas canceladas

El menú de reservas canceladas muestra la lista de todas las reservas que fueron canceladas por el usuario y que fueron registradas por Bot. Por medio de este menú se puede obtener

la información de esas reservas y posteriormente ser analizadas por otros medios si así lo desea el administrador del sistema.

Dado que esta es una de las ventanas que más datos maneja se ofrece la posibilidad de filtrar por año y por el nombre de evento para poder contar con un mejor orden.

La ventana cuenta también con opciones útiles como permitir ordenar los valores por cada uno de los parámetros de la tabla, además de permitir transferir los datos a una hoja Excel en caso de ser necesario el manipularlos para usarlos como fuentes de datos.

La ventana también ofrece un botón para imprimir la tabla que se muestra en caso de querer contar con una versión física de la misma.

QR Reservations
Facultad de Ingeniería UCSG

LISTA DE LAS RESERVAS CANCELADAS

Convertir a archivo Excel Imprimir Filtrar evento

idReserva	idActividad	nombreActividad	cedula	horaInicio	horaFin	horaCierreReserva	QRCode	estadoReserva
5	225	PRUEBA ACTIVIDAD	0941422404	4/12/2021 13:00	4/12/2021 15:00	4/12/2021 15:15	VXPJOWQWZSD5KD8403S3U6RDB	0
25	245	Evento de prueba	0941422404	10/12/2021 15:00	15/12/2021 17:00	15/12/2021 17:15	8004R6ATW4DTJUDLEC4MB2CAQ	0
35	245	Evento de prueba	0941422404	10/12/2021 15:00	15/12/2021 17:00	15/12/2021 17:15	XT2UAM1WR6LZR8N4WB9EDXA2U	0
55	245	Evento de prueba	2300494099	10/12/2021 15:00	15/12/2021 17:00	15/12/2021 17:15	X1UJA5RM9KBEC83OUPMIR8GHV	0
65	245	Evento de prueba	0941422404	10/12/2021 15:00	15/12/2021 17:00	15/12/2021 17:15	0ZP9BEDY23XYTF943LX6XE8CG	0
5	245	Evento de prueba	0941422404	10/12/2021 15:00	15/12/2021 17:00	15/12/2021 17:15	1MQTV3B6RAK8PUV89HU48MUT4	0
15	245	Evento de prueba	0941422404	10/12/2021 15:00	15/12/2021 17:00	15/12/2021 17:15	4PV2QIFD97A30GZPY3KM1247	0
5	245	Evento de prueba	0941422404	10/12/2021 15:00	15/12/2021 17:00	15/12/2021 17:15	4FCB0UJ3GQIN0PJ30G2XIFSEK	0
25	265	aaaa	0941422404	14/12/2021 16:52	15/12/2021 16:52	15/12/2021 17:07	EH4OG1YV808WUPHFVJBXT09M0	0
35	265	aaaa	0941422404	14/12/2021 16:52	15/12/2021 16:52	15/12/2021 17:07	20GHVPCDH64DB2V3HT8U8N7JX	0
45	315	Prueba Horarios	0941422404	14/12/2021 20:00	16/12/2021 20:30	16/12/2021 20:45	3UQHK3710WW3FNBQ88NEW4JOL	0
55	335	Evento Prueba 2	0941422404	15/12/2021 15:00	17/12/2021 15:00	17/12/2021 15:15	KUOXPF7016XZE892CC21NRQPCG	0
75	345	Evento Prueba 3	0941422404	15/12/2021 15:00	20/12/2021 15:00	20/12/2021 15:15	JUCCR32JE1PRBJKYE040HX6IE	0
85	345	Evento Prueba 3	0702012733	15/12/2021 15:00	20/12/2021 15:00	20/12/2021 15:15	ML7665PO3QJPEWVAM8K3UGKS	0
115	365	Evento prueba cupos	0941422404	18/12/2021 8:00	19/12/2021 8:00	19/12/2021 8:15	GF902DLU2ZE553DWVEZY8B0T	0
25	385	Prueba 2023	0953545589	21/12/2023 10:00	22/12/2023 10:00	22/12/2023 10:15	8B75KW741P7W5KVFAH1CVRBTP	0
15	375	Prueba 2022	0931198782	21/12/2022 10:00	22/12/2022 10:00	22/12/2022 10:15	3FHUO6S81TOMP54RUZPSPV025	0
35	375	Prueba 2022	0941422404	21/12/2022 10:00	22/12/2022 10:00	22/12/2022 10:15	ZQ2JYF9412ZSN9X513KKOUNOVC	0

3.2.4 Agregar Actividad

El menú de agregar actividad permite generar nuevas actividades, eventos, entre otros, que serán añadidos al Bot y estarán disponibles para reserva.

