



# Sistema WEB para el Manejo de Indicadores de los Juicios en el Tribunal de Justicia Administrativa

---

## TESIS

QUE PARA OBTENER EL GRADO DE  
MAESTRO EN SISTEMAS COMPUTACIONALES

PRESENTA:  
ANDRÉS HERNÁNDEZ JIMÉNEZ

DIRECTORA:  
MTRA. RAMONA EVELIA CHÁVEZ VALDEZ

CO-DIRECTOR:  
MC. J. REYES BENAVIDES DELGADO

VILLA DE ÁLVAREZ, COLIMA AGOSTO DEL 2019





# EDUCACIÓN

SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO

Instituto Tecnológico de Colima

"2019, Año del Caudillo del Sur, Emiliano Zapata"

Villa de Álvarez, Colima, **Julio/2019**

OFICIO No. DEPI-197/2019

**C. ING. ANDRÉS HERNÁNDEZ JIMÉNEZ**  
**PASANTE DE LA MAESTRÍA EN SISTEMAS COMPUTACIONALES**  
**PRESENTE**

La División de Estudios de Posgrado e Investigación de acuerdo al procedimiento para la obtención del Título de Maestría de los Institutos Tecnológicos y habiendo cumplido con todas las indicaciones que la comisión revisora hizo a su trabajo profesional denominado "**Sistema WEB para el Manejo de los Indicadores de los Juicios en el Tribunal de Justicia Administrativa**", por la opción de tesis, que para obtener el grado de Maestro en Sistemas Computacionales será presentado por Usted, tiene a bien concederle la **AUTORIZACIÓN** de impresión de la tesis citada.

Sin otro particular por el momento, aprovecho la ocasión para enviarle un cordial y afectuoso saludo.

**ATENTAMENTE**

*Excelencia en Educación Tecnológica*

**RAMONA EVELIA CHÁVEZ VALDEZ**  
**JEFA DE LA DIVISIÓN DE ESTUDIOS DE POSGRADO**  
**E INVESTIGACIÓN**



Secretaría de  
Educación Pública  
INSTITUTO TECNOLÓGICO  
De Colima  
División de Estudios de  
Posgrado e Investigación

C.p. Archivo.

"2019, 30 Años de la convención sobre los derechos del niño"  
Av. Tecnológico No. 1, Villa de Álvarez, Col., C.P. 28976

Tel. 01 (312) 314 0933, 314 0683, 312 6393, 312 9920

www.tecnm.mx | www.itcolima.edu.mx



## **Epígrafe**

*“Las personas no son recordadas por el número de veces que fracasan, sino por el número de veces que tienen éxito”.*

**Thomas Alva Edison.**

## ***Agradecimientos***

Quiero agradecer en estas líneas a todas las personas que son parte de mi día a día y que fueron parte de ésta travesía, a todos aquellos que estuvieron junto a mí, pero sobre todo a los que seguirán a mi lado.

### ***A mi madre***

Me encuentro enteramente agradecido a ella y siempre lo estaré, pese a la situación que nos encontremos, por ejemplo, la enfermedad y las carencias, siempre ha encontrado la forma de apoyarme, siempre ha encontrado las palabras para hacer que me motive a realizar los deberes, aunque las palabras duelan, sé que lo dice por algo. Bien dicen que las palabras de un padre o una madre son sabias.

### ***A mi pareja***

A ti que me has acompañado durante este proceso, que me has dado palabras de aliento, o me has reconfortado cuando me encontraba exhausto, y que me regañas cuando casi no duermo por cumplir con mis responsabilidades, sé que hemos hecho sacrificios, y estos sacrificios nos darán grandes beneficios, espero que siempre estés a mi lado, ya que será una alegría compartir todos y cada uno de mis logros contigo.

### ***A mis compañeros***

A ustedes amigos y compañeros, y en especial a aquellos que estuvieron junto a mí en este proceso, muchas gracias por ser parte de mi historia, sé que aunque puede no parecer mucho, cada momento, chiste o plática que tuvimos juntos forman parte de quien soy.

### ***A mis profesores***

A ustedes profesores, doctores y personal laboral de ésta escuela, agradezco todo el tiempo y esfuerzo que aplicaron para que yo aprendiera, sé que todo lo absorbi de conocimiento en el tiempo que pase con ustedes, tiene un propósito y que será complemento de mi ser para superar los retos que vengan en el futuro.

Finalmente, quiero agradecer a mi directora de tesis, la maestra Ramona Evelia Chávez Valdez, a quien conocí hace más de 7 años cuando realizaba mis estudios de ingeniería, una persona dedicada, amable y abierta a ayudar siempre, sin importar si la persona a la que le ofrezca su ayuda lo agradezca o amerite.

## **Resumen**

En esta investigación se realizó el proceso de reingeniería para corregir fallos y aplicar mejoras al Sistema de Seguimiento de Demandas (SiSeDe) implantado en el Tribunal de Justicia Administrativa del estado de Colima, México. Se trata de un sistema de información con características web que facilita el seguimiento a los proyectos de sentencias, enfocándose en un módulo de indicadores de productividad para mejorar los tiempos de atención a los juicios de carácter administrativo. Para controlar el proceso de ingeniería de software se empleó el Proceso Unificado Ágil, por sus características para manejar proyectos pequeños, cambiantes y de cierto riesgo. Los resultados residen en un software que presenta con elementos estadísticos el tiempo de atención a la demanda, considerando métricas de evaluación de procesos por tiempo, así como las métricas establecidas por la Ley Administrativa del estado de Colima, México. Los actores responsables de la impartición de justicia administrativa, con apoyo de las señales de semáforos, pueden visualizar en el sistema la productividad de los actores que intervienen en la demanda, identificando cuántas demandas han atendido y cuántas se tienen en prevención o en rezago. Estos datos permiten a la dependencia incorporar estrategias para mejorar la atención de los procesos de justicia.

## **PALABRAS CLAVE**

Indicadores, Juicios, Productividad, Reingeniería, Seguimiento

## **Abstract**

In this research, the process of reengineering was carried out to correct faults and apply improvements to the Follow-up System of Demands (SiSeDe) implanted in the Administrative Court of the State of Colima, Mexico. It is an information system with web features that facilitates the follow-up of sentencing projects, focusing on a module of productivity indicators to improve the time of attention to administrative trials. To control the software engineering process, the Agile Unified Process was used, due to its characteristics to handle small, changing and risky projects. The results reside in a software that presents the time of attention to the demand with statistical elements, considering metrics of evaluation of processes by time in addition to those established by the Administrative Law of the State of Colima, Mexico. The actors responsible for administering administrative justice, with the support of signals like traffic lights, can visualize in the system the productivity of the actors that intervene in the demand, identifying how many demands they have attended and how many are in prevention or in lag of attendance. These data allow the administrative personal to incorporate strategies to improve the attention of the justice processes.

## **KEYWORDS**

Follow-up, Indicators, Productivity, Reengineering, Trials

## ÍNDICE

CAPÍTULO I Investigación preliminar .....	1
1.1 La naturaleza del problema .....	1
1.2 El contexto del problema .....	2
1.3 Revisión de la literatura .....	5
1.4 Propuesta de solución .....	6
1.5 Justificación .....	9
1.6 Objetivos .....	9
1.6.1 General .....	9
1.6.2 Específicos .....	9
1.7 Hipótesis .....	10
1.8 Organización del documento .....	11
CAPÍTULO II ESTADO DEL CAMPO DE CONOCIMIENTO .....	12
2.1 Marco Histórico .....	12
Uso de los Sistemas de Información en juicios .....	12
Métodos o planes de cálculo de productividad .....	13
2.2 Marco Contextual .....	13
2.3 Marco Teórico .....	15
2.3.1 Indicadores KPI .....	15
2.3.2 MySQL .....	15
2.3.3 Bases de datos Relacionales .....	16
2.3.4 PHP .....	16
2.3.5 Laravel .....	16
2.3.6 JavaScript .....	17
2.3.7 Framework .....	17
2.3.8 HTML .....	18
2.3.9 CSS .....	18
2.3.10 Bootstrap .....	18
2.3.11 Talend Open Studio .....	19
2.3.12 Enterprise Architect Project .....	19
2.3.13 Balsamiq Muckups .....	19
2.3.14 Re ingeniería de Software .....	19
2.3.15 Modelado de requisitos .....	21

2.3.16 Modelo de clases .....	21
2.3.17 Diseño de interfaces.....	21
2.3.18 Pruebas de Software.....	24
2.3.19 PHPUnit .....	24
2.3.20 Filezilla .....	24
2.3.21 Dominios y Subdominios.....	25
CAPÍTULO III MÉTODOS EMPLEADOS .....	26
3.1 Metodología .....	26
3.2 Materiales y equipo .....	28
CAPÍTULO IV DESARROLLO DEL SISTEMA .....	31
4.1 Modelado de Casos de Uso.....	31
4.3 Modelo de Requisitos .....	32
4.3.1 Requerimientos funcionales .....	32
4.3.2 Requerimientos No Funcionales .....	34
4.4 Modelo de Clases .....	34
4.5 Modelo de Base de Datos.....	35
4.5.1 Diagrama de base de datos .....	35
4.5.2 Diccionarios de datos .....	37
4.6 Diseño de Interfaces.....	39
4.6.1 Modelado .....	39
4.6.2 Creación de las nuevas interfaces .....	41
4.7 Diseño del modelo de medición por indicadores .....	42
4.7.1 Indicadores generales de actividad .....	42
4.7.2 Medición estadística.....	43
4.8 Codificación .....	44
4.9 Pruebas .....	45
4.9.1 Pruebas Unitarias.....	45
4.9.2 Pruebas de Integración .....	47
CAPÍTULO V Resultados .....	49
5.1 SiSeDe y el módulo de Indicadores de Productividad Laboral .....	49
5.2 Configuración y Carga del sistema en el servidor.....	52
5.2.1 Carga de los archivos que componen el proyecto .....	54
5.2.2 Configuración del dominio.....	56
5.2.3 Creación y montaje de la base de datos .....	58

5.2.4 Enlace de configuraciones .....	60
5.2.5 Pruebas de despliegue.....	60
5.2.6 Configuraciones adicionales .....	62
CAPÍTULO VI CONCLUSIONES .....	64
6.1 Conclusiones .....	64
Bibliografía .....	65
Anexos .....	68
1. Artículo de publicación Científica.....	68

## ÍNDICE DE FIGURAS

Figura 1 Organigrama Actual del TJA .....	3
Figura 2 Diagrama de flujo del proceso de seguimiento de demanda.....	4
Figura 3 Modelo conceptual de la integración de indicadores de productividad al SI SiSeDe .....	8
Figura 4 Ciclo de vida del software acorde a PUA .....	27
Figura 5 Caso de Uso general del SiSeDe.....	32
Figura 6 Requerimientos funcionales del sistema .....	33
Figura 7 Requerimientos No funcionales del sistema .....	34
Figura 8 Diagrama de Clases del sistema.....	35
Figura 9 Diagrama de base de datos del proyecto SiSeDe.....	36
Figura 10 Bosquejo general del menú de usuario .....	40
Figura 11 Bosquejo de la representación de Indicadores .....	41
Figura 12 Plantilla de diseño AdminLTE .....	42
Figura 13 Procedimiento obtención de días hábiles.....	44
Figura 14 Código de la vista obtención de seguimiento de expediente.....	45
Figura 15 Creación de una prueba unitaria .....	45
Figura 16 Prueba unitaria de emails.....	46
Figura 17 Ejecución de prueba unitaria.....	46
Figura 18 Prueba de integración de paquetes.....	47
Figura 19 Seguimiento paso a paso de demandas .....	49
Figura 20 Visualización del semáforo.....	50
Figura 21 Información estadística general .....	51
Figura 22 Información estadística individual .....	51
Figura 23 Características del servidor.....	53
Figura 24 Panel de administración del servidor.....	54
Figura 25 Panel de cuentas FTP.....	55
Figura 26 Administrador de archivos del servidor .....	56
Figura 27 Dominio de un sitio web .....	57
Figura 28 Creación de subdominio.....	57
Figura 29 Código para creación de usuario de base de datos .....	58
Figura 30 Script para creación de base de datos .....	59
Figura 31 Montaje de la base de datos .....	59
Figura 32 Configuración del proyecto.....	60



Figura 33 Prueba de visualización .....	61
Figura 34 Registro de demanda .....	62
Figura 35 Configuración de parámetros de PHP .....	63

Índice de tablas

Tabla 1 Top Ten de lenguajes Open Source en Github y SourceForge .....	17
Tabla 2 Diccionario de datos de la tabla expediente .....	38
Tabla 3 Diccionario de datos de la tabla de seguimiento .....	39
Tabla 4 Parámetros de evaluación para el semáforo .....	43
Tabla 5 Grupos de pruebas aplicados al sistema.....	47

# ***CAPÍTULO I Investigación preliminar***

En este capítulo se presenta el contexto del problema, referentes a sistemas para la gestión de los procesos jurídicos relacionados con la justicia administrativa y la eficiencia en la atención de los procesos relacionados.

## ***1.1 La naturaleza del problema***

Dentro de toda organización, uno de los objetivos principales es conducirse hacia la eficiencia organizacional, una de las formas para validar la eficiencia es con el establecimiento de indicadores y las evaluaciones sistemáticas que se realicen a éstos, dado que estos sirven como referencia para evidenciar el crecimiento de la institución. El desempeño en si se presenta como un constructo clave en la dirección estratégica (March & Sutton, 1997), que se mide periódicamente, en el caso de estudio de este documento es con fundamento en los indicadores de productividad laboral establecidos, lo que a su vez determina el nivel de eficiencia de la misma.

La productividad es “aquella relación entre lo producido y los medios empleados, tales como mano de obra, materiales, energía, etc...” (RAE, 2017). Con base en esto podemos decir que productividad es la capacidad que se tiene para dar más (más resultados, más producido, más dinero, más beneficios, etc.) esto sin tener que aumentar los recursos implicados. Se expresa como una medida para conocer qué tan bien un país, industria o unidad de negocios utiliza los recursos.

Acorde a la investigación realizada por (Mejía, 2007) la productividad concebida como estrategia de gestión en las obras, se convierte en un indicador importante del desarrollo constructivo, ya que relaciona intrínsecamente diversos factores claves que inciden directamente sobre el desempeño de los procesos, como la calidad, la seguridad, el costo, el tiempo, la planeación y el control.

Considerando lo que mencionan (March & Sutton, 1997; Mejía, 2007), se tiene entendido que la productividad en una organización es una métrica substancial que permite evaluar el rendimiento de los elementos que ejercen en dicha institución, y que además de ser un indicador del estado de la misma, puede servir como índice para producir estrategias de control o de mejora, que permitan mejorar los procesos y acciones llevadas a cabo en la empresa, para proveer a ésta en el incremento del desempeño presentado antes, durante y después de aplicar dichos planes de trabajo.

En el estado de Colima, México, en el Tribunal de Justicia Administrativa (TJA) se presenta el caso de rezago en la atención de los expedientes que se presentan ante la dependencia, ya que con el paso de los años el número de demandas han aumentado, mientras que, el personal laboral de la dependencia sigue siendo el mismo, lo que refleja un crecimiento no proporcional entre los casos atendidos con

la cantidad de personal dedicado. Por ejemplo, en el año 2014, se atendieron 1050 demandas, en 2015 se recibieron y procesaron 1034, en 2017 se registraron 1128 asuntos respectivamente (TCA del Estado de Colima, 2017). Esos datos son la base para proyectar estrategias que mejoren la productividad del personal que interviene, así como la eficiencia en la atención de las demandas que se reciben.

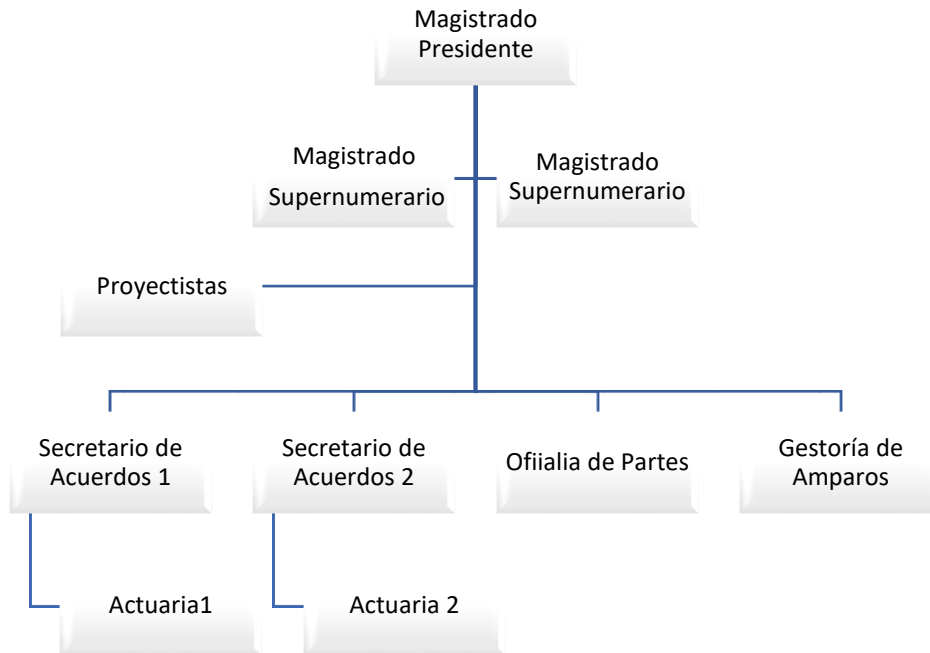
Para solventar dicha problemática se propone un Sistema de Información (SI) que haga uso de tecnologías para optimizar la gestión de los procesos; ello se sustenta en el censo del Catálogo de Dependencias e Instituciones de Interés Público CADIIP 2014-2015 realizado por el Instituto Nacional de Estadística y Geografía en México (INEGI, 2014), que reveló que las innovaciones tecnológicas, están siendo utilizadas como instrumentos para la gestión de los procesos de impartición de justicia, como se observa en los Tribunales Federales y en algunos Tribunales Estatales Administrativos del país. De acuerdo con las 19 instituciones disponibles en las 32 entidades que conforma el país, únicamente en dos entidades (Guanajuato y Sinaloa) se han implantado un SI en un ambiente WEB para el seguimiento de los procesos que atienden dichas dependencias.

Actualmente, las acciones están encaminadas para que el TJA de Colima se anexe a los estados antes mencionados, a través de la implementación de un SI que recopila y gestiona los procesos que se realizan cuando un ciudadano antepone una demanda a una institución de rubro local, estatal o federal.

Siendo esto el motivo y razón del desarrollo de esta investigación aplicada, proponiendo la creación de un módulo de medición de indicadores basado en tiempos de procesamiento, que se agregó al Sistema de Seguimiento de las Demandas (SiSeDe) que ya se encuentra implementado en el TJA, el cual otorga al usuario un tablero de información (Dashboard) en donde se muestra el estado actual de los procesos y en general de cada una de las demandas que se atienden.

## ***1.2 El contexto del problema***

Actualmente el TJA del Estado de Colima pasó de ser un tribunal unitario a un tribunal colegiado, esto significa un cambio estructural y de delegación de responsabilidades para los usuarios internos que laboran en el tribunal, ya que ahora se tiene un Magistrado Presidente el cual cuenta con el apoyo de dos Magistrados Supernumerarios que le ayudarán en la toma de decisiones, esto se puede observar en la Figura 1, la cual representa la estructura orgánica activa de esta institución.



