



INSTITUTO POLITÉCNICO NACIONAL



ESCUELA SUPERIOR DE CÓMPUTO

INGENIERÍA DE SOFTWARE

Proyecto: Asistente Turístico de Sitios Culturales
(guiAR)

Nombres de los Integrantes:

- ✓ Contreras Méndez Brandon
- ✓ Díaz Matus Ricardo
- ✓ Martínez Olivares Vicente Jafet
- ✓ Reyes Rojas Roberto

Grupo: 3CM11

ÍNDICE GENERAL

Capítulo 1: IDENTIFICACIÓN E INTRODUCCIÓN AL SISTEMA	5
Objetivo General	5
Objetivos Específicos	5
Descripción	5
Funciones Principales y Aspecto Innovación	5
Justificación (Complejidad; social e innovación)	6
Paradigma de desarrollo	7
Estado del Arte	8
Capítulo 2: OBTENCIÓN DE LA INFORMACIÓN	9
Recolección de Requisitos	9
Cuestionario y Muestreo	9
Capítulo 3: ADMINISTRACIÓN DEL PROYECTO DE SOFTWARE	12
II. ANÁLISIS DE RIESGOS	12
Riesgo del Software y su Clasificación	12
Plan RSGR	16
III. MÉTRICAS Y ESTIMACIÓN	18
Métrica basada en funciones	19
Cálculo de los Puntos de Función	20
Valor de Ajuste de la Complejidad	21
Modelo COCOMO II	22
IV. AGENDA (Diagrama de Gantt & Diagrama de PERT)	25
V. RECURSOS (Hardware, Software & Personal)	28
VI. ORGANIZACIÓN DEL PERSONAL	37
Descentralizado democrático (DD)	37
Descentralizado controlado (DC)	37
Centralizado controlado (CC)	37
VII. MECANISMOS DE SEGUIMIENTO Y CONTROL	39
Manejo de la Calidad del Proyecto	39

Capítulo 4: ANÁLISIS Y DISEÑO DEL SISTEMA ORIENTADO A OBJETOS, MEDIANTE EL USO DE UML	42
VIII. METODOLOGÍA PROPUESTA DE DESARROLLO	42
Análisis de Flujo de Datos de Gane & Searson	43
Metodología Ágil “Scrum”	47
IX. ANÁLISIS Y DISEÑO DEL SISTEMA Y MODELADO CON UML	50
Requisitos Funcionales	50
Requisitos No funcionales	50
Modelado de Casos de Uso de la aplicación	50
Reglas del Business	54
Modelado con UML	55
Diagrama de Caso de Uso “guiAR”	55
Diagrama de Clases “guiAR”	56
Diagrama de Secuencia “guiAR”	56
Diagrama de Estados “guiAR”	57
Diagrama de Actividades “guiAR”	58
Capítulo 5: ADMINISTRACIÓN DE LA CALIDAD	59
X. CALIDAD DEL PROYECTO DE SOFTWARE “guiAR”	59
XI. NORMAS ISO RELACIONADAS AL PROYECTO “guiAR”	63
ISO/IEC-9126 Modelo De Calidad Del Producto De Software.....	63
ISO/IEC-14598 Proceso De Desarrolladores	63
ISO/IEC 2501 División De Requisitos De Calidad	63
ISO 9000 Sistemas De Gestión De La Calidad	64
ISO 21503 División De Modelo De Calidad	64
XII. NORMAS IEEE RELACIONADAS AL PROYECTO “guiAR”	65
830-1998-IEEE Documentación de prueba de software	65
1016-IEEE Actividades en el proceso de las pruebas de software	65
1471-IEEE Arquitectura De Software De Sistemas Intensivos	65
1233-1998 – IEEE Desarrollo/especificaciones de los requisitos del sistema ...	65
1012-1998 - IEEE Verificación y validación de procesos de software	65

Capítulo 6: PROTOTIPADO, COMPOSICIÓN E INTEGRACIÓN DEL SISTEMA	66
XIII. MODELADO Y DISEÑO DEL SISTEMA	66
Planificación del Proceso	66
Entrada, Proceso y Salida del Sistema	66
Prototipo de Prediseño de la Aplicación Móvil “guiAR”	67
Prototipo de Diseño Final de la Aplicación Móvil “guiAR”	71
Capítulo 7: ASEGURAMIENTO E IMPLEMENTACIÓN DE LA CALIDAD	73
XIV. IMPLEMENTACIÓN DE CALIDAD/ SATISFACTORIA	73
Creación del Centro de Información	73
Topología de Red	75
Diagrama de Estructura para diseñar sistemas modulares	77
Modelado de red	78
Creación de un diagrama de conectividad de hub	79
Capacitación de los Usuarios	80
Conversión del Sistema	81
Evaluación del Sistema	82
Modelo de Negocio	84
Business Model Canvas o Modelo de Negocio CANVAS	84
Capítulo 8: INTERFAZ, IMPLEMENTACIÓN Y VALIDACIONES	86
XV. DESARROLLO, EJECUCIÓN Y PRUEBAS	86
Algoritmo empleado para la Aplicación Móvil “guiAR”	86
Monetización del proyecto de software “guiAR”	92
App gratuita con publicidad in-App	92
Paid o Pago por descarga	94
Capítulo 9: ASPECTO DE MERCADOTECNIA	97
Logo de la Aplicación.....	97
Slogan	97
Infografía	98
Tríptico	99
Folleto	100

Licencia	101
Contrato	103
Póliza de Garantía	106
ANEXOS	107
Técnica De Recolección De Requisitos	107
Obtención De Resultados A Partir Del Cuestionario	109
Cálculos	112
Referencias Bibliográficas	114

Capítulo 1: IDENTIFICACIÓN E INTRODUCCIÓN AL SISTEMA

Asistente Turístico de Sitios Culturales “guiAR”

Ámbito

Primordialmente para el sector turístico y cultural.

Objetivo General

Desarrollar una herramienta de software mediante el reconocimiento de imágenes de los lugares turísticos más importantes del Centro Histórico de la Ciudad de México.

Objetivos Específicos

1. Implementar un sistema de reconocimiento de imágenes
2. Emplear audios en los principales idiomas para la descripción de los lugares.
3. Guiar a todas aquellas personas extranjeras o de provincia a conocer los lugares más importantes de la Ciudad de México.
4. Facilitar la información necesaria de cada sitio turístico implementado.

Descripción

“guiAR” es una aplicación móvil con reconocimiento de imágenes de diversos sitios culturales, históricos e importantes de la Ciudad de México, el cuál despliega información resumida relevante e interesante, cómo también en audios en los principales idiomas, tales como el Inglés, Chino, Español, etc.....

Funciones Principales y Aspecto Innovación

La implementación de este software es para móviles, estando en alguna parte de la Ciudad de México, y haya alguna atracción o sitio turístico importante. Mediante el uso de la cámara trasera del móvil se debe enfocar a dicha sitio y se despliega información relevante e importante, cómo algunos datos históricos y de la misma manera una opción con audios en los principales idiomas para conocer de forma rápida la información acerca de este sitio.

Justificación (Complejidad; social e innovación)

La realización de software impacta directa y benéficamente al sector turístico y cultural puesto que la información de cada sitio turístico debe ser transparente y con claridad, al desarrollar este software el usuario le bastará abrir su cámara, apuntar hacia el lugar y tendrá la información al instante, de esta manera el cliente no tendrá problemas con estar buscando en diferentes sitios los datos interesantes, mapa, etc., del sitio turístico donde se encuentre.

El complejo por resolver con esta propuesta de Proyecto es con el uso de las tecnologías, cómo es el reconocimiento de imágenes con la ayuda de la cámara trasera de un smartphone, se lleve a cabo la implementación innovadora de un asistente, acerca de los sitios turísticos-culturales más importantes y mejor visitados del Centro Histórico.

Lo anterior deriva, dado que hemos visto de manera personal y vivencial, al momento de asistir en algún momento de nuestra etapa a visitar uno o diversos sitios culturales, o en su defecto turísticos, el comportamiento de personas extranjeras, cuando se encuentran enfrente de lo antes mencionado y no saben absolutamente nada, estén perdidos o confundidos por estar en un lugar que no es país de origen, no comprenden o dominan el idioma español, por lo que la comunicación en su idioma natal es complicado establecer un contacto de confianza con alguna persona residente de la Ciudad de México y que por ende sabe todo lo relacionado de ese lugar.

Cabe destacar que existen sistemas (aplicaciones móviles), tales como son Google Lens, SMIITY y Word Around Me, que poseen el mismo enfoque de ofrecer información (No completa y muy básica), datos (no muy relevante), y la función de visualizar lo que hay alrededor acerca de lugares turísticos en tiempo real, entre otras funciones parecidas a las que ofrece Google Maps, para todos aquellos visitantes, viajeros, conocedores, turistas. Sin embargo, sólo están disponibles e implementadas para países de Europa y Estados Unidos de América. Y para nuestro país, México no está disponible y por ende ninguno que ofrezca información sobre los diversos lugares turísticos y culturales que existen a lo ancho, de la República Mexicana algún sistema, asistente, aplicación para móviles o similares.

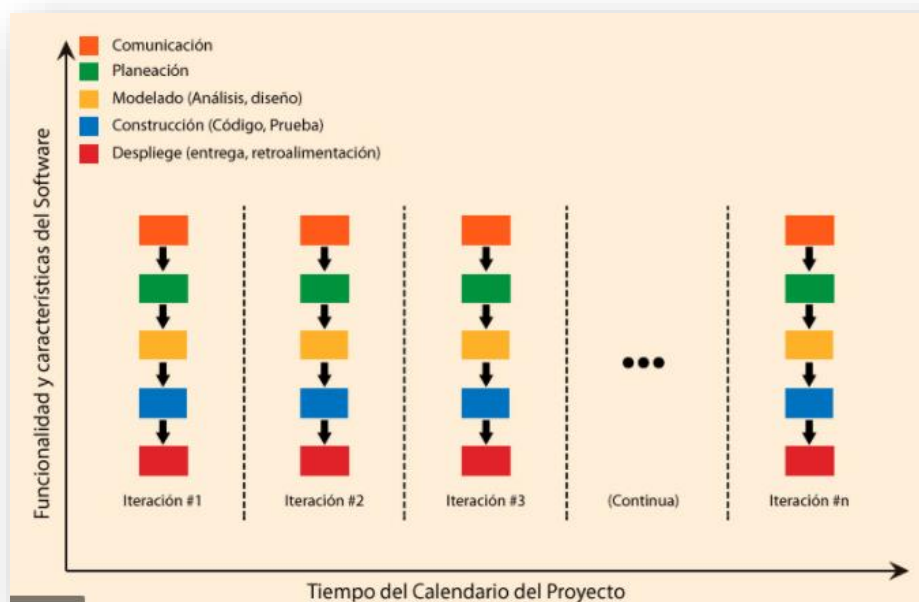
Por lo que decidimos dar una tentativa de solución y mejora de lo existente, que es una aplicación móvil, pero enfocado en guiar y facilitar información necesaria e interés, de cada sitio turístico implementado. a todas aquellas personas extranjeras o de provincia a conocer los lugares más importantes de la Ciudad de México.

Paradigma de desarrollo

Se ha escogido el modelo Iterativo e Incremental, ya que es el mejor paradigma de acuerdo con las necesidades que posee el equipo, es decir, mediante retroalimentación del cliente (profesor), el equipo podrá ir corrigiendo e implementando nuevas funciones al proyecto inicial, para así, poder entregar una versión final de calidad.

Las ventajas del presente paradigma Iterativo e Incremental, para la realización del Proyecto en turno, se enlistan a continuación:

- ⇒ Se reduce el tiempo de desarrollo inicial, ya que se implementa la funcionalidad parcial.
- ⇒ Se puede gestionar de manera natural los cambios que van apareciendo durante el proyecto. La finalización de cada iteración es el lugar natural donde el cliente puede proporcionar su retroalimentación, tras examinar el resultado obtenido
- ⇒ Proporciona todas las ventajas del modelo en cascada realimentado, reduciendo sus desventajas sólo al ámbito de cada incremento.
- ⇒ Los modelos iterativos e incrementales disminuyen riesgos. ya que estos modelos se basan en la retroalimentación sobre los avances.
- ⇒ Resulta más sencillo acomodar cambios al acotar el tamaño de los incrementos
- ⇒ Se puede gestionar las expectativas del cliente (requisitos desarrollados, velocidad de desarrollo, calidad) de manera regular, puede tomar decisiones en cada iteración.



Estado del Arte

No.	Sistema	Características
1	Google Lens	<ol style="list-style-type: none"> 1. Disponible para dispositivos Android y iOS. 2. Ofrece información sobre el lugar, como datos curiosos del monumento. 3. Ofrece diferentes datos del lugar como los horarios de apertura o la información de contacto. 4. Es gratuito. 5. Solo es necesario abrir la cámara del dispositivo, seleccionar el icono de "lens" y seleccionar el filtro de lugares (solo en dispositivos Android).
2	Smiity	<ol style="list-style-type: none"> 1. Disponible para dispositivos Android y iOS. 2. Disponible en inglés y portugués. 3. Los usuarios pueden conocer noticias, eventos, senderos, puntos de interés, eventos especiales, etc. 4. Tiene la función de realidad aumentada que permite visualizar los puntos de interés que se encuentran alrededor, como información del lugar, distancia y rutas de acceso. 5. No es necesario tener conexión a internet (pero se necesita descargar con anticipación todo el contenido). 6. Es gratuita.
3	World Around Me (WAM)	<ol style="list-style-type: none"> 1. Disponible para dispositivos Android y iOS. 2. Se pueden realizar búsquedas por voz. 3. Brinda información como reseñas de usuarios, datos de contacto, rango de precios, etc. 4. Se puede elegir ver los resultados en realidad aumentada, en una lista o en un mapa para navegar. 5. Se puede obtener la distancia a un punto de interés actualizada en tiempo real mientras se camina. 6. Cuenta con una versión de pago.

Capítulo 2: OBTENCIÓN DE LA INFORMACIÓN

Recolección de Requisitos

En este apartado se aplicarán dos técnicas de recolección de requisitos, cómo él es cuestionario y el muestreo, este último se hablará posteriormente, después de priorizar la técnica del cuestionario, la revisión de los sitios/lugares turísticos del Centro Histórico de la Ciudad de México más concurridos, sin embargo, para este momento se comenzó a cuestionar si estas técnicas eran las mejores para el proyecto, por lo que se planteó una nueva técnica, el cuestionario. Con la pregunta uno, se pretende saber el rango de edad de las personas que serían los potenciales clientes de nuestra app, la segunda pregunta será para saber a qué género le interesaría más una app como la nuestra. La tercera pregunta será una especie de pregunta filtro, a aquellas personas que no sepan que es realidad virtual se les proporcionará una breve explicación acerca de esta tecnología. De la pregunta 4 a la 6 se tiene el objetivo de saber qué Sistema Operativo y que versión del mismo conviene desarrollar la app. Las preguntas 7 y 8 tienen como propósito saber que tan buena o mala fue la experiencia de los usuarios al usar apps con tecnología de AR. De las preguntas 9 a 18 tienen como objetivo saber las preferencias de los potenciales usuarios, así como saber si está bien planteada la problemática a resolver.

Las preguntas 19 a 20 se hacen para saber si los potenciales usuarios estarían dispuestos a pagar por nuestra app y cual sería lo máximo que pagarían por ella, esto sirve para saber cuál es la mejor forma para financiar la app y si puede llegar a ser rentable.

A la par del cuestionario, aplicaremos la técnica del muestreo.

Primero determinaremos los datos que deberán ser recopilados, esto lo haremos con la ayuda de nuestro cuestionario.

Nos interesa la información básica como lo es la edad y el género de nuestros potenciales clientes, a su vez también es importante saber la versión del S.O de los potenciales usuarios, esto para que la aplicación pueda ser usada por la mayor cantidad de dispositivos posibles, dada la cantidad de usuarios del S.O y de las limitaciones del mismo.

Además, dado que la aplicación será desarrollada en base a la Ciudad de México, nos interesa saber ciertos datos como la frecuencia con la que los usuarios visitan la ciudad, si han tenido dificultades al momento de buscar lugares y cuáles son los lugares que más les interesan de acuerdo con el propósito de la aplicación.

Otra cosa que nos interesa saber es si las características propuestas de la aplicación son del agrado del público, esto servirá para agregar, modificar o incluso descartar alguna de esas ideas y ofrecerle al usuario la mejor experiencia posible.

Por último, pero no menos importante, nos interesa saber si las personas están dispuestas a pagar por una aplicación como la nuestra y en caso de que sí, saber cuánto es lo máximo que estarían dispuestas a pagar, es importante saber esto para determinar la forma de financiar el proyecto. El siguiente paso será determinar la población de la cual obtendremos las muestras. Esta información la obtendremos de la página de la secretaria de turismo de la Ciudad de México.

De acuerdo con los principales indicadores turísticos, la cantidad de turistas en el año 2019 es de:

Turistas Nacionales: 10,150,039

Turistas Internacionales: 3,755,044

Total de turistas en la CDMX: 13,905,083

Ahora, la misma página de la secretaria, ofrece información sobre los perfiles de los turistas que visitaron la CDMX, (los datos más actualizados de esta información corresponden al año 2017). A continuación, enlistaremos la información más relevante para el proyecto del perfil prototípico del turista que visita la CDMX.

En general el turista que visita la CDMX...

- 1) Es un adulto joven; tanto hombres (49%), como mujeres (51%), con edad promedio de 38 años.
- 2) Cuenta con acceso, uso y apropiación de internet (89%), (86% turistas nacionales y 99% turistas internacionales); lo utiliza prácticamente para todo, incluso para la consulta de información turística (76% para turistas nacionales y 92% turistas internacionales) y la planeación de sus viajes (65% turistas nacionales y 90% turistas internacionales).
- 3) El principal motivo para visitar CDMX es por vacaciones (33%), seguido por visitar a familiares y amigos (21%); en menor medida por trabajo o negocios (13%).
- 4) Lo que más gusta a los visitantes de la CDMX es su Patrimonio Histórico y Cultural (36%), su Arquitectura y Urbanismo (20%), así como la Diversidad Turística (14%) y su Gastronomía (14%).
- 5) Al turista que viaja a CDMX le gusta pasear; declara que durante su estancia le apetece Visitar Lugares Turísticos (24%), Conocer la Ciudad (19%) y Visitar Museos (18%).
- 6) El gasto promedio diario de los turistas es de \$788 MXN.
- 7) La mayoría de los turistas internacionales vienen del continente americano (78%). El país que genera más visitantes a CDMX es Estados Unidos (25%).
- 8) Un 35% son viajeros que han visitado anteriormente CDMX.

9) El gasto promedio diario de los viajeros foráneos es de \$2,058 MXN.

Tenemos varios datos que nos van a servir para nuestro proyecto, pero para tomar el tamaño de la muestra, ocuparemos el perfil de las personas con acceso, uso y apropiación de internet, que lo usen para la consulta de información turística, que son el 76% para turistas nacionales y el 92% para turistas internacionales.

Usamos este dato ya que supondremos que, al tener acceso a internet, esta población tendrá un smartphone con el cual puedan consultar la información turística. Tomando en cuenta la cantidad de turistas que visitaron la CDMX en 2019, tenemos que los turistas con acceso, uso y apropiación de internet, que lo usen para la consulta de información turística es de:

Turistas nacionales: 7,714,028.

Turistas internacionales: 3,454,639.

Total de turistas: 11,168,667. Este será el tamaño de la población.

El siguiente paso será tomar un margen de error de ± 0.02 , un nivel de confianza de 95% y la desviación estándar que, dado que aún no hemos realizado una encuesta, se toma un valor sugerido de 0.5, que garantiza que el tamaño de la muestra es lo suficientemente grande.

Ahora se procede a calcular el tamaño de la muestra, con la siguiente formula:

$$\text{Tamaño de la muestra} = \frac{(Z) * (\text{Desviación estándar}) * (1 - \text{Desviación estándar})}{(\text{Margen de error})}$$

Donde z se obtiene de acuerdo con el nivel de confianza de acuerdo con los siguientes valores.

90 % $\rightarrow Z = 1,645$

95 % $\rightarrow Z = 1,96$

99 % $\rightarrow Z = 2,576$

Sustituyendo los datos tenemos que el tamaño de la muestra debe de ser de:

Tamaño de la muestra = 385 personas

El propósito de las técnicas de recolección propuestas no es más que nada que medir la necesidad de implementación y por ende creación de la aplicación, así como tener conocimiento si de lado de la comunidad la descargará, en sus dispositivos móviles.

Capítulo 3: ADMINISTRACIÓN DEL PROYECTO DE SOFTWARE

II. ANÁLISIS DE RIESGOS

El análisis y la administración del riesgo son acciones que ayudan al equipo de software a entender y manejar la incertidumbre, el riesgo se preocupa por los acontecimientos futuros.

Los pasos para identificar los riesgos son:

- ✓ **Identificación de riesgos:** Es reconocer qué puede salir mal.
- ✓ **Clasifican los riesgos:** Después de analizar, se determina la probabilidad e impacto.
- ✓ **Plan de Anti-Riesgo:** Se desarrolla un plan para manejar aquellos que tengan alta probabilidad y alto impacto.

Riesgo de Software

Cuando se analizan los riesgos es importante cuantificar el nivel de incertidumbre y el grado de pérdidas asociados con cada riesgo.

Clasificación del Riesgo del Software.

- ↳ **Riesgos del Proyecto:** Amenazan el plan del proyecto, es decir, si los riesgos del proyecto se vuelven reales, es probable que el calendario del proyecto se deslice y que los costos aumenten. Los riesgos del proyecto identifican potenciales problemas de presupuesto, calendario, personal (tanto técnico como en la organización), recursos, participantes y requisitos, así como su impacto sobre un proyecto de software.
- ↳ **Riesgos Técnicos:** Amenazan la calidad y temporalidad del software que se va a producir. Si un riesgo técnico se vuelve una realidad, la implementación puede volverse difícil o imposible e identifican potenciales problemas de diseño, implementación, interfaz, verificación y mantenimiento.
- ↳ **Riesgos Empresariales:** Amenazan la viabilidad del software que se va a construir y con frecuencia ponen en peligro el proyecto o el producto
- ↳ **Riesgos Conocidos:** Son aquellos que pueden descubrirse después de una evaluación cuidadosa del plan del proyecto, del entorno empresarial o técnico donde se desarrolla el proyecto y de otras fuentes de información confiables.
- ↳ **Riesgos Predecibles:** Infieren con la experiencia en proyectos anteriores como la rotación de personal, pobre comunicación con el cliente, disolución del esfuerzo del personal conforme se atienden las solicitudes de mantenimiento.

- ✍ **Riesgos Impredecibles:** Son el comodín en la baraja. Pueden ocurrir y lo hacen, pero son extremadamente difíciles de identificar por adelantado.
- ✍ **Riesgos Genéricos:** Son una amenaza potencial a todo proyecto de software.
- ✍ **Riesgos Específicos:** Se identifican y ocurren solamente por quienes tienen clara comprensión de la tecnología, el personal y el entorno específico del software que se construye.

Después que se explicó la definición del Análisis de Riesgos cómo su respectiva clasificación, toca mencionar que nuestro Proyecto “guiAR”, recae en 4 posibles tipos de Riesgo, cómo son el técnico, genéricos, específicos e impredecibles, cómo el primero, dado que al momento de realizar la respectiva implementación y desarrollo de la aplicación pueden llegar a complicarse en algunas partes tales cómo el diseño, interfaz, verificación, pruebas, etc.... el segundo es un riesgo que siempre estará presente en cualquier proyecto de software a realizar, puesto que surgirán diversas amenazas y complejos, difíciles de poder controlar, sino se lleva a cabo una muy buena planeación y documentación, cómo su correspondiente implantación, el tercero va relacionado si y sólo sí al analizar el plan del proyecto, que rubros que lo integran son o se consideran puntos débiles que requerirán una reformulación o en su defecto un cambio drástico, para que de esa manera no afecte a toda la planeación completa del proyecto en turno. Y el cuarto está presente casi al momento de concluir toda la planeación del proyecto o a la mitad del mismo, puesto que son muy complicados de identificar a simple vista o análisis, al inicio del plan del proyecto.

El impacto de cada promotor de riesgo sobre el componente de riesgo se divide en una de 4 categorías de impacto:

- **Catastrófico:** La falla para satisfacer el requisito resultaría en fallo en la misión. Degradación significativa para no lograr el rendimiento técnico y Software que no responde o no puede tener apoyo.
- **Crítico:** Falla para satisfacer el requisito degradaría el rendimiento del sistema hasta un punto donde el éxito de la misión sería cuestionable. Cierta reducción en rendimiento técnico y Demoras menores en modificaciones de software.
- **Despreciable:** Falla para satisfacer requisitos crearía inconvenientes o impacto no operativo. No reducción en rendimiento técnico y Software fácilmente soportable.
- **Marginal:** Falla para satisfacer los requisitos resultaría en degradación de misión secundaria Reducción mínima a pequeña en rendimiento técnico y Apoyo de software receptivo.

En la siguiente tabla, se visualiza el análisis de riesgos, el cual especifica los diversos riesgos, dividiéndolos por categoría, la probabilidad que sucedan o que se lleguen a presentar, el impacto que tendrá a corto plazo en el presente proyecto que se desarrollará, y un plan de respaldo, en el caso de que llegue a ocurrir, dar a posibles soluciones.

Análisis de Riesgo					
ID	Riesgo	Categoría	Probabilidad	Impacto	Plan RSGR
R-01	Deficiencia en la comunicación con los integrantes del equipo.	PD	70%	2	Hacer un calendario de actividades y tener constante comunicación.
R-02	Problemas en la salud en alguno de los integrantes del equipo.	SC	15%	2	Distribuir las actividades restantes entre los demás integrantes.
R-03	Los integrantes no posean el mismo conocimiento en los lenguajes y herramientas de desarrollo.	SC	30%	3	Hablarlo en conjunto con el equipo de trabajo las debilidades y fortalezas de cada uno de los integrantes y llegar a acuerdos flexibles.
R-04	Limitante en los entregables, cómo la entrega final de la aplicación funcional.	BI	65%	2	Solicitar al cliente paciencia y no generar presiones.
R-05	La PC y/o laptop, no cuenten con los suficientes recursos necesarios.	DE	30%	3	Comunicar y establecer los requisitos sobre el software a ocupar , antes de iniciar el desarrollo.
R-06	Los integrantes no dispongan de tiempo prolongado para la realización del Proyecto.	PD	55%	1	Comunicarlo con los demás integrantes y llegar acuerdos para no generar atrasos en el Proyecto.

R-07	Problemas y complejos al momento de la codificación e implementación de la aplicación.	DE y TB	80%	1	Hacer una reunión del equipo completo para dar posibles soluciones, para la resolución a las dificultades presentadas.
R-08	Problemas con la energía eléctrica e Internet en algunos de los integrantes.	DE	25%	1	Distribuir las actividades restantes entre los demás integrantes.
R-09	La conexión del hardware sufra algún accidente.	DE	10%	2	Tener algún soporte del hardware de forma inmediata.
R-10	El usuario final tenga dificultades para utilizar la aplicación.	BI	5%	3	Brindarle al usuario un manual audiovisual.

Valores de Impacto: 1—catastrófico 2—crítico 3—marginal 4—despreciable

Categorías de los Riesgos: BI (Impacto del Negocio), DE (Medio de Desarrollo), SS (Capacitación y Tamaño del Personal), PD (Definición del Proceso), PS (Tamaño del Producto), SC (Características de los Participantes), TB (Tecnología a Crear).

Ahora bien, para justificar y comprender la columna, los valores de probabilidad de ocurrencia y los valores de impacto de la tabla anterior. A continuación se visualiza una en el siguiente cuadro, los valores de probabilidad de ocurrencia. Donde dichos valores de probabilidad de ocurrencia utilizados en la tabla de análisis de riesgos, el 1% es la probabilidad de ocurrencia muy bajo y el 99% es la probabilidad de ocurrencia muy alto. Cabe mencionar que el porcentaje de 0% y el porcentaje del 100% no son tomados como valores de probabilidad.

Probabilidad del Análisis de Riesgo	
Porcentaje de Probabilidad	Indicador o Nivel de Factor
1% ~ 15%	Muy Bajo
16% ~ 29%	Bajo
30% ~ 44%	Bajo-Medio
45% ~ 55%	Medio
56% ~ 70%	Medio Alto
71% ~ 89%	Alto
90% ~ 99%	Muy Alto

Plan RSGR

En esta sección, se ejemplifican la ficha de información de los riesgos con sus respectivos planes de RSGR especificados. El motivo por el que se tienen que considerar, es en caso de que algún riesgo llegase a suceder, para que así no nos tome de sorpresa o de manera imprevista a tal grado de poder disminuir y erradicar los riesgos enlistados en la tabla de la sección anterior y así llevar a cabo el desarrollo e implementación del proyecto en turno.

Ficha de Información de Riesgo		
ID: R-01	Probabilidad: 70%	Impacto: Crítico
Riesgo: Deficiencia en la comunicación con los integrantes del equipo.		
Alternativas a seguir: <ul style="list-style-type: none">✓ Hacer uso de medios electrónicos, para designar actividades, dar avisos, enviar avances, etc. entre los integrantes del equipo.✓ Realizar un cronograma de actividades entre los integrantes del equipo.✓ Realizar reuniones virtuales remotas para tomar decisiones democráticas.		

Ficha de Información de Riesgo		
ID: R-02	Probabilidad: 15%	Impacto: Crítico
Riesgo: Problemas en la salud en alguno de los integrantes del equipo.		
Alternativas a seguir: <ul style="list-style-type: none">✓ Distribuir las actividades restantes entre los demás integrantes.✓ Avisar y mantener contacto previo con los integrantes del equipo sobre la situación de salud del compañero afectado.		

Ficha de Información de Riesgo		
ID: R-03	Probabilidad: 30%	Impacto: Marginal
Riesgo: Los integrantes no posean el mismo conocimiento en los lenguajes y herramientas de desarrollo		
Alternativas a seguir: <ul style="list-style-type: none">✓ Dialogar en conjunto con el equipo de trabajo, las debilidades y fortalezas de cada uno de los integrantes.✓ Tomar algún curso “relámpago”, vía plataformas online u otros medios.		