Los parámetros necesarios para generar actividades para reserva son:

- Nombre de la actividad
- Hora de inicio (incluyendo el día)
- Hora de fin (incluyendo el día)
- Cantidad de cupos disponibles

3.2.4.1 Validaciones

- El nombre de la actividad no debe superar los 100 caracteres.
- La hora y día de inicio y el día y la hora de finalización no debe ser la misma.
- La hora y el día de finalización no debe ser antes que el de inicio.
- El campo de cupos disponibles únicamente debe contener números.
- Ningún campo debe quedar en blanco.

AgregarActividad

QR Reservations

Facultad de Ingeniería UCSG

AGREGAR NUEVA ACTIVIDAD

13/12/2021 19:04:22

Nombre de la actividad

Hora de inicio (yyyy-mm-dd hh:mm:ss)

Hora de finalización (yyyy-mm-dd hh:mm:ss)

Cupos Disponibles

Añadir Actividad

3.2.5 Editar actividades

El menú de editar actividades le permite al administrador del sistema modificar el estado de las actividades disponibles, de manera que así estas puedan ocultarse para ser reservadas por el Bot o cancelarse en caso de ser necesario. Posteriormente si el administrador así lo desea, las actividades que aun tengan una fecha de inicio que aún no haya pasado, pueden ser reactivadas de ser necesario. Si una actividad es cancelada, todos los usuarios que tenían reservas activas dentro de esa actividad serán notificados vía correo electrónico de la cancelación y se les permitirá realizar una nueva reserva por medio del Bot de Telegram. Todas las reservas canceladas podrán ser consultadas mediante la ventana reservas canceladas previamente mencionada.

3.2.5.1 Validaciones

- Todas las actividades que aparezcan dentro de la lista desplegable deberán tener un tiempo y fecha de inicio mayor al tiempo y fecha actual.
- Todas las actividades que aparezcan dentro de la lista desplegable deberán tener un tiempo y fecha de inicio y un tiempo y fecha de finalización entre los cuales se encuentre el tiempo y fecha actual.
- Los eventos activos únicamente podrán ser desactivados.

- Los eventos inactivos únicamente podrán ser reactivados.

Editar_Actividades

QR Reservations

Facultad de Ingeniería UCSG

EDITAR ACTIVIDADES

Buscar Actividad

Nombre de la actividad:

Hora de inicio:

Hora de finalización:

Cupos disponibles:

Estado:

Reactivar Actividad

Desactivar Actividad

3.2.6 Lista de las personas registradas

En el menú de personas registradas se muestra una lista de los datos de todos los usuarios que hayan realizado una reserva alguna vez. De esta manera el administrador puede usar estos datos para contactar a los usuarios o si lo desea usar estos datos para otras campañas.

La ventana cuenta también con opciones útiles como permitir ordenar los valores por cada uno de los parámetros de la tabla, además de permitir transferir los datos a una hoja Excel en caso de ser necesario el manipularlos para usarlos como fuentes de datos.

La ventana también ofrece un botón para imprimir la tabla que se muestra en caso de querer contar con una versión física de la misma.

VisualizarPersonas

QR Reservations

Facultad de Ingeniería UCSG

LISTA DE LAS PERSONAS REGISTRADAS

Convertir a archivo Excel Imprimir

	Cedula	Nombre	Apellido	Celular	Correo	Provincia	Canton
▶	0700461809	Pedro	Aguilar	0982266659	fercastro66@yahoo.com.ar		
	0702012733	Fernando	Castro	0982266659	fercastro66@yahoo.com.ar		
	0909884587	Silvia	Camillo	0997056946	contacto@rolcar.com.ec		
	0913216818	María	Lopez	0949235667	maria.lopezt@gmail.com		
	0931198782	Zamarys	Roldan	0991475007	yosoyjofre@gmail.com		
	0941422404	Jofre	Roldán	0994016357	jofre.roldancamillo@gmail.com	Guayas	Guayaquil
	0953545589	Marcos	Zambrano	0994475681	yosoyjofre@gmail.com		
	2300494099	Christian	Ramirez	0995538213	christian.ramirez01@cu.ucsg.edu.ec		

3.2.7 Actualización de datos del usuario

En el menú de actualización de datos del usuario se le ofrece la posibilidad al administrador del sistema de editar datos de usuario o directamente eliminarlos del sistema.

La ventana tiene todos los controles bloqueados excepto el de ingreso de número de cedula. Para desbloquear los otros controles el administrador deberá ingresar un número válido que ya se encuentre registrado en el sistema y dar clic en “Buscar persona”. Si se encuentra los datos del usuario, se rellenarán los otros campos automáticamente y se habilitarán para ser editados.