*Figura 1 Organigrama Actual del TJA*

Como se puntualizó anteriormente, las acciones y responsabilidades de cada uno de los individuos que laboran en el tribunal corresponde a una parte dentro del proceso de seguimiento de las demandas, desde el punto de vista organizacional para el Magistrado Presidente se convierte en un factor de estudio, dado que es necesario verificar la productividad de cada una de las áreas para así acotar la diferencia de lo que el tribunal produce con el crecimiento exponencial de recepción de demandas. Es en este proceso de seguimiento donde se pretende localizar aquellos puntos de inflexión que pueden ralentizar los tiempos de respuesta a las demandas, siendo este el motivo por el que se desea implementar un sistema de indicadores que muestren los tiempos de actividad de cada uno de los actores y los tiempos del juicio en forma general.

El proceso de recepción y atención de demandas por parte del Tribunal se establece como se presenta en la Figura 2.

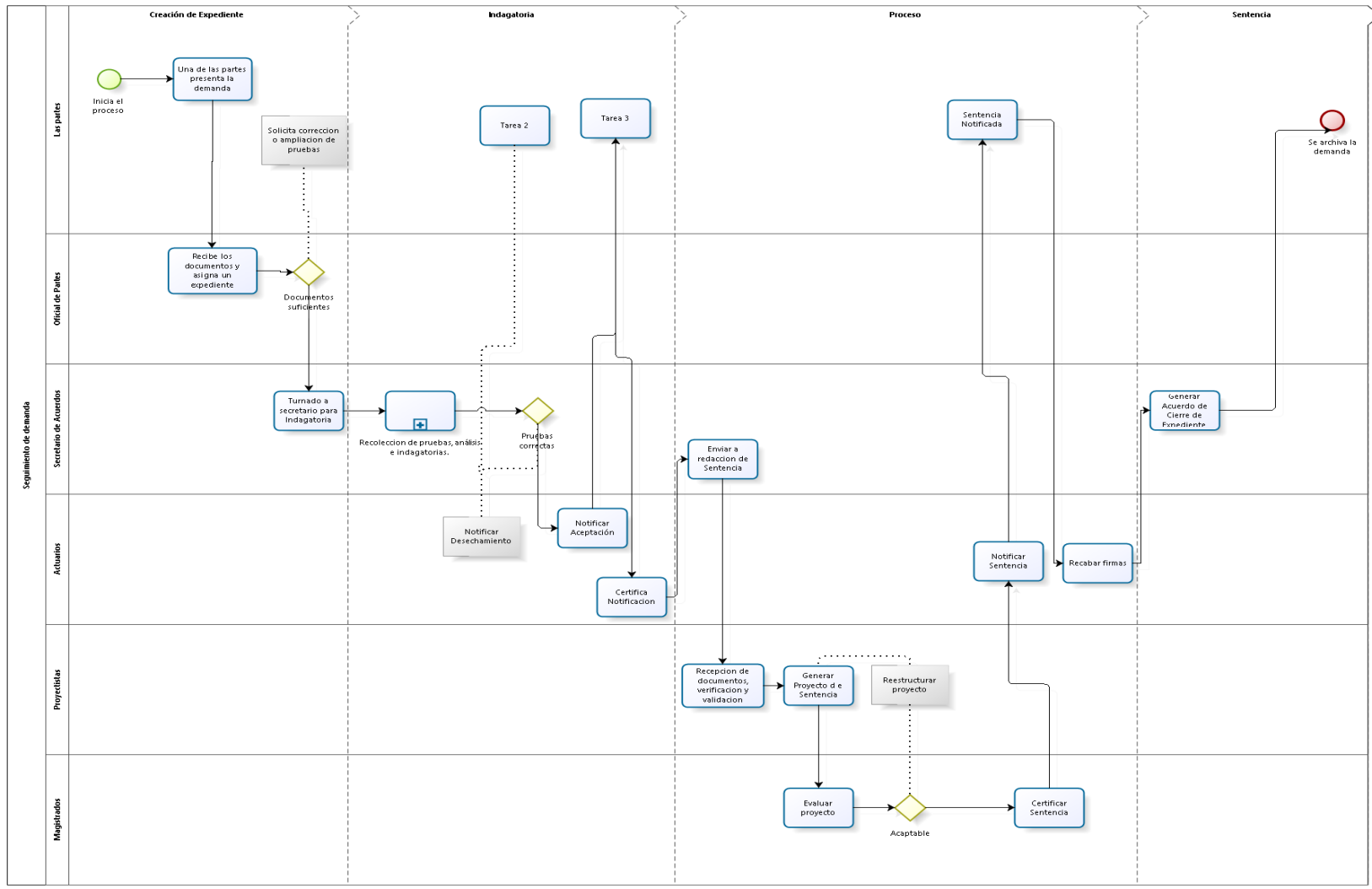


Figura 2 Diagrama de flujo del proceso de seguimiento de demanda

Como se observa en la Figura 2, el proceso de seguimiento de demandas comienza cuando un usuario presenta un oficio ante la Oficialía de Partes la cual se encarga de hacer la recepción y validación de la misma mediante la asignación de un folio de expediente, este folio establece el número de expediente por el cual se relacionará al asunto a atender; el siguiente proceso entra por parte de los Secretarios de Acuerdos, siendo ellos los encargados de evaluar y procesar el recurso de demanda, esto mediante la revisión de las pruebas ofrecidas como parte del expediente y procediendo a redactar un acuerdo de procedencia o improcedencia del expediente, el cual una vez realizado debe turnarse a los actuarios para notificar a los interesados sobre el resultado (parcial o definitivo) del expediente en curso. Una vez que sean notificados las partes (demandante y demandado), se procede a establecer un periodo de retroalimentación el cual depende de la situación por la que se interpuso dicho recurso legal, al finalizarse el periodo de retroalimentación, si existen repuestas de las partes se analizan las mismas y se realiza el mismo procedimiento por parte de los Secretarios de Acuerdos hasta que todas las dudas, puntos de vista o aclaraciones sean suficientes para que se pueda dictaminar una sentencia y esta sea notificada nuevamente por los Actuarios.

Los encargados de proponer el dictamen de una sentencia a un expediente son los Proyectistas, siendo estos los que analizan a detalle cada una de las evidencias presentadas, así como todos los acuerdos realizados por los Secretarios, con la finalidad de obtener un resultado al pleito o alegato en cuestión, siguiendo lo establecido en el marco legal. Finalmente, es labor de los Magistrados el verificar si el proyecto de sentencia es consistente a lo establecido en lo que dictaminan las leyes, lo que genera una sentencia y solución a la problemática que se pretende solucionar mediante el Juicio Administrativo en cuestión.

### ***1.3 Revisión de la literatura***

Cuando se habla del concepto de productividad debemos tener en cuenta que para todo tipo de industria existen formas para evaluar su desempeño, por ejemplo, (Rojas, Eduardo, & Gaona, 2001) desarrollaron una aplicación que permitía a los usuarios (Empresas PYME) determinar el estado actual de su productividad indicando cuál de los insumos tangibles o medibles de la empresa son los más representativos y sobre cuales se debe de enfatizar la estrategia de mejoramiento de productibilidad, esta aplicación se basa en el Modelo de Productividad Total (MPT) desarrollado por David J. Sumanth en 1990.

Lo común es encontrar aplicaciones para el incremento de la productividad en un área que involucre el desarrollo de algún producto o servicio, que conlleve un intercambio de valores, es decir, que exista flujo de efectivo que motive a la organización a generar más, sin consumir tanto. Sin embargo, el concepto de

mejora que representa el cálculo de la productividad de las organizaciones, se ve implicado en todo tipo de organización, sin importar su índole (pública o privada). En el trabajo desarrollado por (Vázquez & Estrada, 2016) se obtuvo un modelo de medición del desempeño en centros públicos y de investigación en la región de Guanajuato, Gto. México, en el cual utilizan los factores de la organización para generar indicadores de medición de desempeño en función de la capacidad de gestión de la organización a nivel negocio con ayuda de factores de gestión comercial, gestión financiera y de producción que presenta la empresa.

Una vez establecido esto, podemos decir que, si hablamos de la medición de productividad durante los procesos legales, por el momento aún no existen herramientas tecnológicas o de *software* que permitan la medición de ésta, ya que existen diversos factores tanto cuantificables como no cuantificables que intervienen en el cálculo de la misma.

#### **1.4 Propuesta de solución**

Como se mencionó anteriormente es necesario establecer un modelo que permita realizar la evaluación de la productividad de los actores que participan en el proceso de demanda, y esto es posible con la ayuda de las herramientas específicas diseñadas para evaluar el desempeño de un proceso o actividad, es decir, hacer uso Indicadores Clave de Desempeño (KPI's por sus siglas en ingles), usando aquellos que se basen en la medición del tiempo transcurrido entre procesos, uno que resulta cómodo de aplicar es el la métrica de rango DAU/MAU (Daily Active Users/ Monthly Active Users por sus siglas en ingles), las cuales se definen como la Proporción de Usuarios Activos Diarios (DAU) a Usuarios Activos Mensuales (MAU) y ésta, mide la rigidez de su producto, es decir, la frecuencia con la que la gente interactúa con su producto. DAU es la cantidad de usuarios únicos que interactúan con su producto en una ventana de un día. MAU es la cantidad de usuarios únicos que interactúan con su producto en un período de 30 días (GECKOBOARD, 2017).

La proporción de DAU a MAU es la proporción de usuarios activos mensuales que interactúan con su producto en una ventana de un solo día.

La clave que se define para calcular la relación DAU / MAU es definir qué "activo" es para su producto. Puede ser cualquier cosa, desde una compra (para comercio electrónico o aplicaciones móviles), páginas vistas / videos vistos / comentarios (para medios / editorial) o inicio de sesión / uso de productos (para empresas SaaS o aplicaciones móviles).

La fórmula que se utiliza para poder calcular la relación DAU/MAU es:

$$\frac{\text{tiempo empleado en el subproceso}}{\text{tiempo total de la demanda}} = \% \text{ de relación DAU/MAU}$$

La métrica de mayor peso para efectos de este estudio, se encuentra definida en la Ley de Justicia Administrativa (H. Congreso del Estado de Colima, 2018), en sus artículos 52, 60, 64, 71 y 91 establece las pautas de los tiempos que deberán de tomar los procesos de seguimiento de demanda, estableciendo que los procesos de recepción, verificación de documentos, preparación de acuerdos para desechar o admitir el recurso de reclamación, deberán realizarse dentro de los primero tres días hábiles a partir de su recepción, se estipula también que si se encontrase una petición o recepción de un amparo, el tiempo para estos procesos obtiene una ampliación de cinco días laborales. Los tiempos de notificación se establecen a un máximo de tres días hábiles, para finalizar, el proceso de redacción de sentencias y aplicación de las mismas cuentan con un periodo máximo de 20 días hábiles para su acción y de igual forma si existiese alguna presentación de amparo por alguna de las partes se presenta una ampliación de tiempo para recepción de documentos, investigación de fondo y aclaraciones con un límite máximo de 15 días. Por lo tanto, en el mejor de los casos una demanda puede circundar entre un periodo de un mes en todo su proceso, y en el peor de los casos, cuando se presente aclarativas o amparos, un tiempo estimado de cuatro meses como máximo.

En la Figura 3 se presenta el modelo conceptual de integración de indicadores de productividad en el sistema SiSeDe, en éste se muestra el flujo de información propio del seguimiento de demandas; existe una interface de acceso por cada actor que participa en el proceso, la información fluye de manera vertical de acuerdo al marco legal del TJA; con lo que se almacena en una base de datos genérica al proceso de demandas y sentencias. Una vez que se han almacenado los datos, se procede a la evaluación de éstos para:

- Determinar la productividad por expediente, considerando si éste se encuentra dentro de los tiempos normativos que establece la Ley de Justicia Administrativa.
- Medir el desempeño de cada uno de los actores que intervienen en el proceso de la demanda, siempre y cuando sean aquellos que laboren en el TJA.
- Contabilizar los expedientes procesados por periodo de tiempo, generalmente por mes y año.
- Visualizar los propios indicadores a los usuarios en los módulos correspondientes, dependiendo sus credenciales y nivel de acceso a la información.





Figura 3 Modelo conceptual de la integración de indicadores de productividad al SI SiSeDe

## **1.5 Justificación**

Teniendo en cuenta el principio de que este trabajo representa la mejora y continuación de un módulo del proyecto SiSeDe para el TJA del estado de Colima, y dado que en junio del 2017 se liberó la primera versión del sistema, su justificación está basada en las siguientes razones:

- Mejorar la eficiencia y eficacia de la atención de las demandas que recibe el TJA.
- Informar a los empleados del TJA de los tiempos de atención de una demanda para solicitar apoyo cuando este sea necesario.
- Proporcionar un medio de seguimiento inmediato por parte del Magistrado Presidente en la atención de las demandas y tomar decisiones al respecto.
- Ofrecer un medio que proporcione transparencia de la productividad de todos los empleados y de la cantidad de demandas procesadas, permitiendo que se creen estrategias de trabajo, tanto individuales como de equipo.
- Contar con la infraestructura tecnológica que sirva de base para acceder en el corto plazo a lo que será la plataforma de juicio en línea.

## **1.6 Objetivos**

### **1.6.1 General**

Desarrollar e implementar el módulo de indicadores de productividad en los juicios administrativos del TJA para su integración al Sistema de Seguimiento de Demandas (SiSeDe) ya existente.

### **1.6.2 Específicos**

Dentro de las actividades necesarias para lograr el objetivo general se tienen las siguiente:

- Investigar los diversos factores que determinan la productividad de los procesos del TJA, mediante la ingeniería de requisitos, para establecer una base teórica que fundamente el nuevo módulo de indicadores.
- Aplicar el proceso de ingeniería de software con base a una metodología formal, para mejorar la calidad de la propuesta.
- Conocer los procesos que implementa el SiSeDe instalado en el TJA para identificar la estructura lógica e incorporar la estructura correspondiente al nuevo módulo.
- Probar el nuevo módulo de indicadores, con el propósito de verificar el funcionamiento del mismo y obtener retroalimentación por parte de los usuarios finales.

- Capacitar en el uso del SiSeDe a los usuarios del TJA a fin de que identifiquen errores y mejoras en el sistema actual, y así garantizar la funcionalidad del nuevo módulo.
- Implementar el módulo de indicadores de productividad, con el fin de ayudar a la toma de decisiones.

### **1.7 Hipótesis**

Dada la temática que presenta esta tesis la hipótesis declarada es:

**El seguimiento a los indicadores de productividad, mediante un sistema informático, permite al Tribunal de Justicia Administrativa realizar evaluaciones periódicas de la eficiencia en la atención de los juicios que interponen los actores externos a fin de mantener la mejora continua.**

#### **Variable Independiente**

Las variables independientes involucradas son los actores externos al proceso de la demanda, así como los documentos que se presentan como parte del expediente.

#### **Variable Dependiente**

La variable dependiente son los tiempos de registro y de corte de cada actor dentro del proceso de la gestión jurídico administrativo, lo que permite percibir el tiempo empleado en la investigación, proceso y respuesta de las demandas.

## **1.8 Organización del documento**

El presente documento de tesis, en el que se realiza el proceso de reingeniería en el sistema SiSeDe se encuentra organizado de la siguiente forma;

En el capítulo II “Estado del campo del conocimiento” se muestran los marcos histórico, contextual y teórico de los proyectos existentes que sean similares a la propuesta de trabajo.

En el capítulo III “Metodología aplicada” se describe la metodología utilizada durante el desarrollo de este proyecto de tesis.

En el capítulo IV, “Desarrollo del sistema” se presenta de forma detallada el procedimiento realizado para la reingeniería del SiSeDe que se encuentra implementado en el TJA del estado de Colima.

En el capítulo V “Resultados” se presentan las pruebas realizadas al SI para comprobar el correcto funcionamiento, así como el análisis de los resultados.

Finalmente, en el capítulo VI “Conclusiones” se presenta la recopilación de las conclusiones sobre el proceso de reingeniería e implementación de indicadores de productividad en el seguimiento de demandas.

## ***CAPÍTULO II ESTADO DEL CAMPO DE CONOCIMIENTO***

En el presente capítulo se desarrolla la indagación de alternativas o herramientas computacionales, así como diversos modelos de medición que son útiles para las organizaciones en el control de sus actividades para fines de eficiencia y eficacia, lo que les permite realizar toma de decisiones con base a los indicadores que se establezcan.

### ***2.1 Marco Histórico***

Para que sea posible el entendimiento del cómo se integran los procesos asistidos por Sistemas de Información, es necesario integrar el conocimiento de los usos y aplicaciones que se han implementado con la finalidad de mejorar los servicios. Uno de los aspectos que se tiene en consideración es el término adaptado para la integración de SI en la impartición de justicia conocido como E-Justice; el otro es aquel enfoque de los SI como apoyo para la toma de decisiones mediante el cálculo de productividad, por lo que se encuentra lo siguiente.

#### Uso de los Sistemas de Información en juicios

Durante la última década, la incorporación de Tecnologías de la Información y las Comunicaciones (TIC) en las instituciones de los sistemas de justicia se ha incrementado, esto es debido a una serie de factores que escapan a lo netamente judicial, y esto es observable enormemente en los países latinoamericanos, tanto a nivel privado como en instituciones públicas. Esto indica que la incorporación de las TIC son un elemento de reforma y modernización en la impartición de justicia (Flórez, 2011).

Como se mencionó anteriormente, existen evidencias de que la incorporación de las TIC en los procesos judiciales es un avance en la impartición de justicia, por ejemplo, Flórez (2011) , expone como el uso de las Tecnologías de Información (TI) en sistemas judiciales en Colombia, favoreció el ingreso de las mismas desde sitios de difícil acceso, detallando entre otros la situación general de la infraestructura informática y de telecomunicaciones, y da ejemplo del cómo dos de las mayores instituciones de Justicia Colombianas enfrentan este reto y aplican buenas prácticas, el Consejo Superior de la Judicatura colombiana y El Centro de Documentación Judicial.

Otro ejemplo se observa en el que expuso Silva (2012) en su ponencia Proceso Judicial Electrónico en donde expone una experiencia concreta en el contexto de la justicia en Brasil, donde el Consejo Nacional de Justicia realizó la implantación de un sistema de tramitación electrónica de Procesos Judiciales, desarrollado bajo un ambiente de programación en Java, este permitía tramitar y ejecutar un caso en el marco de competencia disciplinaria de la institución.

No obstante, esta tendencia no es propiamente de los países Sud-Americanos como se mencionó en los casos anteriores, ya que incluso en México, existen instituciones que hacen uso de SI para complementar sus actividades de impartición de justicia, por ejemplo, dentro de las 32 entidades que componen al país, por lo menos dos ya han implementado un SI en un ambiente WEB para mejorar la atención brindada al contexto de la impartición de justicia como lo son los estados de Guanajuato y Sinaloa (INEGI, 2014), los cuales pese a sus diferencias en amplitud de servicios, se prestan como herramienta de trabajo que facilita el incremento de productividad de dichas instituciones.

### Métodos o planes de cálculo de productividad

En el ámbito del cálculo y medición de productividad existe una gran controversia respecto a cómo definirla, ya que la mayoría de las herramientas disponibles que ayudan a establecer indicadores que midan la productividad de una empresa o institución, están enfocadas en el ámbito financiero, es decir, se preocupan por los costos tanto de insumos/mano de obra respecto a las ventas, lo que para una organización que provee un servicio que no tiene coste para el usuario, en resumen un organismo público, el tipo de métricas que establecen no concuerdan con la misma.