Ficha de Información de Riesgo		
ID: R-04	Probabilidad: 65%	Impacto: Crítico
Riesgo: Limitante en los entregables, cómo la entrega final de la aplicación funcional.		
Alternativas a seguir: <ul style="list-style-type: none">✓ Solicitar al cliente paciencia y no generar presiones.✓ Dividir la carga de trabajo entre los integrantes del equipo, mediante los rubros, tópicos que se hayan comprendido de la mejor manera.		

Ficha de Información de Riesgo		
ID: R-05	Probabilidad: 30%	Impacto: Marginal
Riesgo: La PC y/o laptop, no cuenten con los suficientes recursos necesarios.		
Alternativas a seguir: <ul style="list-style-type: none"> ✓ Establecer los requisitos sobre el software a ocupar, antes de iniciar el desarrollo. ✓ Llevar a cabo las actualizaciones en el sistema pertinentes para un funcionamiento óptimo de la PC y/o laptop. ✓ Verificar las características mínimas necesarias para el funcionamiento del software. 		

Ficha de Información de Riesgo		
ID: R-06	Probabilidad: 55%	Impacto: Catastrófico
Riesgo: Los integrantes no dispongan de tiempo prolongado para la realización del Proyecto.		
Alternativas a seguir: <ul style="list-style-type: none"> ✓ Comunicarlo con los demás integrantes y llegar acuerdos para no generar atrasos en el Proyecto. 		

Ficha de Información de Riesgo		
ID: R-07	Probabilidad: 80%	Impacto: Catastrófico
Riesgo: Problemas y complejos al momento de la codificación e implementación de la aplicación.		
Alternativas a seguir: <ul style="list-style-type: none"> ✓ Hacer una reunión virtual remota con el equipo completo para dar posibles soluciones, para la resolución a las dificultades presentadas. 		

Ficha de Información de Riesgo		
ID: R-08	Probabilidad: 10%	Impacto: Marginal
Riesgo: Problemas con la energía eléctrica e Internet en algunos de los integrantes.		
Alternativas a seguir: <ul style="list-style-type: none"> ✓ Distribuir las actividades restantes entre los demás integrantes. 		

Ficha de Información de Riesgo		
ID: R-09	Probabilidad: 5%	Impacto: Marginal
Riesgo: La conexión del hardware sufra algún accidente.		
Alternativas a seguir: <ul style="list-style-type: none"> ✓ Tener algún soporte del hardware de forma inmediata. ✓ Revisar que no existan daños físicos en la conexión. ✓ Verificar que el sistema se encuentre en óptimas condiciones para un buen funcionamiento. 		

Ficha de Información de Riesgo		
ID: R-10	Probabilidad: 70%	Impacto: Crítico
Riesgo: El usuario final tenga dificultades para utilizar la aplicación.		
Alternativas a seguir: <ul style="list-style-type: none"> ✓ Brindarle al usuario un manual audiovisual. ✓ Informarle de manera clara y concisa acerca de cómo se debe operar el sistema, de esta manera se evitan confusiones por parte del usuario. 		

III. MÉTRICAS Y ESTIMACIÓN

Un elemento clave de cualquier proceso de ingeniería es la medición. Pueden usarse medidas para entender mejor los atributos de los modelos que se crean y para valorar la calidad de los productos o sistemas sometidos a ingeniería que se construyen.

Las métricas de producto ayudan a los ingenieros de software a obtener comprensión acerca del diseño y la construcción del software que elaboran, al enfocarse en atributos mensurables específicos de los productos de trabajo de la ingeniería del software.

Métrica basada en funciones

La métrica de punto de función (PF) puede usarse de manera efectiva como medio para medir la funcionalidad que entra a un sistema. Al usar datos históricos, la métrica PF puede entonces usarse para:

- 1) *Estimar el costo o esfuerzo requerido para diseñar, codificar y probar el software;*
- 2) *Predecir el número de errores que se encontrarán durante las pruebas, y*
- 3) *Prever el número de componentes y/o de líneas fuente proyectadas en el sistema implementado.*

Los puntos de función se derivan usando una relación empírica basada en medidas contables (directas) del dominio de información del software y en valoraciones cualitativas de la complejidad del software. Los valores de dominio de información se definen en la forma siguiente:

- **Número de entradas externas (EE).** Cada entrada externa se origina de un usuario o se transmite desde otra aplicación, y proporciona distintos datos orientados a aplicación o información de control.
- **Número de salidas externas (SE).** Cada salida externa es datos derivados dentro de la aplicación que ofrecen información al usuario. En este contexto, salida externa se refiere a reportes, pantallas, mensajes de error, etc. Los ítems de datos individuales dentro de un reporte no se cuentan por separado.
- **Número de consultas externas (CE).** Una consulta externa se define como una entrada en línea que da como resultado la generación de alguna respuesta de software inmediata en la forma de una salida en línea (con frecuencia recuperada de un ALI).
- **Número de archivos lógicos internos (ALI).** Cada archivo lógico interno es un agrupamiento lógico de datos que reside dentro de la frontera de la aplicación y se mantiene mediante entradas externas.
- **Número de archivos de interfaz externos (AIE).** Cada archivo de interfaz externo es un agrupamiento lógico de datos que reside fuera de la aplicación, pero que proporciona información que puede usar la aplicación.

Para calcular puntos de función (PF), se usa la siguiente relación:

$$PF = \text{conteo total} \times [0.65 + 0.01 \times \Sigma (Fi)]$$

Estos valores de dominio se cuantifican y en nuestro proyecto hemos obtenido los siguientes resultados:

- ✓ **Número de entradas externas: 3**
 - ↗ login 2 (usuario/contraseña)
 - ↗ toma de fotografía de sitio de interés 1
- ✓ **Número de salidas externas: 7**
 - ↗ conexión exitosa 1
 - ↗ error (conexión, datos) 3
 - ↗ registro exitoso 1
 - ↗ envío de registro 1
 - ↗ envío de datos de foto 1
- ✓ **Número de consultas externas: 2**
 - ↗ consulta de perfil 1
 - ↗ consulta de información de sitio 1
- ✓ **Número de archivos lógicos internos: 1**
 - ↗ base de datos 1
- ✓ **Número de archivos de interfaz externos: 3**

- ↗ guía de uso de la app
- ↗ interfaz gráfica de la app
- ↗ smartphone

Cálculo de los Puntos de Función					
Valor de Dominio	Conteo	Factor Ponderado			
		Simple	Promedio	Complejo	Subtotal
Entradas Externas (EE)	3	3	4	6	12
Salidas Externas (SE)	7	4	5	7	35
Consultas Externas	2	3	4	6	8
Archivos Lógicos Internos (ALI)	1	7	10	15	10
Archivos de Interfaz Externos (AIE)	3	5	7	10	21
Conteo Total Aproximado					86

A continuación, se muestra una tabla el Valor de Ajuste de la Complejidad, el cual comprende de **14 preguntas** y se responde usando una escala que va desde e **0 (no importante o aplicable)** a **5 (absolutamente esencial)**. Los valores constantes en la ecuación y los factores ponderados que se aplican a los conteos de dominio de información.

Valor de Ajuste de la Complejidad	
Preguntas de Factores de Ajuste de Valor (FAV)	Escala 0 (no importante aplicable) a 5 (absolutamente esencial).
1. ¿El sistema requiere respaldo y recuperación confiables?	5
2. ¿Se requieren comunicaciones de datos especializadas para transferir información hacia o desde la aplicación?	4
3. ¿Existen funciones de procesamiento distribuidas?	0
4. ¿El desempeño es crucial?	5

5. ¿El sistema correrá en un entorno operativo existente enormemente utilizado?	5
6. ¿El sistema requiere entrada de datos en línea?	5
7. ¿La entrada de datos en línea requiere que la transacción de entrada se construya sobre múltiples pantallas u operaciones?	4
8. ¿Los Archivos Lógicos Internos (ALI) se actualizan en línea?	5
9. ¿Las entradas, salidas, archivos o consultas son complejos?	4
10. ¿El procesamiento interno es complejo?	5
11. ¿El código se diseña para ser reutilizable?	5
12. ¿La conversión y la instalación se incluyen en el diseño?	5
13. ¿El sistema se diseña para instalaciones múltiples en diferentes organizaciones?	0
14. ¿La aplicación se diseña para facilitar el cambio y su uso por parte del usuario?	5
Cuenta Total $\Sigma (Fi)$	57

$$PF = \text{conteo total} \times [0.65 + 0.01 \times \Sigma (Fi)]$$

$$PF = 86 \times [0.65 + 0.01 \times 57]$$

$$PF = 105$$

Modelo COCOMO II

En su libro clásico acerca de “economía de la ingeniería de software”, Barry Boehm, introdujo una jerarquía de modelos de estimación de software que llevan el nombre COCOMO, por **CO**nstructive **CO**st **MO**del: **Modelo Constructivo de Costos**. El modelo COCOMO original se convirtió en uno de los modelos de estimación de costo más ampliamente utilizados y estudiados en la industria. Evolucionó hacia un modelo de estimación más exhaustivo, llamado COCOMO II. Como su predecesor, COCOMO II en realidad es una jerarquía de modelos de estimación que aborda las áreas siguientes:

- ✓ **Modelo de composición de aplicación.** Se usa durante las primeras etapas de la ingeniería de software, cuando son primordiales la elaboración de prototipos de las interfaces de usuario, la consideración de la interacción del software y el sistema, la valoración del rendimiento y la evaluación de la madurez de la tecnología.
- ✓ **Modelo de etapa temprana de diseño.** Se usa una vez estabilizados los requisitos y establecida la arquitectura básica del software.
- ✓ **Modelo de etapa post-arquitectónica.** Se usa durante la construcción del software.

Los modelos COCOMO están definidos para tres tipos de proyectos:

- **Modo Orgánico:** Proyectos de software relativamente pequeño y sencillo.
- **Modo Semiacoplado:** Proyectos de software intermedio, con variados niveles de experiencia, que deben satisfacer requisitos poco o medio rígido.
- **Modo Empotrado:** proyectos de software que deben ser desarrollados en un conjunto de hardware, software y restricciones operativas muy restringidas.

Debido al que el proyecto es mediano, por lo que no requiere de restricciones operativas muy restringidas, se ha seleccionado el Modelo Orgánico, puesto que sí tiene requisitos poco rígidos.

En la siguiente tabla se puede visualizar el uso de COCOMO Básico debido a la complejidad del trabajo.

Modelo COCOMO	Factores			
Proyecto de Software	A_b	B_b	C_b	D_b
Orgánico	3.2	1.05	2.5	0.38
Semiacoplado	3.0	1.12	2.5	0.35
Empotrado	2.8	1.2	2.5	0.32

Como todos los modelos de estimación para software, los modelos COCOMO II requieren información sobre dimensionamiento. Como parte de la jerarquía del modelo, están disponibles tres diferentes opciones de dimensionamiento: puntos objeto, puntos de función y líneas de código fuente.

El PF obtenido con valor de 105 sería el número aproximado de funciones con el cual contará el sistema.

Calculamos las Líneas de Código para nuestro proyecto expresadas en miles de Líneas de Código (KLDC) haciendo uso de la siguiente tabla.

LOC por punto de función

Lenguaje de programación	Promedio	Mediana	Bajo	Alto
Access	35	38	15	47
Ada	154	—	104	205
APS	86	83	20	184
ASP 69	62	—	32	127
Ensamblador	337	315	91	694
C	162	109	33	704
C++	66	53	29	178
Clipper	38	39	27	70
COBOL	77	77	14	400
Cool:Gen/IEF	38	31	10	180
Culprit	51	—	—	—
DBase IV	52	—	—	—
Easytrieve+	33	34	25	41
Excel47	46	—	31	63
Focus	43	42	32	56
FORTRAN	—	—	—	—
FoxPro	32	35	25	35
Ideal	66	52	34	203
IEF/Cool:Gen	38	31	10	180
Informix	42	31	24	57
Java	63	53	77	—
JavaScript	58	63	42	75
JCL	91	123	26	150
JSP	59	—	—	—
Lotus Notes	21	22	15	25
Mantis	71	27	22	250
Mapper	118	81	16	245
Natural	60	52	22	141
Oracle	30	35	4	217
PeopleSoft	33	32	30	40
Perl	60	—	—	—
PL/I	78	67	22	263
Powerbuilder	32	31	11	105
REXX	67	—	—	—
RPG II/III	61	49	24	155
SAS	40	41	33	49
Smalltalk	26	19	10	55
SQL	40	37	7	110
VBScript36	34	27	50	—
Visual Basic	47	42	16	158

El LOC de Python no está determinado en la tabla anterior pero algunos documentos indican que es menor o igual al lenguaje de programación Perl, por lo que para la realización de los cálculos vamos a tomar el promedio LOC de Perl que serían 60 líneas promedio por función.

KLDC= (PF) * (Promedio de Líneas por Función)

KLDC= (105) * (60)= 7,5387

KLDC=((105) * (60)) / 1000=7.5387 miles de líneas de código aproximadas.

por lo que: **Estimación de líneas de código = 7,5387 / KLDC=7.5387**

Para este caso, se aplicará el modelo COCOMO de tipo Orgánico. Por consiguiente, los coeficientes que usaremos serán las siguientes:

Para realizar el **cálculo del esfuerzo (E)** del COCOMO Básico Orgánico nos apoyaremos de la siguiente fórmula:

$E = A_b * [KLDC]^{B_b} = (3.2) * (7.5387)^{(1.05)}$

E = 27.5794 =28 persona/mes

Para realizar el cálculo de la duración del proyecto (D) en meses del COCOMO básico Orgánico nos apoyaremos de la siguiente fórmula:

$D = C_b * E^{D_b}$

$D = (2.5) * (27.5794)^{(0.38)}$

D = 8.9891= 9 meses

Finalmente, para calcular el número de personas necesarias (N), nos apoyaremos de la siguiente fórmula:

$N = E/D$

N = (32)/9 meses)

N = 3.11111 =3 personas

Debido a que solo contamos con 6 meses para la realización del proyecto debemos ajustar la duración del proyecto en la fórmula anteriormente realizada para obtener el valor justo de personas que se requieren para realizar el proyecto en el tiempo definido.

$N = E/D$

N = (28)/(6 meses)

N = 4.66666 =5 personas

IV. AGENDA (Diagrama de Gantt)

No.	Actividad	Agosto~Septiembre 2021													
		30	31	02	06	07	09	13	14	20	21	23	27	28	30
1	Plan de Proyecto														
1.1	I. INTRODUCCIÓN														
1.2	Ámbito														
1.3	Objetivo General y Específico														
1.4	Descripción del Proyecto														
1.5	Funciones Principales														
1.6	Justificación														
1.7	Paradigma de Desarrollo														
1.8	Estado del Arte														
2	II. ANÁLISIS DE RIESGOS														
3	III. MÉTRICAS Y ESTIMACION														
4	IV. AGENDA (Identificación de Tareas Gráfico Gantt & PERT)														
5	V. RECURSOS (Lenguajes de Desarrollo)														
6	VI. ORGANIZACIÓN DEL PERSONAL														
7	VII. MECANISMOS DE SEGUIMIENTO Y CONTROL (FORMATO DE TQM)														
8	VIII. ANEXOS														
8.1	Cálculos														
8.2	Técnicas de Recolección de Requisitos														
9	1^{era} Entrega de la Documentación del Proyecto														

No.	Actividad	Octubre~Noviembre 2021													
		04	05	07	11	12	14	18	21	25	26	28	08	11	15
8	VIII. METODOLOGÍA DE PROPUESTA DE DESARROLLO														
8.1	Análisis de Flujo de Datos de Gane & Searson														
8.2	Metodología Ágil "Scrum"														
9	IX. ANÁLISIS Y DISEÑO DEL SISTEMA Y MODELADO CON UML														
9.1	Requisitos Funcionales y Requisitos NO Funcionales														
9.2	Modelado de Casos de Uso														
9.3	Reglas del Business														
9.4	Modelado con UML Diagrama de Caso de Uso Diagrama de Clases Diagrama de Secuencia Diagrama de Estados Diagrama de Actividades														
10	X. CALIDAD DEL PROYECTO DE SOFTWARE														
11-12	XI. XII. NORMAS ISO & IEEE RELACIONADAS AL PROYECTO														
13	XIII. MODELADO Y DISEÑO DEL SISTEMA														
13.1	Entrada, Proceso y Salida del Sistema														
13.2	Prototipo de Prediseño de la Aplicación Móvil														
13.3	Prototipo de Diseño Final de la Aplicación Móvil "guiAR"														
14	XIV. IMPLEMENTACIÓN DE CALIDAD/ SATISFACTORIA														
15	XV. DESARROLLO, EJECUCIÓN Y PRUEBAS														
15.1	2da Entrega de la Documentación del Proyecto														

Diagrama de PERT

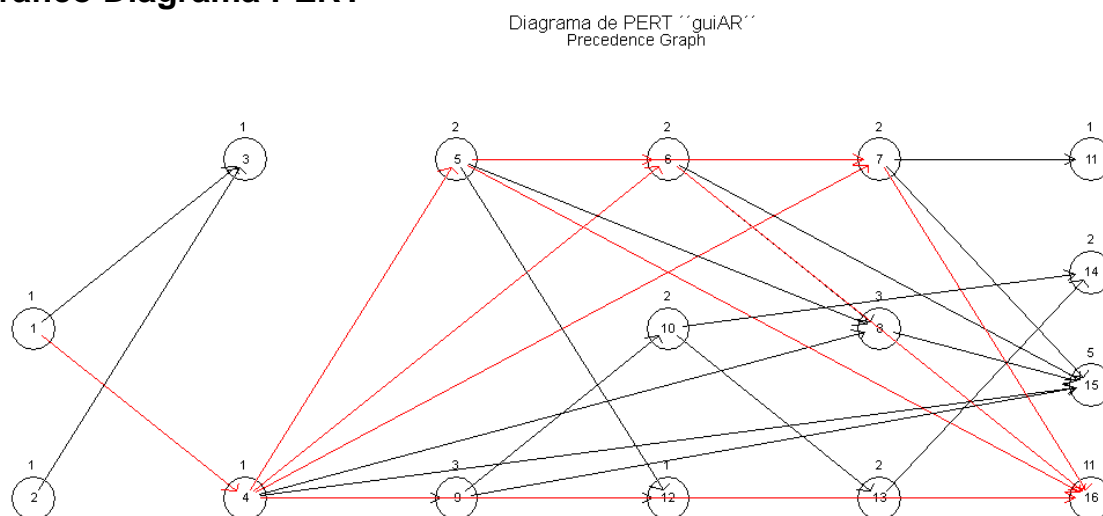
Lista Matriz de Actividades

Diagrama de PERT ""guiAR""								
	Activity time	Prec 1	Prec 2	Prec 3	Prec 4	Prec 5	Prec 6	Prec 7
1	1							
2	1							
3	1							
4	1	2						
5	2	4						
6	2	5	4					
7	2	5	6	4				
8	3	4	6	5				
9	3	4						
10	2	9						
11	1	7						
12	1	4	5					
13	2	10						
14	2	13	10					
15	5	4	6	7	8	9		
16	11	4	5	6	7			

Tabla PERT

Diagrama de PERT ""guiAR"" Solution						
	Activity time	Early Start	Early Finish	Late Start	Late Finish	Slack
Project	19					
1	1	0	1	0	1	0
2	1	0	1	17	18	17
3	1	1	2	18	19	17
4	1	1	2	1	2	0
5	2	2	4	2	4	0
6	2	4	6	4	6	0
7	2	6	8	6	8	0
8	3	6	9	11	14	5
9	3	2	5	10	13	8
10	2	5	7	13	15	8
11	1	8	9	18	19	10
12	1	4	5	18	19	14
13	2	7	9	15	17	8
14	2	9	11	17	19	8
15	5	9	14	14	19	5
16	11	8	19	8	19	0

Gráfico Diagrama PERT



V. RECURSOS (Hardware, Software & Personal)

En la siguiente tabla, se muestra el tipo de recurso de hardware a utilizar en el presente proyecto, para su codificación, pruebas de ejecución, cómo su respectiva finalización.

Recursos de Hardware	
PC HP All in One Modelo: 24-dd0018la Pantalla: 23.8 pulgadas, Full HD Procesador: AMD Ryzen. Memoria: 8 GB. Disco duro: 1 TB.	
Laptop HP Thin&Light Modelo: 13-BB0502LA Pantalla: 15´pulgadas Procesador: Intel Core i3 Memoria: 8 GB Disco duro: 1TB	

Smartphone con Sistema Operativo Android

Modelo: Indistinto.

Diagonal de la pantalla: 15 cm.

Resolución: 1080 x 1920 píxeles.

Procesador: Indistinto.

Almacenamiento: 16 GB.

Memoria RAM: 4 GB

Sistema operativo: Android 6.0



Recursos Software

Lenguajes y Herramientas para el Desarrollo

La idea principal del proyecto es tener una base de datos en la que se localice toda la información que el usuario necesita, información que ha sido recopilada por medio de los cuestionarios aplicados. Por lo que la licencia de un SGBD es fundamental en el proyecto, se tiene pensado usar MongoDB por fines prácticos, sin embargo, las especificaciones todavía siguen en investigación ya que la demanda del producto al ser poca no se requiere de un gran espacio en memoria ni un potente ordenador que sea capaz de procesar muchas solicitudes al mismo tiempo, solicitudes de las que se hablará después.

Python

Se trata de un lenguaje dinámico que cualquier profesional de IT o desarrollador web debe tener en su radar. Se implementa en diversas plataformas y sistemas operativos y permite automatizar procesos. A continuación, desarrollaremos sus innumerables usos y posibilidades.

Python se define como un “lenguaje de programación versátil, multiplataforma y multiparadigma que se destaca por su código legible y limpio”. Cuenta con una licencia de código abierto que permite su utilización en distintos contextos de forma gratuita. A su vez, se emplea en plataformas de alto tráfico como Google, YouTube o Facebook. Python atrae por su sencillez y exactitud en la sintaxis, ya que se trata de un lenguaje como cualquier otro, pero a nivel informático.

Su principal objetivo es la automatización de procesos, lo que hará de las tareas algo mucho más simple. En este sentido, Python crea un código con gran legibilidad, que ahorra tiempo y recursos. Uno de sus puntos fuertes es que “comprueba los errores sobre la marcha” para solucionarlos cuando afectan a la memoria, lo que mantiene la integridad de la matriz y evita las complicaciones a la hora de escribir el código.



FireBase

Es un conjunto de herramientas orientadas a la creación de aplicaciones de alta calidad, al crecimiento de los usuarios y a ganar más dinero. Personalmente describo la plataforma como una suite de diferentes aplicaciones que nos harán más fácil el desarrollo de nuestra aplicación. Veamos, uno por uno, los servicios más destacados que nos ofrece FireBase para trabajar de una manera más sencilla en nuestra aplicación es:

Base de datos Real time (Tiempo Real)

Con la base en tiempo real de FireBase podrás guardar todos los datos que requiera tu aplicación. Se lleva muy bien con React y su patrón reactivo que permite actualizar los datos en los componentes auto mágicamente. Los datos se almacenan en formato JSON y se pueden agregar reglas para permitir requests con token o solo desde una URL.

Autenticación

Es un servicio que nos simplifica el inicio de sesión y la gestión de la misma, en nuestra aplicación. Si la usamos en aplicaciones web es extremadamente fácil de configurar, sobre todo si usamos el proveedor de Google, aun así, si usamos otros de los disponibles (Correo/Contraseña, Teléfono, Facebook, Twitter, GitHub, Anónimo) también es muy fácil, sólo es un paso más en el caso de las redes.

Almacenamiento

Este servicio es muy bueno para aplicaciones que requieran guardar archivos del usuario. También nos sirve si queremos subir estáticos ya que existe un botón desde la interfaz o podemos programar algo.

Laboratorio de test (pruebas) para Android

Los desarrolladores de Android se habrán planteado la encrucijada que produce muchas veces a la hora de probar una aplicación. Existen múltiples modelos de dispositivos con diferentes versiones y diseño del OS, así como diferentes sensores. Este servicio nos ofrece la posibilidad de probar los test de nuestra aplicación en los entornos que configuremos (Dispositivo, versión del OS).

Informes sobre fallos

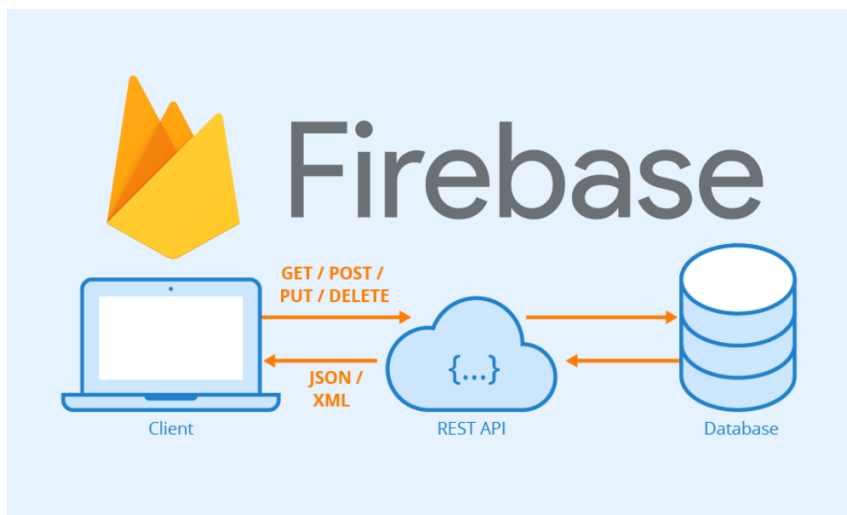
Con esta utilidad podemos detectar los errores que aparezcan en nuestras aplicaciones de iOS y Android, así como poder erradicarlos a tiempo. Esta herramienta próximamente dará paso a Crashlytics de Fabric, que se integrará como una herramienta más.

Notificaciones

Este servicio, como su nombre nos indica, nos permitirá gestionar el envío de notificaciones a nuestros usuarios con la diferencia de que estas podrán ser programadas acorde a diferentes parámetros.

Configuración remota

Este añadido nos permitirá básicamente modificar el comportamiento y la apariencia de nuestra aplicación sin que nuestros usuarios tengan que tocar nada.



MongoDB

MongoDB es una base de datos distribuida, basada en documentos y de uso general que ha sido diseñada para desarrolladores de aplicaciones modernas y para la era de la nube. Ninguna otra ofrece un nivel de productividad de uso tan alto.

MongoDB es una base de datos orientada a documentos. Esto quiere decir que, en lugar de guardar los datos en registros, guarda los datos en documentos. Estos documentos son almacenados en BSON, que es una representación binaria de JSON.

Una de las diferencias más importantes con respecto a las bases de datos relacionales, es que no es necesario seguir un esquema. Los documentos de una misma colección concepto similar a una tabla de una base de datos relacional, pueden tener esquemas diferentes.



Características:

- La forma más natural y productiva de trabajar con datos.
- Admite matrices y objetos anidados como valores.
- Trabaje con esquemas dinámicos y flexibles.
- Lenguaje de consulta rico y expresivo que permite filtrar y ordenar por cualquier campo, independientemente de cómo esté incrustado en un documento.
- Admite agregaciones y otros casos de uso modernos, como búsqueda de gráficos o texto, y búsqueda basada en información geoespacial.
- Las propias consultas son también JSON, por lo que se programan fácilmente. Olvídense de concatenar cadenas para generar consultas SQL de forma dinámica.
- Lenguaje de consulta rico y expresivo que permite filtrar y ordenar por cualquier campo, independientemente de cómo esté incrustado en un documento.
- Admite agregaciones y otros casos de uso modernos, como búsqueda de gráficos o texto, y búsqueda basada en información geoespacial.
- Las propias consultas son también JSON, por lo que se programan fácilmente. Olvídense de concatenar cadenas para generar consultas SQL de forma dinámica.

IDE

Otra licencia que aún no se menciona es nuestra IDE por utilizar para la programación correspondiente del proyecto, por lo que se usará las siguientes:

1. **Visual Studio Code** es un editor de código redefinido y optimizado para crear y depurar aplicaciones web y en la nube modernas. y está disponible para Windows, macOS y Linux. Viene con soporte integrado para JavaScript, TypeScript y Node.js y tiene un rico ecosistema de extensiones para otros lenguajes (como C++, C, Java, Python, PHP, Go) y tiempos de ejecución (como .NET y Unity).
2. **Flutter** es el kit de herramientas de UI de Google para realizar hermosas aplicaciones, compiladas nativamente, para móvil, web y escritorio desde una única base de código. Ayuda a rápida y fácilmente experimentar, construir UIs, añadir funcionalidades, y corregir bugs más rápido. Experimenta tiempos de recarga por debajo de un segundo, sin perder el estado, en emuladores, simuladores, y dispositivos para iOS y Android.
3. **Dart** es un lenguaje open source desarrollado en Google con el objetivo de permitir a los desarrolladores utilizar un lenguaje orientado a objetos y con análisis estático de tipo. Se diseñó con el objetivo de hacer el proceso de desarrollo lo más cómodo y rápido posible para los desarrolladores. Por eso, viene con un conjunto bastante extenso de herramientas integrado, como su propio gestor de paquetes, varios compiladores, un analizador y formateador. Además, la máquina virtual de Dart y la compilación *Just-in-Time* hacen que los cambios realizados en el código se puedan ejecutar inmediatamente.
4. **Postman** es una plataforma de colaboración para el desarrollo de API. Las características de Postman simplifican cada paso de la creación de una API y optimizan la colaboración para que pueda crear mejores API, más rápido. Sabiendo los lenguajes también a ocupar:



Cabe mencionar que las licencias de uso de cada una de las IDE's mencionadas son gratis, por lo que la decisión de uso está más enfocada en el lenguaje a utilizar y las necesidades que cubre el lenguaje.