Una vez se hayan habilitado los controles el administrador tendrá dos opciones:

- Actualizar información: En caso de existir algún cambio con respecto a la información que se rellenó automáticamente, una ventana emergente preguntándole al administrador si confirma que desea realizar los cambios aparecerá y una vez este acepta se realizarán los cambios en el sistema.
- Eliminar usuario: Si el administrador desea puede dar clic en esta opción y una ventana emergente preguntándole si confirma que desea realizar la eliminación de usuario, en caso de ser afirmativa el usuario será removido del sistema.

En caso de que el administrador desee realizar más de una modificación/eliminación, existe un botón de “limpiar campos” para que rápidamente se limpien todos los campos y pueda continuar con el siguiente.

3.2.7.1 Validaciones

- El número de cedula únicamente debe contener números. Caso contrario se limpiará el cuadro de texto y se mostrará un error por pantalla.
- Al momento de editar un usuario ningún cuadro debe estar vacío, caso contrario se mostrará un error por pantalla y no continuará la transacción.
- La provincia y la ciudad seleccionada deben pertenecerse a sí mismas, caso contrario no se podrá editar al usuario.

The screenshot shows a web application window titled 'ActualizarInformación'. The main heading is 'QR Reservations' and the subtitle is 'ACTUALIZACIÓN DE DATOS DEL USUARIO'. Below the heading, it says 'Facultad de Ingeniería UCSG'. The form contains the following fields and controls:

- Cédula:** A text input field followed by a 'Buscar Persona' button.
- Nombre:** A text input field.
- Apellido:** A text input field.
- Celular:** A text input field.
- Correo:** A text input field.
- Provincia:** A dropdown menu.
- Ciudad:** A dropdown menu.

At the bottom of the form, there are three buttons: 'Limpiar Campos', 'Actualizar Información', and 'Eliminar Usuario'.

3.2.8 Lista de actividades activas

El menú de actividades activas muestra por defecto una tabla con todas las actividades que se encuentren activas al momento de la consulta. Sin embargo, en caso de que el administrador lo requiera, la ventana cuenta con una opción que permite visualizar las actividades pasadas o deshabilitadas.

La ventana cuenta también con opciones útiles como permitir ordenar los valores por cada uno de los parámetros de la tabla, además de permitir transferir los datos a una hoja Excel en caso de ser necesario el manipularlos para usarlos como fuentes de datos.

La ventana también ofrece un botón para imprimir la tabla que se muestra en caso de querer contar con una versión física de la misma.



The screenshot shows a web application window titled "Actividades" with a dark red header. The main content area has a dark red background with the text "QR Reservations" and "Facultad de Ingeniería UCSG". On the right, it says "LISTA DE ACTIVIDADES ACTIVAS". There are two buttons: "Convertir a archivo Excel" and "Imprimir". A checkbox labeled "Mostrar actividades pasadas" is checked. Below the buttons is a table with the following data:


	id	nombreActividad	horaInicio	horaFin	cuposTotales	cuposDisponibles	activa
▶	325	Evento Prueba 1	14/12/2021 20:00	16/12/2021 20:00	100	100	1
	345	Evento Prueba 3	15/12/2021 15:00	20/12/2021 15:00	125	125	1

3.2.9 Consultar actividades por usuario

Le permite al administrador del sistema consultar todas las actividades que haya tomado un usuario en base a su número de cedula. El administrador solo inserta el numero en el campo indicado y el programa automáticamente generara una lista con estos valores.

La ventana cuenta también con opciones útiles como permitir ordenar los valores por cada uno de los parámetros de la tabla, además de permitir transferir los datos a una hoja Excel en caso de ser necesario el manipularlos para usarlos como fuentes de datos.

La ventana también ofrece un botón para imprimir la tabla que se muestra en caso de querer contar con una versión física de la misma.



nombreActividad	horaInicio	horaFin
aaaa	14/12/2021 16:52	15/12/2021 16:52
Evento de prueba	10/12/2021 15:00	15/12/2021 17:00
Evento Prueba 2	15/12/2021 15:00	17/12/2021 15:00
Evento Prueba 3	15/12/2021 15:00	20/12/2021 15:00
Evento prueba cupos	18/12/2021 8:00	19/12/2021 8:00
Prueba 2022	21/12/2022 10:00	22/12/2022 10:00
Prueba 2023	21/12/2023 10:00	22/12/2023 10:00
PRUEBA ACTIVIDAD	4/12/2021 13:00	4/12/2021 15:00
Prueba Horarios	14/12/2021 20:00	16/12/2021 20:30

3.3 Lector de códigos QR

El lector de códigos QR es el encargado de leer los códigos QR generados por el Bot para verificar si una reserva es válida o no. Para realizar este proceso el programa se ejecuta en una raspberry pi que corre el sistema operativo DietPi y realiza la lectura de los códigos QR usando una Pi Camera que tiene conectada en su puerto serial.

3.3.1 Validaciones

- Si la reserva esta correcta, se mostrara un mensaje por pantalla indicando los datos de la reserva incluyendo el número de cédula de la persona que reservo.