Uno de los SI que se encuentran disponibles para realizar la medición de la productividad es aquel conocido como EffiWork proveído por la empresa WorkMeter, este software de pago por uso proporciona a las organizaciones una herramienta que permite medir la productividad de los empleados por medio de métricas que evalúan el tiempo de ejecución de procesos dentro de las actividades del mismo, además de que permite crear planes y objetivos de trabajo los cuales se implementan como indicadores mismos para la evaluación de la productividad (WorkMeter, 2018).

Por ejemplo Matilla & Chalmeta (2007) desarrollaron un trabajo en el cual describe la implementación de la metodología CMI-IRIS en un sistema de medición de rendimiento que se basa en un Cuadro de Mando Integral Empresarial, dentro de las actividades que marca la metodología se encuentran la definición y validación de sistema de indicadores, el diseño y la mejora de los procesos, así como la definición y comprobación del sistema mediante monitoreo, todo esto enfocándose en las actividades relacionadas con el marco de la empresa.

## **2.2 Marco Contextual**

Como se mencionó anteriormente, en el área de sistemas en un ámbito jurídico ha sido un factor creciente durante la última década en los 34 países miembros de la Organización de Estados Americanos (OEA), en los cuales México es un integrante, esto es debido a que las herramientas que proveen los SI en este entorno favorece

a la optimización y mejora de los servicios de impartición de justicia en materia legal. Lo anterior es un indicio de que el desarrollo de SI a la medida provee una herramienta útil para agilizar los procesos de justicia, lo que da lugar al e-Justice. Ahora bien, veamos casos de países que han implementado este tipo de aplicaciones.

En su trabajo “Diseño y desarrollo de un sistema de información judicial para centros judiciales integrados del órgano judicial de El Salvador: Implantación en el centro Judicial de Soyapango”, Wolfgang (2011) expone el caso de éxito de todo el proceso de desarrollo de una plataforma en formato digital que permite el acceso y procesamiento de información del ámbito judicial mediante el desarrollo de un nuevo modelo de gestión judicial, posibilitando la dedicación exclusiva de cada juez a su jurisdicción (juzgar y ejecutar lo juzgado), asignar secretarios que se encargaran de las oficinas de sustanciación de causas enfocadas en el análisis jurídico y finalmente adjuntar el apoyo de una serie de oficinas comunes especializadas en el levantamiento de decretos, autos, sentencias, actas, etc.. , cabe señalar que el modelo implementado para este sistema nunca se había implementado en ningún país de América latina, por lo que su construcción fue realizada desde cero. Este sistema permite integrar todas las funciones jurídicas y administrativas de un tribunal, manejar expedientes electrónicos con apoyo de una firma digital, es replicable y sobre todo permite la escalabilidad para ajustarse a cualquier tipo de materia o juzgado.

Otro caso similar sucedió en Colombia, donde (Galivis, González, & Hernán, 2011) realizaron un proyecto de implementación de desarrollo tecnológico para construir una plataforma web que permitiera administrar y divulgar información sobre la oferta de servicios de justicia en ese país, para ello utilizaron la metodología de desarrollo de programación extrema con lo que obtuvieron tres productos: un tesoro de oferta de justicia, un directorio de instituciones y finalmente una plataforma donde se visualizaban reportes sobre la oferta de justicia.

No obstante, en México sucede algo similar, como se ha mencionado anteriormente, actualmente se tienen implantado un producto de TI enfocado al servicio de justicia, como lo es en los estados de Guanajuato y Sonora, en donde en el primero se implementa una herramienta en la que se puede seguir el proceso de juicio en línea y obtener asistencia jurídica bajo el mismo medio digital (TCAGTO, 2017); y en el segundo, de igual forma implementan el lineamiento del juicio digital, asesoramiento en tiempo real, un boletín electrónico que provee retroalimentación a los usuarios de la plataforma, entre otros (TCASIN, 2017).

Con lo anteriormente expuesto, es posible decir que hasta el momento, no se encontró evidencia de que exista algún SI que sea igual al que se propone desarrollar, ya que la mayoría de los encontrados se enfocan en el proceso de impartición de justicia o en el permitir que los usuarios externos a los tribunales tengan acceso a la información general o a los servicios de justicia misma, más sin embargo, ninguno de ellos incorpora un medio que permita evaluar el rendimiento y productividad de la organización misma.

## **2.3 Marco Teórico**

Para el desarrollo de la propuesta en cuestión, que en vista del usuario final será la implementación de una nueva interface de visualización, se presentan los siguientes elementos teóricos que le dan soporte.

### 2.3.1 Indicadores KPI

El acrónimo KPI está formado por las iniciales de los términos Key Performance Indicator (cuya traducción al español es la de indicador clave de desempeño, o también conocido como indicadores de gestión), estos son métricas que nos ayudan a identificar el rendimiento de una determinada acción o estrategia, ya que indican el nivel de desempeño en base a los objetivos fijados por la organización (Espinoza, 2017).

Los KPI's se agrupan gráficamente en cuadros de mando para que los directivos puedan ser ágiles en la toma de decisiones, el cuadro de mando debe incluir los principales indicadores para la empresa lo que provee una forma visual de la información deseada para guiar el rumbo de la institución sobre un plan establecido; sin embargo, para poder definir un indicador necesita reunir ciertas características; sea medible, cuantificable, específico, temporal y sobre todo relevante (Gómez, 2017), por lo tanto, se puede indicar que para establecer tus propios indicadores es necesario responder a las siguientes preguntas básicas:

- ¿Qué queremos medir?
- ¿Por qué medimos este dato?
- ¿Realiza el seguimiento de los resultados de uno de nuestros objetivos?
- ¿Es un factor clave para la empresa?
- ¿Quién es el responsable de supervisar?
- ¿Con qué periodicidad conviene supervisar?

### 2.3.2 MySQL

MySQL es tanto SGDB como una base de datos (DB) que es considerado como el más popular de los gestores de código abierto. Con su probado rendimiento,



fiabilidad y facilidad de uso se ha convertido en la elección de sistemas de DB líder para aplicaciones basadas en una plataforma WEB y de aplicaciones del alto perfil (Oracle, 2017).

### 2.3.3 Bases de datos Relacionales

Las bases de datos relacionales (RDB siglas en inglés de Relational Data Base) hacen uso de tablas (a las que se les conoce como relaciones) que representan gráficamente con una estructura rectangular formada por filas y columnas; donde cada columna almacena información sobre una propiedad determinada de la tabla (lo que se conoce como atributo), de igual forma, en estas tablas, cada fila (también conocidas como tuplas) posee una ocurrencia o ejemplar de la instancia o relación representada por la tabla.

Las RDB ha tenido su mayor proliferación y aceptación por los desarrolladores de software, lo que las posiciona en primer lugar en el mercado por ser las más indicadas para las tradicionales aplicaciones de gestión. (Vela, Cavero, & Marcos, 2001)

### 2.3.4 PHP

PHP son las siglas de Pre Hypertext Processor , o bien pre-procesador de hipertexto, está enfocado principalmente a la programación de scripts del lado del servidor, es utilizado en proyectos de código abierto que se encuentran disponibles en la red como lo son Drupal, WordPress, Moodle, Joomla, entre otros (OpenSource, 2017), sin embargo, esto no sería posible sin la aportación de Zend Engines, la empresa que se encarga de fortalecer PHP ante su competencia, dedicando año tras año diversos recursos humanos para lograrlo (Zend Engine, 2015), debido a la creciente demanda en el mercado que conllevan los sitios WEB, de móviles y servicios en la nube, se realizó la actualización del motor o núcleo principal de PHP llevándolo a la versión 7.0 la cual conlleva notables mejoras en el rendimiento del mismo, ya que las aplicaciones desarrolladas bajo esta versión consiguen un rendimiento mayor a las que se realizaban en versiones anteriores.

### 2.3.5 Laravel

Laravel es uno de los frameworks de código abierto más fáciles de asimilar para PHP. Es simple, muy potente y tiene una interfaz elegante y divertida de usar. Fue creado en 2011 y tiene una gran influencia de frameworks como Ruby on Rails, Sinatra y ASP.NET MVC. El objetivo de Laravel es el de ser un framework que permita el uso de una sintaxis refinada y expresiva para crear código de forma sencilla, evitando el “código espagueti” y permitiendo multitud de funcionalidades. Aprovecha todo lo bueno de otros frameworks y utiliza las características de las últimas versiones de PHP.

La mayor parte de su estructura está formada por dependencias, especialmente de Symfony, lo que implica que el desarrollo de Laravel dependa también del desarrollo de sus dependencias (TAYLOR OTWELL, 2017).

### 2.3.6 JavaScript

JavaScript (Js) es uno de los lenguajes de programación más utilizados que son almacenados en las plataformas SourceForge y Github de carácter público para proyectos Open Source, en la siguiente tabla se puede observar el top Ten de los lenguajes orientados a proyectos Open Source alojados en las plataformas antes mencionadas.

Tabla 1 Top Ten de lenguajes Open Source en Github y SourceForge

SourceForge			Github		
Lenguaje	# proyectos	%	Lenguaje	# repositorios	%
Java	53,575	23	<b>JavaScript</b>	1,666,302	22
C++	43,189	19	Java	1,413,447	19
PHP	33,789	15	Ruby	888,679	12
C	31,837	14	Python	814,449	11
C#	17,053	7	PHP	697,898	9
Python	16,585	7	CSS	529,392	7
<b>JavaScript</b>	13,884	6	C++	439,423	6
Perl	10,012	4	HTML	432,546	6
Unix Shell	4,775	2	C	386,232	5
VB .NET	4,050	2	C#	356,856	5
<b>Total</b>	<b>228.749</b>	<b>100</b>	<b>Total</b>	<b>7,625224</b>	<b>100</b>

Lo anterior demuestra que Js es uno de los lenguajes de programación más utilizados en el mundo, si bien, este nació ante las necesidades de los desarrolladores de crear aplicaciones WEB más robustas y parecidas a aquellas de escritorio (M, 2010), sus raíces se remontan a 1993, cuando Netscape Comunicaciones incluía en su Navegador de internet una tecnología de scripting llamada LiveScript, la que permitía la interactividad del usuario con el navegador mismo, lo que lo hizo muy popular, en sí, a Js se considera un lenguaje expresivo al que se le es fácil de descargar funcionalidades en tiempo real de las aplicaciones en ambos lados, el del cliente y el del servidor (Mesbah, 2014).

### 2.3.7 Framework

En general, con el término framework, nos estamos refiriendo a una estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. En otras palabras, un framework se puede considerar como una aplicación genérica incompleta y configurable a la que podemos añadirle las últimas piezas para construir una aplicación concreta (Gutiérrez, 2008).

### 2.3.8 HTML

HTML son las siglas de Hiper Text Markup Lenguaje, o lenguaje de marcado de hipertexto, con el que se es posible crear un documento digital que es interpretado por los navegadores de internet como una página WEB. Para el desarrollo de una aplicación que provea sus servicios a través de internet, el diseño de la apariencia, es decir la interface gráfica que comprende los aspectos de la página web.

### 2.3.9 CSS

Las hojas de estilo aparecieron después que el lenguaje de etiquetas (SGML) alrededor del año 1970, desde su creación, se observó la necesidad de definir un mecanismo que permita aplicar de forma consiente y simple, diferentes estilos a los documentos electrónicos. El auge de los lenguajes de hojas de estilo se produjo con el boom de Internet y el avance y desarrollo del lenguaje HTML, la guerra de los navegadores y la falta de un estándar para la definición de estilos dificultaba la creación de los documentos para que compartieran la misma apariencia en los diferentes navegadores (Eguiluz, 2015).

Con este motivo específico, el organismo W3C (World Wide Web Consortium), encargado de crear estándares relacionados con la WEB, propuso crear un lenguaje de hojas de estilo que se ajuste específicamente para el lenguaje HTML. Por lo que entre finales de 1994 y 1995 Lie y Bos se unieron para definir un lenguaje que unificara lo mejor que ofrecía cada una de las propuestas obtenidas en años anteriores y lo denominaron CSS (cascading style sheets por sus siglas en inglés), con lo que en 1996 la W3C publicó la primera recomendación oficial conocida como CSS level1.

Sucesivamente, con el paso del tiempo se fueron incorporando mejoras al estándar establecido en 1996, surgiendo así en 1998 la segunda recomendación, el CSS level 2, cabe señalar que la versión utilizada por los navegadores actuales es la versión 2.1 la cual recibió actualizaciones en 2009,

### 2.3.10 Bootstrap

Bootstrap es uno de los frameworks de front-end más populares utilizado en proyectos de código abierto en el mundo, este fue originalmente creado por un diseñador y desarrollador en Twitter a mediados de 2010 por Mark Otto y Jacob Tornton. Antes de ser conocido como lo es actualmente, éste se presentaba como Twitter Blueprint. El que originalmente se publicó el viernes 19 de agosto de 2011 ha tenido más de veinte versiones, incluyendo dos re-escrituras principales en su código lo que representan las versiones 2 y 3, en las que se han añadido funcionalidades a todo el marco como una hoja de estilo opcional. Finalmente,

basándose en la tercera versión, se reescribe la biblioteca una vez más para que esta responda de forma predeterminada con un enfoque móvil (Bootstrap, 2015).

### 2.3.11 Talend Open Studio

Talend Open Studio es una herramienta robusta que provee utilidad para procesos ETL (Extract, Transform and Load por sus siglas en inglés). Este paquete de software desarrollado bajo el lenguaje de programación Java, permite conectar y transformar datos desde múltiples orígenes, desde ficheros hasta bases de datos en los diferentes gestores e incluso aquellas en la nube. Talend permite diseñar y desplegar con facilidad y rapidez el proceso ETL con ayuda de herramientas gráficas y la generación de código nativo (Talend, 2017).

### 2.3.12 Enterprise Architect Project

Este es un *software* de plataforma avanzada para modelado y diseño en tiempo real, potenciado por UML abarca completamente el ciclo de vida de desarrollo de software, incluye herramientas que provén una estructura para el modelado de negocio, diseño de software, ingeniería en sistemas, arquitectura de empresas, gestión de requisitos, testeo, entre otros. Esta herramienta completa permite integrar al equipo de trabajo y dar vida a la visión de trabajo compartido (Sparxsystems, 2017).

### 2.3.13 Balsamiq Muckups

Una de las mejores herramientas para crear prototipos, bocetos o wireframes es Balsamiq Muckups (ISDI, 2014). Cuando queremos crear un proyecto, lo normal es querer realizar sketch, bocetos o dibujos rápidos para tener una imagen visual de cómo será nuestra interface gráfica, lo que en un principio es un esbozo en papel, más adelante sirve para poder crear el proyecto lo más semejante al boceto, toda mejora se puede agregar, a fin de cuentas, la creación de la estructura de una interfaz es lo que se denomina Wireframing.

Este programa de escritorio programado en Flex y Air es instalable en plataformas Windows, Linux, Mac OS X e incluso cuenta con una versión WEB, cuenta con objetos prediseñados, funcionalidades de dibujo libre, en fin, una herramienta capaz de generar bocetos y exportarlos en formato PNG, PDF y más, lo que facilita la creación de esquemas (Santamaría, 2013).

### 2.3.14 Re ingeniería de Software

La reingeniería de sistemas heredados tiene por finalidad el reestructurar o transformar viejos sistemas en aplicaciones más fáciles de mantener, con entornos más agradables e integrados en nuevas plataformas de Hardware/Software. (Álvarez, Mateos, & Moreno, 2005).

La reingeniería debe ser entendida como un proceso mediante el cual se mejora un software existente haciendo uso de técnicas de ingeniería inversa y reestructuración de código. Para llevar a cabo la reingeniería del Software se puede realizar a través del modelo Cíclico. En algunas ocasiones, estas actividades se producen de forma secuencial y lineal, pero esto no siempre es así. Por ejemplo, puede ser que la ingeniería inversa (la comprensión del funcionamiento interno de un programa) tenga que producirse antes de que pueda comenzar la reestructuración de documentos (Isoftwareunesum, 2011).

La Reingeniería del software se puede definir como: “modificación de un producto software, o de ciertos componentes, usando para el análisis del sistema existente técnicas de Ingeniería Inversa y, para la etapa de reconstrucción, herramientas de Ingeniería Directa, de tal manera que se oriente este cambio hacia mayores niveles de facilidad en cuanto a mantenimiento, reutilización, comprensión o evaluación.”

Pasos de la reingeniería:

- **Análisis de Inventario.** - Todas las organizaciones de software deberían tener un inventario de todas sus aplicaciones. El inventario tal vez no sea más que un modelo en una hoja de cálculo que contenga información que proporcione una descripción detallada (tamaño, edad, importancia para el negocio) de las aplicaciones activas. Es importante señalar que el inventario deberá visitarse con regularidad, el estado de las aplicaciones puede cambiar en función del tiempo y, como resultado, cambiarán las prioridades para la reingeniería.
- **Reestructuración de código.** - Algunos sistemas heredados tienen una arquitectura de programa relativamente sólida, pero los módulos individuales han sido codificados de una forma que hace difícil comprenderlos, comprobarlos y mantenerlos. En estos casos, se puede reestructurar el código ubicado dentro de los módulos sospechosos. Para llevar a cabo esta actividad, se analiza el código fuente mediante una herramienta de reestructuración, se indican las violaciones de las estructuras de programación estructurada, y entonces se reestructura el código (esto se puede hacer automáticamente). El código reestructurado resultante se revisa y se comprueba para asegurar que no se hayan introducido anomalías. Se actualiza la documentación interna del código.
- **Reestructuración de datos.** - Un programa que posea una estructura de datos débil será difícil de adaptar y de mejorar. De hecho, para muchas aplicaciones, la arquitectura de datos tiene más que ver con la viabilidad a largo plazo del programa que el propio código fuente. A diferencia de la reestructuración de código, que se produce en un nivel relativamente bajo de abstracción, la estructuración de datos es una actividad de reingeniería a gran escala. En la mayoría de los casos, la reestructuración de datos comienza por una actividad de ingeniería inversa. La arquitectura de datos actual se analiza minuciosamente y se definen los modelos de datos necesarios. Se identifican los objetos de datos y atributos y, a continuación, se revisan las estructuras de datos a efectos de calidad.

- **Ingeniería directa.** - En un mundo ideal, las aplicaciones se reconstruyen utilizando un “motor de reingeniería” automatizado. En el motor se insertaría el programa viejo, que lo analizaría, reestructuraría y después regeneraría la forma de exhibir los mejores aspectos de la calidad del software. Después de un espacio de tiempo corto, es probable que llegue a aparecer este “motor”, pero los fabricantes de CASE han presentado herramientas que proporcionan un subconjunto limitado de estas capacidades y que se enfrentan con dominios de aplicaciones específicas. Lo que es más importante, estas herramientas de reingeniería cada vez son más sofisticadas.

### 2.3.15 Modelado de requisitos

El modelado de requisitos tiene como objetivo delimitar el sistema y capturar todas aquellas funcionalidades que debe ofrecer desde la perspectiva del usuario. Este modelo funciona en ciertos aspectos como un contrato entre el desarrollador y el cliente, dado que proyecta lo que el cliente desea según la percepción del desarrollador (IEEE, 2004).