Recursos Personal

Teniendo en cuenta el Proyecto a desarrollar e implementar. A continuación, se presentan algunas de las actividades como información el cual especifica el cargo, las funciones, los requerimientos y habilidades necesarios para ocupar el puesto, entre algunas otras características que desarrollarán tres distintos puestos dentro del Proyecto del Asistente Turístico "guiAR" de acuerdo con el organigrama presentado en la siguiente sección.

Nombre del cargo: Director general		Objetivo: Prever, organizar, mandar, coordinar y controlar las actividades del Proyecto.
Requerimientos y habilidades mentales: <ul style="list-style-type: none"> ✎ Capacidad de descubrir y potenciar las capacidades de quienes forman su equipo. ✎ Capacidad para delegar funciones entre los miembros del equipo. ✎ Habilidad para la comunicación y adaptación a diversos locutores, subordinados, superiores, etc. ✎ Capacidad de resolver problemas. ✎ Capacidad para adaptarse positivamente a los problemas. ✎ Habilidades técnicas básicas. ✎ Empatía y habilidades sociales. ✎ Fluidez en la toma de decisiones. 	Edad: 30 años en adelante.	
	Sexo: Indistinto	
	Funciones: <ul style="list-style-type: none"> • Brindar orientación a sus subordinados para lograr los objetivos de una forma más productiva. • Planificar estrategias enfocadas a mejorar los procesos internos y externos de la organización con el propósito de enriquecer sus operaciones. • Organizar los recursos que se emplean en cada una de las áreas que conforman. • Liderar al capital humano. • Controlar y organizar los procesos que se llevan a cabo en el interior del Proyecto. 	
Responsabilidad: Es el responsable de supervisar todas las funciones administrativas de la empresa. Una gran parte implica liderar y dirigir a los empleados.		Conocimientos: Competencias a nivel avanzado en inglés. Tener entre 2 a 4 años de experiencia en gerencia o planificación, desarrollo y gestión de proyectos.

Actividades por realizar:

- ✎ Comunicar los deseos de sus superiores a los empleados administrativos.
- ✎ Toma de decisiones: analizar situaciones y determinar las mejores medidas.
- ✎ Ayudar a mejorar a los empleados o eliminar a aquellos que no cumplen con los estándares de la empresa.
- ✎ Atender la selección y contratación: juzgar a los candidatos potenciales a ser contratados.
- ✎ Mantener relaciones cordiales con los clientes que requieren servicios de la empresa.

Nombre del cargo: Desarrollador de Software	Objetivo: Llevar a cabo la realización del Proyecto desde 0, hasta su culminación y entrega.
Requerimientos y habilidades mentales: <ul style="list-style-type: none">➤ Responsabilidad.➤ Trabajo en Equipo.➤ Capacidad para adaptarse al ambiente.➤ Ser Productivo, organizado y competitividad.➤ Habilidad para la comunicación.➤ Compromiso en el cargo/puesto.	Edad: 22 años en adelante.
	Sexo: Indistinto
	Funciones: <ul style="list-style-type: none">➤ Lectura y análisis de informes de investigación y sugerencias de los clientes.➤ Integrar los componentes del software con los programas de terceros.➤ Producir un código limpio y eficiente basado en las especificaciones.➤ Proceso de desarrollo de aplicaciones o apps móviles.
Mantiene contacto con: Director General, administrador de proyectos, supervisor de ventas y gerente de operaciones.	Conocimientos: Competencias a nivel avanzado en Inglés. Tener entre 2 años o más de experiencia en: Conocimientos de lenguajes de codificación y Experiencia con bases de datos y sistemas de mapeo objeto-relacional.

	Estar familiarizado con las metodologías de desarrollo.
Responsabilidad: Es el responsable directo en la implementación de los proyectos de software requeridos.	
Actividades para realizar: <ul style="list-style-type: none"> ➤ Diseñar en un entorno de software orientado a pruebas. ➤ Ingenio y capacidad para solucionar problemas. ➤ Crear documentación técnica para referencia e informe. ➤ Aprender nuevos lenguajes y tecnologías. ➤ Solucionar problemas, depurar y actualizar el software existente. 	

Nombre del cargo: Gestor de Personal.	Objetivo: Contratación de personal, relaciones humanas, evaluación y desempeño del personal.
Requerimientos y habilidades mentales: <ul style="list-style-type: none"> ✓ Capacidad de adaptarse a distintos ambientes. ✓ Manejo de situaciones conflictivas. ✓ Capacidad de comunicación. ✓ Capacidad de evaluar a una persona. ✓ Habilidad de leer a las personas. 	Edad: 22 años en adelante.
	Sexo: Indistinto
	Funciones: <ul style="list-style-type: none"> ✓ Contratación de personal. ✓ Manejo de situación conflictivas. ✓ Evaluación de empleados. ✓ Informe de rendimiento.
Responsabilidad: Es el responsable contratar personas para los distintos puestos de la empresa. Así como de evaluar el desempeño de los empleados.	Conocimientos: Competencias a nivel avanzado en inglés. Conocimiento básico de psicología, experiencia mínima de 1 año en empresas.

Actividades por realizar:

- ✓ Contratación de nuevo personal.
- ✓ Informar el desempeño de la empresa.
- ✓ Realizar evaluaciones continuas al personal.
- ✓ Mantener buenas relaciones en el ambiente laboral.
- ✓ Informar el despido o la contratación de un empleado al Director General.

VI. ORGANIZACIÓN DEL PERSONAL

La Gestión de un Proyecto implica la planificación, supervisión, control del personal, del proceso y de los eventos que ocurren mientras evoluciona el desarrollo del software, desde el inicio del proyecto hasta la finalización del mismo. Se concentra en las 4P: Personal, Producto, Proceso y Proyecto. Por lo que los paradigmas y organización de equipos de desarrollo, son los siguientes:

↳ **Descentralizado democrático (DD).**

- No tiene un jefe permanente. Se nombran coordinadores de tareas a corto plazo y se sustituyen por otros para diferentes tareas.
- Las decisiones y los enfoques se hacen por consenso.
- La comunicación entre los miembros del equipo es horizontal.

↳ **Descentralizado controlado (DC).**

- Tiene un jefe definido que coordina tareas específicas y jefes secundarios que tienen responsabilidades sobre subtareas.
- La solución de problemas es una actividad del grupo, pero la implementación de soluciones se reparte entre los subgrupos por el jefe de equipo.
- La comunicación entre subgrupos e individuos es horizontal.
- También hay comunicación vertical a lo largo de la jerarquía de control.

↳ **Centralizado controlado (CC).**

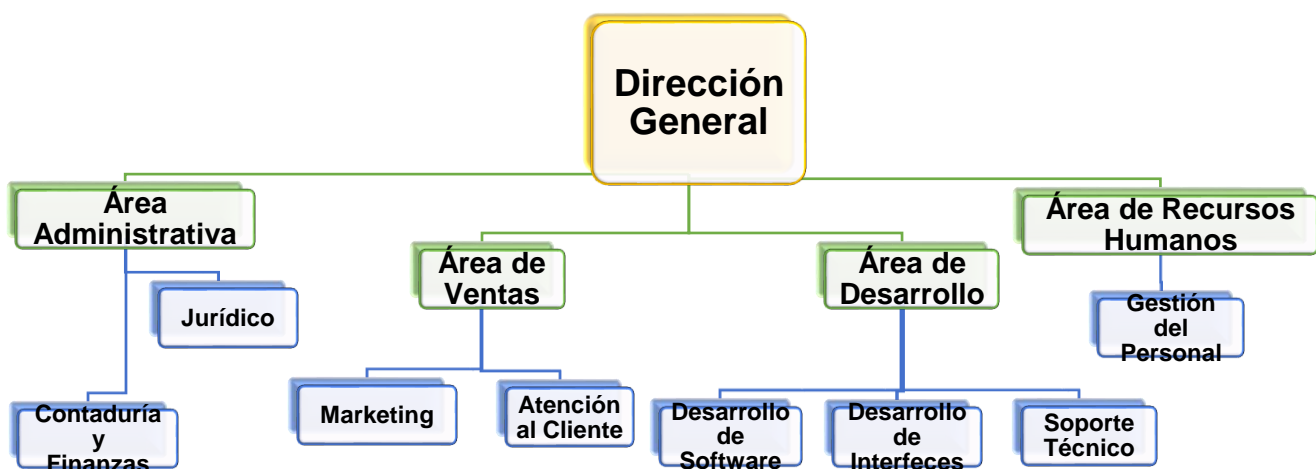
- El jefe de equipo se encarga de la solución de problemas a alto nivel y la coordinación interna del equipo.
- La comunicación entre el jefe y los miembros del equipo es vertical.

Existen 7 factores de un proyecto que deben considerarse al planificar el organigrama de equipos.

- ✓ **Dificultad del problema.**
- ✓ **Tamaño del programa en líneas de código o puntos de función.**
- ✓ **Tiempo de vida del equipo.**
- ✓ **Grado en que el problema puede ser modularizado.**
- ✓ **Calidad requerida y fiabilidad del sistema que se va a construir.**
- ✓ **Rigidez de la fecha de entrega.**
- ✓ **Grado de comunicación requerido para el proyecto.**

En nuestro proyecto, como seguramente en la mayoría o en todos, se usará la organización de equipo “**Descentralizado Democrático (DD)**”, puesto que no habrá un jefe como tal, sino que todos haremos una parte del proyecto, organizándonos en repartir tareas específicas a cada miembro del equipo, teniendo la opción de sustituirlas por otras, dependiendo del conocimiento que tenga cada integrante. Es importante mencionar que las decisiones se toman en conjunto y por democracia, también que todos los integrantes tenemos la misma información, permitiendo una organización cooperativa y colaborativa, de esta manera, todos estaremos en contexto de lo que vayamos haciendo, teniendo una mejor comunicación.

Seguidamente se modelo la organización del personal de una manera más ilustrativa como fue dividida por áreas/departamentos, dado que nos dará una mejor visualización y una mayor organización.



VII. MECANISMOS DE SEGUIMIENTO Y CONTROL

Manejo de la Calidad del Proyecto

La administración de la calidad total (TQM) es esencial en todos los pasos de desarrollo de sistemas. De acuerdo con Evans y Lindsay (2004), los principales elementos de la TQM tienen sentido sólo cuando ocurren en un contexto organizacional que apoye un esfuerzo de calidad integral. Es en este contexto en el que se unen los elementos de enfoque hacia el cliente, la planeación y el liderazgo estratégico, la mejora continua, el otorgamiento de poderes y el trabajo en equipo para cambiar el comportamiento de los empleados y, en última instancia, el curso de la organización.

Los analistas de sistemas deben estar conscientes de los factores que impulsan el interés en la calidad. Es importante tener en cuenta que el compromiso cada vez mayor de las empresas con la TQM se adapta extraordinariamente bien a los objetivos generales para el análisis y diseño de sistemas.

La calidad total consiste en los esfuerzos de toda una organización para instalar y crear un clima permanente donde los empleados mejoren continuamente su capacidad para proporcionar productos y servicios que los clientes encuentren con un valor particular. Se enfoca en asegurar que las normas internas y los estándares de proceso reduzcan los errores.

"Total" enfatiza que todos los demás departamentos además de producción, como contabilidad, ventas, marketing y diseño e ingeniería están obligados a mejorar sus operaciones.

Los ejecutivos están obligados a gestionar activamente la calidad a través de la financiación, la capacitación, la dotación de personal y el establecimiento de objetivos.

El producto recibido por los consumidores debe de estar en las correctas condiciones para su uso, además de mejorar todos los procesos internos, buscando producir bienes sin defectos desde un inicio.

Deberemos de seguir los siguientes principios básicos que establece la calidad total:

1. Enfocarse en el cliente.
2. Participación de los trabajadores.
3. Centrado en el proceso.
4. Sistema integrado.
5. Enfoque estratégico y sistemático.
6. Toma de decisiones basada en hechos.
7. Comunicación.
8. Mejora continua.

Una adecuada gestión de calidad total supone:

1.- Planificar la calidad

Precede al inicio de toda actividad e implica el desarrollo de los productos y procesos que mejor satisfagan las necesidades de los clientes, este punto consta de 3 etapas

A. Determinar quiénes son los clientes a los que debemos dirigirnos y cuáles son sus necesidades.

Esto ya se ha establecido previamente en el proyecto. Los clientes a los cuales nos dirigimos son turistas nacionales y extranjeros que cuenten con un dispositivo móvil, que visiten la CDMX por motivo de vacaciones. En cuanto a la necesidad a satisfacer, esta será ofrecer un guía turístico el cual mostrará la mayor cantidad de puntos de interés de la ciudad además de mostrar en primera instancia, información histórica de los lugares, además de información general del sitio.

B. Proceder al diseño del producto o servicio teniendo en cuenta los resultados de la fase anterior en lo que a determinación de sus necesidades se refiere

Las funciones de esta aplicación serán el mostrar mediante realidad aumentada, información relevante e importante de las atracciones o sitios turísticos de la CDMX, teniendo como opción que dicha información sea mencionada en forma de audio en los principales idiomas, esto solo enfocando dicho sitio con la cámara trasera del móvil.

C. Llevar a cabo los procesos adecuados para lograr las características del producto que se hayan fijado.

Para lograr este punto, el proyecto se deberá de realizar lo más apegado a este documento, respetando las fechas de la agenda del proyecto y teniendo muy en cuenta el análisis de los riesgos.

2.- Controlar la calidad

Basándonos en las posibles desviaciones que se hayan producido en la realización de los procesos, para lo cual asumiremos el nivel fijado por la planificación como el nivel estándar, se llevarán a cabo dos acciones.

A. Evaluación de las desviaciones en calidad.

Esta parte muy posiblemente se ira actualizando con el tiempo, de momento encontramos que se ha dado una desviación en la calidad en cuanto a los tiempos que se habían estipulado en la agenda.

B. Toma de medidas necesarias para la corrección de dichas desviaciones.

De momento las medidas propuestas para la corrección de las desviaciones son el tener mayor comunicación entre el equipo y muy posiblemente reagendar y acelerar las actividades planeadas.

3.- Mejorar la calidad

Se trata de la actividad sistemática y organizada que trata de corregir las deficiencias originadas en la etapa de planificación, para poder elevar los niveles de calidad en futuras planificaciones, los objetivos que se esperan para esta fase son:

- A.** Establecimiento de una infraestructura capaz de asegurar mejoras de calidad sistemáticas.
- B.** Elaboración de los pertinentes "proyectos de mejora" identificación y selección de las mejoras de calidad prioritarias, las más apremiantes.

Capítulo 4: ANÁLISIS Y DISEÑO DEL SISTEMA ORIENTADO A OBJETOS, MEDIANTE EL USO DE UML

VIII. METODOLOGÍA PROPUESTA DE DESARROLLO

Análisis de Flujo de Datos de Gane & Searson

En esta sección se llevará a cabo el siguiente Análisis de Gane & Searson, mediante la creación de diagramas de flujo lógicos. Dicho análisis se tomó en cuenta, como una metodología posible a seguir para el presente Proyecto, dado que estos diagramas son herramientas estructuradas de análisis y diseño, las cuales permiten al desarrollador comprender el sistema y los subsistemas en forma visual, como un conjunto de flujos de datos interrelacionados.

El desarrollador de sistemas extrae los procesos, orígenes, almacenes y flujos de datos de las narrativas o historias de la organización que contaron los usuarios o que revelaron los datos, y utiliza una metodología arriba-abajo para dibujar primero un diagrama de flujo de datos a nivel de contexto del sistema con una vista más amplia. Después se dibuja un diagrama de flujo de datos lógico de nivel 0. Se muestran los procesos y se agregan los almacenes de datos. A continuación, el desarrollador crea un diagrama hijo para cada uno de los procesos en el Diagrama 0. Las entradas y salidas permanecen constantes, pero los almacenes de datos y los orígenes cambian. Al expandir el diagrama de flujo de datos original, el desarrollador de sistemas se puede concentrar en descripciones más detalladas del movimiento de datos en el sistema. Después, el desarrollador realiza un diagrama de flujo de datos físico a partir del diagrama de flujo de datos lógico y lo particiona para facilitar la programación. Se analiza cada proceso para determinar si debe ser manual o automatizado.

A continuación se enlistan los pasos y las representaciones gráficas del almacenamiento y la transformación del movimiento de los datos, del Proyecto de Software “guiAR”:

Paso 1: Hacer una lista de actividades de la empresa y usarla para determinar los siguientes elementos.

De todo lo visto anteriormente, desde la introducción y planteo del problema tenemos que:

Entidades Externas:

1. Usuarios (Turistas)
2. Sitios/Lugares Turísticos

Flujos de Datos:

1. Usuario escoge un sitio turístico de su agrado.
2. Se obtiene de la base de datos los diversos sitios turísticos del Centro Histórico almacenados.
3. Se obtiene el reconocimiento del sitio/lugar, mediante el uso de la cámara trasera del smartphone.
4. Envío a la data a enlazar.
5. Se entrega al usuario (turista) la respuesta a la petición que realizó.
6. Despliega la información relaciona al sitio o lugar, tal como son: datos, relevantes, históricos, e interesantes, como también un extra, como puntos de recomendación, transporte, etc...
7. Despliegue de la información relaciona al sitio o lugar, tal como son: datos, relevantes, históricos, e interesantes, mediante la utilización de audio en los principales idiomas.
8. Actualización de otro sitio/lugar turístico-cultural.

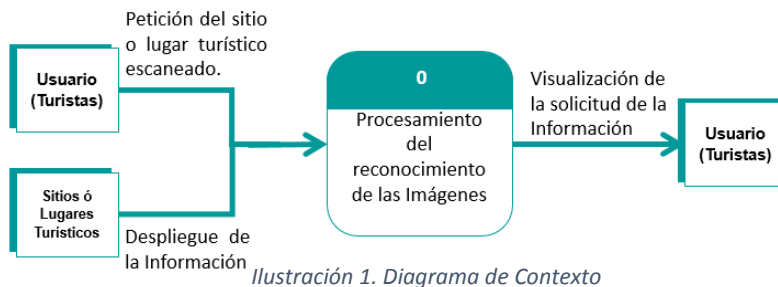
Procesos:

1. Reconocimiento y Lectura de las imágenes.
2. Solicito el nombre del lugar turístico.
3. Solicito la información y todo lo referente del sitio turístico.
4. Conectar la figura (imagen), el nombre, los datos importantes, relevantes del sitio/lugar turístico-cultural.
5. Realizar la petición del sitio o lugar turístico escaneado.
6. Desplegar y recopilar información extra a ejecutar.
7. Habilitación de la opción audio, para comunicar los datos importantes, relevantes del sitio/lugar turístico-cultural.
8. Actualizar otro sitio/lugar turístico-cultural.

Almacenes de datos:

1. Base de datos del Asistente Turístico de Sitios Culturales “guiAR”.
Contiene y posee los datos de los sitios/lugares turísticos-culturales, de los cuales se recolectará y se harán uso de la información más relevante, interesante e histórica, es la base de datos principal.

Paso 2: Crear un diagrama de contexto que muestre las entidades externas y los flujos de datos que entran y salen del sistema.



Paso 3. Dibujar el Diagrama 0, el siguiente nivel.

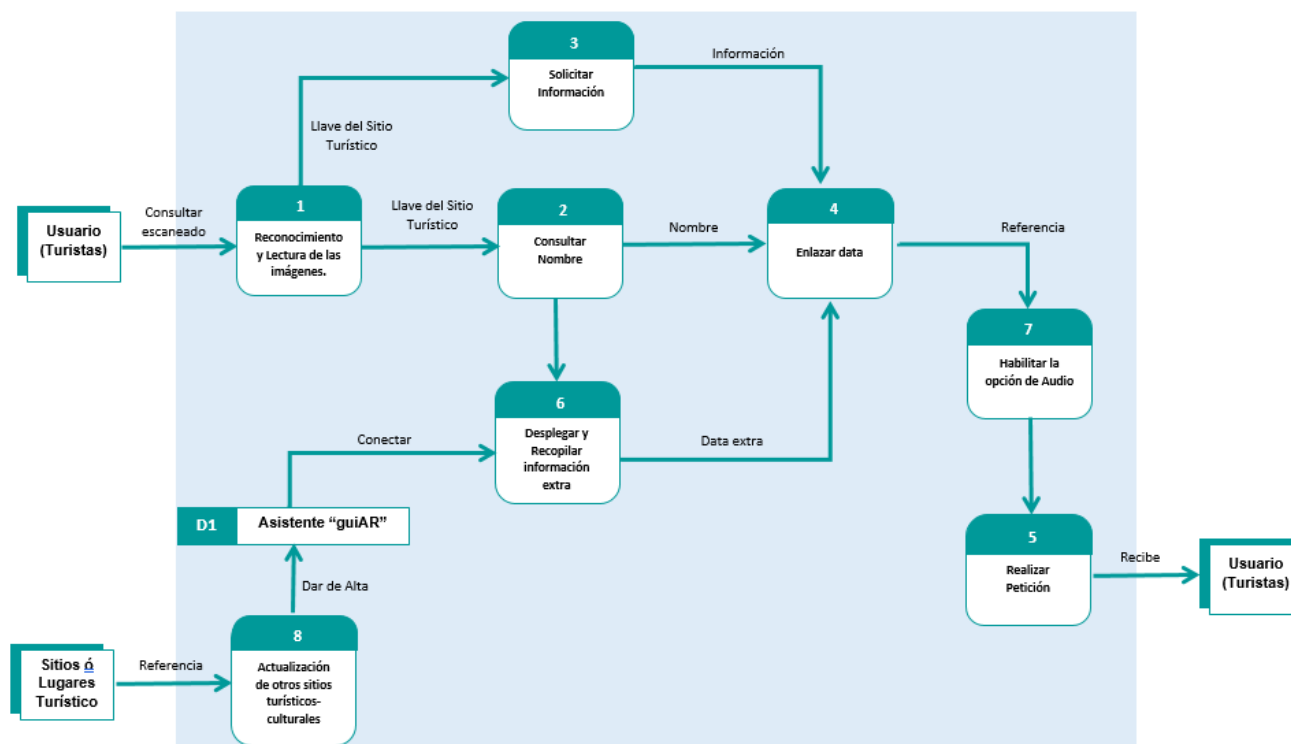


Ilustración 2. Diagrama 0, el siguiente nivel.

Paso 4. Crear un diagrama hijo para cada uno de los procesos en el Diagrama 0.

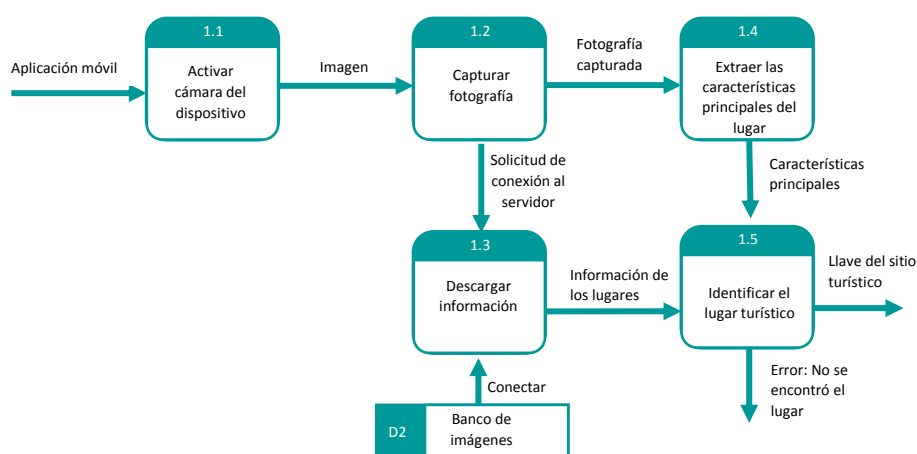


Ilustración 3. Diagrama hijo del proceso 1.

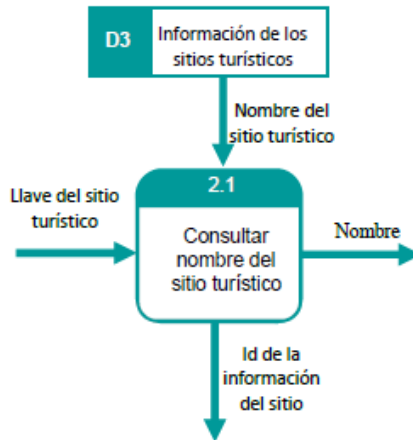


Ilustración 4. Diagrama hijo del proceso 2.

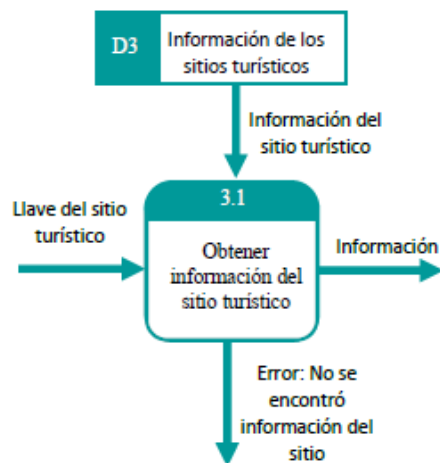


Ilustración 5. Diagrama hijo del proceso 3..

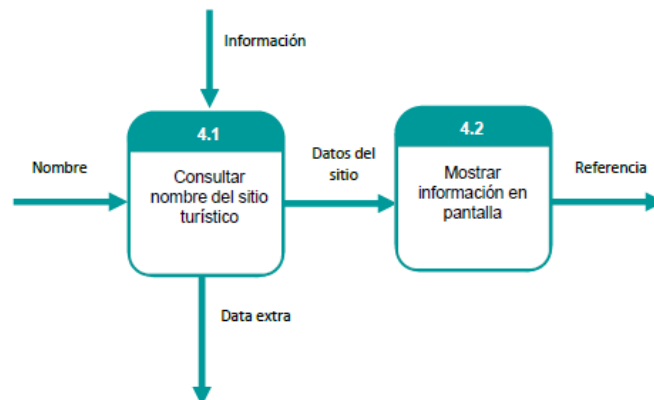


Ilustración 5. Diagrama hijo del proceso 4.

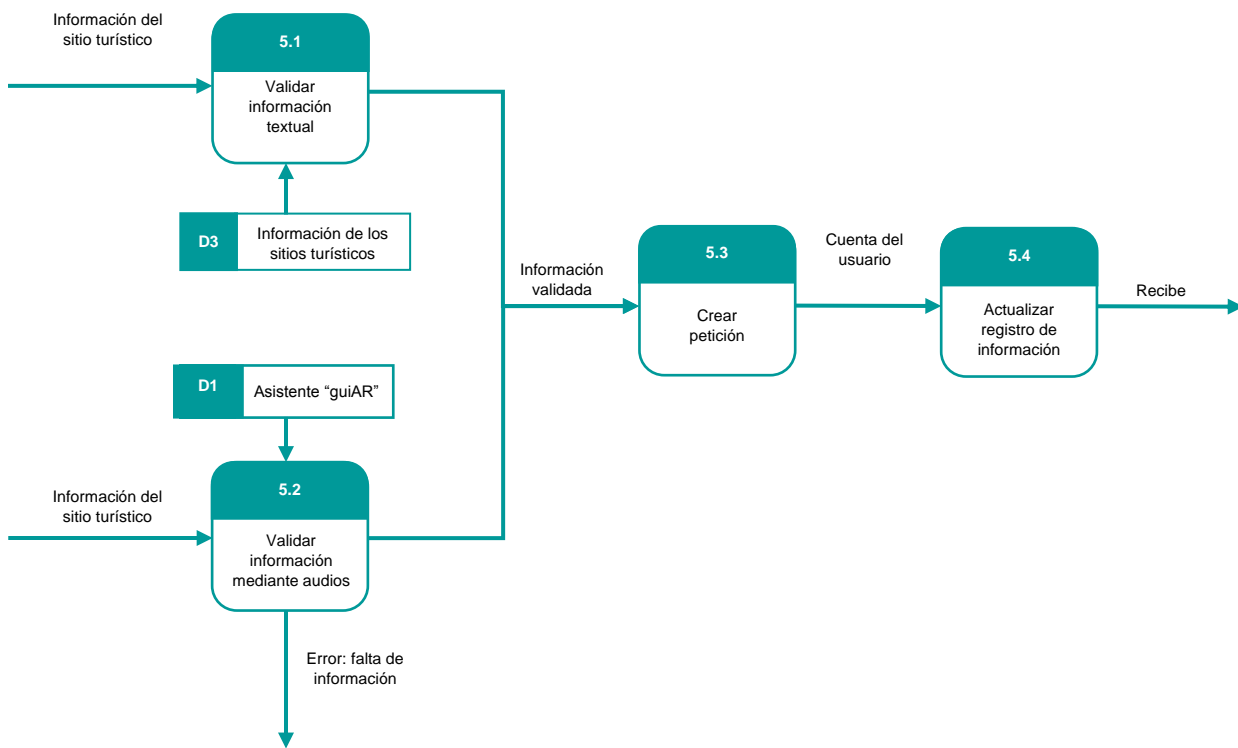


Ilustración 6. Diagrama hijo del proceso 5.

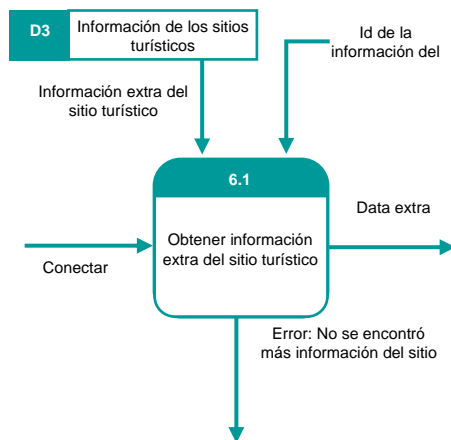


Ilustración 7. Diagrama hijo del proceso 6.

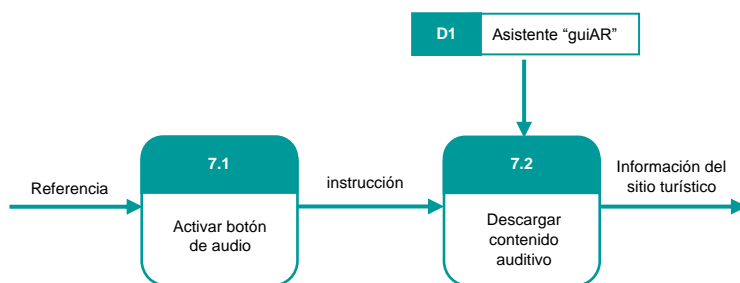


Ilustración 8. Diagrama hijo del proceso 7.