- Si la reserva esta correcta y es usada, el estado de la reserva cambiara en la base de datos a “utilizado” o 1.
- Si la reserva es incorrecta o esta usada, se mostrará un mensaje por pantalla indicando que la reserva no es válida junto al mensaje encriptado dentro del código QR mostrado.
- Si se reincide con mostrar la reserva inválida, el sistema simplemente ignorara el código QR.



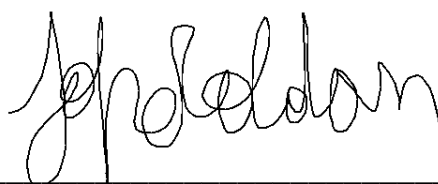
DECLARACIÓN Y AUTORIZACIÓN

Yo, **Roldán Carrillo Jofre Ronny**, con C.C: # **0941422404** autor/a del trabajo de integración curricular: **Diseño e implementación de un sistema de acceso vía móvil, web y escritorio para el control de reservas y aforos usando Bots de Telegram y códigos QR** previo a la obtención del título de **Ingeniero en Ciencias de la Computación** en la Universidad Católica de Santiago de Guayaquil.

1.- Declaro tener pleno conocimiento de la obligación que tienen las instituciones de educación superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de titulación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la SENESCYT a tener una copia del referido trabajo de titulación, con el propósito de generar un repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Guayaquil, 4 de marzo de 2022

f. 

Nombre: **Roldán Carrillo Jofre Ronny**

C.C: **0941422404**

REPOSITORIO NACIONAL EN CIENCIA Y TECNOLOGÍA

FICHA DE REGISTRO DE TESIS/TRABAJO DE TITULACIÓN

TEMA Y SUBTEMA:	Diseño e implementación de un sistema de acceso vía móvil, web y escritorio para el control de reservas y aforos usando Bots de Telegram y códigos QR		
AUTOR(ES)	Roldán Carrillo Jofre Ronny		
REVISOR(ES)/TUTOR(ES)	Castro Aguilar Gilberto Fernando		
INSTITUCIÓN:	Universidad Católica de Santiago de Guayaquil		
FACULTAD:	Ingeniería		
CARRERA:	Ingeniería en Ciencias de la Computación		
TÍTULO OBTENIDO:	Ingeniero en Ciencias de la Computación		
FECHA DE PUBLICACIÓN:	4 de marzo de 2022	No. PÁGINAS:	105
ÁREAS TEMÁTICAS:	Gestión de eventos masivos, automatización y optimización de procesos, mejoras de atención orientadas al usuario.		
PALABRAS CLAVES/KEYWORDS:	Chatbot, Telegram, Nube, Automatización, Python, C#,		
RESUMEN/ABSTRACT:	<p>El presente trabajo de investigación tiene como objetivo principal resolver la necesidad de automatización y modernización de los procesos de obtención de datos, gestión reservas y control de aforos de la facultad de Ingeniería de la Universidad Católica Santiago de Guayaquil mediante el desarrollo de un sistema tecnológico completo, el cual estará conformado por un chatbot desarrollado alrededor del API oficial de Telegram para atender las distintas peticiones y cancelaciones de reservas por parte de los usuarios mediante el uso de códigos QR, además de usarse como herramienta de obtención de datos personales del mismo, adicionalmente estará conformado de un software desarrollado en el lenguaje de programación Python que se ejecutara en una raspberry pi conectada a una pi camera, el cual tiene el objetivo de leer los códigos QR generados por el Bot con la finalidad de reconocer las reservas activas y obtener rápidamente información sobre la validez de las mismas, también contará con un panel de administración donde se podrá revisar de manera manual el estado de las reservas, gestionar los usuarios y obtener reportes de información, entre otras funcionalidades. La implementación de estas tecnologías otorgará a la facultad de Ingeniería de la Universidad Católica Santiago de Guayaquil una mejor administración de los eventos y servicios ofertados además de permitir la creación de un banco de datos que posteriormente podrá ser usado para la creación de nuevas campañas u ofertas que este ente desee ofrecer.</p>		
ADJUNTO PDF:	<input checked="" type="checkbox"/> SI	<input type="checkbox"/> NO	
CONTACTO CON AUTOR/ES:	Teléfono: +593994016357	E-mail: jofre.roldan@cu.ucsg.edu.ec	
CONTACTO CON LA INSTITUCIÓN (COORDINADOR DEL PROCESO UTE):	Nombre: Toala Quimí, Edison José		
	Teléfono: +593-9-90976776		
	E-mail: edison.toala@cu.ucsg.edu.ec		
SECCIÓN PARA USO DE BIBLIOTECA			
Nº. DE REGISTRO (en base a datos):			
Nº. DE CLASIFICACIÓN:			
DIRECCIÓN URL (tesis en la web):			