Acorde a lo establecido en el SWEBOK (Software Engineering Body of Knowledge por sus siglas en inglés, los requisitos funcionales son declaraciones de los servicios o funcionalidades que prestará el sistema, en la forma en la que reaccionará a determinados insumos, en cambio, cuando se habla de los requisitos no funcionales, se habla de aquellos que no se refieren a las funciones específicas suministradas por el sistema, sino, de las propiedades del sistema (rendimiento, seguridad, disponibilidad, etc.), es decir no hablan de lo que se hace en el sistema, sino de cómo se hacen.

### 2.3.16 Modelo de clases

Pressman (2010) menciona que el modelado basado en clases representa los objetos que manipulará el sistema, es decir las operaciones) que se aplicarán a los objetos para efectuar la manipulación, las relaciones entre los objetos y las colaboraciones que tienen lugar entre las clases dirigidas entre ellos.

### 2.3.17 Diseño de interfaces

El diseño de interfaz de usuario es el resultado de definir la forma, función, usabilidad, ergonomía, imagen de marca y otros aspectos que afectan a la apariencia externa de las interfaces de usuario en sistemas de todo tipo (computadoras de uso general, sistemas de control, dispositivos de comunicación móviles, software de sistemas, software de aplicaciones, sitios web, etc. El objetivo final del diseño de la interfaz de usuario es hacer que la interacción entre el usuario y el sistema del que es interfaz sea tan simple y eficiente como sea posible, en términos de cumplimiento de los objetivos del usuario. Sigue por ello una filosofía de diseño centrado en el usuario.

La interfaz de usuario como tal es la parte visible de la aplicación front-end (traducible al español como interfaz) con la que el usuario interactúa a fin de usar el software. El usuario puede manipular y controlar el software, así como el hardware por medio de las interfaces de usuario, es decir, es parte del software y está diseñada de tal manera que se espera proveer al usuario con un conocimiento sobre la percepción del software. La Interfaz de usuario también aporta una plataforma fundamental para la interacción entre los humanos y el ordenador.

La UI puede ser gráfica, en forma de texto, audiovisual, dependiendo del hardware subyacente y su combinación con el software. La UI puede ser un hardware, un software o una combinación de ambos.

El software suele ser más popular cuando su UI es:

- Atractiva
- Fácil de usar
- De respuesta rápida
- Clara de comprender
- Coherente en toda la pantalla de interfaz

La UI se divide en dos categorías:

- Interfaz de línea de comandos
- Interfaz gráfica de usuario (GUI)

La interface de línea de comandos es aquella que nos permite comunicarnos con el programa con una serie de instrucciones (comandos) escritos de forma tal que el programa interpreta estos como una serie de órdenes que debe ejecutar, a diferencia de la anterior, la interface gráfica de usuario o GUI se enfoca en la presentación e interacción del software con el usuario de una forma más simplificada y gratificante para el mismo.

Durante el proceso de modelado de una interface de usuario se utilizan todas las herramientas que el diseñador conozca, para que con ayuda de estas intente plasmar una idea general con bosquejos la solución al problema de los usuarios.

Shneiderman & Plaisant (2005) establecen normas de oro para el diseño de las interfaces las cuales instituyen lo siguiente:

- **Esfuerzo de coherencia** - Las secuencias coherentes de acciones son requeridas en situaciones similares. Se debe usar terminología idéntica en menús, prompts, y pantallas de ayuda. Los comandos coherentes deben ser empleados en de punta a cabo.

- **Posibilitar el uso de atajos a usuarios frecuentes** - El deseo del usuario por reducir el número de interacciones se incrementa con la frecuencia de uso. Las abreviaciones, funciones clave, comandos escondidos, y facilidades macro son muy útiles para un usuario experto.
- **Ofrecer retroalimentación informativa** - Para cada acción del operador, debe haber alguna retroalimentación de sistema. Para acciones menores y frecuentes, la respuesta debe ser modesta, mientras que, para acciones no frecuentes y mayores, la respuesta debe ser más substancial.
- **Diseñar diálogo para permitir el cierre** - Las secuencias de acciones se deben organizar en grupos con un inicio, núcleo, y final. La retroalimentación informativa cuando se completa un grupo de acciones da a los operadores una gran satisfacción de logro, un sentimiento de alivio, la señal de abandonar planes y opciones de contingencia de sus mentes, y esto indica que el camino a seguir es claro y se pueden preparar para el siguiente grupo de acciones.
- **Ofrecer tratamientos de error simples** - Diseñe tanto como sea posible el sistema para que el usuario no haga errores graves. Si se comete un error, el sistema debe poder detectarlo y ofrecer mecanismos simples y comprensibles para tratar el error.
- **Permitir deshacer acciones fácilmente** - Esta característica reduce la ansiedad, ya que el usuario sabe que los errores se pueden deshacer. Esto motiva la exploración de opciones nuevas y no familiares. Las unidades de reversibilidad pueden ser una sola acción, una entrada de datos, o un grupo de acciones.
- **Permitir la ubicación interna de control** - Los operadores con experiencia desean de todo corazón poder sentir que tienen el control del sistema y que el sistema responde a sus acciones. Diseñe el sistema de modo que los usuarios puedan iniciar acciones en vez de ser los que responden.
- **Reducir la carga de memoria a corto plazo** - La limitación humana para procesar información con una memoria a corto plazo hace necesario mantener la pantalla de forma simple, así como mostrar las páginas de forma coherente, reducir la frecuencia de 'window-motion' y asignar un



tiempo de formación suficiente para los códigos, mnemónicos, y las secuencias de acciones.

### 2.3.18 Pruebas de Software

Las pruebas de software son una dinámica que permite la verificación del comportamiento de un programa en un conjunto finito de casos de prueba, los cuales deben ser debidamente seleccionados de por lo general infinitas ejecuciones de dominio, contra el comportamiento esperado, es decir son una serie de actividades que se realizan con el propósito de encontrar los posibles fallos de implementación, calidad o usabilidad de un programa u ordenador; comprobando el comportamiento del mismo en el camino.

Existen múltiples tipos de pruebas que pueden aplicarse a un producto de software, las que regularmente se realizan durante la construcción del mismo. Comúnmente, los tipos de pruebas más utilizados son aquellas que nos permitan realizar la comprobación de módulos individuales o el conjunto de dichos módulos, estas son las pruebas unitarias y las de integración.

Las pruebas unitarias son una forma en la que se comprueba nuestro código a nivel individual para asegurarnos que funcionen correctamente por separado, lo que nos proporciona un plus de estabilidad al código porque se puede asegurar que el fragmento no tiene ningún fallo; por otro lado, realizar pruebas de integración es el término que se aplica a la unificación de múltiples pruebas que complementen un módulo general o una función específica que se desea evaluar (Abad, 2005).

### 2.3.19 PHPUnit

Aunque existen muchas herramientas de software que nos permitan realizar dichas pruebas, en el lenguaje de programación PHP la más comúnmente utilizada es PHPUnit (Bergmann, 2019), dicha herramienta se encuentra disponible para su uso de forma libre, es decir es una herramienta OpenSource, dicho factor es un punto a favor sobre el framework de programación Laravel el cual incluye dentro de los repositorios que se instalan junto a él la herramienta descrita anteriormente en su versión 4.8.16.

### 2.3.20 Filezilla

FileZilla es el nombre de un programa, de un software de vital importancia para las compañías que necesitan alojar y compartir archivos en internet. Se trata de una herramienta pensada para aprovechar los protocolos FTP, lo que permite la

descarga y el envío de archivos a gran velocidad a través de un servidor dedicado o compartido.

Es un software de código abierto, lo que implica que su descarga es totalmente gratuita y que incluso puede modificarse para ajustarse a las necesidades de cada consumidor, siempre y cuando no se haga con fines lucrativos. Por su nombre, son muchos los que lo relacionan de un modo u otro con Mozilla, pero ambos tienen procedencias completamente diferentes.

Con el paso de los años, y con numerosas actualizaciones, este cliente ha implementado la compatibilidad con protocolos más modernos y seguros como SFTP, lo que permite el cifrado de toda la comunicación para evitar la intervención de los datos y su monitorización por parte de terceros. Además de eso, ofrece la posibilidad de conectar con servidores remotos usando VPN.

Una herramienta completa y gratuita que se usa en muchísimas empresas debido a su versatilidad y utilidad. FileZilla es una plataforma que nació con el mero fin de facilitar el almacenamiento de ficheros para su posterior uso en servidores a través del protocolo FTP, pero con el tiempo se ha convertido en un software casi imprescindible para realizar numerosas gestiones online (NeoAttack, 2018).

### 2.3.21 Dominios y Subdominios

Simplificándolo mucho, el dominio es el nombre único y exclusivo que se le asigna a una página web en Internet. Sería algo así como el equivalente de la matrícula de un automóvil, pero aplicado a una página web.

Gracias al dominio que se vincula a la página web, es posible que aquellos usuarios que busquen en Internet la página por el nombre de la web la encontrarán fácilmente y podrán acceder a tu contenido. según Rubén (2014) el dominio se utiliza para “humanizar” la Red y hacer la navegación más sencilla y accesible para las personas.

Suponemos que te resultará más sencillo recordar, por ejemplo, el dominio google.com que los doce dígitos que puede contener una dirección IP multiplicado por las decenas de portales y páginas web que visitas a diario.

Obviamente, el funcionamiento a nivel técnico es algo más complejo que eso, ya que lo que realmente se encuentra tras un nombre de dominio es una dirección a un servidor DNS (de las siglas en inglés de Sistema de Nombres de Dominio), que a su vez enlaza con la dirección IP del ordenador en el que están alojados los archivos que forman tu página web, creando así un nexo de unión entre el nombre de dominio que los usuarios escriben en su navegador para encontrar una página web.

De la misma forma, un subdominio es el método por el cual realizamos subdivisiones dentro de nuestro sitio web, las cuales pueden representar acciones completamente diferentes

Todo lo visto anteriormente representa las tecnologías disponibles además del fundamento teórico utilizado en el desarrollo de este proyecto, en estos se pueden localizar herramientas y métodos que son aplicados en la ingeniería de sistemas de información para el desarrollo de un producto de *software*.

## **CAPÍTULO III MÉTODOS EMPLEADOS**

En el presente capítulo se presenta la metodología que fue empleada para el desarrollo de la implementación de un módulo de indicadores de productividad en el sistema SiSeDe. El proceso Unificado Ágil (PUA) es una metodología para desarrollar una propuesta de solución como la que se presentó en el capítulo I en donde se destaca el problema del contexto presente en el Tribunal de Justicia Administrativa del Estado de Colima (TJA).

### **3.1 Metodología**

Para el desarrollo del nuevo módulo que mida la productividad se plantea utilizar la metodología del Proceso Unificado Ágil (PUA), la cual es una versión simplificada de la metodología de Proceso Racional Unificado, en donde se describe un enfoque simple y fácil de entender para el desarrollo de software de aplicaciones de negocio utilizando técnicas y conceptos ágiles, las cuales incluyen el desarrollo guiado por pruebas, el modelado ágil, la gestión de cambio ágil y la refactorización de base de datos para mejorar la productividad del producto informático (Ambler, 2014).

Acorde a lo establecido por Ambler (2014), la metodología del Proceso Unificado Ágil consta de cuatro fases que el proyecto atraviesa de forma secuencial:

1. **Iniciación:** el objetivo de esta fase es identificar el alcance inicial del proyecto, una arquitectura potencial para el sistema y obtener, si procede, financiación para el proyecto y la aceptación por parte de los promotores del sistema.
2. **Elaboración:** mediante esta fase se pretende identificar y validar la arquitectura del sistema.
3. **Construcción:** el objetivo de esta fase consiste en construir software desde un punto de vista incremental basado en las prioridades de los participantes.
4. **Transición:** en esta fase se valida y despliega el sistema en el entorno de producción.

A lo largo de las cuatro fases, se desarrollan actividades relativas a siete disciplinas de manera iterativa:

- Modelado: su objeto es entender la lógica de negocio de la aplicación, el dominio del problema del proyecto e identificar una solución viable para el dominio del problema.
- Implementación: transformar los modelos en código ejecutable y realizar pruebas básicas, en particular pruebas unitarias.
- Pruebas: realizar una evaluación de los objetivos para asegurar la calidad. Esto incluye encontrar defectos, validar que el sistema funciona como fue diseñado y verificar que los requisitos se cumplen.
- Despliegue: planear la entrega del sistema y ejecutar el plan para hacer que el sistema quede disponible para los usuarios finales.
- Gestión de la configuración: gestionar el acceso a los artefactos del proyecto. Esto incluye, además de la traza de versiones de los artefactos, el control de cambios y la gestión de los mismos.
- Gestión del proyecto: dirige las actividades que tienen lugar dentro del proyecto, incluyendo gestión de riesgos, dirección del personal y coordinación.
- Entorno: apoyar el resto del esfuerzo asegurando que los procesos, métodos y herramientas están disponibles para el equipo cuando los necesitan.

Lo anterior, representa el ciclo de vida del desarrollo de software según la visualización del proceso unificado ágil, este se puede representar como se observa en la Figura 4.

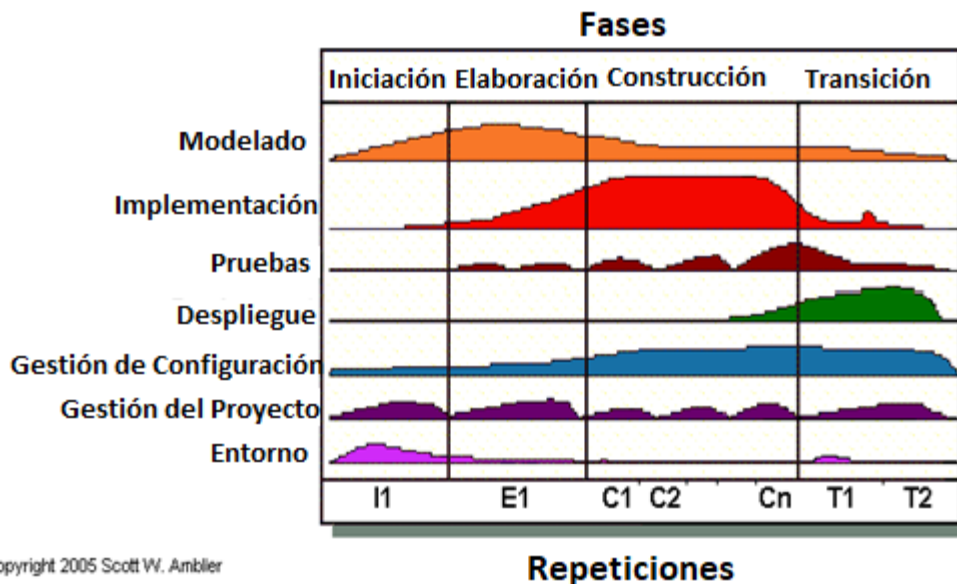


Figura 4 Ciclo de vida del software acorde a PUA

Además del uso de la metodología es necesario realizar el proceso de actualización del software anterior proveído por Fernández & Moreno (2017) el cual refleja la estructura del proceso de seguimiento de demanda que aplican en el TJA, es

necesario implementar el proceso de reingeniería de software, apoyado con los métodos incluidos en ésta como son *la ingeniería inversa, la reestructuración de documentos y la implementación de mejoras* lo que requirió del uso de herramientas que permitan realizar la reestructuración de datos, de interfaces y de procesos.

Con base en la forma en la se programan y organizan los procedimientos con ayuda de la metodología PUA, la cual establece iteraciones, que a su vez se definieron y distribuyeron con ayuda y colaboración de los interesados, obteniendo como resultado nueve iteraciones las cuales se distribuyeron entre las fases de elaboración, construcción y transición indicadas en la metodología.

Durante la fase de iniciación se definió con ayuda de los interesados los cambios y mejoras necesarias para el proyecto, lo que estipulo la razón de ser y su alcance; con lo que se definieron las iteraciones basadas a un periodo de una sesión cada 3 semanas, fecha en la cual se realizaría la entrega de los módulos modificados a los interesados.

La fase de elaboración se realizó en 3 iteraciones, en las cuales se estipulo el plan de mejora del proyecto, así como las modificaciones necesarias a la arquitectura del sistema, resultando de esta fase los diagramas de proceso de negocio, reglas del negocio, de clases, de datos, de interfaces, y de despliegue.

Durante la fase de construcción, se realizó la reestructuración y normalización de la base de datos implementada en el sistema, así como la reestructuración del código central de la aplicación dado el cambio de framework de programación y la mejora de las interfaces de usuario, esto con ayuda de las herramientas ofrecidas por EAP, Laravel y MariaDB, los cuales devolvieron como evidencias los scripts de creación de base de datos, los nuevos casos de uso y la estructura base de la aplicación.

Finalmente, la fase de transición se desplegó para el proyecto en un entorno real, para lo cual se realizaron pruebas integrales del sistema tanto en forma local como en la plataforma en internet, con lo que si capacitó a los usuarios en el uso y manejo de la nueva versión de la plataforma la cual se encuentra liberada y establecida en su versión alfa, cabe señalar que esta fase se realizó en 7 iteraciones las cuales establecían la entrega de módulos individuales a cada uno de los actores, finalizando con una capacitación general la cual involucraba la operación de todos los interesados completando la fase, obteniendo como resultados el dominio o sitio web habilitado para el seguimiento de demandas los cuales incluyen las funciones tanto de actores internos como externos al TJA.

### **3.2 Materiales y equipo**

Para poder realizar la reingeniería del motor de base de datos, se utilizo la herramienta Talend Open Studio, la cual es una herramienta de código abierto desarrollada bajo el lenguaje de programación Java®, que permite realizar los

procesos ETL lo que establece un medio para filtrar, organizar y reestructurar un modelo de base de datos a otro almacén.

Finalmente, para el desarrollo del nuevo módulo que se anexó al sistema SiSeDe se hizo uso del Sistema Gestor de Base de datos MariaDB, JavaScript (Kemp, 2009), Apache, PHP 5.6 con ayuda del Framework Laravel v 5.4 (TAYLOR OTWELL, 2017),. Es importante mencionar que las herramientas tecnológicas utilizadas fueron seleccionadas por ser de código abierto la cual no se necesita licencia comercial para su utilización además se pueden emplear de forma sencilla, disminuyendo el despliegue complicado de comandos (Samra, 2015), lo que para aquellas instituciones con presupuesto reducido y que desean aumentar su productividad ,resulta conveniente (Prokofyeva & Boltunova, 2017).

Además comparando ésta tecnología propuesta con las utilizadas en los tribunales ya referenciados podemos mencionar que JSP(Java Server Pages) en comparación con PHP es un lenguaje robusto, superando a otros en desempeño; sin embargo, existen estudios realizados por los autores (Trent, Tatsubori, Suzumura, Tozawa, & Onodera, 2008) en su artículo Performance comparison of PHP and JSP as server-side scripting languages que afirman “Las extensiones PHP (escritas en C) y las bibliotecas de clases Java, una vez compiladas muestran menos variación en rendimiento.”, por lo que lo hace un lenguaje ideal para el desarrollo de SI web; además de contar con un Framework que nos proporciona una arquitectura de software Modelo Vista Controlador y CSS (Cascading Style Sheets) que nos favorecen por facilitar la reutilización de código; con la finalidad de implementarlas mejoras al en el SiSeDe en menor tiempo, como lo afirman los autores (Sierra, Acosta, Ariza, & Salas, 2013) en su investigación “Estudio y análisis de los Framework en PHP basados en el modelo vista controlador para el desarrollo de software orientado a la web”.