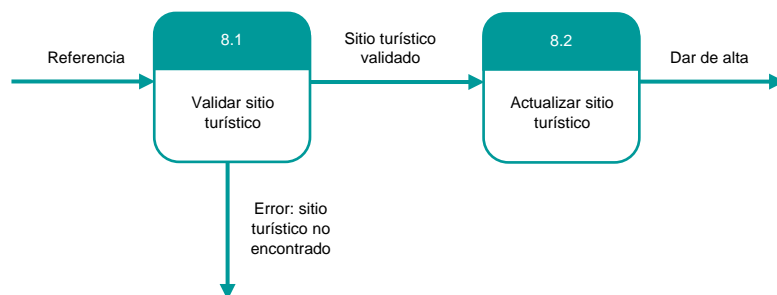


Ilustración 10. Diagrama hijo del proceso 8.

Metodología Ágil “Scrum”

En este apartado se abordará una Metodología Ágil, la cual se desarrollará y se llevará acabo con apoyo y apego en el Presente Proyecto de Software, dado que en la sección u apartado anterior se inició por una Metodología de “Análisis de Flujo de Datos de Gane & Searson”, cómo una pre-metodología a seguir. No obstante, en el equipo de trabajo se discutió, sobre cuál sería la mejor opción y orientación al presente Proyecto y se eligió por una Metodología ágil, dado que es una metodología de desarrollo de software que se basa en valores, principios y prácticas básicas. Los cuatro valores son comunicación, simpleza, retroalimentación y valentía. Recomendamos que los analistas de sistemas adopten estos valores en todos los proyectos que emprendan y no sólo cuando adopten la metodología ágil.

SCRUM o melé es otra de las metodologías ágiles. Desarrollada por Ken Schwaber, Jeff Sutherland y Mike Beedle, en Boston E.U.A en 1993.

Fundamentalmente, scrum se refiere al trabajo en equipo, donde desarrollan un proyecto con un plan de alto nivel que se puede modificar al instante, a medida que avanza el tiempo del Proyecto. Los miembros del equipo de desarrollo de sistemas deben tener en cuenta que el éxito del proyecto es lo más importante y que su éxito individual es secundario. El líder del proyecto tiene sólo un poco de influencia sobre los detalles. El equipo de sistemas trabaja dentro de un estricto periodo (30 días para el desarrollo). Podemos describir los componentes de la metodología de scrum así:

1. Acumulación (backlog) de productos, en donde se deriva una lista a partir de las especificaciones de los productos.
2. Acumulación de corrida (sprint), una lista que cambia en forma dinámica sobre las tareas que se van a completar en la siguiente corrida.
3. Corrida, un periodo de 30 días en donde el equipo de desarrollo transforma la acumulación en software que se puede demostrar.
4. Scrum diario, una reunión breve en donde, los miembros del equipo necesitan explicar lo que hicieron desde la última reunión, si se toparon con obstáculos y lo que planean hacer antes del siguiente scrum diaria.
5. Demo, software funcional que se puede demostrar al cliente.

Ahora bien, consideraremos a **SCRUM** como un marco de trabajo en el que se pueden crear productos o desarrollar proyectos de forma iterativa e incremental. Está formado por un conjunto de prácticas y reglas que dan respuesta a los siguientes principios de desarrollo ágil.

- ↳ Trabajan basando la calidad del resultado en el conocimiento tácito de las personas más que en el explícito de los procesos y la tecnología empleada.
- ↳ Estrategia de desarrollo incremental a través de iteraciones (**sprints**) y revisiones.
- ↳ Existe una gestión evolutiva del avance.

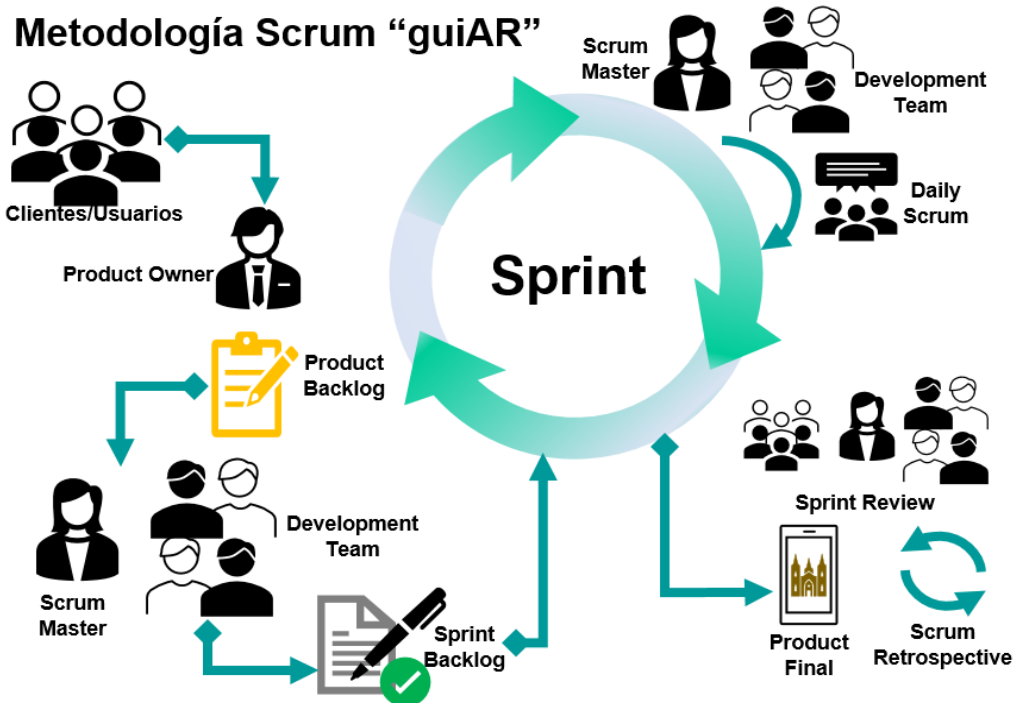
Se comienza con la visión general de lo que se desea obtener y a partir de ella se especifica y da detalle a las partes de mayor prioridad, y que se desean obtener cuanto antes.

Cada ciclo de desarrollo o iteración (sprint) están acotadas en el tiempo, es decir, finalizan en una fecha determinada independientemente de si el trabajo ha finalizado por completo o no. Normalmente los equipos escogen la duración de Sprint (no mayor a un mes) y la mantienen para todos sus sprints hasta que son capaces de emplear sprints más cortos.

El equipo monitoriza la evolución de cada sprint en reuniones breves diarias donde se revisa en conjunto el trabajo realizado por cada miembro y el previsto para el día en curso. La duración de la reunión (**scrum diario**) no debe exceder los 15 minutos, y se realiza apoyándose de un tablero o pizarra con información de las tareas del sprint y el trabajo pendiente en cada una.

No se trata de un grupo de trabajo con asignaciones específicas de tareas y responsabilidades que sigue un proceso o pautas, sino de equipos con espíritu de colaboración siguiendo un propósito en común: conseguir el mayor valor posible para la visión del cliente. En el equipo

- ✓ Todos conocen y comprenden la visión del propietario del producto.
- ✓ Comparten de forma conjunta el objetivo de cada sprint y la responsabilidad.
- ✓ Todos los miembros participan en las decisiones, respetan opiniones y aportes de todos en el proyecto en desarrollo.



IX. ANÁLISIS Y DISEÑO DEL SISTEMA Y MODELADO CON UML

Requisitos Funcionales

1. Validar imagen
2. Analizar la imagen del sitio turístico
3. Encender la cámara
4. Mostrar información del sitio turístico
5. Desplegar información mediante audio
6. Registrar usuario

Requisitos No funcionales

1. Estar disponible todas las veces que el usuario intente acceder
2. Ser usable, es decir, presentar un diseño "responsive"
3. Ser eficiente, es decir, el tiempo de respuesta al usuario
4. Tener rendimiento
5. Ser seguro

Tabla y Modelado de Casos de Uso de la aplicación

a) ¿Cuántos actores tiene el sistema?

Usuario 1 (cliente, persona humana que hace uso de la aplicación).

Usuario 2 (administrador, persona humana que controla el uso del sistema).

b) Casos de uso

- 1) Iniciar sesión
- 2) Dar permisos para abrir la cámara del smartphone
- 3) Enforzar y Analizar la imagen del Sitio/Lugar Turístico
- 4) Visualizar la información relevante, del Sitio/Lugar Turístico
- 5) Puntos de Recomendación
- 6) Actualizar otro sitio/lugar turístico

c) Tercera pregunta según el inciso b)

- ↗ ¿Cómo iniciar sesión?
- ↗ ¿Cómo dar permisos para abrir la cámara?
- ↗ ¿Cómo Enforzar un Sitio/Lugar Turístico?
- ↗ ¿Cómo Analizar la imagen del Sitio turístico?
- ↗ ¿Cómo Visualizar la información relevante, del Sitio/Lugar Turístico?
- ↗ ¿Cómo Habilitar información mediante audio?
- ↗ ¿Cómo se recomiendan otros puntos de interés?
- ↗ ¿Cómo se Actualizar otro sitio/lugar turístico?

NOMBRE	01.- Iniciar sesión	
DESCRIPCION	El usuario tendrá que contar con una cuenta para tener un registro de sus datos.	
PRECONDICION	El usuario debe contar con un dispositivo móvil con la aplicación instalada.	
SECUENCIA PRINCIPAL	01	El usuario ingresa a la aplicación.
	02	El usuario ingresa sus datos de acceso a la aplicación.
	03	El usuario indicara que desea enfocar un Sitio/Lugar Turístico.
ERROES / ALTERNATIVAS	01	El usuario no cuenta con acceso a internet.
	02	El dispositivo del usuario no cuenta con batería suficiente.
POSCONDICION	Ninguna	
NOTAS	Ninguna	

NOMBRE	02.- Abrir la cámara del smartphone	
DESCRIPCION	El usuario tendrá que dar permisos para habilitar la cámara del smartphone.	
PRECONDICION	El usuario indico que quiere enfocar un sitio turístico de su preferencia.	
SECUENCIA PRINCIPAL	01	El usuario apunta al lugar turístico enfrente de él.
ERROES / ALTERNATIVAS	01	El dispositivo del usuario no posee una cámara con buena resolución.
	02	El reconocimiento de imagen, no tiene un enfoque correcto y claro del lugar turístico.
POSCONDICION	Ninguna	
NOTAS	Ninguna	

NOMBRE	03.- Enfocar y Analizar la imagen del Sitio/Lugar Turístico				
DESCRIPCION	El usuario indico que quiere enfocar un sitio turístico de su preferencia.				
PRECONDICION	El usuario desea saber todo lo relacionado al sitio turístico en turno.				
SECUENCIA PRINCIPAL	<table border="1"> <tr> <td>01</td><td>El sistema inicia el reconocimiento de imagen del lugar.</td></tr> <tr> <td>02</td><td>El sistema busca el reconocimiento de imagen del lugar.</td></tr> </table>	01	El sistema inicia el reconocimiento de imagen del lugar.	02	El sistema busca el reconocimiento de imagen del lugar.
01	El sistema inicia el reconocimiento de imagen del lugar.				
02	El sistema busca el reconocimiento de imagen del lugar.				
ERROES / ALTERNATIVAS	<table border="1"> <tr> <td>01</td><td>El reconocimiento de imagen, no tiene un enfoque correcto y claro del lugar turístico.</td></tr> <tr> <td>02</td><td>El reconocimiento de imagen no se encuentra dentro de la base de datos</td></tr> </table>	01	El reconocimiento de imagen, no tiene un enfoque correcto y claro del lugar turístico.	02	El reconocimiento de imagen no se encuentra dentro de la base de datos
01	El reconocimiento de imagen, no tiene un enfoque correcto y claro del lugar turístico.				
02	El reconocimiento de imagen no se encuentra dentro de la base de datos				
POSCONDICION	Ninguna				
NOTAS	Ninguna				

NOMBRE	04.-Visualizar la información relevante, del Sitio/Lugar Turístico						
DESCRIPCION	El usuario indico que desea saber información relevante del sitio turístico de su preferencia.						
PRECONDICION	El usuario desea saber todo lo relacionado al sitio turístico en turno.						
SECUENCIA PRINCIPAL	<table border="1"> <tr> <td>01</td><td>Se despliega en forma resumida, datos, reseñas, antecedentes históricos importantes del lugar en turno.</td></tr> <tr> <td>02</td><td>Se posee una opción de habilitación mediante audio en diversos idiomas, sobre los datos, reseñas, antecedentes históricos importantes del lugar en turno.</td></tr> </table>	01	Se despliega en forma resumida, datos, reseñas, antecedentes históricos importantes del lugar en turno.	02	Se posee una opción de habilitación mediante audio en diversos idiomas, sobre los datos, reseñas, antecedentes históricos importantes del lugar en turno.		
01	Se despliega en forma resumida, datos, reseñas, antecedentes históricos importantes del lugar en turno.						
02	Se posee una opción de habilitación mediante audio en diversos idiomas, sobre los datos, reseñas, antecedentes históricos importantes del lugar en turno.						
ERROES / ALTERNATIVAS	<table border="1"> <tr> <td>01</td><td>La información acerca del sitio turístico, no sea y no se muestre de forma correcta.</td></tr> <tr> <td>02</td><td>Al momento de reproducir un audio en un idioma seleccionado, no sea el verdadero.</td></tr> <tr> <td>03</td><td>La información acerca del sitio turístico, no se detecten dentro de la base de datos.</td></tr> </table>	01	La información acerca del sitio turístico, no sea y no se muestre de forma correcta.	02	Al momento de reproducir un audio en un idioma seleccionado, no sea el verdadero.	03	La información acerca del sitio turístico, no se detecten dentro de la base de datos.
01	La información acerca del sitio turístico, no sea y no se muestre de forma correcta.						
02	Al momento de reproducir un audio en un idioma seleccionado, no sea el verdadero.						
03	La información acerca del sitio turístico, no se detecten dentro de la base de datos.						
POSCONDICION	Ninguna						
NOTAS	Ninguna						

NOMBRE	05.- Puntos de Recomendación				
DESCRIPCION	Es una sección que se le recomienda al usuario si desea visitar, recorrer, ir a otros puntos que estén alrededor del sitio turístico.				
PRECONDICION	Dicha sección estará ahí siempre y cuando el usuario desee ir a algún punto de recomendación.				
SECUENCIA PRINCIPAL	<table border="1"> <tr> <td>01</td><td>El sistema inicia la recomendación de diversos o algunos puntos de interés, alrededor de la zona del sitio turístico visitado.</td></tr> </table>	01	El sistema inicia la recomendación de diversos o algunos puntos de interés, alrededor de la zona del sitio turístico visitado.		
01	El sistema inicia la recomendación de diversos o algunos puntos de interés, alrededor de la zona del sitio turístico visitado.				
ERROES / ALTERNATIVAS	<table border="1"> <tr> <td>01</td><td>Los puntos de interés cercanos al sitio turístico, no sean los correctos.</td></tr> <tr> <td>02</td><td>Los puntos de interés no se encuentren dentro de la base de datos.</td></tr> </table>	01	Los puntos de interés cercanos al sitio turístico, no sean los correctos.	02	Los puntos de interés no se encuentren dentro de la base de datos.
01	Los puntos de interés cercanos al sitio turístico, no sean los correctos.				
02	Los puntos de interés no se encuentren dentro de la base de datos.				
POSCONDICION	Ninguna				
NOTAS	Ninguna				

NOMBRE	06.- Actualizar otro sitio/lugar turístico								
DESCRIPCION	El usuario indico que desea enfocar y saber todo lo relacionado a otro sitio turístico de su preferencia.								
PRECONDICION	El usuario desea conocer todo lo referido a otro sitio turístico en turno.								
SECUENCIA PRINCIPAL	<table border="1"> <tr> <td>01</td><td>El usuario apunta al lugar turístico enfrente de él.</td></tr> <tr> <td>02</td><td>El sistema inicia el reconocimiento de imagen del lugar.</td></tr> <tr> <td>03</td><td>Se despliega en forma resumida, datos, reseñas, antecedentes históricos importantes del lugar en turno.</td></tr> </table>	01	El usuario apunta al lugar turístico enfrente de él.	02	El sistema inicia el reconocimiento de imagen del lugar.	03	Se despliega en forma resumida, datos, reseñas, antecedentes históricos importantes del lugar en turno.		
01	El usuario apunta al lugar turístico enfrente de él.								
02	El sistema inicia el reconocimiento de imagen del lugar.								
03	Se despliega en forma resumida, datos, reseñas, antecedentes históricos importantes del lugar en turno.								
ERROES / ALTERNATIVAS	<table border="1"> <tr> <td>01</td><td>El reconocimiento de imagen, no tiene un enfoque correcto y claro del lugar turístico.</td></tr> <tr> <td>02</td><td>El reconocimiento de imagen no se encuentra dentro de la base de datos</td></tr> <tr> <td>03</td><td>La información acerca del sitio turístico, no sea y no se muestre de forma correcta.</td></tr> <tr> <td>04</td><td>El dispositivo del usuario no posee, con una cámara con buena resolución.</td></tr> </table>	01	El reconocimiento de imagen, no tiene un enfoque correcto y claro del lugar turístico.	02	El reconocimiento de imagen no se encuentra dentro de la base de datos	03	La información acerca del sitio turístico, no sea y no se muestre de forma correcta.	04	El dispositivo del usuario no posee, con una cámara con buena resolución.
01	El reconocimiento de imagen, no tiene un enfoque correcto y claro del lugar turístico.								
02	El reconocimiento de imagen no se encuentra dentro de la base de datos								
03	La información acerca del sitio turístico, no sea y no se muestre de forma correcta.								
04	El dispositivo del usuario no posee, con una cámara con buena resolución.								
POSCONDICION	Ninguna								
NOTAS	Ninguna								

Reglas del Business

Código	Nombre	Descripción	Dinámicas	Estáticas
Reglas de Acceso				
RAN01	Usuarios (Turistas)	Solo los usuarios que han sido registrados en la BD podrán acceder a las funcionalidades de la aplicación	<input type="checkbox"/>	<input checked="" type="checkbox"/>
RAN02	Usuarios NO registrados	Los futuros usuarios deberán llenar un formulario para añadirlos a la BD	<input type="checkbox"/>	<input checked="" type="checkbox"/>
RAN03	Desarrolladores	Serán designados por el representante del equipo al que se integrará el nuevo developer, se entregará solicitud a su supervisor	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Código	Nombre	Descripción	Dinámicas	Estáticas
Reglas de Permisos				
RPN01	Modificación de la BD	Solo los miembros autorizados pueden modificar la BD	<input type="checkbox"/>	<input checked="" type="checkbox"/>
RPN02	Visualizar la información relevante, del Sitio/Lugar Turístico	Corresponde a la descripción relevantes, reseñas históricas, datos curiosos, de cada uno de los Lugares Turísticos.	<input type="checkbox"/>	<input checked="" type="checkbox"/>

RPN03	Ingreso al Proyecto de Software	Cualquier requerimiento emitido y aceptado por el equipo de "guiAR" puede incorporarse al grupo del proyecto de software que implementan en "guiAR"	<input type="checkbox"/>	<input checked="" type="checkbox"/>
--------------	---------------------------------	---	--------------------------	-------------------------------------

Modelado con UML

Diagrama de Caso de Uso "guiAR"

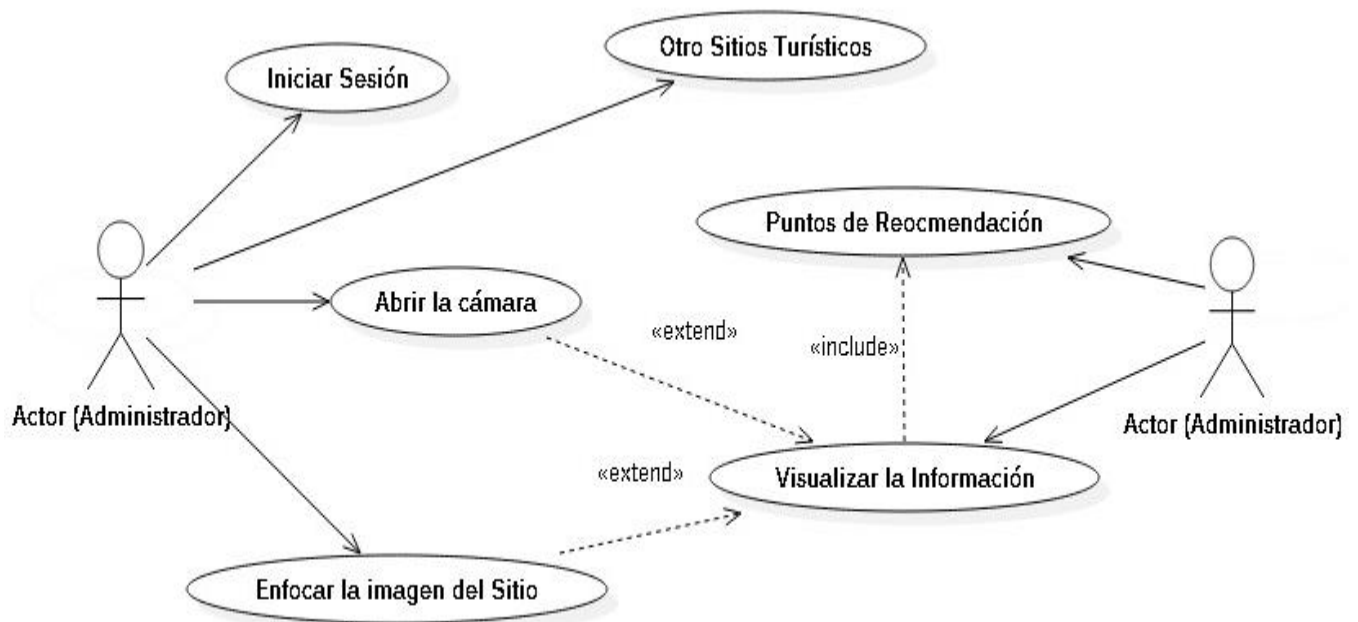


Diagrama de Clases “guiAR”

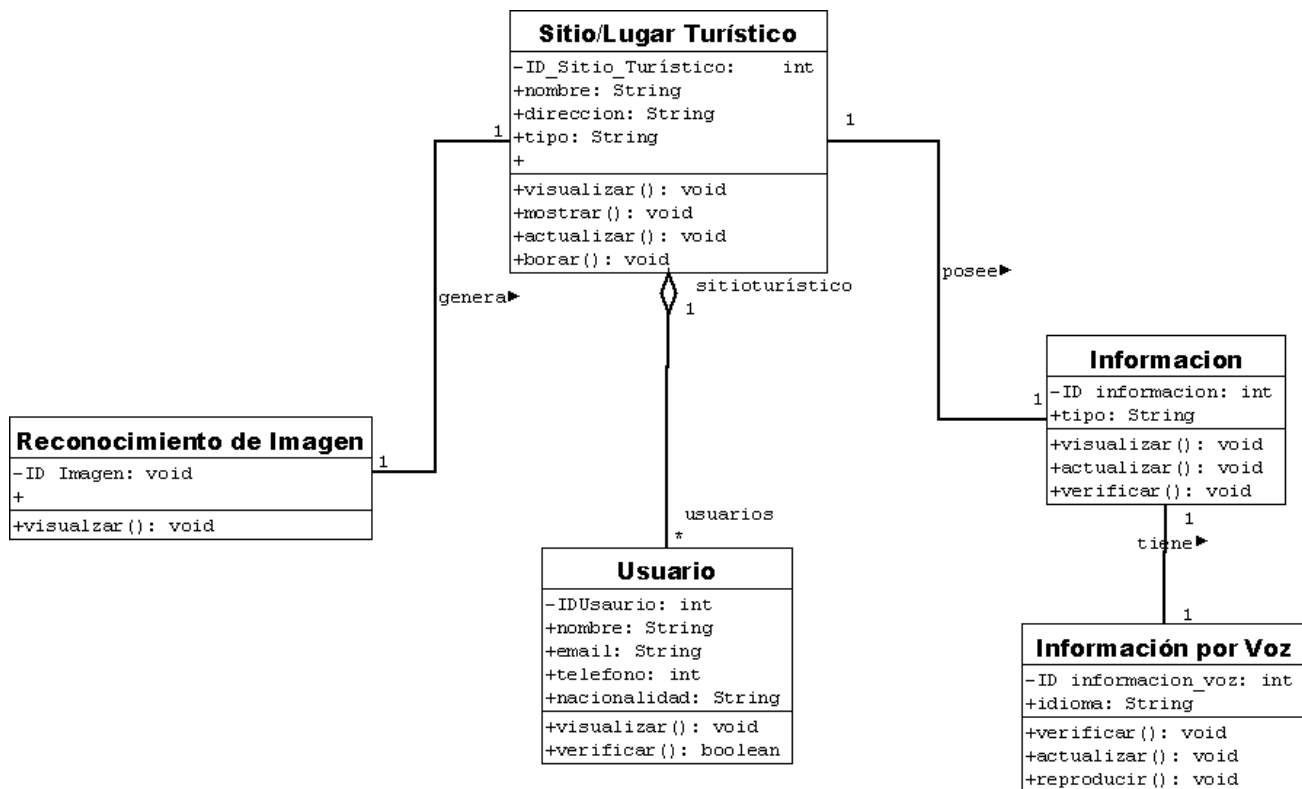


Diagrama de Secuencia “guiAR”

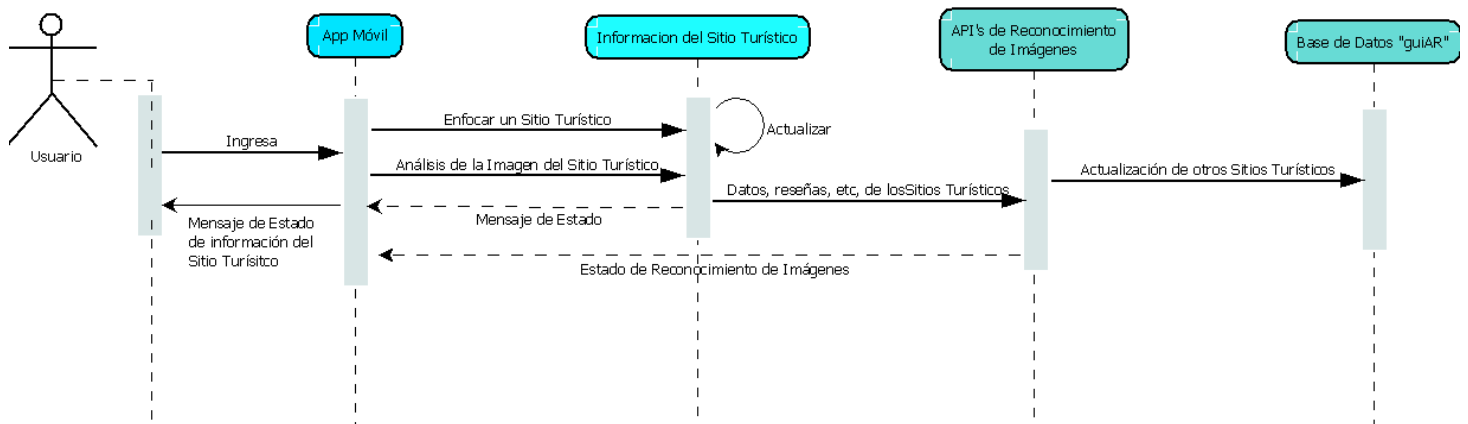


Diagrama de Estados “guiAR”