## ***CAPÍTULO IV DESARROLLO DEL SISTEMA***

En este capítulo se presentan los artefactos correspondientes a la aplicación de la ingeniería de software, utilizando el Proceso Unificado Ágil, que sirven como guía y referencia técnica para el desarrollo del proyecto.

### ***4.1 Modelado de Casos de Uso***

La recopilación de requerimientos incluida en el planteamiento y análisis del problema permitió representar los escenarios del proceso de administración de justicia mediante casos de uso, que se atendieron en iteraciones; estos permitieron identificar y visualizar a los actores que interactúan en el sistema y delimitar los alcances del mismo. En la Figura 5, se observa *<registrar demanda>*, éste se considera el caso de uso principal del sistema, porque representa la misión del organismo y da origen a la justificación del proyecto; además, *<registrar demanda>* gestiona de forma global la interacción de cada uno de los actores del sistema, conforme al ciclo de seguimiento de la demanda en atención a las normativas definidas en el TJA y la Ley de Justicia Administrativa del estado de Colima. Derivado del proceso de registrar demandas, se monitorean los historiales de movimientos de atención de cada demanda con la finalidad de *<generar indicadores>* a partir de la evaluación de los tiempos de cada actividad; se calcula el tiempo usado y la eficiencia de atención de la demanda en forma general y específica.



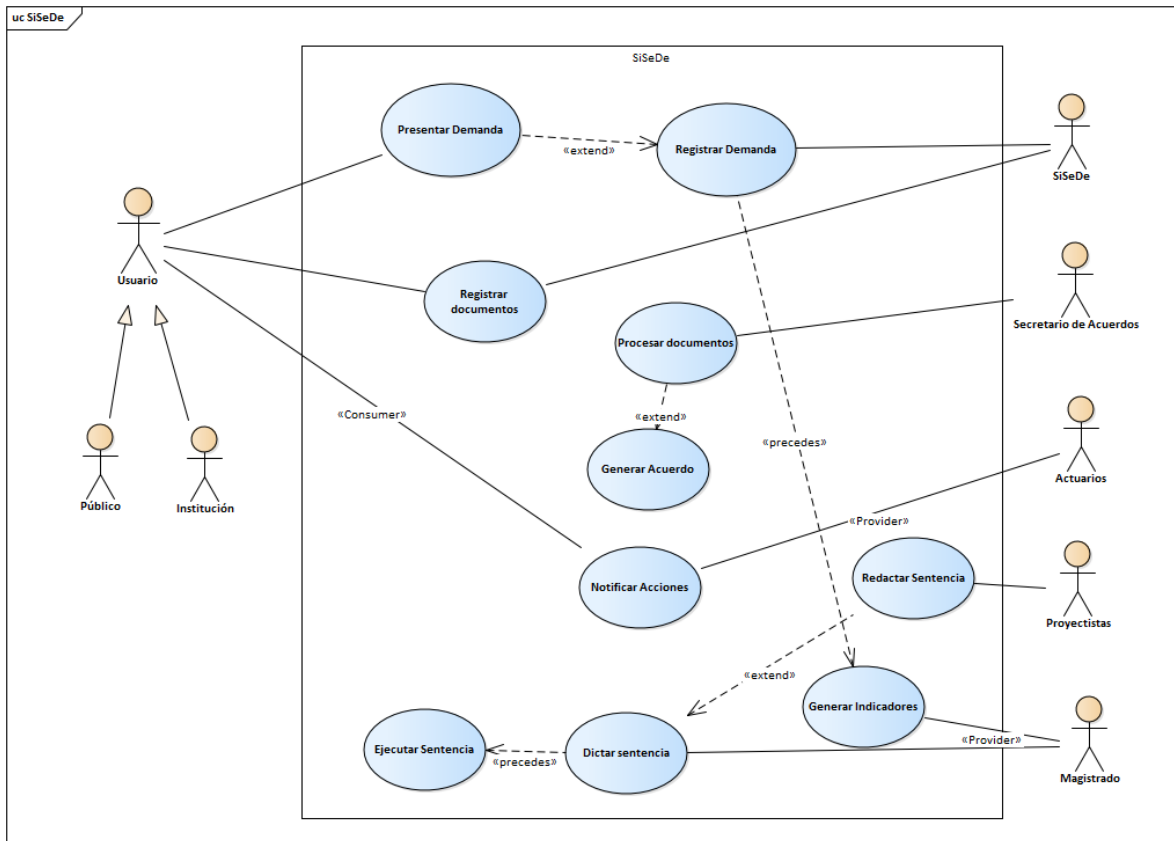


Figura 5 Caso de Uso general del SiSeDe

### 4.3 Modelo de Requisitos

Atendiendo los casos de uso identificados en esta sección se presenta el modelado de requisitos funcionales y no funcionales, que corresponden al proyecto en desarrollo.

#### 4.3.1 Requerimientos funcionales

La Figura 6 muestra los requisitos funcionales del proyecto SiSeDe/ Indicadores de productividad, destacando el <RF19 semáforos> y el <RF20 estadística>; estos requisitos representan las funcionalidades que se agregan al Sistema como herramienta para la medición de la productividad en el seguimiento de las demandas.



Figura 6 Requerimientos funcionales del sistema

### 4.3.2 Requerimientos No Funcionales

La Figura 7 considera algunos de los requisitos no funcionales que se detectaron en el desarrollo de la aplicación, destaca la operación del sistema desde cualquier navegador y dispositivo computacional.

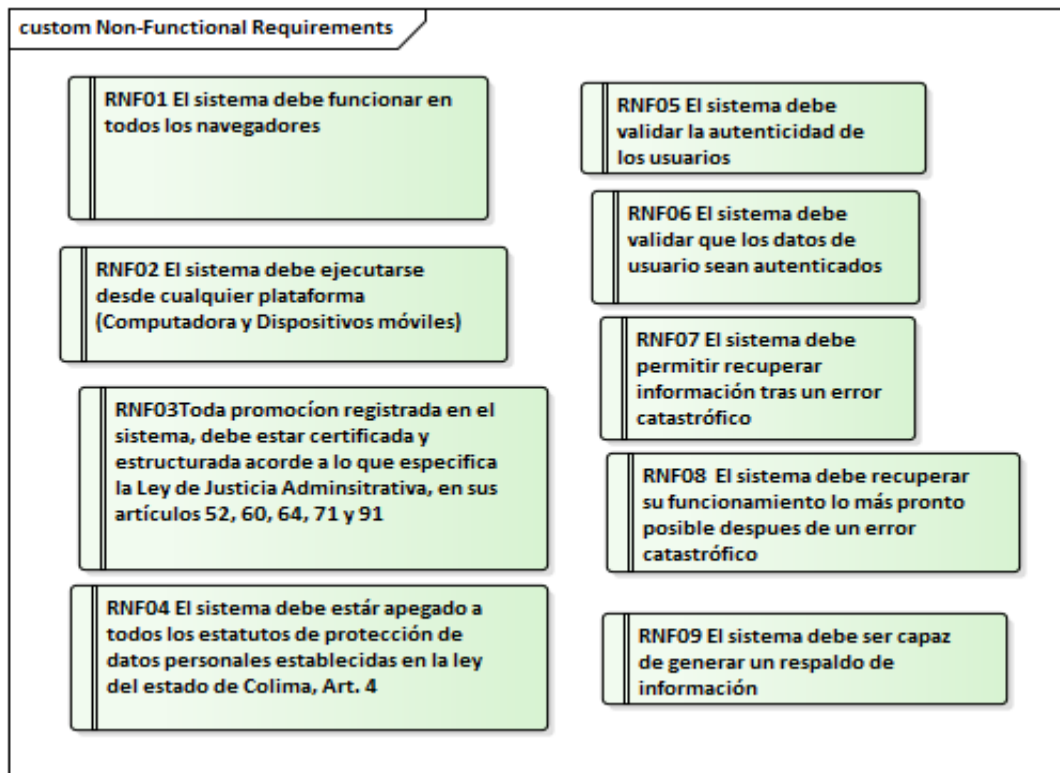


Figura 7 Requerimientos No funcionales del sistema

### 4.4 Modelo de Clases

En la Figura 8 se presenta el diagrama de clases que representa todos los objetos y sus relaciones que interactúan en el sistema; la clase <seguimiento> contiene los atributos necesarios para llevar el registro histórico de los movimientos registrados que derivan en elementos indispensables para la aplicación de los indicadores de medición de tiempos de procesos, se asumen como datos de entrada para la medición de productividad (uso del semáforo); mientras que, la clase <expediente> permite recolectar la información general del mismo, permitiendo realizar con esta la evaluación formal y estadística de expedientes atendidos.



almacén de información que utiliza el SiSeDe/Indicadores de productividad; la relación que se establece entre las tablas <seguimiento> y <expediente> permiten obtener la información que satisface al caso de uso de <generar indicadores> presentado anteriormente, elemento que complementa la funcionalidad extendida del proyecto, la cual se enfoca en la medición de productividad.

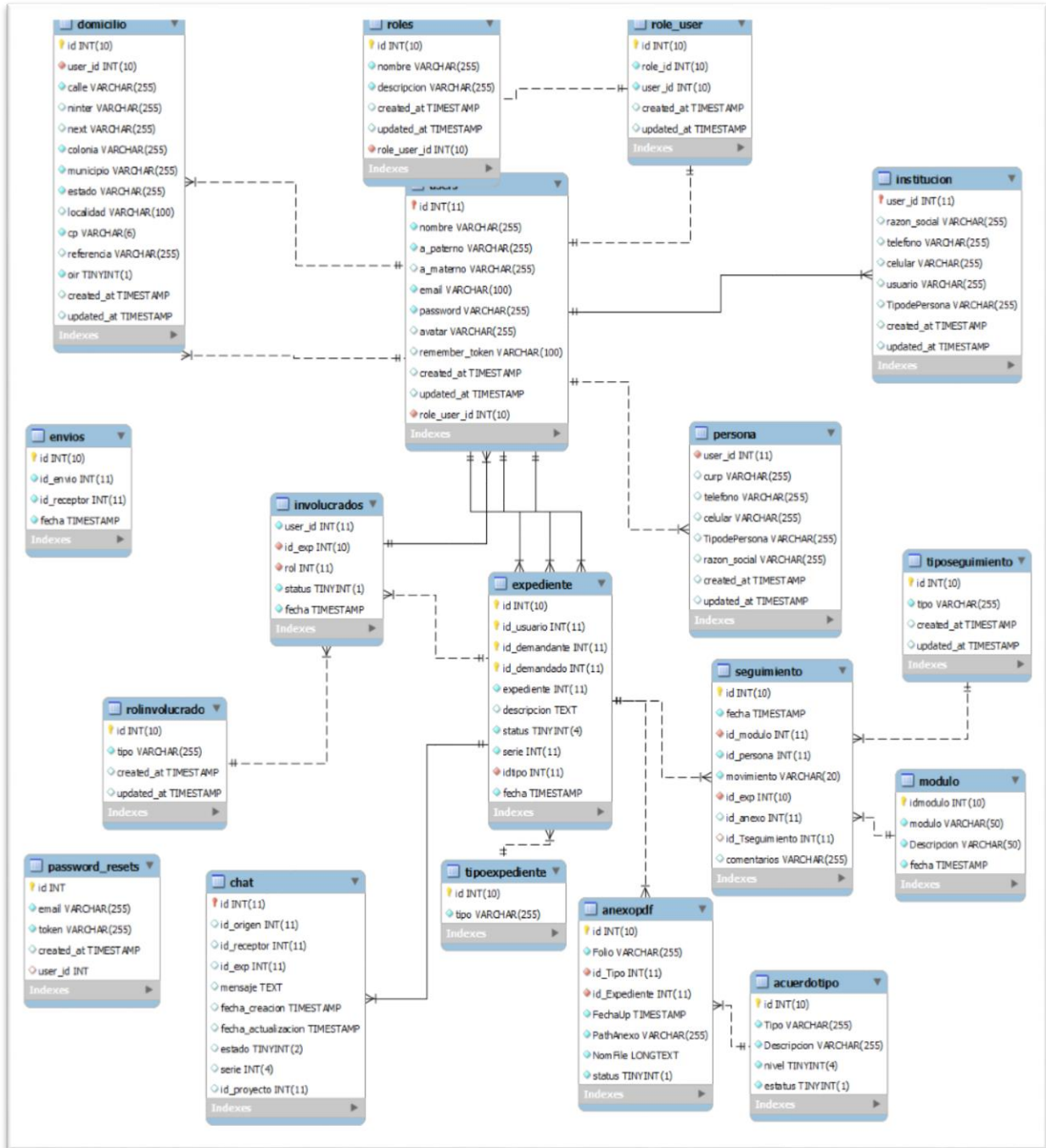


Figura 9 Diagrama de base de datos del proyecto SiSeDe

#### 4.5.2 Diccionarios de datos

El diccionario de datos que se presenta en la Tabla 2 y Tabla 3 es el complemento del modelo de datos, en él se especifican los elementos (propiedades) que debe contener cada una de las tablas, así como una descripción sencilla y concisa de la estructura que debe integrar a la misma, es decir, se especifica el tipo de dato, motivo del uso de ese dato, el formato, restricciones, llaves primarias, foráneas o candidatas y finalmente las referencias a las mismas.

En las Tablas 2 y 3, se presenta el diccionario de datos de las entidades <expediente> y <seguimiento> que precisan las características de los datos que se almacenan; destacando en la Tabla 2 los campos *expediente*, *fecha* y *estatus*; de la Tabla 3 los campos *id\_tiposeg*, *fecha* e *id\_exp*, pues son los elementos que permiten realizar el cálculo de la productividad y el uso del semáforo.

Tabla 2 Diccionario de datos de la tabla expediente

Tabla expediente									
Contiene las los datos relacionados al expediente tales como los identificadores de aquellos involucrados en el, la descripción general del caso, el estatus del expediente, etc.									
Nombre de la Propiedad	Descripción	Tipo de dato	Formato	Valor nulo (S/N)	Valor Default	Restricción	Indexar (S/N)	PK, CK o FK	Clave externa (S/N) y referencia
id	NI de registro de la tabla	int		N		id>0 AI	Si	PK	
id_usuario	Especifica el usuario que registro la demanda	int		N				FK	User(id)
id_demandante	Especifica el usuario demandante	int		N				FK	Users(id)
id_demandado	Especifica el usuario demandado	int		N				FK	Users(id)
expediente	Especifica el folio del expediente	int		N					
descripcion	Establece la descripción general del expediente	text		S					
status	Establece el estado de la demanda, e nivel donde se encuentra	tinyint		N		0=registrado 1=validado por oficial ... 15=archivado			
serie	Establece el año de registro	int		N		Año del registro			
idtipo	Establece el tipo de acción de la demanda	int		N				FK	Tipoexpediente (id)
fecha	Establece la fecha de recepción de expediente	timestamp	Mm-dd-aa HH:mm:ss	N					

Tabla 3 Diccionario de datos de la tabla de seguimiento

Tabla Seguimiento									
Contiene las los datos relacionados todo movimiento registrado para cada uno de los expedientes.									
Nombre de la Propiedad	Descripción	Tipo de dato	Formato	Valor nulo (S/N)	Valor Default	Restricción	Indexar (S/N)	PK, CK o FK	Clave externa (S/N) y referencia
id	Nº de registro de la tabla	int		N		id>0 AI	Si	PK	
Fecha	Establece la fecha y hora del movimiento	Timestap	Mm-dd-aa HH:mm:ss	N					
Id_modulo	Establece el módulo de origen del movimiento	int		N				FK	Modulo(id)
Id_persona	Establece el identificador de la persona que hace el movimiento	int		N				FK	User(id)
Movimiento	Establece una descripción del movimiento	Varchar(20)		N		Entrada o Salida		FK	Expediente(id)
Id_exp	Establece el expediente de referencia	int		N					
Id_anexo	Establece el identificador del documento	int		N				FK	Anexopdf(id)
Id_tiposeg	Establece que tipo de seguimiento se aplica	int		N				FK	Tiposeguimiento(id)
Comentarios	Establece un campo de comentarios	text		S					

## 4.6 Diseño de Interfaces

Con ayuda de los puntos que establecen los requerimientos y la perspectiva de los usuarios, se hizo uso de la herramienta Balsamiq Mockups, la cual permitió realizar un esbozo rápido de las ideas que se planteaban para el diseño de las nuevas interfaces de usuario, las cuales se apoyan en un diseño minimalista.

### 4.6.1 Modelado

Como se puede observar en la Figura 10, los usuarios visualizarán los menús de navegación dentro del sistema usando un escritorio con una sección de menú en el lado izquierdo y el área de trabajo en la parte derecha.





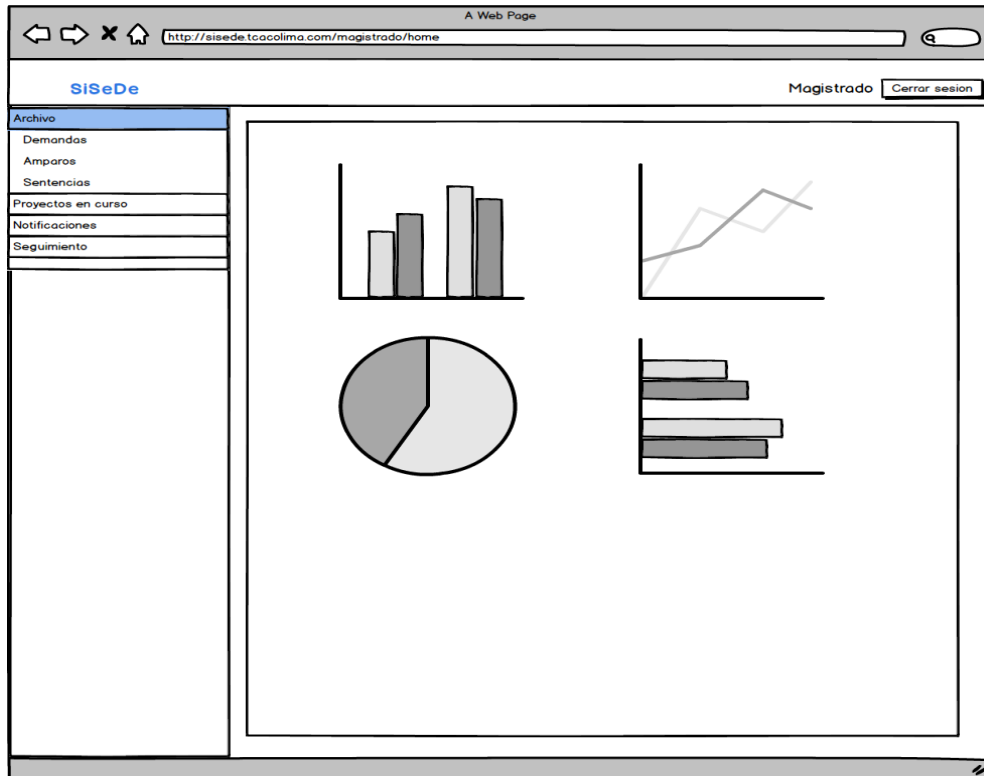


Figura 11 Bosquejo de la representación de Indicadores

#### 4.6.2 Creación de las nuevas interfaces

Para la creación de las nuevas interfaces, dado a la similitud de funcionalidades que ofrecen las plantillas prediseñadas que existen en el mercado, se procedió a utilizar una que maneje una estructura similar a la que surgió de los bosquejos, esto fue la plantilla Admin LTE en su versión 2.4, desarrollada por Abdullah Almsaeed ésta presenta una interface minimalista, hace uso de formas semejantes a las que utiliza el sistema operativo de Windows en su explorador de archivos, incluye un menú de accesos directos en su panel izquierdo dejando la sección central y derecha para el contenido de la aplicación, muchos de los requisitos como la interface responsiva y manejo accesible a los contenidos son proporcionados por ésta plantilla, el diseño general de lo que nos proporciona se puede observar en la Figura 12.