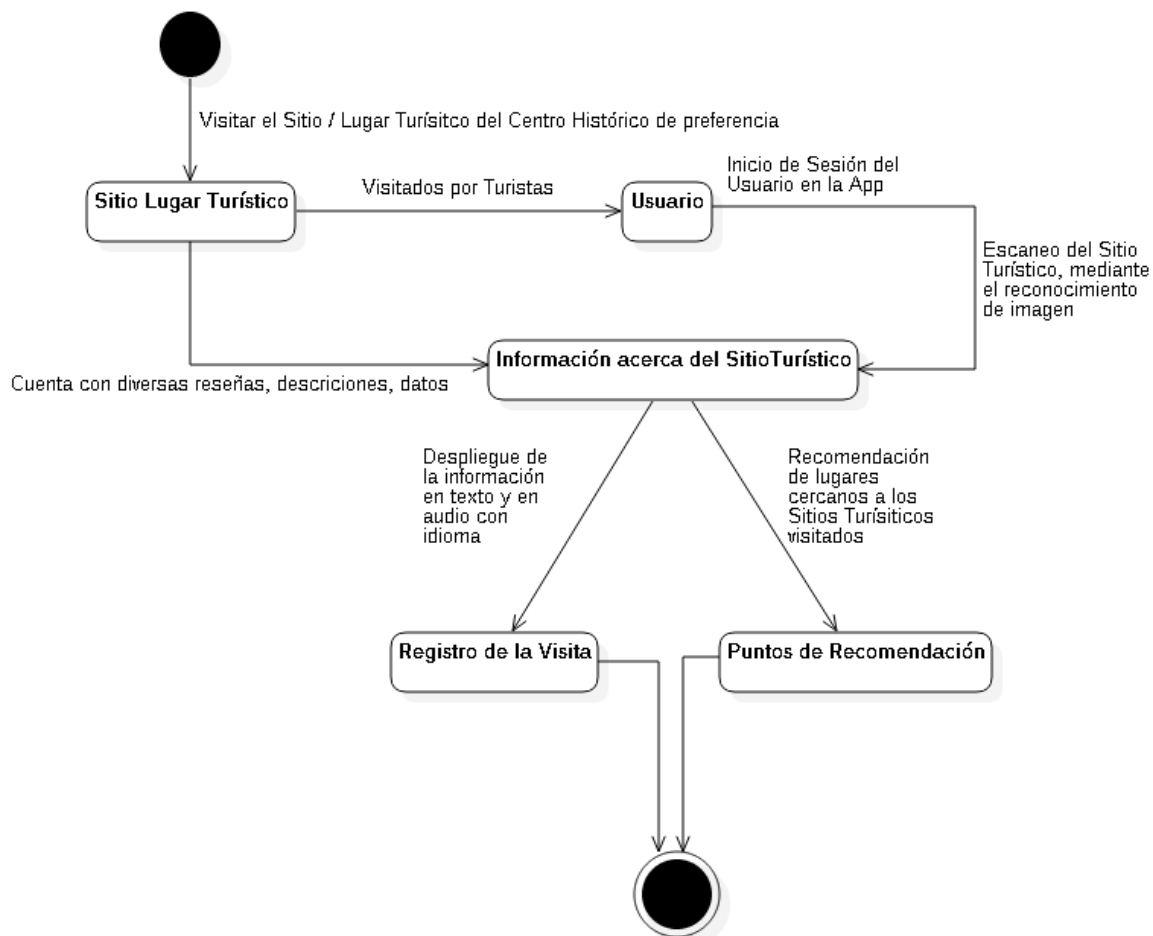
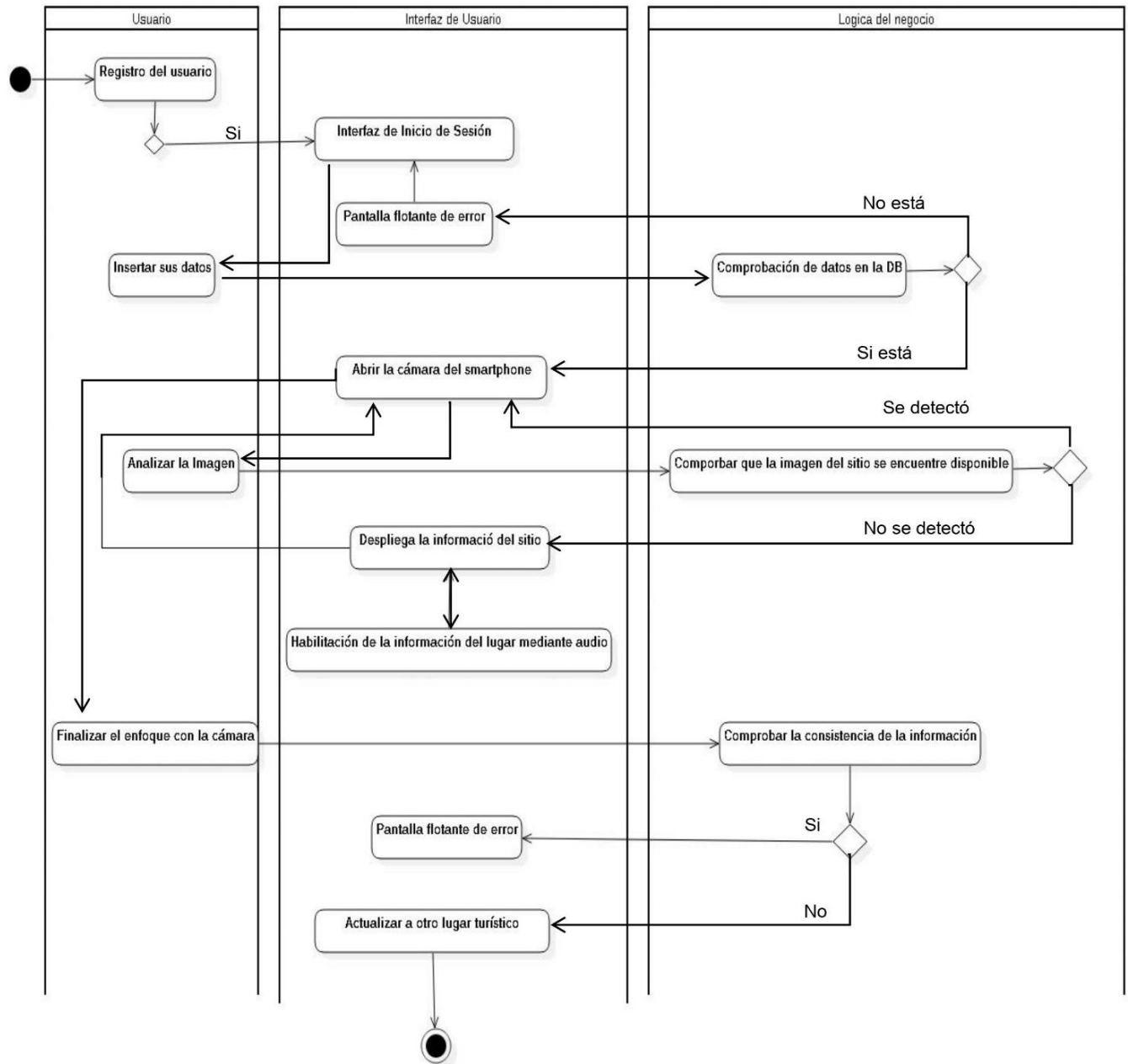


Diagrama de Actividades “guiAR”



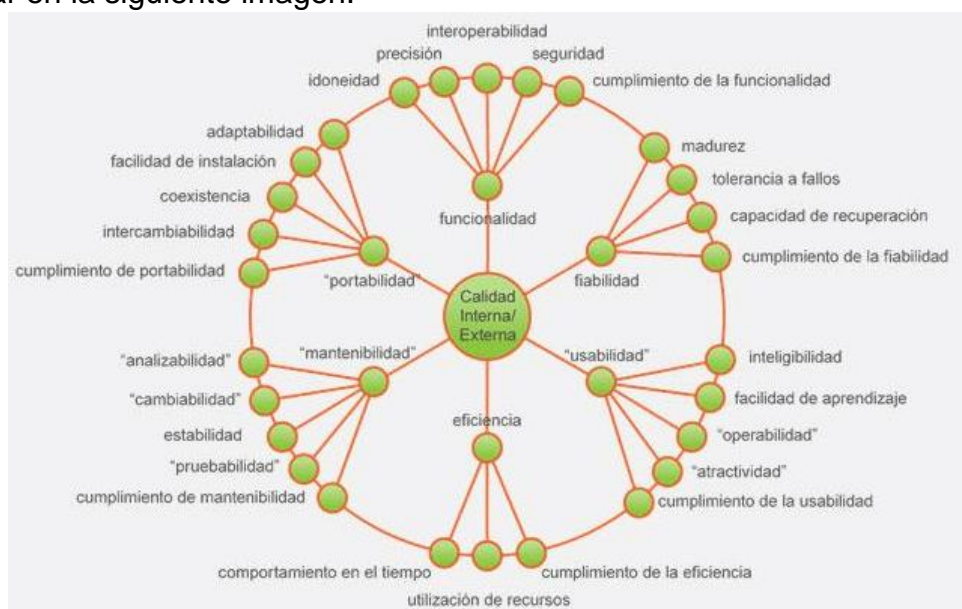
Capítulo 5: ADMINISTRACIÓN DE LA CALIDAD

X. CALIDAD DEL PROYECTO DE SOFTWARE “guiAR”

El concepto de calidad de software, según Pressman se asocia a la "concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos con los estándares de desarrollo plenamente documentados y con las características implícitas que se espera de todo software desarrollado profesionalmente", con base en los requisitos funcionales y no funcionales identificados en la etapa de análisis del sistema, insumo principal para implementar dichos requisitos con los atributos mínimos de calidad, fomentando la aplicación de procesos estandarizados y criterios necesarios en cada una de sus etapas, así se fomenta que el avance en el ciclo de vida del software minimice el riesgo de fracaso del proyecto. Por su parte, el Instituto de Ingenieros Eléctricos y Electrónicos (IEEE, 1990) define calidad de software como "el grado con el que un sistema, componente o proceso cumple los requerimientos especificados y las necesidades o expectativas del cliente o usuario", denotando que el énfasis radica en los requisitos específicos del sistema y en la búsqueda de la satisfacción del cliente.

La obtención de un software con calidad implica la utilización de metodologías o procedimientos estándares para el análisis, diseño, programación y prueba del software que permitan uniformar la filosofía de trabajo, en aras de lograr una mayor confiabilidad, mantenibilidad y facilidad de prueba, a la vez que eleven la productividad, tanto para la labor de desarrollo como para el control de la calidad del software.

La calidad es sinónimo de eficiencia, flexibilidad, corrección, confiabilidad, mantenibilidad, portabilidad, usabilidad, seguridad e integridad, como se puede observar en la siguiente imagen.



Es por eso, que a continuación se van a describir estos conceptos, respecto a nuestro proyecto “guiAR”.

Funcionalidad: El software cumple y provee las funciones que satisfacen las necesidades explícitas (conocer sobre un sitio turístico en particular) e implícitas (una interfaz amigable para el usuario) cuando es utilizado.

Veamos los criterios de la funcionalidad: en primer lugar, tenemos a la **idoneidad**, en este caso, el software provee un adecuado conjunto de funciones que cumplen las tareas y objetivos especificados por el usuario, es decir, obtener la información necesaria y pertinente de un sitio turístico de interés por parte del usuario.

Después tenemos a la **precisión**, en este caso, el software hace los procesos pertinentes y entrega los resultados solicitados con precisión o de forma esperada, siempre y cuando el hardware lo permita (la cámara) y exista información del sitio turístico en nuestra base de datos.

Seguimos con la **interoperabilidad**, la cual no es una capacidad de nuestro software, ya que no interactúa con uno o más sistemas específicos.

Después respecto a la **seguridad**, nuestro software tiene la capacidad para proteger la información y los datos de manera que los usuarios o los sistemas no autorizados no puedan acceder a ellos para realizar operaciones, esto se logra gracias a que cuando el usuario se registra por primera vez tendrá que confirmar el correo electrónico que especificó y crear una contraseña segura y con esto podrá iniciar sesión de manera segura.

Finalmente, el último punto, **cumplimiento de la funcionalidad**, el software tiene la capacidad de cumplir los estándares referentes a la funcionalidad, ya que cumple la mayoría de los criterios especificados.

Fiabilidad: El sistema es capaz de desempeñar las funciones especificadas, usándose durante un periodo determinado. Veamos cada una de las características: Nuestro sistema tiene la capacidad de **madurez** ya que satisface las necesidades de fiabilidad en condiciones normales, por ejemplo, el usuario puede apuntar a diferentes sitios turísticos para conocer más acerca de ellos.

Nuestro sistema no tiene la capacidad de **tolerancia a fallos**, ya que no podría operar en presencia de fallos de hardware como la cámara o software como el no encontrar el sitio turístico elegido por el usuario porque no existe en nuestra base de datos.

Nuestro sistema si tiene la **capacidad de recuperación**, ya que, por ejemplo, cuando no se encuentra un sitio turístico, el sistema se restablece al estado deseado por el fallo ocurrido y nuevamente el usuario puede apuntar a otro sitio turístico de su interés.

Finalmente, el **cumplimiento de la fiabilidad** en nuestro sistema es el correcto dada las funciones deseadas por el usuario.

Usabilidad: El software además de cubrir los requerimientos funcionales para los cuales fue creado, también es de fácil uso para el usuario, ya que el software debe ser el mejor manual de uso para sí mismo, es decir que no se requiera un manual de uso externo al software puesto que este mismo guía al usuario y permite la intuitividad de su uso.

Veamos cada uno de sus criterios: nuestro software cuenta con **inteligibilidad**, ya que le permite al usuario entender como puede ser usado para condiciones particulares, por ejemplo, cuando ocurre algún error, también tiene **facilidad de aprendizaje** ya que es intuitivo el uso y le va mostrando mensajes claros para que el usuario elija lo que desea, de la misma manera, tiene **operabilidad** debido a que el software no solo cumple con cubrir la necesidad del usuario sino que también le ayuda a que el proceso que debe cumplir sea rápido, por lo tanto, para el usuario operar cada funcionalidad del sistema software es simple. Igualmente, tiene la característica de **atractividad**, ya que la presentación del software está acorde a la personalidad de la app, es decir, el diseño es el apropiado para una aplicación de este tipo, con los colores, botones, posiciones adecuadas, etc. Finalmente, el **cumplimiento de la usabilidad** en nuestro sistema es el correcto, cumple con las expectativas de los usuarios.

Eficiencia: El software tiene esta habilidad para responder a una petición de usuario con la velocidad apropiada. Veamos sus atributos: el **comportamiento en el tiempo** es una capacidad que tiene el software ya que proporciona tiempos de respuesta, tiempos de proceso y potencia apropiados, por ejemplo, cuando el usuario hace la petición de mostrar la información del sitio turístico, la app lo hace con buena velocidad, para que el usuario la tenga al instante de hacer la petición. También hace una buena **utilización de recursos**, usa lo necesario para llevar a cabo cada función. Finalmente, respecto al **cumplimiento de la eficiencia**, nuestro producto software se adhiere a las normas o convenciones relacionadas con la eficiencia como veremos más adelante.

Mantenibilidad: Se refiere a la habilidad del software para que el usuario invierta el mínimo esfuerzo para mantenerlo o mejorarlo. Veamos sus atributos:

Nuestro software tiene la capacidad de **analizabilidad**, ya que permite hacerle diagnósticos de deficiencias o causas de los fallos en el software, o para identificar las partes que han de ser modificadas.

También tiene la capacidad de **confiabilidad**, permite que una determinada modificación sea implementada.

Igualmente, la capacidad de **estabilidad** está presente porque evita efectos inesperados debidos a modificaciones del software, siempre y cuando, haya una implementación correcta de estas y sea para mejorar la aplicación.

La capacidad de **pruebabilidad** está ligada a la anterior porque permite que el software modificado sea validado.

Respecto al **cumplimiento de la mantenibilidad**, nuestro software lo hace bien porque se adhiere a las normas o convenciones relacionadas con la mantenibilidad.

Portabilidad: Se refiere a la habilidad del software para ser transferido de un ambiente a otro y funcionar en este. En nuestro sistema, existe portabilidad porque mediante la app puede funcionar en cualquier teléfono Android considerando una cámara relativamente buena para que funcione correctamente la realidad aumentada. Veamos sus características:

Existe **adaptabilidad** porque puede ser adaptado a diferentes entornos especificados, también tiene la **facilidad de instalación** porque simplemente se puede instalar como cualquier otra app directamente, tiene la capacidad de **coexistencia** porque puede funcionar compartiendo mismos recursos con el sistema, sin que afecte el resultado o funcionamiento del mismo, por ejemplo, cuando se oprima la opción de escuchar la información con audio, se podría también escuchar una canción del teléfono a la par y no habría inconvenientes.

También tiene la capacidad de **intercambiabilidad** porque puede ser usado en lugar de otro producto software, para el mismo propósito, en el mismo entorno, por ejemplo, se pueda usar la app para abrir la cámara propia del celular y tomar fotografías, aunque no sea para esto, se podría hacer.

Por lo tanto, el software tiene el **cumplimiento de la portabilidad**, por todo lo anterior ya mencionado.

XI. NORMAS ISO RELACIONADAS AL PROYECTO “guiAR”

ISO/IEC-9126 Modelo de calidad del producto de software

En esta norma, se establecen atributos que permiten calificar si un producto de software maneja de manera adecuada el conjunto de funciones que satisfagan Las necesidades para las cuales fue diseñado.

Ya que nuestra aplicación será usada a una gran cantidad de usuarios (Nacionales y extranjeros) es prácticamente obligatorio cumplir con esta norma.

Recordemos que las características que requiere nuestra aplicación, como lo marca esta norma son:

1. Funcionalidad:

2. Confiabilidad

3. Usabilidad

4. Eficiencia

5. Mantenibilidad

6. Portabilidad

7. Satisfacción

ISO/IEC-14598

La serie de estándares de la ISO/IEC 14598 proveen métodos para las mediciones y evaluaciones de la calidad del producto del software. No describen ni los métodos para evaluar los procesos de producción.

Esta norma está diseñada para ser usada por peritos, adquisidores y evaluadores independientes, particularmente para aquellos que se responsabilizan de la evaluación de productos de software. Los resultados de la evaluación producidos por la aplicación de los estándares de la ISO/IEC 14598 podremos usarlos para medir el “acatamiento” de los requisitos y para hacer mejoras en donde sea necesario.

Los resultados de la evaluación también podremos usarlos para establecer las relaciones fundamentales entre las métricas internas y las externas. También se puede usar los resultados de la evaluación para determinar cómo los procesos pueden ser mejorados por medio del estudio y la evaluación de la información de la calidad del producto.

ISO/IEC 25000

El objetivo de esta norma es organizar, enriquecer y unificar las series que cubren dos procesos principales: especificación de requisitos de calidad del software y evaluación de la calidad del software, soportada por el proceso de medición de calidad del software.

La ISO/IEC 25000 es el resultado de la evolución de otras normas en especial de las normas ISO 9126 e ISO 14598 y está compuesta por 5 divisiones

Dado que esta es la evolución de las normas que se mencionaron anteriormente, tendremos dos opciones para nuestro proyecto, la primera es guiarnos por las normas ISO 9126 e ISO 14598, y la otra es optar por usar esta norma, lo cual sería lo óptimo para garantizar la calidad de nuestro software mediante las normas actualizadas.

ISO 9000

Esta norma se ocupará en nuestro proyecto ya que es un documento el cual describirá todos los requisitos necesarios para crear y mantener un sistema de gestión de la calidad.

El aplicar esta norma, nos aportara beneficios como la capacidad de que nuestra aplicación cumpla con requisitos legales, facilitar oportunidades para mejorar la satisfacción del cliente, entre otros.

ISO 21500

Esta norma funcionara más como una guía para elaborar el proyecto, ya que su objetivo es recomendar a las distintas compañías una forma profesional de elaborar proyectos cumpliendo con los estándares más exigentes, demandados por el mercado global. Esta norma ofrece una descripción de alto nivel acerca de los diferentes conceptos y procesos que componen las buenas prácticas en dirección y gestión de proyectos.

XII. NORMAS IEEE RELACIONADAS AL PROYECTO “guiAR”

829-1998-IEEE Documentación de prueba de software

Se describe un conjunto de documentos básicos de pruebas de software. Se compone de un contenido de documentos de prueba individuales.

1008-IEEE Actividades involucradas en el proceso de las pruebas de software.

La aplicación de esta norma es muy importante ya se planea el enfoque general, fuentes y calendarios, determina las características a ser probadas, refina el plan general, diseña el conjunto de pruebas, implementa el plan y diseño refinado, ejecuta los procedimientos de prueba, verifica la terminación y evalúa el esfuerzo de la prueba.

1058.1-IEEE

Se hará uso de este estándar ya que especifica el formato y contenidos de los planes para la gestión de proyectos de software. Como lo indica este estándar, no se especifican las técnicas exactas que pueden ser usadas en el desarrollo de los planes de proyectos. Debemos desarrollar un conjunto de prácticas y procedimientos para proporcionar una guía detallada para la preparación y actualización de los planes de la gestión del proyecto.

1233-1998 – IEEE Desarrollo y especificaciones de los requisitos del sistema

Este estándar proporciona orientación para el desarrollo del conjunto de requisitos. El desarrollo incluye la identificación, organización, presentación, y la modificación de los requisitos. También se establecen las condiciones para la incorporación de los conceptos operacionales, las limitaciones de diseño, y los requisitos de configuración de diseño en la especificación. Al ofrecer orientación para el desarrollo del conjunto de requisitos principalmente, es que consideramos que este estándar se acopla a nuestro proyecto.

1012-1998 - IEEE Verificación y validación de procesos de software

Con este estándar, se determina si los productos desarrollados en una determinada actividad se ajustan a los requisitos de esa actividad, y si el software satisface sus necesidades de uso y usuarios previstos. Esto es muy importante ya que se garantiza que el producto final entregue todo lo que ofrece, de manera correcta y no quede a deber nada al usuario final.

Capítulo 6: PROTOTIPADO, COMPOSICIÓN E INTEGRACIÓN DEL SISTEMA

XIII. MODELADO Y DISEÑO DEL SISTEMA

Planificación del Proceso

Nuestro planificador del proceso consistió en las siguientes entradas y/o parámetros:

✓ **Prediseño:** Consistió como una pre-propuesta en cómo sería el diseño visual, e intuitivo. Es por ello por lo que, se realizó para tener una idea más clara, la facilidad del manejo y la apariencia, aquí combinación de colores, tamaño de la letra y la ubicación del espacio. La base de nuestro diseño se establece en la interacción entre el usuario con la aplicación móvil, y a su vez supervisar que se cumplan dicha interacción en los procesos mencionados y planteados en capítulos anteriores.

✓ **Diseño:** Se tomó en cuenta un entorno agradable y cómodo hacia o para el usuario, con colores contrastantes y un diseño fácil y sencillo que pueda utilizar sin problema.

✓ **Desarrollo:** Aquí se realizó toda la programación de la Aplicación, cómo la Base de Datos, de los sitios/lugares turísticos, el reconocimiento de imágenes y la información relevante, reseñas, datos históricos, tanto en forma de texto, cómo en audio en diversos idiomas, para así tener una Aplicación Móvil (prototipo) funcional, así como las verificaciones de prueba y error para saber qué tipo de detalles se tenían que realizar para después visualizar las correcciones y saber de esa manera tener la Aplicación Móvil funcionando.

Tipo de Interfaz de Usuario

La interfaz que “guiAR” incluye, va desde la más empleada, a la más novedosa, puesto que la Aplicación Móvil, es más software, y la interfaz que se relaciona como es:

↳ **Interfaz Gráfica de Usuario:** Se entiende que es un programa (software), de entorno visual de imágenes y objetos mediante el cual una computadora y un usuario interactúan cómo se ha venido planteando con anterioridad la Aplicación Móvil “guiAR” este concepto es clave y fundamental, puesto que la Aplicación mostrará de forma visual y amigable todas las opciones disponibles y posibles para el usuario, como también que ellos puedan interactuar con mayor facilidad, eficacia y rapidez. Sin la necesidad de esperar y hacer largas filas y la posibilidad de errores en el precio de los productos.

Entrada, Proceso y Salida del Sistema

- ✓ **Entrada:** El usuario, ingresa a la aplicación, mediante un dispositivo móvil y comienza a interactuar tanto de manera visual y manual con la aplicación.
- ✓ **Proceso:** Cuando el usuario esté pasando y se encuentre frente a un sitio o lugar turístico y durante su estadía de ese lugar, éste tendrá que sacar su smartphone abrir la cámara e iniciar a enfocar dicho sitio turístico, que desee saber todo lo relacionado de él, lo anterior, mediante el reconocimiento de imágenes implementado en la aplicación, además de realizar el proceso, esté podrá de igual manera ir a otros puntos de interés que estén alrededor y cercanos a la zona de dicho sitio turístico, siempre y cuando lo prefiera.
- ✓ **Salida:** En este punto, ya que el usuario haya decidido enfocar el sitio turístico de su preferencia, le mostrará la información en forma de texto, reseñas, datos históricos más importantes e interesantes ligados a éste de igual manera, dicha información estará disponible, mediante audios en diversos idiomas y esto es para todas aquellas personas que sean extranjeras que no dominen al 100% el idioma español. Y es así cómo finaliza el procedimiento del modo de empleo de la aplicación móvil “guiAR”.

Prototipo de Prediseño de la Aplicación Móvil “guiAR”



Descripción: En esta primera vista, se presenta la pantalla de inicio de la Aplicación del Asistente Turístico de Sitios Culturales “guiAR”.

Descripción: Esta es la segunda vista, donde se visualiza la pantalla de Inicio de Sesión a la aplicación, por parte de los usuarios (turistas). Y un botón para poderse registrarse por primera vez, como también otras formas de Registrarse.



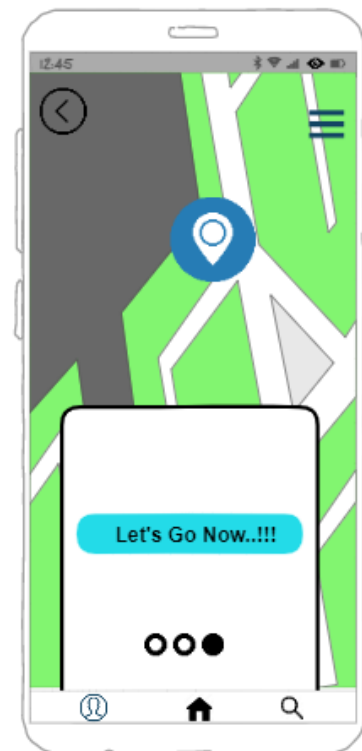
Descripción: Esta se observa la tercera vista de la aplicación, donde va en conjunto con la segunda vista, dado que se encuentra el registro de los datos para que el usuario pueda darse de alta en la aplicación y pueda disfrutar de la misma.

Descripción: Esta es la cuarta vista de la aplicación, donde se le desplegará al usuario, los permisos correspondientes, que concederá para poder hacer uso de esta en su smartphone.



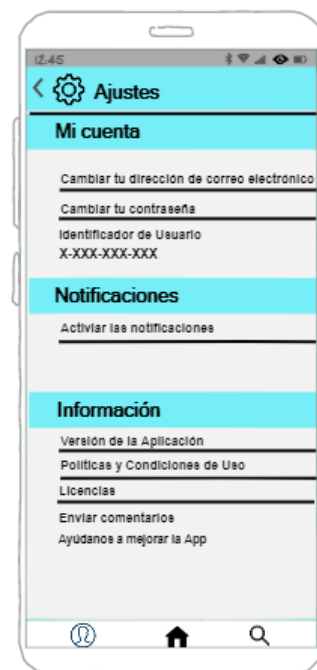
Descripción: En esta quinta vista de la aplicación, se le dará la bienvenida como los agradecimientos al usuario, por ser parte de la aplicación, cómo también un minitutorial, acerca del manejo de la aplicación "guiAR".

Descripción: En esta sexta vista, se presenta la ubicación en donde se encuentra el usuario, y también de poder enfocar al sitio turístico en que se sitúa para iniciar el reconocimiento de la imagen del lugar turístico visitado y se despliegue toda la información relevante, importante, histórica, acerca del mismo.



Descripción: En esta séptima vista se aprecia el acceso al menú de la aplicación, donde se encontrará con diversas opciones que el usuario podrá dirigirse si así lo desea, como son el acceso a la cuenta, guardado, ajustes y ayuda.

Descripción: En esta octava vista, se visualiza la pantalla de las distintas opciones dentro de la pestaña u opción de Ajustes, que el usuario podrá tener acceso, tales como son "Mi cuenta", "Notificaciones", "Información de la App", si así lo desea.

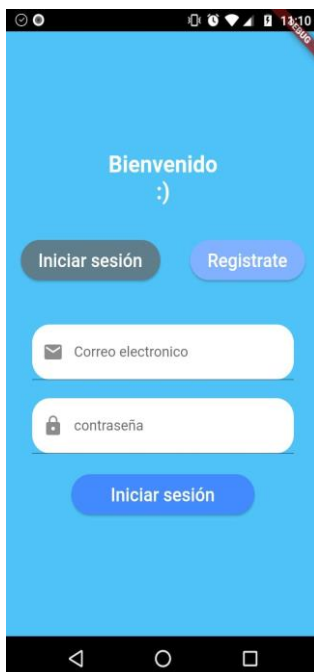


Prototipo de Diseño Final de la Aplicación Móvil “guiAR”



Descripción: En esta primera vista, se presenta la pantalla de inicio de la Aplicación del Asistente Turístico de Sitios Culturales “guiAR”.

Descripción: Esta es la segunda vista, donde se visualiza la pantalla de Inicio de Sesión a la aplicación, por parte de los usuarios (turistas). Y un botón para poderse registrarse por primera vez, como también otras formas de Registrarse.



Descripción: Esta se observa la tercera vista de la aplicación, donde va en conjunto con la segunda vista, dado que se encuentra el registro de los datos para que el usuario pueda darse de alta en la aplicación y pueda disfrutar de la misma.



Descripción: Esta es la cuarta vista de la aplicación, donde se le desplegará al usuario, los permisos correspondientes, que concederá para poder hacer uso de esta en su smartphone.

Descripción: En esta quinta vista de la aplicación, se le dará la bienvenida como los agradecimientos al usuario, por ser parte de la aplicación, cómo también un minitutorial, acerca del manejo de la aplicación “guiAR”.



Descripción: En esta sexta vista de la aplicación, se visualiza el reconocimiento de la imagen del Sitio/ Turístico escaneado. Cómo es el Palacio de Bellas Artes.

Capítulo 7: ASEGURAMIENTO E IMPLEMENTACIÓN DE LA CALIDAD

XIV. IMPLEMENTACIÓN DE CALIDAD/ SATISFACTORIA

El proceso de primero asegurarse de que el sistema de información sea operacional, y permitir que luego tomen los usuarios control de la operación para su uso y evaluación es llamado implementación.

El primer enfoque para la implementación se refiere al movimiento del poder de cómputo a usuarios individuales, poniendo un centro de información (IC) o dándole poder de cómputo y responsabilidad a los grupos a lo largo del negocio con la ayuda de la computación distribuida.

El segundo enfoque para la implementación es el uso de diferentes estrategias para el entrenamiento de los usuarios y el personal del centro de información, incluyendo el hablarles en su propio nivel, usando una diversidad de técnicas de entrenamiento.