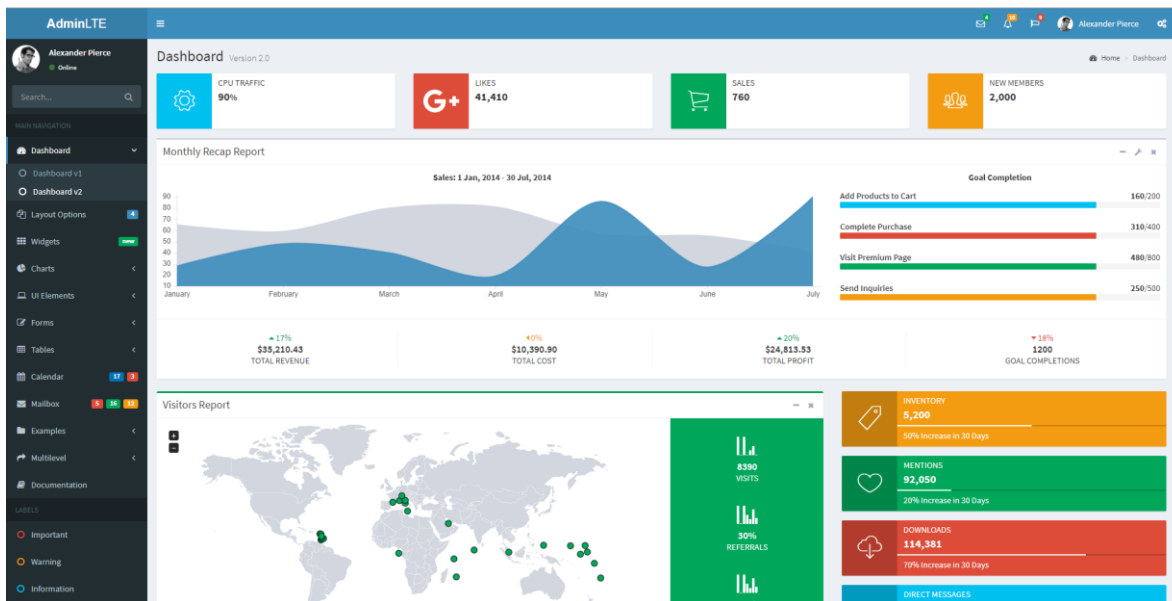


Figura 12 Plantilla de diseño AdminLTE

#### 4.7 Diseño del modelo de medición por indicadores

El proceso de la medición por indicadores, se separa en dos aspectos esenciales, la implementación de los indicadores y la medición estadística que es capaz de generarse del uso de estos, a continuación, se presenta la forma en la que se implementaron ambos conceptos.

##### 4.7.1 Indicadores generales de actividad

Los indicadores de actividad general son aquellos que se refieren al semáforo de actividad, la cual representa mediante colores el estado de avance respecto a cierta actividad específica dentro del proceso de seguimiento de una demanda en los actores internos del TJA, esto se define mediante la evaluación de tiempo entre la fecha de inicio de cada proceso con la salida respectiva.

Este indicador toma en cuenta el momento exacto en el que un fragmento del proceso del seguimiento inicia (por ejemplo: el registro de una demanda por el Oficial de Partes indica el inicio de la fase y el final de la misma se registra cuando éste turna el expediente al siguiente Actor), para ello se definieron los indicadores con base a dos aspectos, el primero es el que establece la Ley de Justicia Administrativa del estado de Colima, el cual dependiendo el estatuto de proceso de la demanda dictamina de entre tres a ocho días hábiles para el proceso de dicha fase; el segundo aspecto que se consideró para establecer este semáforo son las pautas específicas proporcionadas por el Magistrado Presidente, en función de agilizar los procesos se estipularon los parámetros que se pueden observar en la

Tabla 4, los cuales son utilizados para calcular la productividad en función de los días hábiles utilizados para el tiempo de respuesta de cada uno de los actores, se toma en consideración que si el evento no ha finalizado se toma la fecha y hora actual para completar el cálculo de los días consumidos para tal efecto.

*Tabla 4 Parámetros de evaluación para el semáforo*

Usuario\Días Hábiles	Verde	Naranja	Rojo
Oficial de partes	0-2	2-4	>4
Secretario de Acuerdos	0-3	3-6	>6
Actuarios	0-1	1-3	>3
Proyectistas	0-2	2-5	>5

#### 4.7.2 Medición estadística

La medición estadística del módulo de indicadores engloba diferentes aspectos relacionados con la medición procesos dentro de periodos de tiempo, esto es, genera el resumen general de los tiempos de respuesta promedio en cada una de las fases del seguimiento desglosado por expedientes. Anexado a esta compilación de cálculos se integran también los resúmenes de cálculos de promedios de atención por periodo de tiempo como lo son mensuales, semestrales y anuales en donde se puede especificar los parámetros de búsqueda para tal efecto, los cuales incluyen evaluaciones por tipo de demanda, estado de proceso del seguimiento y finalmente de los actores involucrados.

Para poder realizar este cálculo se utiliza la fórmula provista por la proporción DAU/MAU, la cual, cuya propiedad principal es ser adaptable a la situación a la que se requiera aplicar, con lo que es posible realizar los cálculos mencionados anteriormente. Por ejemplo, si se desea obtener la cantidad de expedientes procesados en un periodo de tiempo la fórmula se aplica de la siguiente forma:

$$\frac{N^{\circ} \text{ de expedientes procesados}}{N^{\circ} \text{ de Expedientes recibidos por periodo de tiempo}} = \% \text{ de avance en el periodo}$$

De esa forma modificando los parámetros de entrada para la proporción podemos obtener los datos estadísticos en forma de porcentaje del trabajo realizado por los actores, en caso de que se desee realizar una búsqueda diferente se ajustan dichos indicadores, lo que permite una amplia gama de posibilidades para realizar cálculos y visualizar dichos factores.

## 4.8 Codificación

La codificación del proyecto se realizó con la ayuda del framework Laravel <sup>TM</sup> el cual funciona bajo el lenguaje de programación PHP y en combinación de una base de datos basada en MySql, esto proporcionó la flexibilidad de establecer tanto funciones almacenadas en el motor de base de datos, como aquellas que se ejecutan en el servidor del proyecto, con la finalidad de repartir la carga de trabajo de los procesos.

Dentro de los procesos establecidos, en el motor de base de datos se tienen programadas las consultas que retornan información tanto general como específica, esto implica retomar múltiples datos, agruparlos y retornarlos como un solo elemento.

En el caso de los expedientes es necesario obtener todos los datos del expediente en cuestión, así como generar el cálculo del tiempo de proceso establecido para el indicador de productividad; para aplicar esta funcionalidad se crearon procedimientos almacenados en el gestor en forma de disparadores, los cuales actúan en el momento de la lectura de información para generar el dato requerido para la evaluación que establece el indicador de semáforo.

Dado que el cálculo del tiempo utilizado para cada fase del proceso de la demanda es evaluado en días, es necesario considerar aquellos días no laborales (los cuales no deben ser cuantificados) que han transcurrido desde su registro y diferirlos de aquellos que sí lo son, para esto se utilizó la siguiente función implementada en MySql que se presenta en la Figura 13.

```
SET @FromDate = 'fecha de inicio';
SET @ToDate = 'fecha final';
SET @DiaSemana = 7; -- 7 domingo, 1 lunes
SET @Minimo = CASE WHEN DAYOFWEEK(@FromDate)-1 = 0 THEN 7 ELSE DAYOFWEEK(@FromDate)-1 END;

SELECT datediff(@ToDate, @FromDate) DIV 7 + (CASE WHEN @Minimo = @DiaSemana THEN 1 ELSE 0 END) AS 'CantDias';
```

Figura 13 Procedimiento obtención de días hábiles

Una vez que se han calculado los días hábiles de cada proceso, se realiza el seguimiento general de cada expediente, resultando una colección de información; el procedimiento se puede observar en la Figura 14, la cual presenta la consulta establecida para dicho proceso.

```

1 CREATE ALGORITHM = UNDEFINED DEFINER = `root`@`localhost` SQL SECURITY DEFINER VIEW `v_entradasalida1` AS SELECT
2   `a`.`id_exp` AS `id_exp`,
3   (
4     to_days(`b`.`salida`) - to_days(`a`.`entrada`)
5   ) AS `tiempo`
6 FROM
7   (
8     `v_entradasalida1` AS `a`
9     JOIN `v_salidasalida1` AS `b` ON (
10      (`a`.`id_exp` = `b`.`id_exp`)
11    )
12  )
13 UNION
14 SELECT
15   `seguimiento`.`id_exp` AS `id_exp`,
16   (
17     to_days(now()) - to_days(`seguimiento`.`fecha`)
18   ) AS `tiempo`
19 FROM `seguimiento`
20 WHERE
21   (
22     (
23       `seguimiento`.`id_modulo` = 2
24     )
25     AND (
26       `seguimiento`.`movimiento` = 'Entrada'
27     )
28     AND (
29       NOT (
30         `seguimiento`.`id_exp` IN (
31           SELECT
32             `seguimiento`.`id_exp`
33           FROM `seguimiento`
34           WHERE
35             (
36               (
37                 `seguimiento`.`id_modulo` = 3
38               )
39               AND (
40                 `seguimiento`.`movimiento` = 'Entrada'
41               )
42             )
43           )
44         )
45       )
46     )
47 )
48 )

```

Figura 14 Código de la vista obtención de seguimiento de expediente

## 4.9 Pruebas

Las pruebas que se aplicaron a este desarrollo de software, se realizaron mediante lo establecido en las enseñanzas de la ingeniería de software, lo que implica el realizar pruebas de los elementos de forma celular, para proseguir con el funcionamiento de estructuras complejas, lo que se conoce como pruebas unitarias y de integración; A continuación, se presentan los procesos realizados para dichas pruebas.

### 4.9.1 Pruebas Unitarias

Dentro del framework de programación Laravel, existe una herramienta que nos permite crear de forma rápida nuestros casos de prueba, para ello se realiza la ejecución del comando que se presenta en la Figura 15.

```

C:\xampp\htdocs\laravel>php artisan make:test WelcomeTest
Test created successfully.
C:\xampp\htdocs\laravel>

```

Figura 15 Creación de una prueba unitaria

Una vez creadas nuestras pruebas, se procede a realizar la programación lógica de lo que ésta realizará, por ejemplo, en la Figura 16 se puede observar el caso de prueba que verifica la creación de correos electrónicos.

```
1 <?php
2 declare(strict_types=1);
3
4 use PHPUnit\Framework\TestCase;
5
6 final class EmailTest extends TestCase
7 {
8     public function testCanBeCreatedFromValidEmailAddress(): void
9     {
10         $this->assertInstanceOf(
11             Email::class,
12             Email::fromString('user@example.com')
13         );
14     }
15
16     public function testCannotBeCreatedFromInvalidEmailAddress(): void
17     {
18         $this->expectException(InvalidArgumentException::class);
19
20         Email::fromString('invalid');
21     }
22
23     public function testCanBeUsedAsString(): void
24     {
25         $this->assertEquals(
26             'user@example.com',
27             Email::fromString('user@example.com')
28         );
29     }
30 }
31
```

Figura 16 Prueba unitaria de emails

Siguiendo este procedimiento se realizaron los casos de prueba de todos los módulos, mediante la ejecución de dichos casos; un ejemplo de eso se puede observar en la Figura 17 donde se realiza la verificación de los casos que abarca el grupo de mensajes, el cual realiza la generación de correos electrónicos que sirven como notificación a los usuarios de un cambio de fase del seguimiento de la demanda. Este proceso se evalúa realizando una prueba general que resulta en dos procedimientos asertivos, los cuales indican el envío de dos correos electrónicos, uno por usuario (demandante y demandado).

```
C:\xampp\htdocs\laravel>phpunit --group message
PHPUnit 4.8.16 by Sebastian Bergmann and contributors.

.

Time: 2.12 seconds, Memory: 8.75Mb
OK (1 test, 2 assertions)
C:\xampp\htdocs\laravel>
```

Figura 17 Ejecución de prueba unitaria

#### 4.9.2 Pruebas de Integración

Un ejemplo de la integración de diversos paquetes de pruebas unitarias, el cual es necesario para realizar una prueba de integración se puede observar en la Figura 18, ésta se representa con la prueba realizada para el formulario de inicio de sesión al sistema; para el inicio de sesión al sistema es necesario ingresar un usuario y contraseña, dichos elementos son evaluados y retornan cuatro valores esenciales para la gestión de la sesión del navegador, el id del usuario, su nombre, tipo de usuario y estatus de cuenta, estos valores deben ser verificados para permitir el ingreso al sistema, resultando en el retorno de datos correctos o vacíos según sea el caso.

```
C:\xampp\htdocs\DocumentaQro>phpunit --group paquetes
PHPUnit 4.8.16 by Sebastian Bergmann and contributors.

..
Time: 4.12 seconds, Memory: 15.75Mb
OK (2 tests, 8 assertions)
C:\xampp\htdocs\DocumentaQro>
```

Figura 18 Prueba de integración de paquetes

Como se ha observado, la definición de las pruebas tanto unitarias como de integración es un proceso extenso y minucioso, por ello se resume en subgrupos como los que se muestran en la Tabla 5, la cual incorpora todas las pruebas que se realizaron tanto a nivel unitario como de integración.

Tabla 5 Grupos de pruebas aplicados al sistema

Grupos de Pruebas	
Unitarias	De Integración
Visualización de formularios	Escritura y lectura de información
Visualización de menús	Subida de documentos
Carga de vistas	Llenado de formularios
Lectura de información	Validación de datos
Generación de correos de aviso	Envío de notificaciones
	Cálculo de Indicadores

El desarrollo del proyecto SiSeDe/Indicadores de Productividad permitió controlar el proceso que establece la Ingeniería de Software con la finalidad de asegurar la calidad de producto de software en relación a la funcionalidad proyectada.



La gestión que ofrece la Ingeniería de Software nos permite controlar todos los aspectos que se ven involucrados en el desarrollo del producto informático, esto inicia desde la recolección de requerimientos que se dividen en funcionales y no funcionales, indicando el camino a seguir, es decir, son las pautas que identifican las acciones y limitaciones del proyecto. Con ayuda de los requerimientos podemos generar las bases estructurales de nuestro producto, las cuales abarcan el nivel de información y el nivel visual, esto es nuestra base de datos y nuestras interfaces, sin olvidar el algoritmo que se debe implementar para transformar la información que recolecta el sistema y debe presentar como respuesta al usuario.

## CAPÍTULO V Resultados

En este capítulo se presentan los resultados del sistema propuesto, con base al modelo conceptual presentado en el capítulo uno de este documento y en el caso de estudio citado.

### 5.1 SiSeDe y el módulo de Indicadores de Productividad Laboral

El módulo de productividad laboral que se implementa en el proyecto SiSeDe se compone de tres factores importantes, el primero de éstos involucra la recolección de información para el cálculo de la productividad y valoración estadística, esto incluye recopilar los movimientos que se registran dentro de la demanda; en la Figura 19 se observa de forma gráfica el seguimiento de la demanda en donde se presenta información general y específica, por un lado, de forma general se puede observar el estatus del seguimiento de la demanda de manera puntual, y por otro, permite visualizar los procesos elaborados para dicho expediente, entre ellos la fecha de movimiento, personal que lo realizó, el módulo del sistema del que procede el movimiento y datos elementales para el cálculo de la productividad misma.

Expediente	Creado	Demandado	Demandante	Resumen
406	12-11-2018	Director General de Tránsito Municipal y Seguridad Ciudadana del H. Ayuntamiento Constitucional de Colima	Luis Daniel Olivo Arellano	ASUNTO: Nulidad de boleta de infracción

Fecha Actividad	Movimiento	Módulo	Concepto	Documento	Folio	Responsable	Comentarios
12-07-2018 00:02:01	Entrada	Oficialía de Partes	Validado por el Oficial de Partes			Gabriela Alitzel García Márquez	ASUNTO: Nulidad de boleta de infracción
12-07-2018 00:02:01	Institu	Oficialía de Partes	Se agrego documento	Demanda	406_0	Gabriela Alitzel García Márquez	Anexo de documento <a href="#">Ver</a>
12-07-2018 00:02:01	Institu	Oficialía de Partes	Se agrego documento	Anexo	406_1	Gabriela Alitzel García Márquez	Anexo de documento <a href="#">Ver</a>
12-07-2018 00:02:01	Institu	Oficialía de Partes	Se agrego documento	Anexo	406_2	Gabriela Alitzel García Márquez	Anexo de documento <a href="#">Ver</a>
12-07-2018 00:02:01	Institu	Oficialía de Partes	Se agrego documento	Anexo	406_3	Gabriela Alitzel García Márquez	Anexo de documento <a href="#">Ver</a>
11-07-2018 07:00:00	Salida	Oficialía de Partes	Se turna el expediente al Secretario de Acuerdo			Gabriela Alitzel García Márquez	Se presentó el día 05 de julio de 2018
11-07-2018 07:00:00	Entrada	Secretaría de acuerdos	Se turna el expediente al Secretario de Acuerdo			Jorge Ricardo Anguiano Barbosa	Se presentó el día 05 de julio de 2018
12-07-2018 15:59:42	Institu	Secretaría de acuerdos	El expediente se admite	Admisión de Demanda	406_4	Jorge Ricardo Anguiano Barbosa	Anexo de Documento <a href="#">Ver</a>
12-07-2018 16:00:05	Salida	Secretaría de acuerdos	Se turna al Actuario para su notificación			Jorge Ricardo Anguiano Barbosa	Se turna el Expediente al Actuario para Notificar
12-07-2018 16:00:05	Entrada	Actuarios	Se turna al Actuario para su notificación			Lizeth Yemeli Martínez García	Se recibe el Expediente para Notificar
12-11-2018 13:10:58	Institu	Actuarios	Se envía notificación	Acta de Notificación Simple	406_5	Lizeth Yemeli Martínez García	asdasda <a href="#">Ver</a>

Figura 19 Seguimiento paso a paso de demandas

El segundo factor que se involucra en el proyecto es la integración de indicadores que ayuden a visualizar el estado general de las acciones de cada usuario, esto se realizó con ayuda de la información recolectada en el seguimiento de las demandas, mediante la implementación de las fórmulas provistas por la relación DAU/MAU en conjunción con los parámetros de valoración que se mostraron en la Tabla 4, los cuales se refieren a los tiempos de recepción y proceso de demandas y recursos de alegatos que establece la Ley de Justicia Administrativa del Estado de Colima

en los artículos 52, 60, 64, 71 y 91. Esto permite la visualización de la información mediante indicadores de color como se puede observar en la Figura 20, la cual presenta el estado del proceso individual del seguimiento de una demanda de forma similar a un semáforo vial, donde los colores representan el tiempo de respuesta y atención del proceso en cuestión.