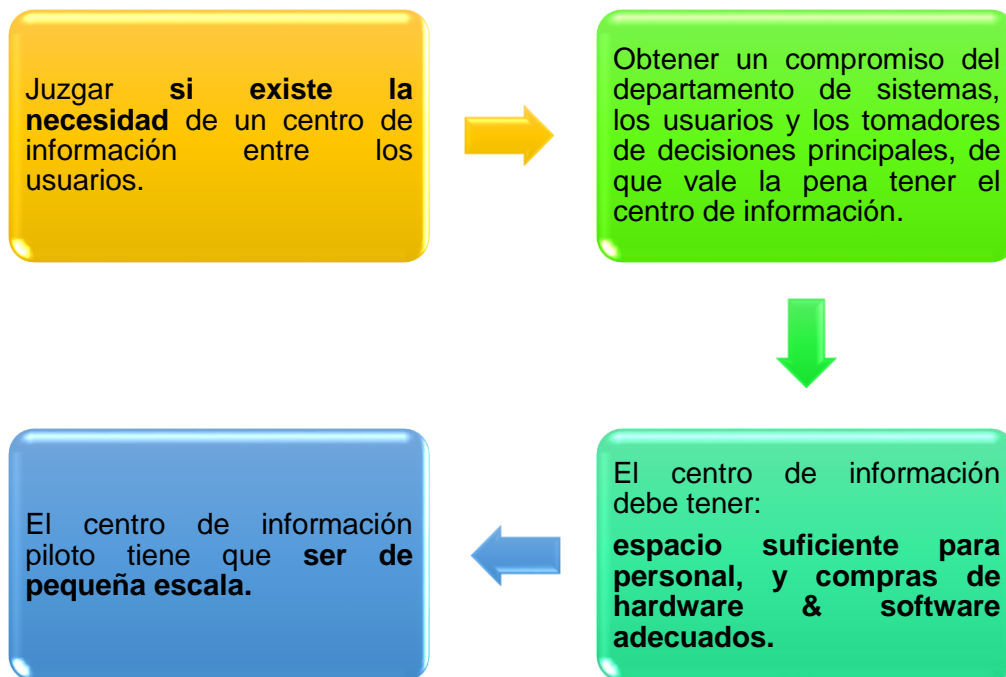
El tercer enfoque para la implementación es la selección de una estrategia de conversión.

El cuarto enfoque para la implementación involucra la evaluación del sistema de información nuevo o modificado o el centro de información. Las evaluaciones vienen del personal del centro de información, usuarios, administración y los mismos analistas.

Creación del Centro de Información

La creación e implementación de un centro de información se propone como una forma para facilitar a los usuarios que satisfagan sus necesidades de información a corto plazo y que, al mismo tiempo, todavía reciban soporte del departamento de sistemas de información.

Los cuatro pasos necesarios para el establecimiento de un centro de información se listan a continuación:



Topología de Red

Red de Estrella: Es la configuración popular para la computación distribuida es la red de estrella. Un mainframe, PC o estación de trabajo se designa como el nodo central. Como tal, se comunican con los nodos menores, pero no se pueden comunicar directamente entre sí. Al requerirse que las PC's se comuniquen entre sí, se lograría con una PC enviando datos al nodo central, el cual pasaría los datos a la segunda PC.

La topología en estrella se aprovecha de la disposición natural del cableado de los edificios. Generalmente, es mejor para distancias cortas y puede ofrecer velocidades elevadas a un número pequeño de dispositivos. En la topología en estrella con pares trenzados es posible utilizar un esquema de detección de colisiones mucho más sencillo. En este caso, la detección de colisiones se basa en magnitudes lógicas en lugar de utilizar niveles de tensión. Se determina que hay colisión si en cualquiera de los concentradores (hubs) hay actividad (señal) en más de una entrada, generándose en este caso una señal especial denominada señal de presencia de colisión.

Especificaciones IEEE 802.3 10 Mbps (Ethernet):

- ↪ **10BASE-T:** utiliza par trenzado no apantallado en una topología en estrella. Dada la alta velocidad y la baja calidad de las transmisiones sobre este tipo de cable, la longitud de cada enlace se restringe a 100 m. Como alternativa se puede utilizar un enlace de fibra óptica, en cuyo caso la longitud máxima es de 500 m.
- ↪ **10BASE-F:** contiene tres especificaciones: una topología en estrella pasiva para la interconexión de estaciones y repetidores con segmentos de hasta 1 km de longitud; un enlace punto a punto que puede ser usado para conectar estaciones o repetidores separados hasta 2 km; y un enlace punto a punto que puede usarse para conectar repetidores a una distancia máxima de 2 km.

A continuación se mostrará una configuración o topología de distribución de red:

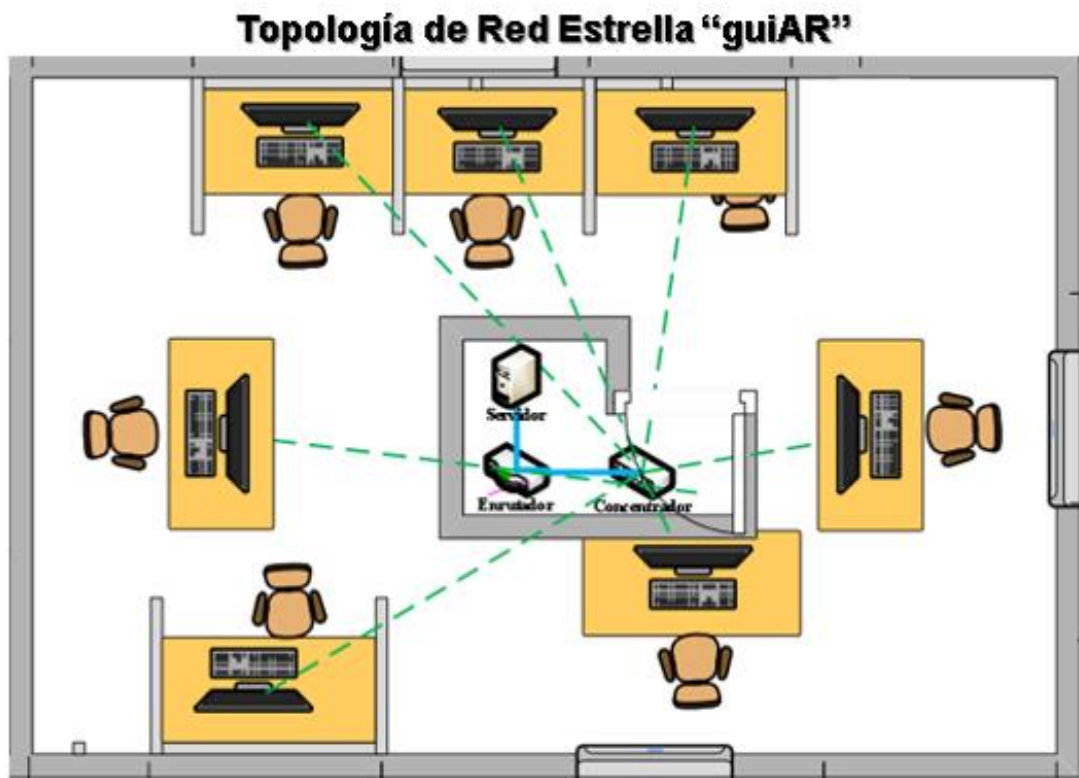


Diagrama de Estructura para diseñar sistemas modulares

Significa ver una descripción amplia del sistema y después dividirla en partes más pequeñas o subsistemas. El diseño descendente permite a los analistas de sistemas determinar primero los objetivos organizacionales globales, así como también determinar cómo se reúnen mejor en un sistema global. Después el analista divide dicho sistema en subsistemas y sus requerimientos.

Una vez que se toma el enfoque del diseño descendente, el enfoque modular es útil en la programación. Este enfoque implica dividir la programación en partes lógicas y manejables llamadas módulos. Este tipo de programación funciona bien con el diseño descendente porque da énfasis a las interfaces entre los módulos y no los descuida hasta el final del desarrollo de sistemas. Idealmente, cada módulo individual debe ser funcionalmente cohesivo de manera que se encargue de realizar una sola función.

El diseño de programa modular tiene tres ventajas principales.

1^{era}: los módulos son más fáciles de escribir y de depurar porque prácticamente son independientes. Rastrear un error en un módulo es menos complicado, debido a que un problema en un módulo no debe causar problemas en otros.

2^{nda}: los módulos son más fáciles de mantener. Normalmente las modificaciones se limitarán a unos módulos y no seguirán en todo el programa.

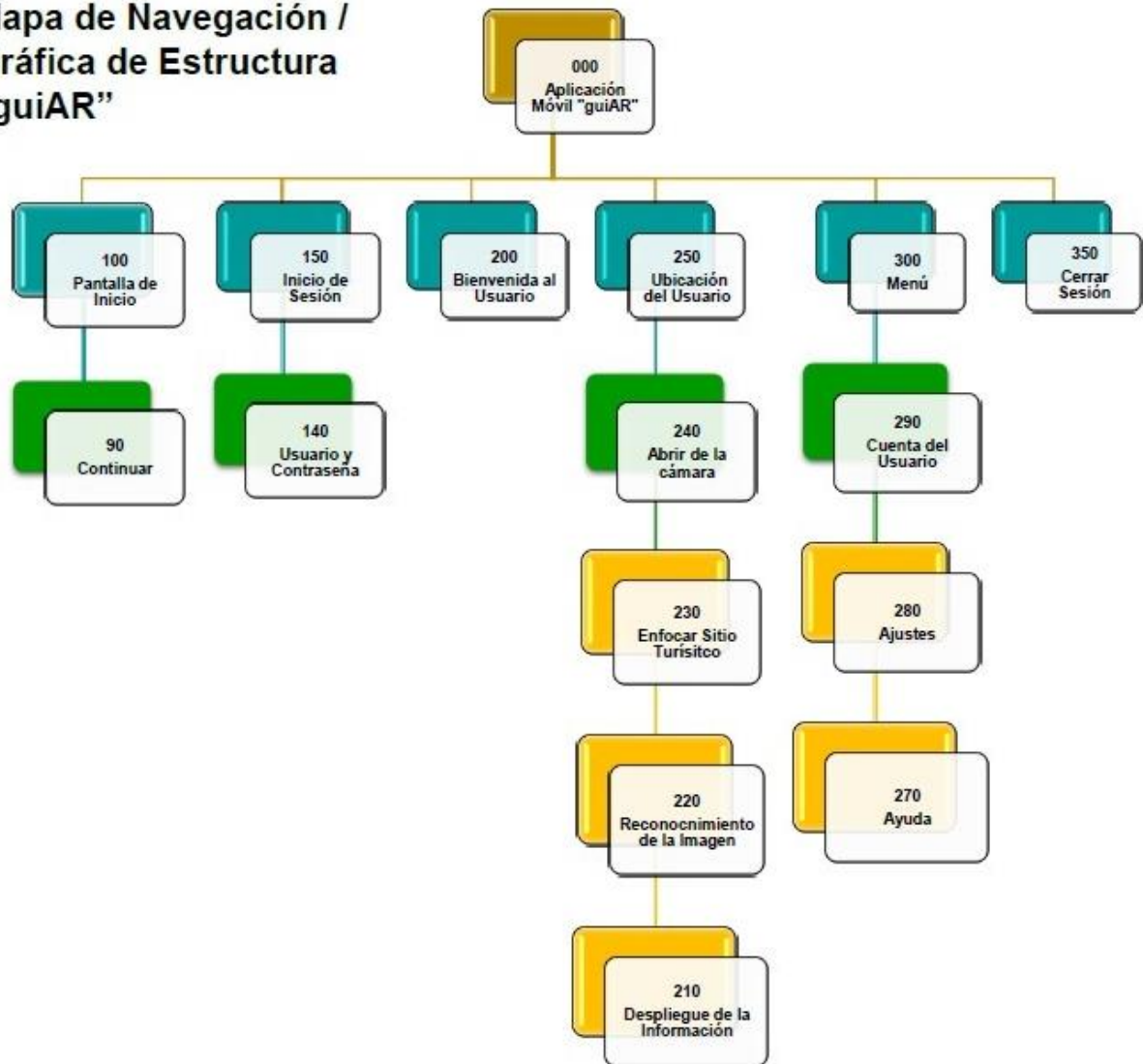
3^{er}: los módulos son más fáciles de entender, debido a que son subsistemas independientes. Por lo tanto, un lector puede adquirir una lista del código de un módulo y entender su función.

A continuación le mostramos algunos lineamientos para la programación modular:

- 1) Mantener cada módulo en un tamaño manejable (lo ideal es que sólo incluya una función).
- 2) Poner atención especial en las interfaces críticas (los datos y las variables de control que se pasan a otros módulos).
- 3) Minimizar el número de módulos que debe modificar el usuario al realizar cambios.
- 4) Mantener las relaciones jerárquicas establecidas en las fases de la metodología descendente.

Un diagrama de estructura fomenta el diseño descendente mediante el uso de módulos:

Mapa de Navegación / Gráfica de Estructura “guiAR”



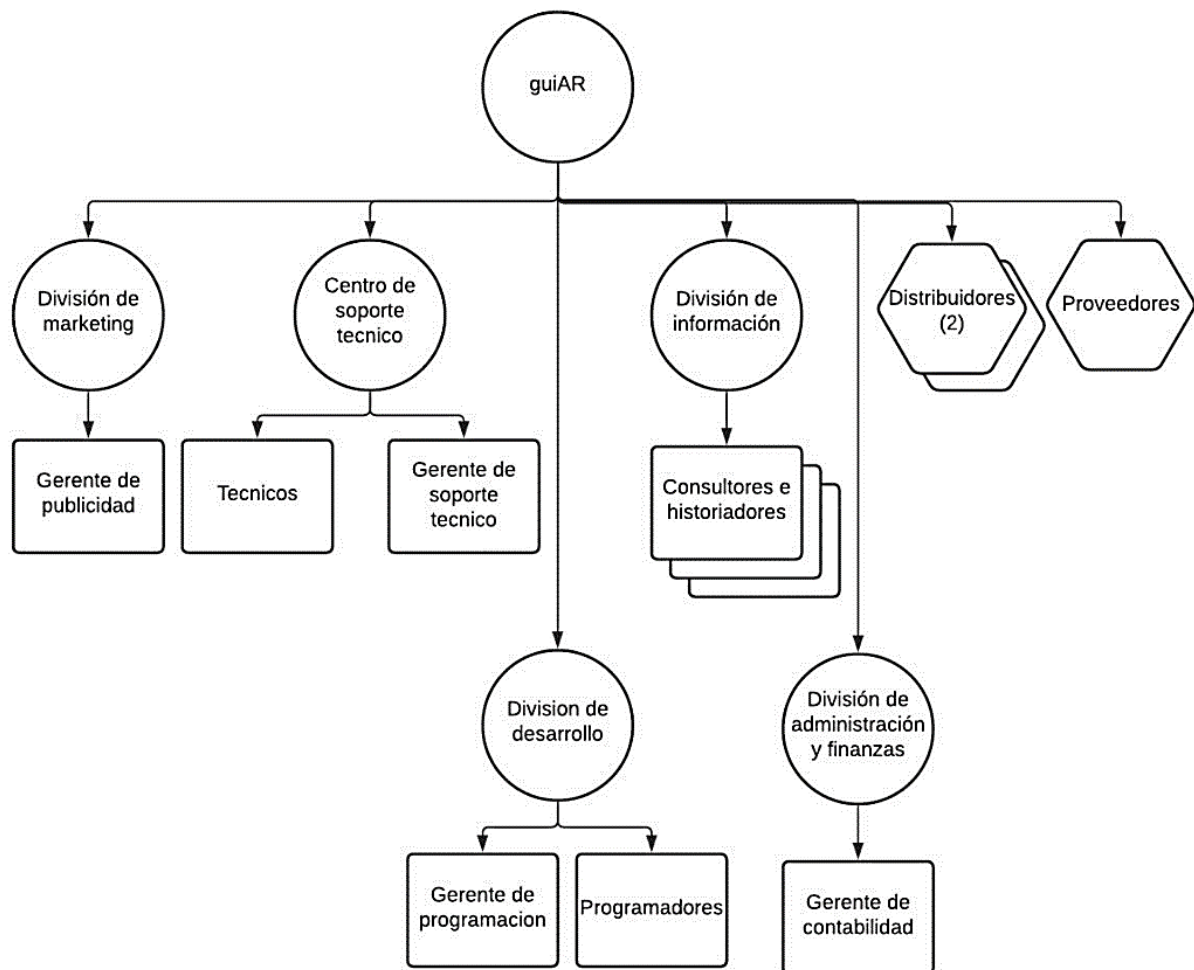
Modelado de red

Debido a que la conexión a una red se ha vuelto muy importante, el diseñador de sistemas necesita tomar en cuenta al diseño de la red. Ya sea que un diseñador de sistemas tenga que decidir sobre las configuraciones de las redes o si se preocupa por el hardware tal como enrutadores y puentes que deben estar en el lugar cuando se conocen las redes, siempre debe tomar en cuenta el diseño lógico de las redes.

Podemos ilustrar el dibujo de un modelo de descomposición de red, tal y como se visualizará más abajo. Se inicia trazando un círculo en la parte superior y nombrándolo como "guiAR". Después se traza varios círculos en el nivel inferior. Dichos círculos representan los hubs (concentradores) para la división de marketing y para cada uno de los tres centros de toma de pedidos y distribución (división americana, división canadiense y división mexicana).

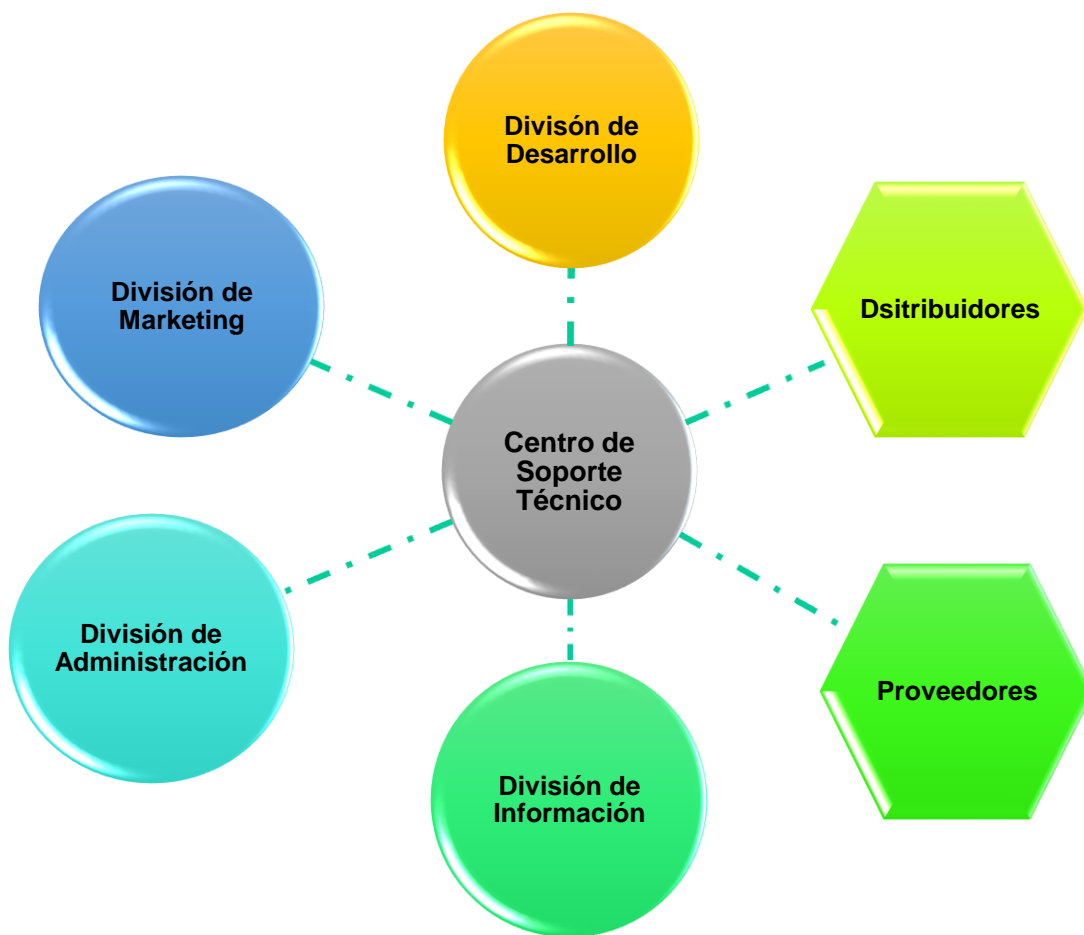
De la misma forma extendemos el trazado otro nivel. Esta vez, podemos agregar las estaciones de trabajo. Por ejemplo, la división de marketing tiene dos estaciones de trabajo conectadas, mientras que la división americana tiene 33 estaciones de trabajo en su LAN (administración, almacén, gerente de entrada de pedido y 30 empleados de entrada de pedido).

Esta red se simplifica con el propósito de proporcionar un ejemplo fácil de entender.



Creación de un diagrama de conectividad de hub

El diagrama de conectividad de hub es útil para mostrar cómo se conectan los hubs principales. En el Asistente Turístico “guiAR”, hay 6 hubs principales conectados entre sí. Además, hay 2 hubs externos (distribuidores y proveedores) que necesitan ser notificados cuando el nivel de inventario baja a un cierto punto, etc. Cada una de las tres divisiones de país se conecta a los 10 proveedores. Sin embargo, la división de marketing no necesita ser conectada a los proveedores. Para producir un diagrama de conectividad de hubs eficaz, empiece dibujando todos los hubs, tal y como se visualizará en breves:

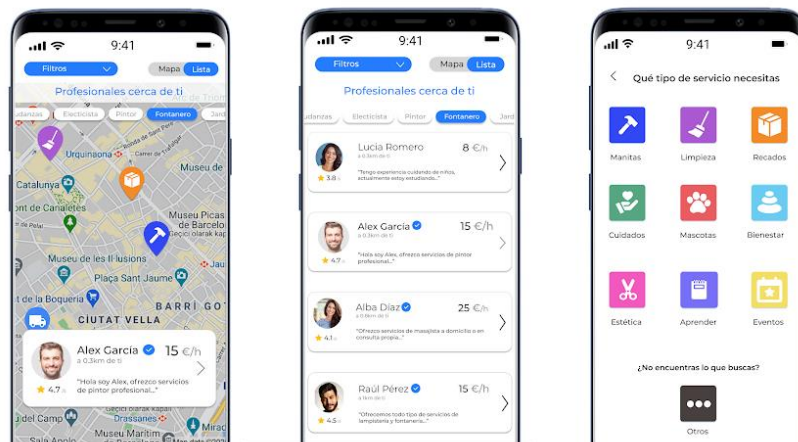


Capacitación de los Usuarios

La capacitación es un proceso educativo en el que participan los analistas de sistemas con los usuarios. El usuario se ha involucrado en todo el ciclo de vida de desarrollo de sistemas, por lo que ahora el analista debe tener una valoración exacta de los usuarios que se deben capacitar. En la implementación de proyectos grandes, el analista normalmente estará manejando la capacitación en lugar de estar involucrado personalmente en ella. Uno de los recursos más valiosos que el analista puede aportar en cualquier situación de capacitación es la habilidad de ver el sistema desde el punto de vista del usuario.

Para llevar a cabo la capacitación de los futuros usuarios, que utilizarán nuestra aplicación móvil del Asistente Turístico de Sitios Culturales, se emplearán diversas estrategias o técnicas de capacitación, dado que cada usuario es diferente. Algunos usuarios aprenden mejor viendo, otros oyendo e incluso otros haciendo. Debido a que normalmente no es posible personalizar la capacitación para un individuo, una combinación de métodos es a menudo la mejor forma de proceder.

Los métodos para aquellos que aprenden mejor viendo incluyen demostraciones de equipo y exposición para manuales de capacitación. Aquellos que aprenden mejor oyendo se beneficiarán de las conferencias sobre los procedimientos, discusiones y sesiones de preguntas y respuestas entre instructores y aprendices. Aquellos que aprenden mejor haciendo necesitan experiencia práctica con la aplicación en un smartphone. Otra forma más práctica y moderna que hoy en día es utilizada y recurrida por muchos usuarios para manejar el uso de una aplicación existente en las tiendas de aplicaciones es ver un vídeo tutorial acerca, en una plataforma de streaming, cómo es YouTube, ya que ahí se explica de forma detallada, explícita, en tiempo real, paso a paso de cómo utilizar de forma correcta una aplicación móvil y también de algunos tips o trucos, para aprovechar al máximo lo que ofrece de forma general y particular, dicha aplicación. Lo que nosotros vemos una estrategia o técnica muy viable, efectiva, eficiente, factible, confiable, etc.. para que nuestros futuros usuarios puedan aprovechar emplear, disfrutar y manejar, sin ningún problema.



Conversión del Sistema

Un tercer apartado para la implementación es convertir físicamente el sistema de información viejo a uno nuevo o modificado (principio de innovación). Existen diversas estrategias de conversión disponibles para los desarrolladores, analistas y también un enfoque en un plan de contingencia que tiene en cuenta diversas variables organizacionales para decidir qué estrategia de conversión llevar a cabo. No se puede subestimar la importancia de diseñar y programar adecuadamente la conversión (la cual con frecuencia tarda muchas semanas), el archivo de respaldo y la seguridad adecuada.

Para la implementación y desarrollo del presente Proyecto de Software, se destinó por la Estrategia de Conversión Gradual o Fases, dado a lo siguiente justificación:

Dicha estrategia de conversión, intenta combinar las mejores características de los dos planes previamente mencionados, sin incurrir en todos los riesgos. En este plan, el volumen de las transacciones manejado por el nuevo sistema aumenta gradualmente conforme el sistema se introduce por fases.

Las ventajas de este método incluyen permitir a usuarios que se involucren gradualmente con el sistema, la posibilidad de descubrir y recuperar errores sin desperdiciar mucho tiempo y la capacidad de agregar características una por una. Las metodologías ágiles tienden a usar esta metodología de conversión.

Otras consideraciones de conversión La conversión también trae consigo otros detalles para el analista, los cuales incluyen los puntos a continuación:

- ✓ Pedir equipo (hasta tres meses antes de la conversión planeada).
- ✓ Pedir cualesquier materiales necesarios que se proporcionan externamente al sistema de información, tal como cartuchos de tinta, papel, formularios impresos previamente y los medios magnéticos.
- ✓ Designar un gerente para supervisar, o supervisar personalmente, la preparación del sitio de la instalación.
- ✓ Planear, fijar y supervisar a programadores y personal de captura de datos que deben convertir todos los archivos y bases de datos relevantes.

Para muchas implementaciones, su papel principal será estimar con precisión el tiempo necesario para cada actividad, nombrar a las personas para manejar cada subproyecto y coordinar su trabajo.

Evaluación del Sistema

A lo largo del ciclo de vida del desarrollo de sistemas, el desarrollador, los directivos y los usuarios han estado evaluando la evolución de los sistemas de información y las redes para proporcionar retroalimentación para su mejora eventual. La evaluación también se necesita para dar seguimiento a la implementación del sistema de información, donde se puede evaluar como exitoso si posee todas estas utilidades. Si el módulo del sistema se juzga como "pobre" al proporcionar alguna de las utilidades, el módulo completo estará destinado a fallar.

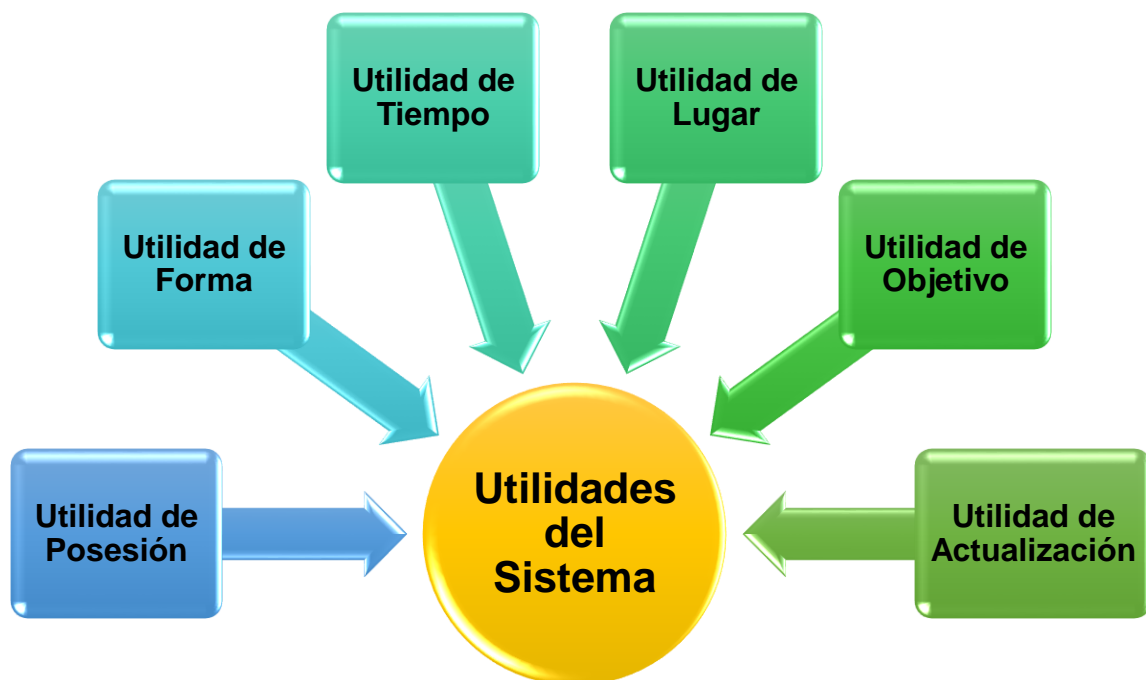
Un logro parcial o "justo" de una utilidad producirá un módulo parcialmente exitoso. Si el módulo del sistema de información se juzga como "bueno" proporcionando cada utilidad, el módulo es un éxito.

Técnicas de Evaluación del Sistema

En reconocimiento de que la evaluación continua de sistemas de información y redes es importante, se han inventado muchas técnicas de evaluación, que para el presente Proyecto de Software se tomará seguirá la que a continuación se menciona:

El enfoque de utilidad del sistema de información

Y esta evalúa el sistema a implementar y desarrollar, al igual que mide el éxito de un sistema desarrollado. Además tiene la función de una guía en el desarrollo de cualesquier proyectos futuros que el analista podría emprender.



El enfoque de utilidad del sistema de información de "quién, qué, cuándo, cómo, dónde y por qué" usado para evaluar el sistema de información de administración de inventario de sangre regional resulta de los juicios subjetivos acerca de la utilidad del sistema de información resumidos en la tabla que se muestra a continuación:

Módulos del Sistema de la Aplicación "guiAR"	Utilidad de Posesión	Utilidad de Forma	Utilidad de Tiempo	Utilidad de Lugar	Utilidad de Objetivo	Utilidad de Actualización
Reconocimiento de las Imágenes del Sitio Turístico						
Despliegue de la Información de cada Sitio Turístico						
Empleo de los audios en idiomas para la descripción de cada Sitio Turístico.						
Registro de Usuarios en la Aplicación						
Inicio de Sesión de los Usuarios, dentro de la Aplicación						
Facilidad de manejo de la Aplicación						
Actualizar otro Sitio Turístico.						