En la Figura 20 se observa el caso de un Secretario de Acuerdos específico, en donde se ejemplifican los tres colores del semáforo, en este caso, para el expediente 400, como se puede observar este se encuentra en verde ya que se ha dado respuesta oportuna dentro de los primeros tres días hábiles como se estipuló en la tabla 4; en cambio, el expediente 406 se muestra en color naranja debido que el proceso de evaluación realizado para éste Actor, ha pasado de los primeros cuatro días hábiles siguientes a la asignación del expediente y no se ha tenido respuesta del mismo; finalmente, en el expediente 9 se puede observar que el indicador se ha tornado rojo, ya que el proceso de seguimiento indica que se ha excedido el tiempo recomendable para la atención oportuna establecida dentro de los parámetros de la Ley de Justicia Administrativa, considerándose como un indicador de qué expedientes requieren mayor atención.

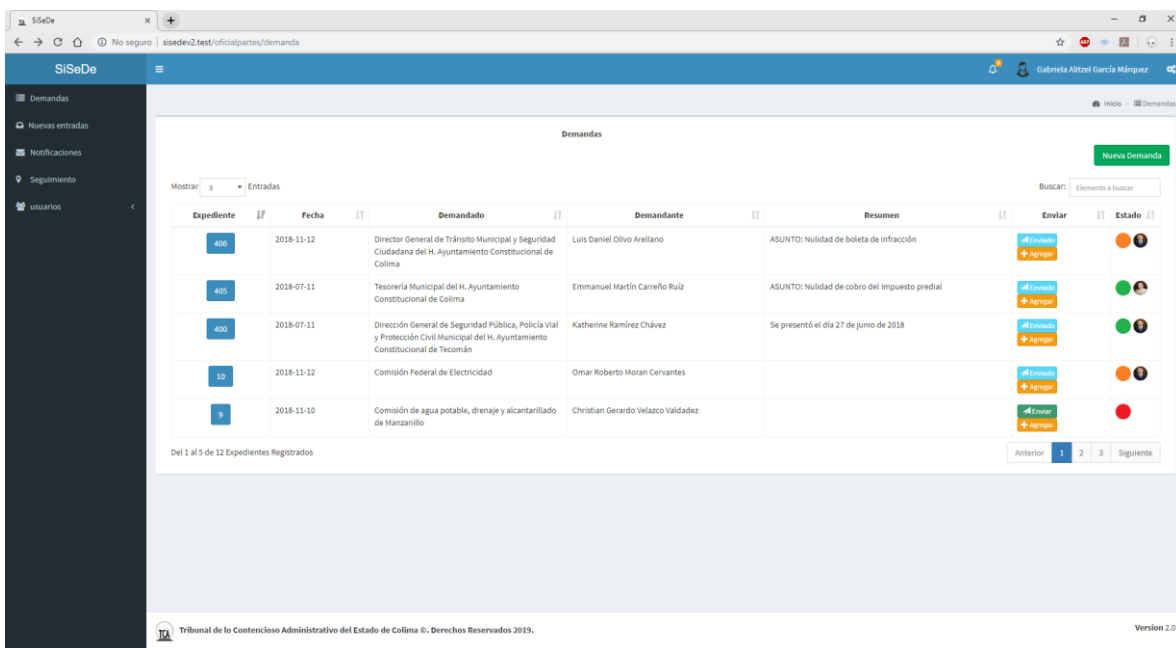


Figura 20 Visualización del semáforo

Por último, el tercer factor es el que indica que el módulo de indicadores de productividad solo estaría completo con la inclusión de un medio cuantificable para representar dicha información, por ello en la Figura 21, los Magistrados, incluido el Presidente, pueden observar de forma gráfica el cálculo de la productividad en el proceso de atención de las demandas, esta información se debe puntualizar acorde a los filtros de Año y Mes. Así, la Figura 21 presenta la información relativa a el proceso de evaluación de productividad para el año 2017 en el mes de noviembre, en donde se obtienen los valores correspondientes a tipo de demandas recibidas,

estado del proceso del seguimiento de las demandas registradas de forma mensual, y finalmente la comparativa anual de los procesos mencionados anteriormente

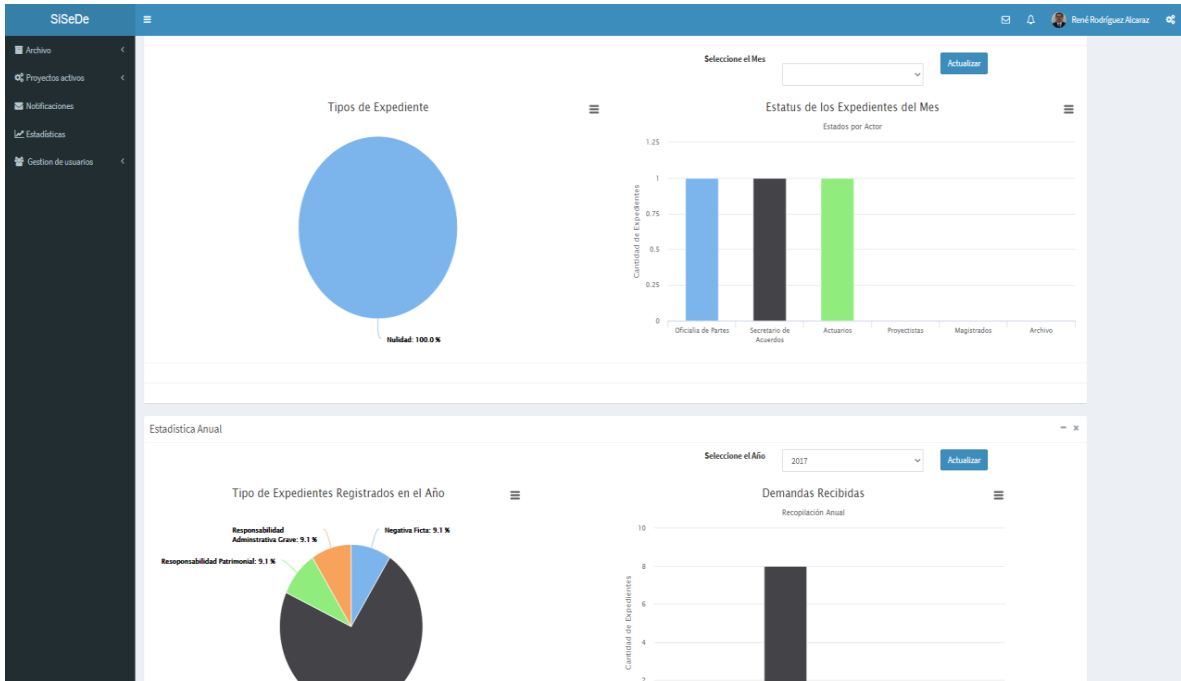


Figura 21 Información estadística general

Finalmente, en la Figura 22 se puede observar la recopilación final del estado de atención de cada uno de los empleados conforme a los valores establecidos anteriormente, se observa el conteo individual mensual de cada uno de ellos, de la misma forma se expone la cantidad de expedientes asignados y el estado general del avance en ellos.

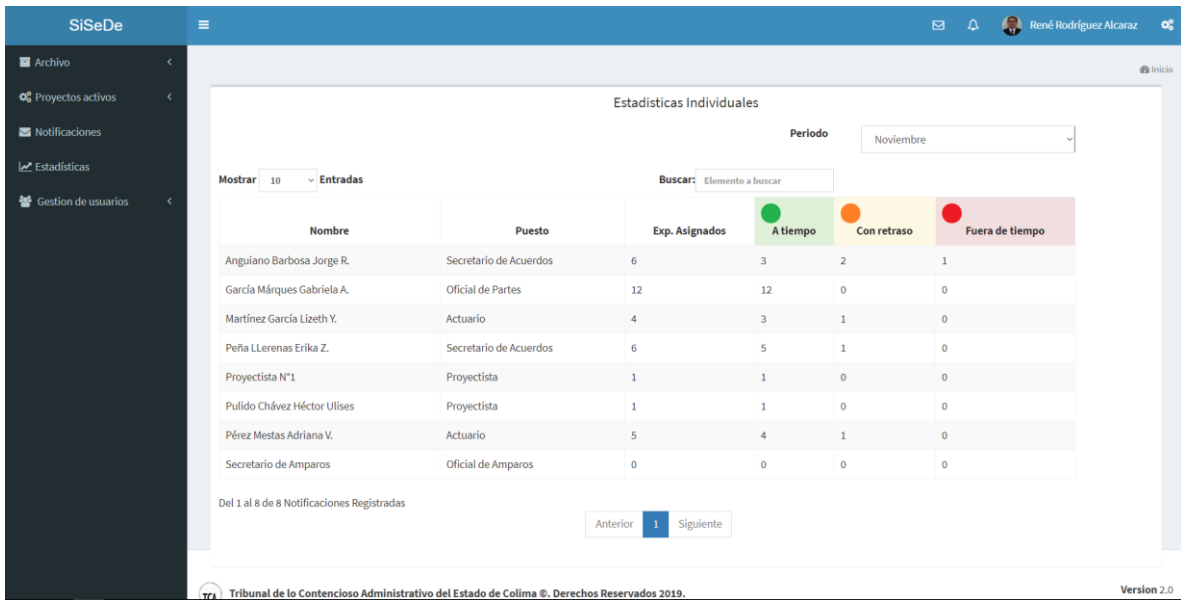


Figura 22 Información estadística individual

## **5.2 Configuración y Carga del sistema en el servidor**

El proceso de configuración y carga del servidor es un proceso simple el cual basta con seguir una serie de pasos para poder establecer el enlace correcto de nuestro proyecto a un sitio en internet al cual se pueda acceder desde múltiples plataformas y dispositivos, esto no limita la conectividad ya que, una vez instanciado nuestro proyecto en internet, el acceso al sistema solo se ve limitado por los permisos de usuario y de si se encuentra una persona registrada en él.

Para comenzar con el proceso de carga en el servidor se tiene establecido que es necesario contar con la prestación de un servicio de almacenamiento y de procesamiento en internet, mejor conocido como *Hosting*; el hosting, o hospedamiento como refleja su significado del inglés al español, es hacer uso de un equipo de cómputo al que se tiene acceso a través de internet mediante el arrendamiento de dicho servicio, para eso se deben evaluar las capacidades de éste (capacidad de almacenamiento, memoria RAM, cantidad de procesos entrantes/salientes, ancho de banda, cantidad de CPU's , plataforma del servidor, servicios instalados, etc.) y así determinar si las capacidades que éste ofrece son las adecuadas para las necesidades que deben cubrirse para los procesos que realiza nuestro proyecto.

En el caso del proyecto propuesto en ésta tesis, se utilizó un hosting proporcionado por la empresa GoDaddy© la cual ofrece un paquete de host que incluye una computadora dedicada con dos procesadores, 1 GB de memoria RAM, ancho de banda ilimitado, espacio de almacenamiento de hasta 3TB sin costo, con un sistema operativo Linux el cual se establece como servidor. Los detalles específicos de este computador arrendado se establecen en la Figura 23 y en la Figura 24, las cuales establecen las características del servidor, así como el panel de gestión del servidor mismo.

# Server Information

## Server Information

Item	Detail
Hosting Package	Ultimate
Server Name	p3plcpnl0885
cPanel Version	70.0 (build 51)
Apache Version	2.4.33
Versión de PHP	5.6.36
MySQL Version	5.6.39-cll-lve
Architecture	x86_64
Operating System	linux
Dedicated IP Address	160.153.57.136
Path to Sendmail	/usr/sbin/sendmail
Path to Perl	/usr/bin/perl
Perl Version	5.10.1
Kernel Version	2.6.32-896.16.1.lve1.4.54.el6.x86_64

*Figura 23 Características del servidor*

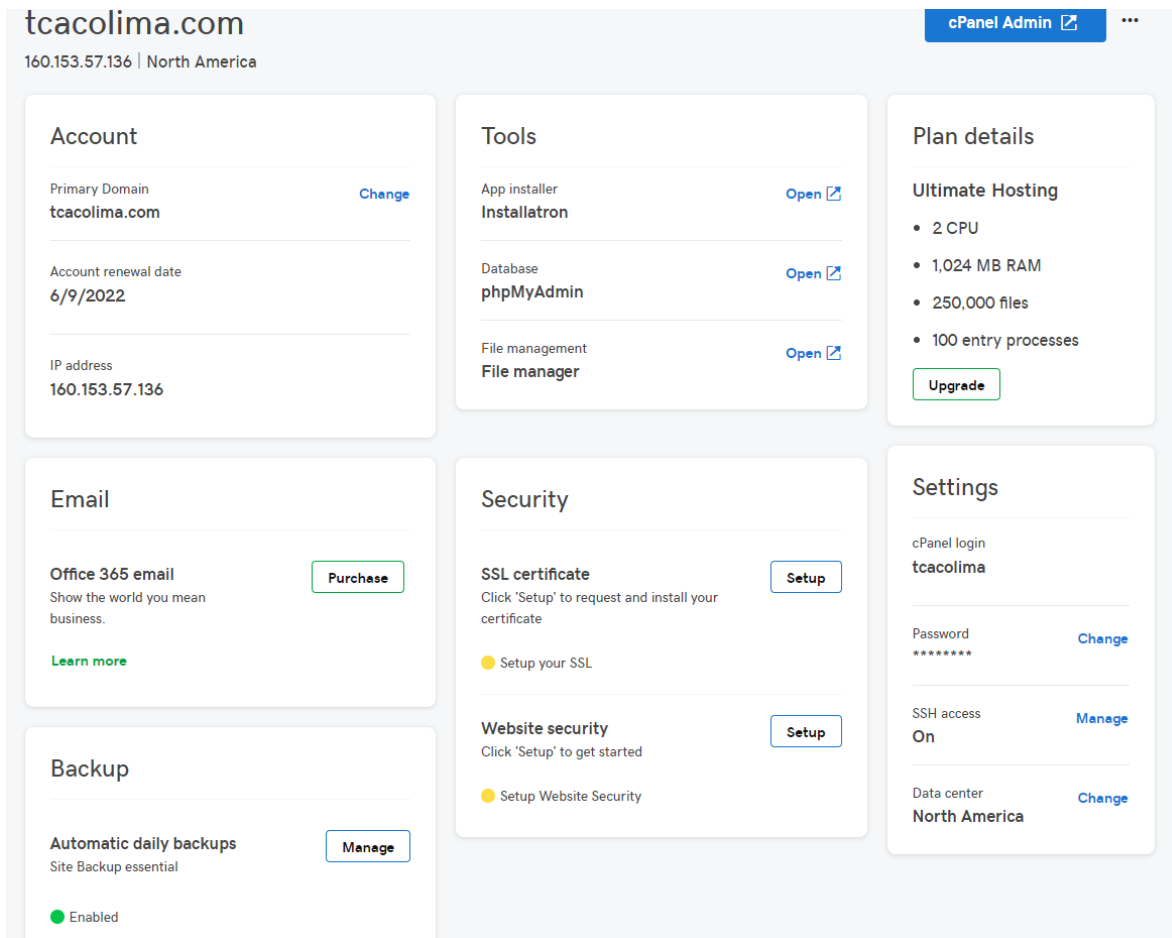


Figura 24 Panel de administración del servidor

Como se mencionó anteriormente, el proceso de carga es simple, basta con cargar los elementos de nuestro proyecto (archivos, bases de datos y configuración) al servidor, para ello es necesario realizar los siguientes pasos:

- Cargar los archivos que componen el proyecto.
- Configurar el dominio de internet.
- Crear la base de datos.
- Enlazar las configuraciones.
- Probar el resultado.

### 5.2.1 Carga de los archivos que componen el proyecto

El proceso de carga de archivos es una actividad simple la cual requiere de protocolos comunes en el ámbito de las tecnologías de internet, como lo es el FTP o Protocolo de Transferencia de Archivos, el cual es un medio que establece cualquier sistema operativo como medio de transferencia remota (mediante internet) permitiendo realizar movimientos de los archivos entre computadores utilizando internet como conducto.

Para poder hacer uso de este protocolo existen reglas de seguridad, las cuales se establecieron para limitar el acceso de esta funcionalidad solo a aquellas personas autorizadas para este propósito, para esto es necesario definir cuentas de acceso, la Figura 25 establece el ejemplo del proceso de creación de cuenta del protocolo que nos permite maniobrar en un directorio especificado para la misma.

Add and configure FTP Accounts to get your website on the internet fast. FTP allows you to manage the files that are associated with your website through an FTP client such as [FileZilla](#). For more information, read the [documentation](#).

### Add FTP Account

Iniciar sesión

Dominio

Contraseña

Password (Again)

Seguridad  Password Generator

Directorio

Cuota  2000 MB  Unlimited

[Create FTP Account](#)

### FTP Accounts

Buscar  [Dirigirse a](#) Page Size: 10

Iniciar sesión	Path	Uso / Cuota	Acciones
SB_385673@tccolima.com	/home/tccolima/public_html	238 / 100 MB	<a href="#">Cambiar contraseña</a> <a href="#">Cambiar cuota</a> <a href="#">Borrar</a> <a href="#">Configurar cliente FTP</a>
SEC_385673@tccolima.com	/home/tccolima/public_html	238 / 100 MB	<a href="#">Cambiar contraseña</a> <a href="#">Cambiar cuota</a> <a href="#">Borrar</a> <a href="#">Configurar cliente FTP</a>
andres@tccolima.com	/home/tccolima/html/sisede	125 / 100 MB	<a href="#">Cambiar contraseña</a> <a href="#">Cambiar cuota</a> <a href="#">Borrar</a> <a href="#">Configurar cliente FTP</a>
masterssede2@tccolima.com	/home/tccolima	163 / 100 MB	<a href="#">Cambiar contraseña</a> <a href="#">Cambiar cuota</a> <a href="#">Borrar</a> <a href="#">Configurar cliente FTP</a>

Figura 25 Panel de cuentas FTP

Una vez que se ha establecido una cuenta y los permisos de usuario para éste, con ayuda de la utilería FileZilla, una herramienta OpenSource que permite manejar FTP, se procede a realizar la carga de los archivos que componen nuestro proyecto a el directorio de acceso público conocido como public\_html, en la cual se debe establecer la totalidad del proyecto, un ejemplo de esto se presenta en la Figura 26 la cual muestra el administrador de archivos del servidor cuando se ha finalizado el proceso de transferencia de datos mediante el protocolo antes mencionado; esto indica el final del proceso de carga de archivos.



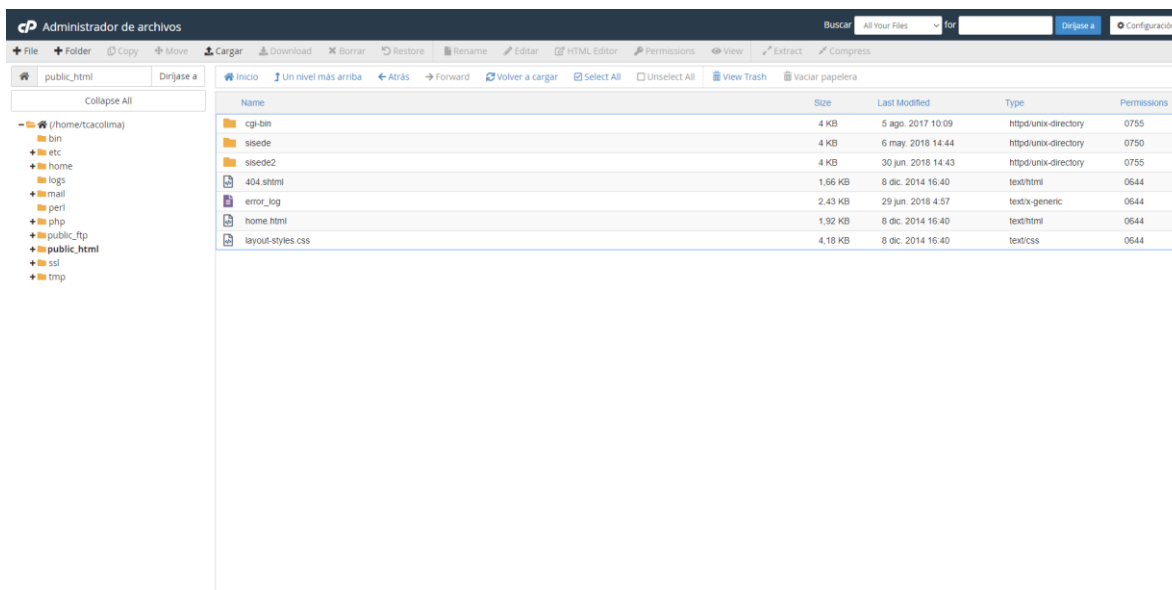


Figura 26 Administrador de archivos del servidor

Este proceso indica el final de la fase de carga de archivos, por lo que se procede a la configuración del dominio que hace referencia al proyecto.