Después de la implementación, el nuevo sistema de información y el enfoque dado u orientado, se deben evaluar. El enfoque de utilidad del sistema de información es un marco de trabajo utilizable y sencillo para evaluar proyectos extensos de sistemas de información y los esfuerzos en proceso. También se puede emplear como una lista de control para supervisar el progreso de sistemas en desarrollo. Además, la evaluación que sigue la implementación permite al desarrollador adquirir ideas sobre cómo proceder con los proyectos de sistemas futuro.

Modelo de Negocio

Un modelo de negocios recoge las bases de creación de valor de un negocio o proyecto. Es una simplificación, un plano, un croquis, que recoge las bases de un negocio o proyecto (concepto al que también suele denominarse “modelo de negocio”, pero en un sentido bien diferente).

Un modelo de negocios no es un plan de negocio o un plan de empresa, donde se recoge la explicación ordenada y sistemática de un proyecto o empresa.

Business Model Canvas o Modelo de Negocio CANVAS

Es un modelo muy visual con el que podremos ordenar nuestras ideas a la hora de definir cuál será nuestro modelo de negocio. Desarrollado por Alexander Osterwalder, se trata de un modelo ideal para determinar y crear modelos innovadores con el objetivo de generar valor para los clientes, definiendo y creando modelos de negocio innovadores a través de cuatro grandes áreas (los clientes, la oferta, la infraestructura y la viabilidad económica) que se desarrollan en nueve divisiones, apartados o casillas.

Algunas ventajas del Modelo Canvas son:

- ✓ Es una herramienta visual y gráfica que permite observar los aspectos más importantes de tu modelo de negocio. Una vez que el lienzo es culminado, se debe dejar a la vista de los socios y el equipo de trabajo para absorber una visión global del negocio.
- ✓ Ofrece un formato cómodo y práctico que puede ser modificado a medida que avanza el negocio y aparecen nuevas posibilidades.
- ✓ Tiene un concepto abierto para probar diferentes hipótesis hasta encontrar el camino más idóneo para la consecución de los objetivos.
- ✓ El Modelo Canvas tiene un diagrama intuitivo, divertido y sencillo. Para la elaboración del lienzo se recomienda usar post-its, rotuladores de colores e imprimir el lienzo en dimensiones grandes para su fácil visualización.
- ✓ Es un método creado para el trabajo en equipo. Coloca el lienzo Canvas en un lugar donde todos los participantes puedan observarlo y fomenta un intercambio de ideas entre todos.

Elementos que conforman un Modelo de Negocios CANVA:

Propuesta de Valor: Producto o servicio que se posee que no tengan los demás y por el que la sociedad (usuarios) estará dispuesta a pagar.

Segmento de Clientes: Se define a los usuarios meta, en ¿En qué problemas tiene? y ¿Qué necesidades tiene?

Canales: Las formas en qué la propuesta de valor va a llegar al segmento de clientes.

Relación con los Clientes: Se define el tipo de relación que se tendrá con él cliente. Dicho cliente es lo más importante.

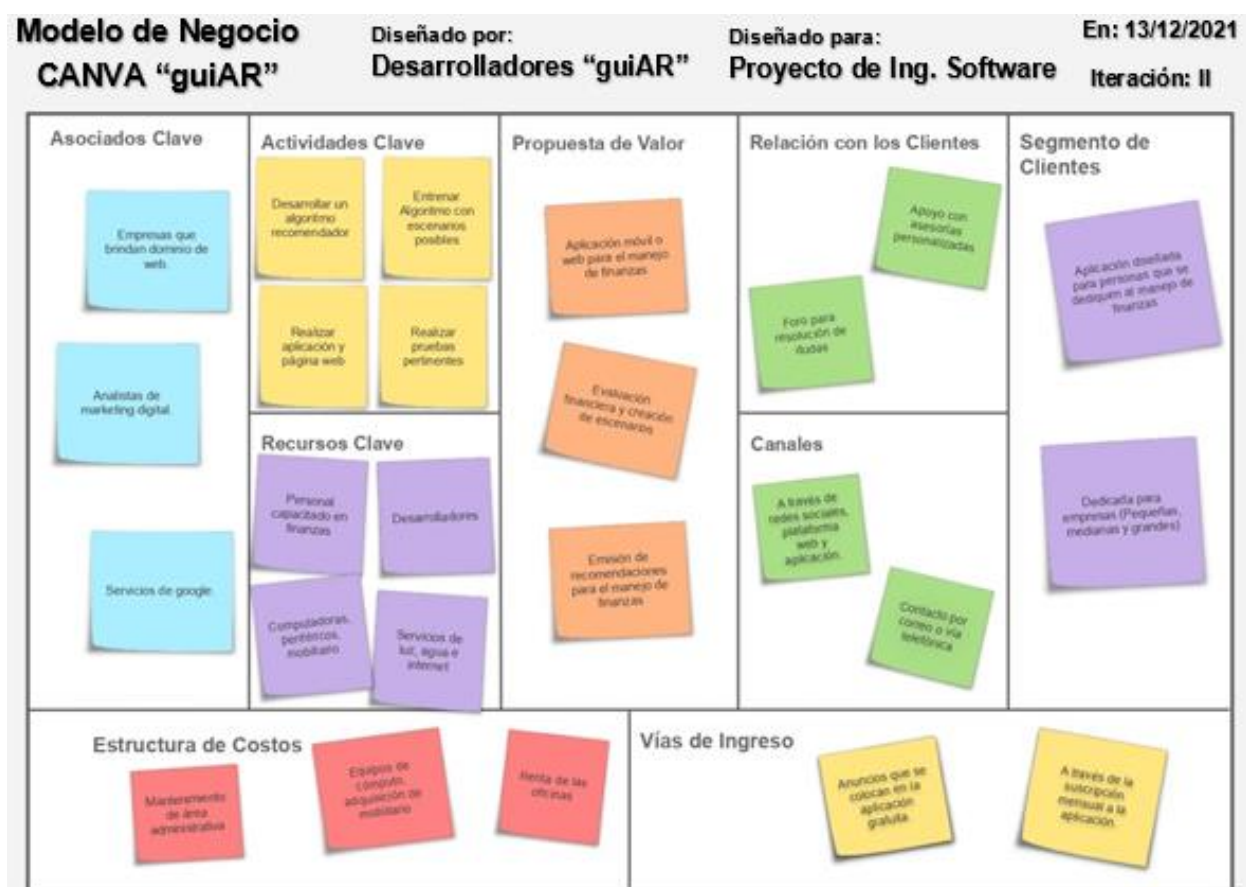
Fuentes/Vías de Ingreso: ¿Cuál es la rentabilidad?, se identifica el cómo y dónde van a llegar los ingresos del Proyecto.

Recursos Clave: Los elementos que se requieren para que funcione el Modelo de Negocio planteado.

Actividades Clave: Son las actividades que corresponde al desarrollo del Modelo de Negocio para que se tenga éxito.

Asociados Clave: Son el conjunto de personas clave con la que se tiene que interactuar para que la propuesta de valor ocurra.

Estructura de Costos: Es la inversión inicial, determinación de costos fijos y variables.



Capítulo 8: INTERFAZ, IMPLEMENTACIÓN Y VALIDACIONES

XV. DESARROLLO, EJECUCIÓN Y PRUEBAS

Algoritmo empleado para la Aplicación Móvil “guiAR”

```
#load system libraries
from SimpleCV.base import *
from SimpleCV.ImageClass import Image, ImageSet, ColorSpace
from SimpleCV.Display import Display
from SimpleCV.Color import Color
from collections import deque
import time
import ctypes as ct
import subprocess
import cv2
import numpy as np
import traceback
import sys

#Globals
_cameras = []
_camera_polling_thread = ""
_index = []

class FrameBufferThread(threading.Thread):
    """
    **SUMMARY**
    This is a helper thread which continually debuffers the camera frames.
    If
    you don't do this, cameras may constantly give you a frame behind, which
    causes problems at low sample rates. This makes sure the frames returned
    by your camera are fresh.
    """
    def run(self):
        global _cameras
        while (1):
            for cam in _cameras:
                if cam.pygame_camera:
                    cam.pygame_buffer = cam.capture.get_image(cam.pygame_buffer)
                else:
```

```

        cv.GrabFrame(cam.capture)
        cam._threadcapturetime = time.time()
        time.sleep(0.04)    #max 25 fps, if you're lucky
def getImage(self):
    return None

def calibrate(self, imageList, grid_sz=0.03, dimensions=(8, 5)):
    """
    **SUMMARY**
    Camera calibration will help remove distortion and fisheye effects
    It is agnostic of the imagery source, and can be used with any camera

    The easiest way to run calibration is to run the
    calibrate.py file under the tools directory for SimpleCV.
    This will walk you through the calibration process.
    **PARAMETERS**
    * *imageList* - is a list of images of color calibration images.
    * *grid_sz* - is the actual grid size of the calibration grid, the unit
    used will be
        the calibration unit value (i.e. if in doubt use meters, or U.S. standard)
    * *dimensions* - is the the count of the *interior* corners in the calibration grid.
        So for a grid where there are 4x4 black grid squares has seven interior corners.
    **RETURNS**
    The camera's intrinsic matrix.
    **EXAMPLE**
    See :py:module:calibrate.py
    """
    # This routine was adapted from code originally written by:
    # Abid. K -- abidrahman2@gmail.com
    # See: https://github.com/abidrahmank/OpenCV-Python/blob/master/Other_Examples/camera_calibration.py

    warn_thresh = 1
    n_boards = 0    #no of boards
    board_w = int(dimensions[0])    # number of horizontal corners
    board_h = int(dimensions[1])    # number of vertical corners
    n_boards = int(len(imageList))
    board_n = board_w * board_h    # no of total corners
    board_sz = (board_w, board_h)    #size of board
    if( n_boards < warn_thresh ):
        logger.warning("FrameSource.calibrate: We suggest using 20 or more images to perform camera calibration!" )

```



```

# creation of memory storages
image_points = cv.CreateMat(n_boards * board_n, 2, cv.CV_32FC1)
object_points = cv.CreateMat(n_boards * board_n, 3, cv.CV_32FC1)
point_counts = cv.CreateMat(n_boards, 1, cv.CV_32SC1)
intrinsic_matrix = cv.CreateMat(3, 3, cv.CV_32FC1)
distortion_coefficient = cv.CreateMat(5, 1, cv.CV_32FC1)

# capture frames of specified properties and modification of m
atrix values
i = 0
z = 0 # to print number of frames
successes = 0
imgIdx = 0
# capturing required number of views
while(successes < n_boards):
    found = 0
    img = imageList[imgIdx]
    (found, corners) = cv.FindChessboardCorners(img.getGrayscaleMatr
ix(), board_sz,
cv.CV_CALIB_CB_ADAPTIVE
_THRESH |
cv.CV_CALIB_CB_FILTER_Q
UADS)
    corners = cv.FindCornerSubPix(img.getGrayscaleMatrix(), corners,
(11, 11),(-1, -1),
(cv.CV_TERMCRIT_EPS + cv.CV_TERMCRIT
_ITER, 30, 0.1))
    # if got a good image, draw chess board
    if found == 1:
        corner_count = len(corners)
        z = z + 1

    # if got a good image, add to matrix
    if len(corners) == board_n:
        step = successes * board_n
        k = step
        for j in range(board_n):
            cv.Set2D(image_points, k, 0, corners[j][0])
            cv.Set2D(image_points, k, 1, corners[j][1])
            cv.Set2D(object_points, k, 0, grid_sz*(float(j)/float(bo
ard_w)))
            cv.Set2D(object_points, k, 1, grid_sz*(float(j)%float(bo
ard_w)))
            cv.Set2D(object_points, k, 2, 0.0)

```

```

        k = k + 1
        cv.Set2D(point_counts, successes, 0, board_n)
        successes = successes + 1

    # now assigning new matrices according to view_count
    if( successes < warn_thresh ):
        logger.warning("FrameSource.calibrate: You have %s good images f
or calibration we recommend at least %s" % (successes, warn_thresh))

    object_points2 = cv.CreateMat(successes * board_n, 3, cv.CV_32FC1)
    image_points2 = cv.CreateMat(successes * board_n, 2, cv.CV_32FC1)
    point_counts2 = cv.CreateMat(successes, 1, cv.CV_32SC1)

    for i in range(successes * board_n):
        cv.Set2D(image_points2, i, 0, cv.Get2D(image_points, i, 0))
        cv.Set2D(image_points2, i, 1, cv.Get2D(image_points, i, 1))
        cv.Set2D(object_points2, i, 0, cv.Get2D(object_points, i, 0))
        cv.Set2D(object_points2, i, 1, cv.Get2D(object_points, i, 1))
        cv.Set2D(object_points2, i, 2, cv.Get2D(object_points, i, 2))
    for i in range(successes):
        cv.Set2D(point_counts2, i, 0, cv.Get2D(point_counts, i, 0))

    cv.Set2D(intrinsic_matrix, 0, 0, 1.0)
    cv.Set2D(intrinsic_matrix, 1, 1, 1.0)
    rcv = cv.CreateMat(n_boards, 3, cv.CV_64FC1)
    tcv = cv.CreateMat(n_boards, 3, cv.CV_64FC1)
    # camera calibration
    cv.CalibrateCamera2(object_points2, image_points2, point_counts2,
                        (img.width, img.height), intrinsic_matrix, distor
tion_coefficient,
                        rcv, tcv, 0)
    self._calibMat = intrinsic_matrix
    self._distCoeff = distortion_coefficient
    return intrinsic_matrix

```

Para esta parte realizaremos una verificación de escritorio de la aplicación para comprobar la forma en que funcionara el sistema.

Al iniciar la aplicación

- ✓ Mostrar en la pantalla, las opciones para registrarse, iniciar sesión o recuperar contraseña
- ✓ Para el registro de usuarios
- ✓ Solicitar nombre de usuario (Sera único de cada usuario)
- ✓ Solicitar nombre y apellido paterno del usuario
- ✓ Solicitar correo electrónico
- ✓ Solicitar número telefónico (opcional)
- ✓ Solicitar contraseña (Mínimo de 8 caracteres con letras y números)
- ✓ Solicitar confirmación de contraseña
- ✓ Mostrar botón para que el usuario pueda leer los términos y condiciones, así como el aviso de privacidad.
- ✓ Mostrar un checkbox para confirmar los términos y condiciones y el aviso de privacidad.
- ✓ Mostrar botón para finalizar el registro.
- ✓ Mostrar opción para registro con cuenta de Google o de Facebook.

Una vez registrado el usuario

- Ya hecho el registro, se le enviara un correo o un SMS al usuario para confirmar la creación de su cuenta.
- Iniciar sesión.
- Solicitar nombre de usuario, correo electrónico o número de teléfono.
- Solicitar contraseña.
- Mostrar opción para recuperar la contraseña.
- Mostrar opción para inicio de sesión con Facebook o con Google.

Recuperar contraseña

- ↗ Solicitar correo electrónico o número de teléfono de la cuenta de la que se desea reestablecer la contraseña.
- ↗ Verificar que el correo ingresado o el número de teléfono este asociado a una cuenta existente.
- ↗ En caso de no existir, mostrar mensaje de error.
- ↗ Si existe, enviar un mensaje al correo ingresado, con un link (habilitado solo por 10 minutos) para poder realizar el cambio de contraseña.
- ↗ En la ventana para cambiar la contraseña, solicitar una solicitar contraseña (Mínimo de 8 caracteres con letras y números) y pedir la confirmación de la misma.

Menú principal

Mostrar pestaña para, escanear sitio, lista de lugares en la CDMX, configuración de la cuenta.

Escanear sitio

- ✓ Al momento de seleccionar esta opción, si no se tiene permiso para acceder a la cámara del dispositivo, solicitar el permiso al usuario
- ✓ La aplicación hará uso de la cámara del dispositivo para escanear los sitios de interés, si no se encuentra ningún sitio, se enviará mensaje del error al usuario, en caso contrario, iniciará la búsqueda del sitio en la base de datos de la aplicación
- ✓ Si la búsqueda del sitio fue exitosa, se mostrará mediante realidad aumentada los datos como el nombre, información, horarios y costos (si aplica) se visualizarán en la pantalla del dispositivo, con la opción de salida por audio en diferentes idiomas como el inglés y el chino
- ✓ Si la búsqueda no tuvo éxito, se mostrará un mensaje de error al usuario.

Lista de lugares en la CDMX

- ✓ **Zócalo**
- ✓ **Palacio de Bellas Artes**
- ✓ **La Torre Latinoamericana**
- ✓ **Hemiciclo a Juárez**
- ✓ **Monumento a la Revolución**

Configuración de la aplicación

- ⇒ Se mostrará un menú con las opciones para cambiar el idioma de la app, cambiar información de la cuenta y una sección de preguntas frecuentes.
- ⇒ La sección de idioma de la app cambiara el idioma de la misma (texto y audio) de acuerdo con la preferencia del usuario.
- ⇒ La sección de información de la cuenta permitirá al usuario cambiar su correo electrónico, número de teléfono, nombre de usuario, datos personales y contraseña, para realizar cualquier modificación, se deberá ingresar la contraseña actual de la cuenta, además de enviarse un correo indicando los cambios que se realizaron.
- ⇒ En la sección de preguntas frecuentes, se mostrarán las principales dudas que han tenido los usuarios y posibles soluciones a la mayoría de los problemas.

Monetización del proyecto de software “guiAR”

Monetizar una app es un gran desafío para las empresas, ya que hace falta conquistar la atención de los usuarios y retenerlos hasta que se conviertan en clientes.

La constante evolución tecnológica y la transformación digital le ha permitido a las personas y las empresas generar ingresos a través de diferentes softwares y plataformas debido a que las aplicaciones se han consolidado como un vehículo que puede conducir hacia los beneficios económicos, la rentabilidad y el cumplimiento de objetivos financieros.

Básicamente, dividimos las apps en dos tipos: las de descarga gratuita y las de pago.

Nosotros utilizaremos dos modelos para monetizar nuestra app:

1. App gratuita con publicidad in-App

La publicidad in-app es uno de los mecanismos que más terreno está ganando como modelo de monetización. Según Statista la publicidad in-app en 2020 ascendió a 117,2 billones de dólares.

En la actualidad, muchas empresas comprenden que divulgar su marca y productos o servicios resulta una muy buena idea. Y esto es grandioso para una aplicación.

Ofrecerles que publiquen en nuestra aplicación nos permitirá generar ingresos de manera directa. Mientras tanto, las compañías pueden alcanzar un mayor volumen de visualizaciones de sus productos o servicios al retener a los usuarios por más tiempo en la plataforma.

Entonces, el uso de la app será gratuito, pero a cambio de ese uso gratuito incluimos pequeños banners de publicidad embebida en la app. O bien saltarán anuncios dentro de la aplicación cada determinado tiempo de uso.

El retorno de la inversión se recupera introduciendo publicidad. De esta manera, podemos obtener un beneficio en función del número de clics (CPC) o por el número de exposiciones (CPI o CPM), cada vez que el usuario haga sobre esa publicidad.

AdMob

Para poner esta publicidad en la app, haremos uso de AdMob, la cual es una empresa de publicidad móvil que es administrada por Google y ayuda a generar ingresos mediante la publicación de anuncios publicitarios y de vídeo.

Los anunciantes pagan a Google por usar AdMob para publicitar sus anuncios dentro de las aplicaciones móviles. De esta forma, podemos generar ingresos de AdMob mediante nuestra app.

En cuanto a los tipos de anuncios en una App existen varios que podemos implementar.

- **Anuncios de texto/banners:** Se trata de un anuncio que ocupa una pequeña parte de la pantalla de nuestro dispositivo. Muy versátil y práctico, pero realmente poco rentable para ganar dinero. Es el que menos CPM (Coste por cada mil visualizaciones) ofrece.
- **Anuncios intersticiales:** Son anuncios que se muestran en pantalla completa y pueden ser tanto imágenes estáticas como videos. Son los que más CPM ofrecen.

El CPM o RPM real que se puede conseguir mezclando estos anuncios con una App usando Admob ronda los 7€, aproximadamente, \$170 MXN.

Entonces tomando que el RPM ronda los 7€ de media, obtenemos que por cada 1,000 anuncios visualizados en nuestra App nos llevaremos 7€.

Ganancias y usuarios diarios en un principio

Supongamos que nuestro objetivo es ingresar, en un principio 10€/día (243 MXN), sabiendo que por cada 1000 visualizaciones vamos a conseguir 7€, $10€/7€$ nos da 1.428. Es decir, nuestra App tiene que mostrar $1000 \times 1.428 = 1428$ anuncios al día para alcanzar los 10€ diarios.

Si necesitamos 1428 impresiones/visualizaciones de anuncios cada día, ¿cuántos usuarios activos vamos a necesitar? Si imaginamos que cada usuario es capaz de ver 5 anuncios en una sesión en nuestra App, ya tenemos la respuesta, $1428/5 = 285$ usuarios.

Por lo tanto, si tenemos 285 usuarios diarios, podremos conseguir al mes 300€ (\$7300 MXN).

Ganancias y usuarios diarios después de cierto tiempo

Ahora, supongamos que nuestro objetivo es ingresar, 50€/día (\$1216 MXN), sabiendo que por cada 1,000 visualizaciones vamos a conseguir 7€, $50€/7€$ nos da 7.142. Es decir, nuestra App tiene que mostrar $1,000 \times 7.142 = 7142$ anuncios al día para alcanzar los 50€ diarios.

Si necesitamos 7142 impresiones/visualizaciones de anuncios cada día y si imaginamos que cada usuario es capaz de ver 5 anuncios en una sesión en nuestra App, necesitaremos $7142/5 = 1428$ usuarios.

Por lo tanto, si tenemos 1428 usuarios diarios, podremos conseguir al mes 1500€ (\$36,501 MXN).

En resumen, necesitamos cerca o más de 1500 usuarios diarios para obtener cada integrante \$10,000 MXN al mes.

2. Paid o Pago por descarga

Quizás, esta es la forma más directa de monetizar una app.

Es decir, que la condición de descarga sea comprar nuestra aplicación. De esta forma estaremos generando ingresos desde el inicio y el usuario tendrá acceso inmediato a todas las funcionalidades y potencialidades.

Los ingresos se generan por una única y primera vez, ya que la persona paga por descargarse la app y usarla tantas veces como desee; por lo tanto, será necesario un esfuerzo considerable de captación para conseguir el mayor número de usuarios posible y asegurar un volumen razonable de ingresos.

Para poder obtener el precio de nuestra app debemos tener en cuenta ciertas consideraciones:

Primero debemos estimar el coste de desarrollo (mano de obra, tecnología, etc.), en este caso, tomemos en cuenta la siguiente tabla.

Desarrolladores	4
Días laborados	90
Horas por día en promedio	4
Horas laboradas en total	1,440
Tarifa promedio (MXN) por hora	100
Total (tarifa x horas)	\$144,000

Por lo tanto, la tarifa de mano de obra que arroja el análisis es de \$108,000, por lo que es el coste de desarrollo ya que las tecnologías que utilizamos son “open source” y no cuestan como tal dinero.

También hay que tener en cuenta el coste que vamos a necesitar para dar a conocer nuestro producto. Para esto tenemos que recurrir a marketing online:

- **ASO On Metadata:** Es el posicionamiento orgánico de la app en los buscadores de apps.
- **ASO Off Metadata:** Consiste en las valoraciones y reviews que hacen los usuarios de nuestra app en Google Play y App Store. Cuantas más y mejores tengamos, mejor será el posicionamiento en los buscadores de apps.
- **SEO para Apps:** También es parte del ASO Off-Metadata y nosotros entendemos muy necesario el SEO para promocionar una app. En este sentido, y por el volumen de búsquedas se hacen a través de la web, se necesitaría una página web landing y un blog optimizado para SEO, con WordPress, por ejemplo, para dar a conocer la app y posicionarla en los buscadores.

Promocionar una app en redes sociales gratuitamente o con Community Manager:
Al igual que lo anterior, esto los puede hacer uno mismo o pagar un Community Manager, cuyo coste dependerá de las horas que le dedique.

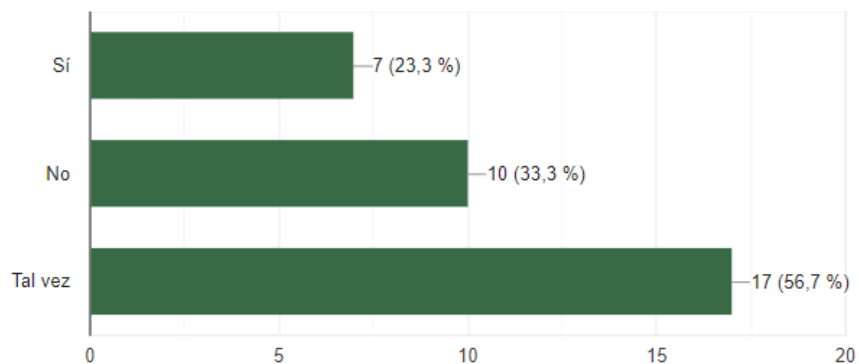
Las anteriores acciones de marketing online, si las extrapolamos a una cantidad de usuarios que queremos conseguir al mes implican que conseguir 4000 instalaciones de una app en un mes costaría \$36,000 MXN.

Entonces, en total, serían \$180,000 MXN

Ahora, viendo las respuestas de nuestro cuestionario como técnica de recolección de requisitos en la pregunta 19 y 20, realizado a 30 personas, contestan lo siguiente:

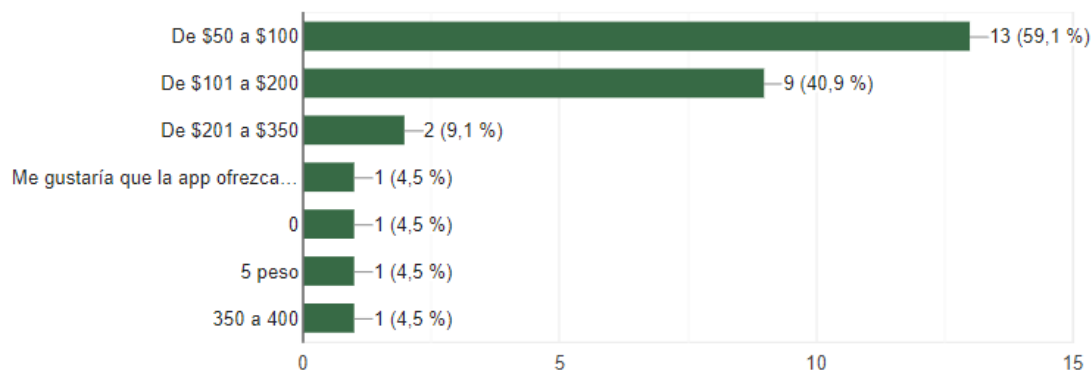
19. ¿Estarías dispuesto a pagar por una aplicación que te ofrezca información importante/relevante sobre algún lugar turístico/cultural?

30 respuestas



20. En caso de que tú pregunta sea afirmativa. ¿En qué rango de precios estás dispuesto a pagar?

22 respuestas



Las personas pagarían por la app, pero un precio entre \$50 y \$100. Ahora, teniendo en cuenta, que en total para hacer la app y el marketing son \$180000 MXN y obtendríamos en promedio 4000 instalaciones, esto nos daría un precio de \$45 MXN, sin embargo, el 30% se lo quedan las tiendas de aplicaciones como PlayStore

(Android) o AppStore (iOS), entonces el precio que realmente tendríamos que publicar sería de **\$65 MXN**.

Por lo tanto, vendiendo la app a este precio, recuperaríamos nuestra inversión en 1 mes, siempre y cuando, logremos las 4000 instalaciones, que, gracias al marketing online, se podría lograr.



Finalmente, observamos en la imagen anterior, que la app que mencionamos en el estado del arte llamada "World Around Me" cuesta en su versión pro, \$235 MXN, por lo que nuestra app será mucho más barata y más accesible a los clientes, teniendo funcionalidades que benefician bastante al usuario en su búsqueda de un lugar/sitio turístico.

Capítulo 9: ASPECTO DE MERCADOTECNIA

Logo de la Aplicación



Slogan

“La CDMX, en la palma de tus manos”



Triptico

La implementación de este software es para móviles, estando en alguna parte de la Ciudad de México, y haya alguna atracción o sitio turístico importante.



Mediante el uso de la cámara trasera del móvil se enfoque a dicha sitio y se despliega información relevante e importante, datos históricos y de la misma manera una opción con audios en los principales idiomas para conocer de forma rápida la información acerca de este sitio.

Diseñadores de la App Móvil

- Contreras Méndez Brandon
- Díaz Matus Ricardo
- Martínez Olivares Vicente Jafet
- Reyes Rojas Roberto

Encuéntrenos

Atención a Usuarios:

guiAR_app@outlook.com

55-64-98-16-05

Av. Juan de Dios Bátiz s/n esq. Av. Miguel Othón de Mendizábal. Colonia Lindavista. Alcaldía: Gustavo A. Madero. C. P. 07738. Ciudad de México.

INSTITUTO POLITÉCNICO NACIONAL

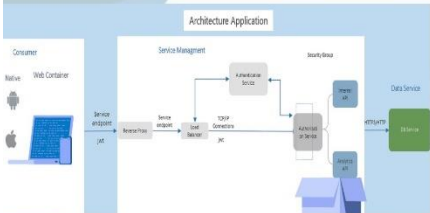
Escuela Superior de Cómputo

“guiAR”



“La CDMX, en la palma de tu mano”

Arquitectura de la App Móvil:



Objetivo:

Desarrollar una herramienta de software mediante el reconocimiento de imágenes de los lugares turísticos más importantes del Centro Histórico de la Ciudad de México.

Especificaciones del Software:

- Sistema Operativo: Android
- Tamaño descarga: 90 MB
- Clasificación: Todos
- Actualización: 16 Diciembre 2021
- Espacio de Almacenamiento del Dispositivo: 8GB +
- Memoria RAM en el Dispositivo: 4GB +
- Disponibilidad de Cámara Trasera

Descripción de la App Móvil:

guiAR, es una aplicación móvil para reconocimiento de imágenes de diversos sitios culturales, históricos e importantes de la Ciudad de México, el cual despliega información resumida relevante e interesante, cómo también en audios en los principales idiomas.

Funciones Principales:

- ✓ Ayudar y guiar a todas aquellas personas extranjeras o de provincia a conocer los lugares más importantes de la Ciudad de México.
- ✓ Desplegar información mediante audio.
- ✓ Facilitar la información necesaria de cada sitio turístico implementado.
- ✓ Enfocar y Analizar la imagen del Sitio/Lugar Turístico.



Aplicación Móvil “guiAR”



*“La CDMX, en la palma
de tu mano”*



Características de la App Móvil:

Enfoca

Se analiza mediante el reconocimiento de imágenes del Sitio/Lugar Turístico.

Brinda Información

Muestra en forma resumida, datos, reseñas, antecedentes históricos importantes del Sitio/Lugar Turístico.



Diseñadores de la App Móvil:

*Contreras Méndez Brandon
Díaz Matus Ricardo
Martínez Olivares Vicente Jafet
Reyes Rojas Roberto*

Multilenguaje

Desplegar audios en los principales idiomas para la descripción de los lugares.

Fácil de Usar

Al abrir la App, se muestra un entorno amigable y entendible para el usuario.

guiAR_app@outlook.com

Teléfono: 55-64-98-16-05

Coordir.ª Martha Rosa Cordero López

Licencia

Acuerdo de Licencia de Software “guiAR”

Artículo 1. Licencia

Los presentes Diseñadores, Desarrolladores y Programadores, le conceden al Usuario una licencia intransferible y NO exclusiva para la Aplicación Móvil, etc. en la forma de la licencia especificada en el siguiente punto (en adelante, denominada "Forma de licencia").

Licencia Única

El Usuario puede utilizar de forma simultánea la Aplicación Móvil, etc. sólo en una cantidad limitada de 2 dispositivos móviles permitidos por la Aplicación, durante el período establecido por la misma en conformidad con el Acuerdo adjunto.

Artículo 2. Restricciones de uso

El Usuario NO deberá/podrá realizar, ni solicitar a un tercero o particular que realice las siguientes acciones:

- I. Llevar a cabo ingeniería inversa, descompilar o desarmar el Software;
- II. Reproducir, alterar, producir, efectuar, traducir o adaptar todo o parte de la Aplicación Móvil, etc. o utilizarlos de cualquier otra forma a menos que se lo permita explícitamente en este Acuerdo;
- III. Distribuir, sublicenciar, arrendar, asignar, etc. la Aplicación Móvil, a algún tercero o entregarlos de cualquier otra forma con o sin compensación a menos que se lo permita explícitamente en este Acuerdo; y
- IV. Eliminar el aviso de derechos reservados de autor y aviso de cualquier otro derecho incluido en la Aplicación Móvil, etc.

Artículo 3. Titularidad de los derechos de propiedad intelectual

Los derechos de autor y cualquier otro derecho de propiedad intelectual relacionados con el Software, etc. son propiedad por completo de los Diseñadores, Desarrolladores y Programadores, o del tercero que se le concederá la licencia para la Aplicación Móvil, etc. NO deberán transferirse al Usuario.

Artículo 4. Notificación de información de actualización

1. En algunas ocasiones, la Aplicación Móvil contiene la función (en adelante, denominada "Notificaciones") para notificar información sobre las actualizaciones disponibles (en adelante, denominada "Información de actualización"), como parches, capacidad ampliada y complemento cuando se ejecuta en un dispositivo conectado a Internet.

2. En los casos donde el Usuario utiliza la Función de notificación, la información sobre el entorno en el que el Usuario utiliza la Aplicación Móvil (incluyendo, pero sin limitarse al nombre, la versión y la configuración de idioma de la Aplicación Móvil) (en adelante, denominada "Información del Usuario ") se envía automáticamente al servidor de datos de la Aplicación Móvil. Los Desarrolladores del Software, podrán conservar y utilizar la Información de la Usuario enviada automáticamente para notificar la Información de actualización, obtener tendencias de uso y mejorar la experiencia, eficacia y calidad de la Aplicación Móvil.

Artículo 5. Instalación y Descarga

La instalación y descarga de la Aplicación Móvil, mediante la plataforma de distribución digital de aplicaciones móviles (Play Store o App Store), significa aceptar los términos de esta licencia. Si usted como Usuario, no está de acuerdo con los términos de esta licencia, está obligado a eliminar y/o borrar los datos inmediatamente, que componen a Aplicación Móvil de todos sus dispositivos de almacenamiento y dejar de usar este producto.

Artículo 6. Cancelación del Acuerdo

- 1) Los Desarrolladores de la Aplicación Móvil poseen la facultad de dar por concluido el presente Acuerdo con efecto inmediato sin ningún aviso y exigencia al Usuario en los casos donde el Usuario viole alguna de las disposiciones de este Acuerdo.
- 2) Si se produce la finalización de este Acuerdo, el Usuario deberá discontinuar de inmediato el uso de la Aplicación Móvil, etc. y eliminar la Aplicación Móvil, etc. y cualquier reproducción de este mismo.

Artículo 7. Ley vigente y competente

- 1) Este Acuerdo se registrará e interpretará según las políticas, acuerdos y condiciones de la Play Store y App Store.
- 2) Cualquier aclaración o controversia en conexión con o que surja de este Acuerdo deberá resolverse conforme a la revisión específica y haciendo comunicación con los Desarrolladores, mediante algún medio electrónico, cómo también en la Tienda de Aplicaciones respectiva (Play Store y App Store), en primera instancia.

Contrato

Contrato de Licencia de Uso Software“guiAR”

REUNIDOS

(1) El equipo de desarrollo (que en este contrato se llama el licenciante); Y

(2) y con domicilio en , (que en este contrato se llama el cliente);

Afirman que tienen capacidad necesaria para hacer este contrato y

ESTABLECEN

1. Que realiza, entre sus actividades, el reconocimiento de imágenes de diversos sitios turísticos y culturales del Centro Histórico de la Ciudad de México, despliegue de información importante interesante, relevante e histórica acerca de los sitios.
2. Que tiene interés en la compra de una aplicación móvil, realizando este contrato de licencia de uso, por el cual, dicha licencia está para que pueda usar aplicación móvil llamado guiar y pactan las siguientes:

CLÁUSULAS

Propósito del contrato

3. En este contrato el licenciante se obliga a ceder el uso, no en exclusiva, al cliente de la aplicación móvil de software llamado guiAR, que incluirá, la aplicación móvil junto con un manual de usuario y su anexo técnico que se agrega al final de este contrato.

Condiciones de los servicios

4. El licenciante realizará la presentación de los servicios derivados de la cesión del uso del programa informático, teniendo en cuenta lo siguiente:
 - a) El licenciante responderá de la calidad del trabajo desarrollado, con la diligencia necesaria que se le pueda exigir en relación con el programa de software objeto del contrato.
 - b) Al licenciante corresponderá la obligación de gestionar y obtener a su cargo todas las licencias, permisos y autorizaciones administrativas, que sean necesarias para la prestación de los servicios ofertados.
 - c) Todos los impuestos, tributos y tasas que se devenguen por la cesión de uso del programa de software, serán por cuenta del licenciante sin perjuicio que en la facturación por los servicios prestados al cliente se incluya, el IVA/IGIC correspondiente.

- d) El licenciante responderá, de los daños y perjuicios ocasionados al cliente por errores o defectos causados por ella misma o el personal a su servicio en la ejecución de este contrato.
- e) Las obligaciones anteriores tendrán que cumplirlas los empleados, colaboradores y posibles subcontratistas del licenciante, que pudieran participar en la prestación de servicios con el cliente.

Licencia de uso

- 5. El licenciante realizará la cesión de la licencia de uso, no exclusiva, prestando los servicios, de la forma siguiente:
 - a) Licenciará al cliente para el uso de la aplicación móvil descrita en este contrato, entendiendo que dicha licencia le es cedida al cliente como licencia personal, no estando autorizado para ceder la misma a terceros.
 - b) La propiedad intelectual de la aplicación móvil licenciado es de exclusiva titularidad del licenciante, incluyendo dicha propiedad intelectual, tanto la aplicación móvil, como su código fuente y la estructura de su base de datos.
 - c) Queda expresamente prohibido al cliente la reproducción total o parcial, modificación, adaptación, mantenimiento, corrección de errores, cesión, venta, alquiler, préstamo, cesión de uso total o parcial, transmisión del derecho de uso, divulgación, publicación, etc., de la aplicación móvil licenciado por cualquier medio, salvo la utilización de mencionada aplicación móvil a través de los empleados y dentro del ámbito de su propia actividad.

Confidencialidad

- 6. El licenciante se obliga a guardar confidencialidad sobre la información facilitada por el cliente y de la que tenga conocimiento como consecuencia de la prestación de los servicios objeto del presente contrato, debiendo mantener dicha confidencialidad durante un período mínimo de 4 años
El incumplimiento de esta cláusula de confidencialidad supondrá la indemnización de los daños y perjuicios por el importe de \$35,000
También deberán cumplir lo anterior, los empleados, colaboradores y posibles subcontratistas de la parte suministradora, que pudieran intervenir en la relación contractual o prestación de servicios con el cliente.

Protección de datos personales

- 7. Para el supuesto que en el desarrollo de la prestación de los servicios objeto del presente contrato, el licenciante accediera a datos de carácter personal del cliente, deberá cumplir las obligaciones incluidas en la legislación sobre protección de datos de carácter personal, siendo de su exclusiva responsabilidad las infracciones que pudiera cometer, si hiciera uso de los datos personales del cliente, con otra finalidad que no sea la de facilitar los servicios que constituyen el objeto de este contrato.

El licenciante, deberá tomar las medidas técnicas y organizativas necesarias, para garantizar la seguridad de los datos de carácter personal y evitar su alteración, pérdida o tratamiento o acceso no autorizado, siendo de su exclusiva responsabilidad el incumplimiento de esta obligación.

Duración

8. El plazo de duración de este contrato es de 4 años, a partir de su firma. Una vez terminado el primer período de duración, se renovará automáticamente por el mismo periodo, salvo que cualquiera de los contratantes comunique al otro su voluntad de no prorrogar el contrato con 30 días de antelación a la fecha de terminación de este.

Precio

9. El precio por la cesión de la licencia de uso de la aplicación móvil guiAR y sus servicios es de \$7,500 anuales, que se podrán pagar a los 15 días de la firma del contrato, mediante el pago en la cuenta bancaria que señale el licenciante, excluido el IVA/IGIC. Para el caso de renovación del contrato, el precio se actualizará conforme a las variaciones que experimente el IPC durante el año anterior.

Mantenimiento

10. En el caso de cualquier avería o mal funcionamiento en la licencia de uso y servicios, el licenciante realizará la revisión y reparaciones necesarias, en el plazo de 72 horas desde que la incidencia en la utilización del programa informático, le haya sido comunicada por el usuario o cliente. El coste de los servicios de mantenimiento y reparación van incluidos en el precio de la utilización de la aplicación móvil, no pudiendo el licenciante hacer facturación alguna por estos trabajos, salvo que las averías y deficiente mantenimiento se haya producido como consecuencia de una incorrecta utilización de la aplicación móvil por parte del cliente, sus trabajadores y empleados.

Firma electrónica

11. Al usar la funcionalidad de e-sign (Notario Electrónico), para los contratos electrónicos creados en la plataforma de Rocket Lawyer, las partes acuerdan que este contrato es la copia original y que les vincula legalmente. Las partes recibirán un e-mail cuando este contrato haya sido firmado y formalizado por las mismas, sirviendo como prueba de su completa validez legal.

Y como prueba de lo convenido ambos firman el presente contrato por duplicado.

En Ciudad de México], a 16 de diciembre de 2021.

.....
Firma. El licenciante

.....
Firma. El cliente

Póliza de Garantía

guiAR

Póliza de Garantía



La presente Póliza Garantiza que la herramienta de software (App Móvil), por el término de 1 año, en todas partes y mano de obra, contra cualquier defecto de programación, diseño y funcionamiento, respectivamente.

FELICITACIONES: La Herramienta de Software (App Móvil), que acaba de adquirir, mediante su descarga ha sido programado, y diseñado, con diversos entornos, herramientas y sistemas de programación de código abierto y sometido a diversas pruebas para comprobar, corroborar y corregir posibles errores, para ofrecerle a usted una herramienta móvil de calidad. Recomendamos leer cuidadosa y lentamente el manual de usuario, el cual indica con claridad el plazo de vigencia de dicha garantía antes de utilizar su nueva App Móvil.

Condiciones para hacer válida la PÓLIZA DE GARANTÍA:

- ✓ Deberá enviar esta tarjeta. En caso de no estar fechada o sellada al momento de la descarga, deberá ser acompañada de la factura de dicha descarga de la App.
- ✓ Si la App Móvil, después de descargar y seguir todas las instrucciones y modos de uso del manual de usuario no funciona correctamente, comuníquese de inmediato a nuestro número o correo electrónico de atención al usuario.

Ficha de Garantía

Datos del Usuario:

Nombre del Usuario: _____

Dirección: _____

Estado o Ciudad: _____

Teléfono/Celular o Fijo: _____

Correo Electrónico: _____

Versión y Tipo de Sistema Operativo: _____

Atención a Usuarios

✉ guiAR_app@outlook.com

☎ 55-87-13-69-09

ANEXOS

TÉCNICA DE RECOLECCIÓN DE REQUISITOS

El cuestionario propuesto contendría preguntas como las siguientes planteadas:

1. ¿A que rango de edad pertenece?
A) 18 a 23
B) 24 a 29
C) Mayor a 30
2. ¿Cuál es su género?
A) Mujer
B) Hombre
C) Otro/Prefiero no contestar
3. ¿Sabes que es la Realidad Virtual?
A) Si
B) No
4. En estos momentos ¿Utilizas algún dispositivo móvil?
A) Si
B) No
5. ¿Qué sistema operativo tiene su dispositivo?
A) Android
B) iOS
C) Otro(especifique)
6. ¿Has utilizado alguna app que haga uso de la realidad virtual?
A) Si
B) No
7. En caso de tener un dispositivo con S.O Android. ¿Cuál es la versión su dispositivo?
A) 4.1 - 4.4.4
B) 6.0 - 6.0.1
C) 7.0 - 7.1.2
D) 8.0 - 8.1
E) No se/Prefiero no responder
F) Otro (especifique)
8. Si la respuesta anterior fue sí. Del 1 al 10 (Donde 1 es pésima y 10 excelente), ¿Qué tan buena ha sido tu experiencia al utilizar apps con este tipo de tecnología?
A) Indicar el número del 1 al 10
9. ¿Consideras útil actualmente el uso de folletos ilustrativos, informativos, acerca de un sitio turístico/cultural?
A) Nunca
B) Raras veces
C) A veces
D) Siempre
10. ¿Has utilizado alguna app relacionada con el turismo o con la cultura?
A) Si
B) No

11. ¿Has visitado diversos lugares de la CDMX en algún momento de tu vida?
- A) Si
B) No
12. Durante tu visita, ¿Pasaste frente a un lugar del cual te interesó saber su historia o información, pero no había manera de saberla o era muy difícil?
- A) No
B) Pocos lugares
C) Algunos lugares
D) Todos los lugares
13. ¿Te encantaría conocer información interesante acerca de algún sitio turístico cuando estás frente de él?
- A) Si
B) No
14. Indica que tipo de información le gustaría que proporcionara la app
- A) Información histórica
B) Eventos
C) Costos
D) Otro (especifique)
15. ¿Qué sitios turísticos/culturales del Centro Histórico de la CDMX, te gustaría saber información importante, histórica, mediante el reconocimiento de imágenes, de una aplicación móvil? **(Abierta)**
16. Si la app hiciera uso de la realidad virtual ¿Qué tipo de información te gustaría que se mostrara primero?
- A) Información histórica
B) Eventos
C) Costos
D) Otro (especifique)
17. ¿En tu dispositivo móvil tienes una aplicación que posea reconocimiento de imágenes de diversos sitios culturales?
- A) Si
B) No
18. Si fueras una persona extranjera que NO domina/entiende el idioma español, ¿Crees que el uso de audios en diversos idiomas te ayudaría a saber tanto el nombre y la descripción del sitio turístico-cultural visitado?
- A) Si
B) No
19. ¿Estás dispuesto a pagar por una aplicación que te ofrezca información importante, relevante sobre algún lugar turístico/cultural?
- A) Si
B) No
C) Tal vez
20. En caso de que tú pregunta sea afirmativa. ¿En qué rango de precios estás dispuesto a pagar?
- A) De \$50 a \$100
B) De \$101 a \$200
C) De \$201 a \$350
D) De \$81 a \$100

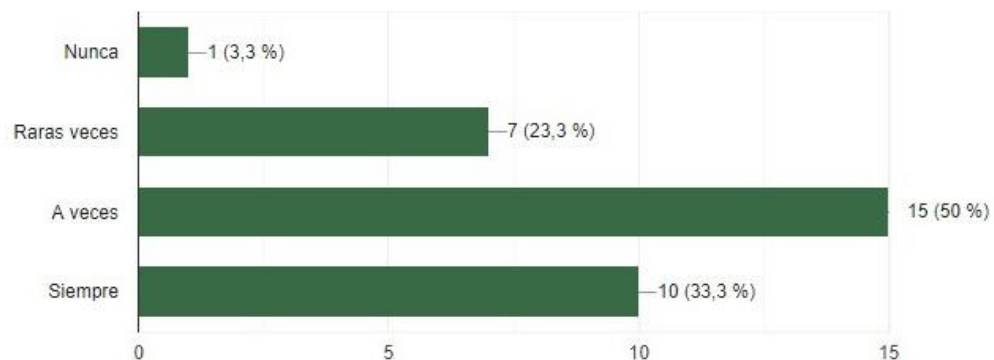
Obtención de resultados a partir del cuestionario

A partir del cuestionario aplicado a una población joven se ha obtenido la siguiente información; información que se presentará en forma de gráficas o datos y se analizarán con cuidado.

Gracias al cuestionario se obtuvo la información, que se mostrará a continuación:

9. ¿Consideras útil actualmente el uso de folletos ilustrativos, informativos, acerca de un sitio turístico/cultural?

30 respuestas

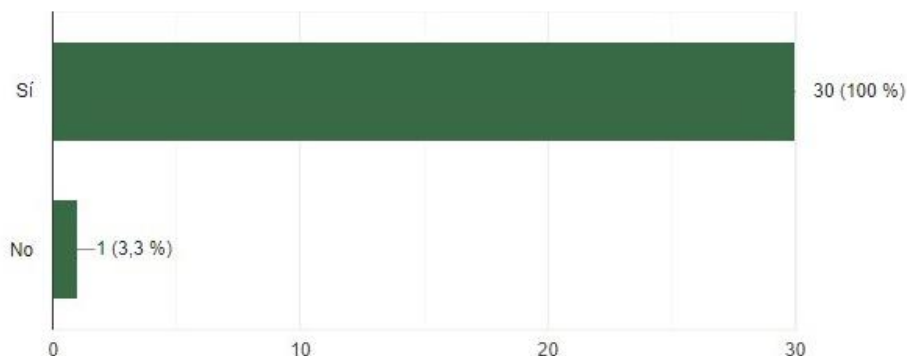


La gráfica anterior representa las experiencias que han tenido los usuarios, sobre la utilidad de los folletos ilustrativos, y donde se dividen en 4 secciones básicas, dándonos el resultado mayoritario que “a veces”, son útiles los folletos, cuando se visita un sitio o lugar turístico y la conclusión de dicha respuesta es dada a que no son muy completos en información del sitio, o suelen ser aburridos, nada atractivos visualmente hablando, sólo por mencionar algunas causas de las cuales la mayoría de los usuarios se destinaron a responder la tercera opción a la presente pregunta.

La siguiente gráfica corresponde si a los usuarios les gustaría saber información importante y relevante acerca de un sitio turístico cuando se encuentre frente de él saber al momento, mediante el apoyo de la cámara de su smartphone para conocer todo lo referente a dicho sitio turístico, y cómo se puede apreciar el 99% de los usuarios encuestados seleccionaron la respuesta afirmativa “Sí”.

13. ¿Te encantaría conocer información interesante acerca de algún sitio turístico cuando estás frente de él?

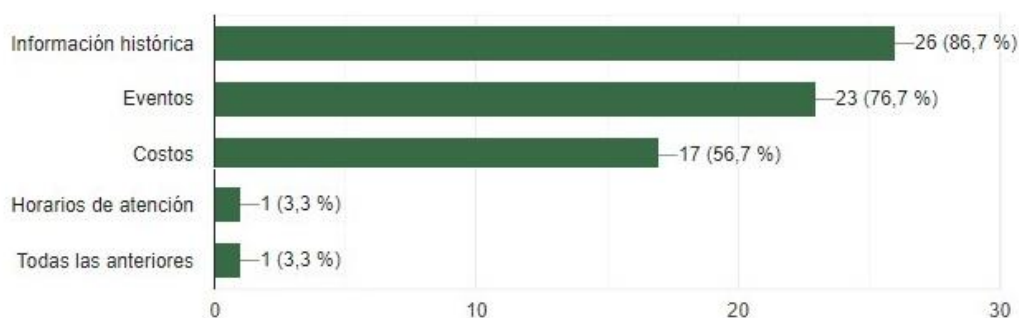
30 respuestas



A continuación se muestra la siguiente gráfica, donde representa las sugerencias sobre el tipo de información que desean saber dentro de la aplicación los usuarios, donde se puede llegar a dividir en 5 secciones básicas:

14. Indica que tipo de información le gustaría que proporcionara la app.

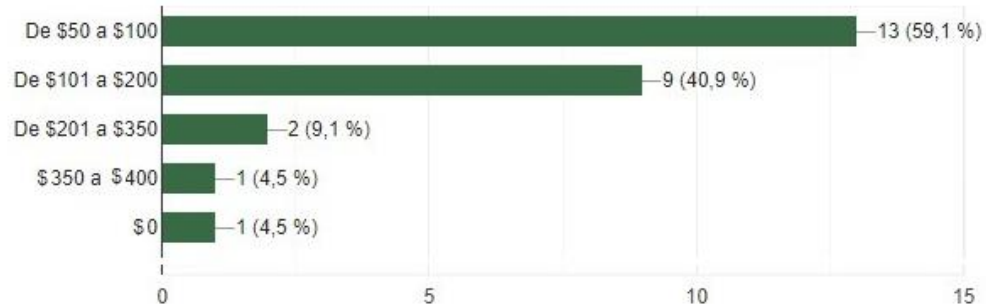
30 respuestas



Como se puede visualizar las 3 opciones más requeridas son información histórica, eventos, que se lleguen a realizar dentro del sitio turístico y los costos, es decir, cuánto cuesta la entrada para poder asistir y por ende disfrutar de todo lo referente al sitio turístico del Centro Histórico de la Ciudad de México.

20. En caso de que tú pregunta sea afirmativa. ¿En qué rango de precios estás dispuesto a pagar?

22 respuestas



El grafico de barras anterior representa las sugerencias que realizaron los usuarios, donde se segmenta en 5 secciones básicas:

1. Gratis (\$0):
El producto es completamente gratuito para que los usuarios puedan usarlo sin necesidad de comprar una licencia de uso indeterminado.
2. \$50 - \$100
La aplicación es de paga y su precio puede variar en el rango de \$50 y \$100.
3. \$101 - \$200
La aplicación es un poco más costosa, pero a su vez su aporte es más para los desarrolladores y diseñadores.
4. \$201 - \$350
La aplicación posee un precio más costoso, pero a su vez su aporte es un poco más para los desarrolladores, diseñadores, y todos aquellos involucrados en el equipo del presente Proyecto de Software.
5. \$350 o más
El precio máximo que puede llegar a tomar la aplicación se localiza dentro de este rango y es la mayor cantidad de dinero que pueden aportar los socios (o lo más rentable, fiable, viable).

CÁLCULOS

Ahora se procede a calcular el **tamaño de la muestra**, con la siguiente formula:

$$\text{Tamaño de la muestra} = \frac{(Z) * (\text{Desviacion estandar}) * (1 - \text{Desviacion estandar})}{(\text{Margen de error})}$$

Donde Z se obtiene de acuerdo con el nivel de confianza de acuerdo con los siguientes valores.

90 % → Z = 1,645

95 % → Z = 1,96

99 % → Z = 2,576

Sustituyendo los datos tenemos que el tamaño de la muestra debe de ser de:

Tamaño de la muestra = 385 personas

Se calculará el Punto de Función, mediante la siguiente fórmula o modelo matemático:

$$\text{PF} = \text{conteo total} \times [0.65 + 0.01 \times \Sigma (Fi)]$$

$$\text{PF} = 86 \times [0.65 + 0.01 \times 57]$$

$$\text{PF} = 105$$

A continuación, se calcula las Líneas de Código para nuestro proyecto expresadas en miles de Líneas de Código (KLDC)

KLDC= (PF) * (Promedio de Líneas por Función)

$$\text{KLDC} = (105) * (60) = 7,5387$$

$$\text{KLDC} = ((105) * (60)) / 1000 = 7.5387 \text{ miles de líneas de código aproximadas.}$$

por lo que: **Estimación de líneas de código = 7,5387 / KLDC=7.5387**

Para realizar el **cálculo del esfuerzo (E)** del COCOMO Básico Orgánico nos apoyaremos de la siguiente fórmula:

$$E = A_b * [\text{KLDC}]^{B_b} = (3.2) * (7.5387)^{(1.05)}$$

$$E = 27.5794 = 28 \text{ persona/mes}$$

Para realizar el cálculo de la duración del proyecto (D) en meses del COCOMO básico Orgánico nos apoyaremos de la siguiente fórmula:

$$D = Cb * E^{Db}$$

$$D = (2.5) * (27.5794)^{(0.38)}$$

$$D = 8.9891 = 9 \text{ meses}$$

Finalmente, para calcular el número de personas necesarias (N), nos apoyaremos de la siguiente fórmula:

$$N = E/D$$

$$N = (32)/9 \text{ meses}$$

$$N = 3.11111 = 3 \text{ personas}$$

Debido a que solo contamos con 6 meses para la realización del proyecto debemos ajustar la duración del proyecto en la fórmula anteriormente realizada para obtener el valor justo de personas que se requieren para realizar el proyecto en el tiempo definido.

$$N = E/D$$

$$N = (28)/(6 \text{ meses})$$

$$N = 4.66666 = 5 \text{ personas}$$

Referencias Bibliográficas:

[1] KENNETH E. KENDALL JULIE E. KENDALL," ANÁLISIS Y DISEÑO DE SISTEMAS" 8^{va} Edición. PEARSON EDUCACIÓN, México, 2011.

[2] Pressman, Roger S. Ph.D, "INGENIERÍA DEL SOFTWARE. UN ENFOQUE PRÁCTICO" 7^{ma} edición. MCGRAW-HILL Interamericana Educación, México, 2010.

[3] Jaramillo Olivares, Daybelis y Salinas Lara, Jorge, Trabajo Terminal :“SARCOMOTOR-Sistema de apoyo en la reducción indirecta de emisores contaminantes de un motor a gasolina”. Escuela Superior de Cómputo del Instituto Politécnico Nacional. ISCCR 127-2015-A005/. Septiembre 2016.Directores: M EN C. Cordero López, Martha Rosa y M EN C. Dorantes González, Marco Antonio.

[4] M EN C. Cordero López, Martha Rosa y M EN C. Dorantes González, Marco Antonio." Técnicas de Recolección de Requisitos". INGENIERÍA DE SOFTWARE. México.

[5] Víctor Diví. (2017). ¿Qué es Dart? inLab FIB
UPC:<https://inlab.fib.upc.edu/es/blog/que-es-el-lenguaje-de-programacion-dart>

[6] López, Sara (2017). Redactora especializada en Digital, Business e Innovación. Firebase: qué es, para qué sirve, funcionalidades y ventajas: <https://www.digital55.com/desarrollo-tecnologia/que-es-firebase-funcionalidades-ventajas-conclusiones>

[7] Gómez, Napoleón (2014). Desarrollador Web Full Stack. ¿Qué es MongoDB?: <https://blog.ida.cl/desarrollo/mongodb-atlas-el-salto-a-la-nube/>

[8] León, Renato, Ferreiro, Osvaldo y Beguería, Sofía. (2021). Área: Gestión de operaciones. Calidad total o TQM, el paradigma que cambió la gestión de la calidad: <https://www.claseejecutiva.uc.cl/blog/articulos/calidad-total-tqm/>

[9] Crespo, Andrea. (Abril 2018). ISO 25000: La calidad del producto software: <https://iso25000.com/index.php/normas-iso-25000/iso-25010>

[10] Monferrer Agu, Raúl (2001). Especificación de Requisitos Software según el estándar de IEEE 830. Ingeniería del Software 5º curso de Ingeniería Informática: http://zeus.inf.ucv.cl/~bcrawford/AULA_ICI_3242/ERS_IEEE830.pdf

[11] Manquillo, Paola Andrea & Arbeláez Polindara, Camila (2015). Estándares de Calidad IEEE. <https://vdocuments.mx/25-estandares-ieee-calidad-de-software.html>

[12] IEEE (Institute of Electrical and Electronics Engineers) El Instituto de Ingeniería Eléctrica y Electrónica). (2005). Standard for Software Configuration Management Plans: <https://ieeexplore.ieee.org/document/1502775>

[13] STALLINGS, WILLIAM. Comunicaciones y Redes de Computadores. Séptima edición. PEARSON EDUCACIÓN, S. A., Madrid, 2004.

[14] KENNETH E. KENDALL JULIE E. KENDALL," ANÁLISIS Y DISEÑO DE SISTEMAS" 6^{ta} Edición. PEARSON EDUCACIÓN, México, 2005.