### 5.2.2 Configuración del dominio

Para especificar qué es el dominio, es necesario comprender que en internet el dominio (sitio web) es la forma a la que nos hacemos referencia a una página o programa montado en un servidor web; la forma de acceder a éstos se designa en dos formas, el primer método es mediante la dirección que nos ofrece el protocolo de direcciones IP la cual tiene asignada el servidor de forma estática y se encuentra visible en la Figura 22 que representa las características del servidor, este medio de acceso no es muy comúnmente utilizada por usuarios finales, más bien es de uso de usuarios de tipo administrador o programadores que dan mantenimiento a la página; el segundo método de acceso es mediante la cadena de identificación (dirección WEB) la cual tiene definido un dominio, el cual se diferencia en el nombre del sitio ya que es el prefijo que sigue a continuación del punto dentro del nombre del mismo, un ejemplo de acceso a un sitio web mediante su cadena de identificación se puede visualizar en la Figura 27 la cual representa el acceso a el sitio Google.com .



Figura 27 Dominio de un sitio web

En resumen, el dominio es la forma de acceso al sitio que se ha montado en el servidor WEB. Dentro de un dominio es posible generar sub-dominios los cuales nos permiten incluir diferentes aplicaciones en un solo sitio, lo que permite reducir el costo de mantenimiento en caso de que se desee incorporar otra aplicación a nuestro sitio.

Para poder realizar este proceso en el panel de control de nuestro servidor, es necesario acceder a la configuración de subdominios, en donde se puede establecer el nombre de la aplicación y la carpeta física de almacenamiento a donde la cadena de identificación colocada en la barra de navegación dirija la acción para inicializar nuestra aplicación mediante el re direccionamiento WEB.

La forma de realizar el registro del sub-domino en el hosting de Godaddy se puede visualizar en la figura 28.

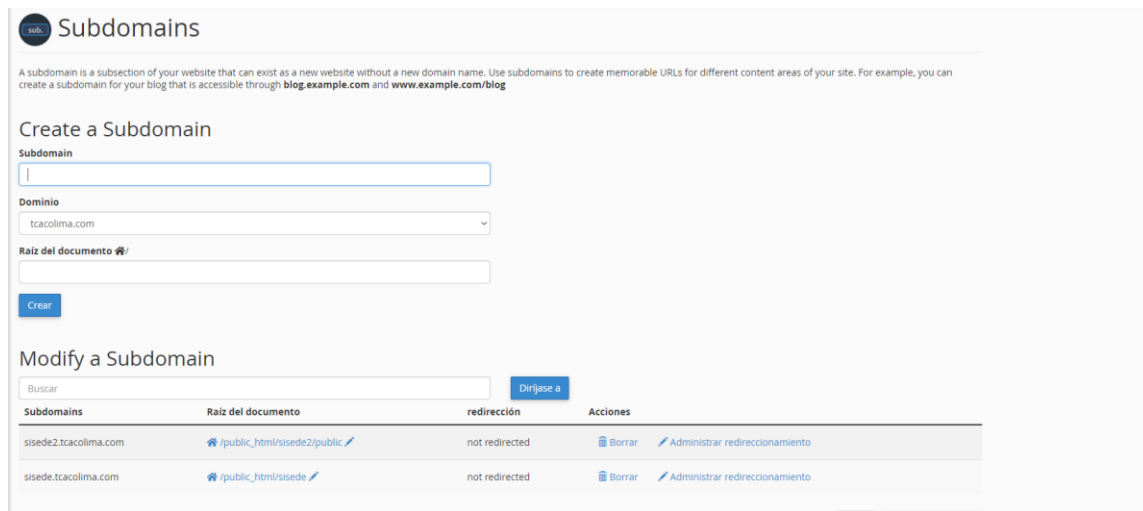


Figura 28 Creación de subdominio

Una vez que se ha creado el subdominio que representará a nuestra aplicación en el dominio que nos proporciona el host, podemos proceder a realizar la creación y configuración de la base de datos que servirá de almacén de información para el programa.

### 5.2.3 Creación y montaje de la base de datos

Dentro de los conocimientos necesario para el manejo de las bases de datos en un sistema gestor dedicado para este propósito, debemos entender que existe como requerimiento principal tener una cuenta de usuario de base datos registrada en el gestor, lo que nos otorga credenciales para poder hacer uso de las bases de información que éste controle.

En el caso de los gestores basados en el motor de base de datos de MySQL ( como es el caso en este proyecto) no es recomendable hacer uso del usuario administrador general (root) ya que esto agrega vulnerabilidad al manejo de la información, por lo que se recomienda crear un usuario de sistema al cual le otorgaremos credenciales y privilegios para gestionar una base de datos, que en ésta ocasión es a la base de datos diseñada para éste proyecto, por lo que se recurre a utilizar un código en el lenguaje SQL que permite la creación de un nuevo usuario y la asignación de privilegios a determinadas bases de datos, el procedimiento se realizó con la ayuda de una aplicación que proporciona una interface gráfica a nuestra base de datos como lo es PhpMyAdmin, la cual se incluye en el panel de configuración de nuestro servidor. Un ejemplo del código utilizado se puede observar en la Figura 29.

```
1 CREATE USER 'nombre_usuario'@'localhost' IDENTIFIED BY 'tu_contrasena';
2 GRANT ALL PRIVILEGES ON * . * TO 'nombre_usuario'@'localhost';
3 FLUSH PRIVILEGES;
```

*Figura 29 Código para creación de usuario de base de datos*

Una vez que se ha registrado el usuario de base de datos que asignaremos a la aplicación se procede a la creación de la base de datos del proyecto con ayuda del script en lenguaje SQL que se generó durante el proceso de modelado de datos, el cual luce como se muestra en la Figura 30, la cual es solo un fragmento de código dada la extensión de nuestra base.

```

1  /*
2  Navicat MySQL Data Transfer
3
4  Source Server          : Mysql
5  Source Server Version : 50720
6  Source Host           : 127.0.0.1:3306
7  Source Database      : sisedev3
8
9  Target Server Type    : MYSQL
10 Target Server Version : 50720
11 File Encoding        : 65001
12
13 Date: 2018-06-27 04:38:26
14 */
15 Create database sisedev3;
16
17 Use sisedev3;
18
19 SET FOREIGN_KEY_CHECKS=0;
20
21 -- -----
22 -- Table structure for acuerdotipo
23 -- -----
24 DROP TABLE IF EXISTS `acuerdotipo`;
25 CREATE TABLE `acuerdotipo` (
26   `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
27   `Tipo` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
28   `Descripcion` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
29   `nivel` tinyint(4) NOT NULL,
30   `estatus` tinyint(1) NOT NULL DEFAULT '1',
31   PRIMARY KEY (`id`)
32 ) ENGINE=InnoDB AUTO_INCREMENT=32 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
33

```

Figura 30 Script para creación de base de datos

Una vez finalizada la ejecución de este Script de creación de base de datos, podemos proceder a realizar una revisión del proceso realizado con ayuda del gestor phpMyAdmin, el cual se puede visualizar en la Figura 31, lo que finaliza el proceso de creación y montaje de la base de datos.

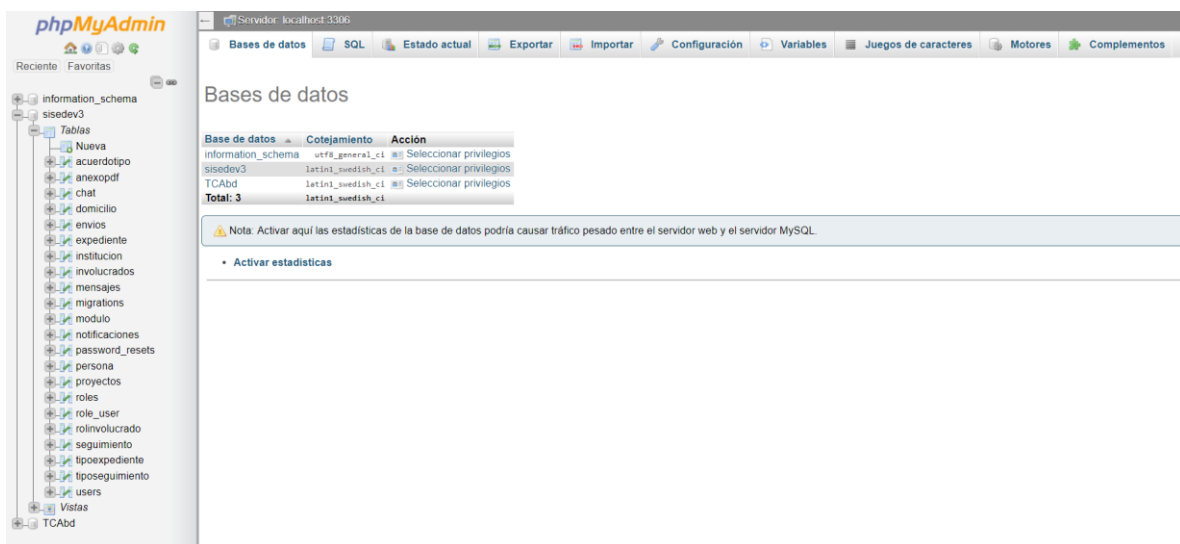


Figura 31 Montaje de la base de datos

## 5.2.4 Enlace de configuraciones

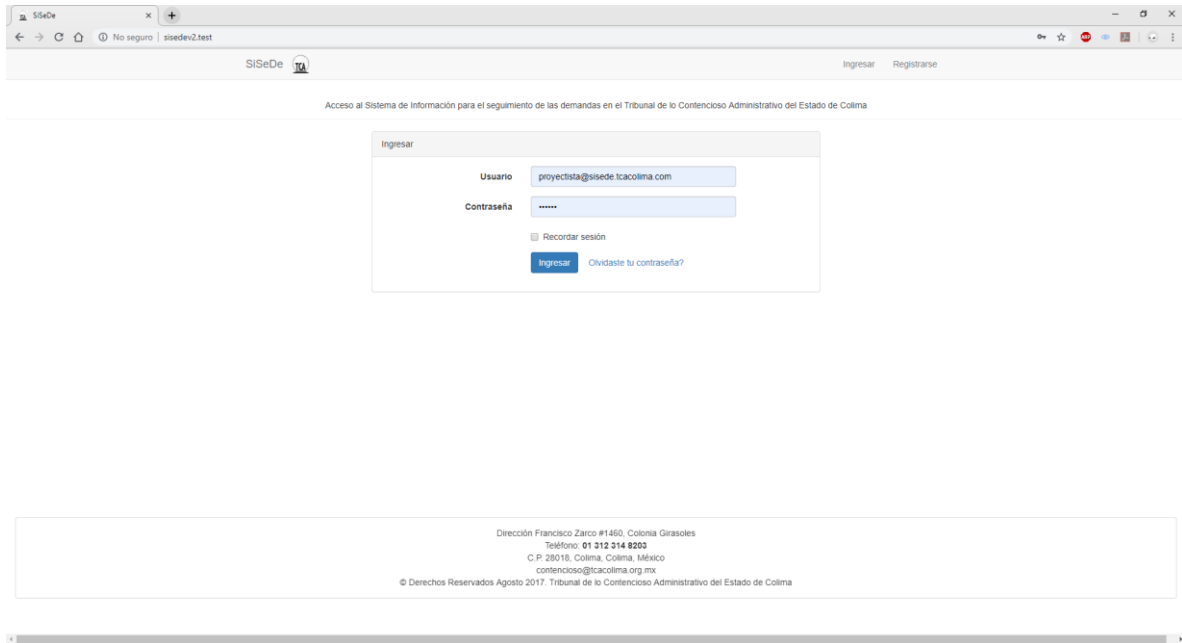
El siguiente proceso del montaje de nuestro proyecto se define gracias a las características que provee el framework de programación Laravel, en donde es necesario definir los elementos de configuración general del proyecto en el archivo designado para dicho fin, este se puede encontrar en la carpeta de nuestro proyecto la cual en procesos anteriores hemos ubicado en el directorio “public\_html”, dentro de éste contenedor de archivos se encuentran la colección de carpetas y documentos que componen nuestro proyecto y dentro de éstos es posible identificar el archivo de configuraciones el cual es denominado “.env” o configuración de entorno ( *environment* en inglés), donde debemos especificar los elementos de acceso para los servicios especiales que utiliza nuestro proyecto como lo es la base de datos y el servidor de correo electrónico, para ello es necesario ingresar las credenciales que se definieron en el proceso de creación de la base de datos. Un ejemplo del contenido del archivo de configuración de entorno se puede observar en la Figura 32; al finalizar el proceso de enlace de configuraciones es posible realizar las primeras pruebas de acceso al sistema o pruebas de despliegue.

```
1 APP_NAME=SiSeDe
2 APP_ENV=local
3 APP_KEY=base64:Tid2Q7kKnsI2NYHU9QnXkZy6F4oQDxEfoqcRb1nz15M=
4 APP_DEBUG=true
5 APP_LOG_LEVEL=debug
6 APP_URL=http://localhost
7
8 DB_CONNECTION=mysql
9 DB_HOST=127.0.0.1
10 DB_PORT=3306
11 DB_DATABASE=sisedev3
12 DB_USERNAME=root
13 DB_PASSWORD=123456789
14
15 BROADCAST_DRIVER=log
16 CACHE_DRIVER=file
17 SESSION_DRIVER=file
18 SESSION_LIFETIME=120
19 QUEUE_DRIVER=sync
```

Figura 32 Configuración del proyecto

## 5.2.5 Pruebas de despliegue

Las pruebas de despliegue del proyecto son elementos esenciales para cerciorase de que la carga y configuración del proyecto se ha realizado con éxito, para esto es necesario ingresar al sitio mediante la dirección WEB y deberá ejecutarse nuestra aplicación sin ningún problema, como ejemplo, en la Figura 33 se observa la página inicial de nuestro proyecto que acorde a lo programado es la página de inicio de sesión de usuario.



*Figura 33 Prueba de visualización*

El procedimiento lógico durante las pruebas de despliegue de un proyecto WEB establecen que deben probarse las funcionalidades establecidas en el producto, es decir realizar una prueba rápida de todas las funcionalidades del mismo para asegurarse que las configuraciones aplicadas en el servidor sean las correctas y aseguren la completa funcionalidad del mismo; es por ello que, en la Figura 34 se presenta la visualización del formulario de registro de demanda al cual pueden acceder tanto los usuarios generales (“Usuarios” e “Instituciones”) como los de tipo “Oficial de Partes”, con lo cual se asegura que se realiza de forma correcta el registro de la información de la base de datos como la carga de documentos al espacio de almacenamiento del servidor.

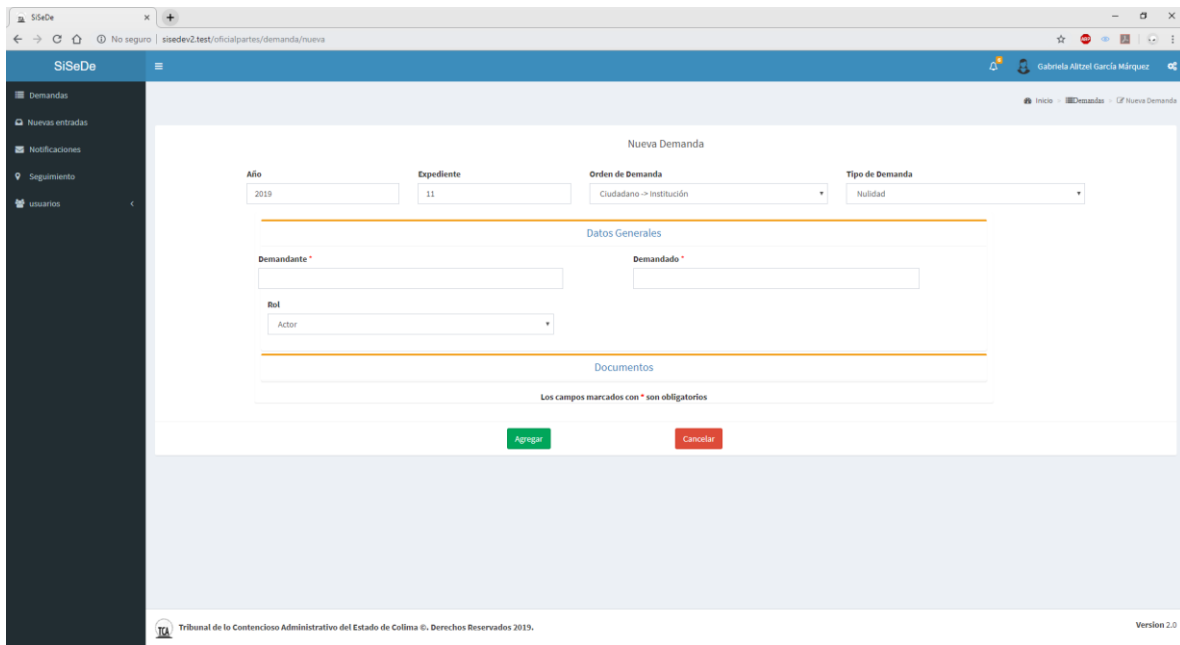


Figura 34 Registro de demanda

## 5.2.6 Configuraciones adicionales

Como proceso final se realizó una configuración extra en el servidor, más específicamente a aquel que se refiere a los servicios que utiliza PHP, para ajustar el límite máximo de carga de archivos que este servicio ofrece; dado que por defecto el límite se encuentra establecido a limitar el proceso de subida de archivos cuando éstos excedan 8MB de almacenamiento (límite no recomendado por la naturaleza de los archivos que se desean almacenar en este proyecto), y teniendo en cuenta que el proceso de subida de un archivo es un proceso que define el tiempo de respuesta de un sitio web, se estableció un máximo considerable para los archivos el cual es de 64 MB lo que garantiza que sea posible realizar la carga de documentos en formato PDF que se requieren en el proceso de seguimiento de demanda.

Este procedimiento de ajuste se realiza en el panel de configuración del servidor, mediante el acceso a los parámetros de configuración de PHP y modificando la variable que establece el tamaño máximo de los archivos para carga (`upload_max_filesize`) como se puede observar en la Figura 35.

PHP Selector | options

Current PHP version: **7.0**

Versión de PHP: 7.0  [Switch To PHP Extensions](#)

allow_url_fopen	On
display_errors	Off
error_reporting	E_ALL
file_uploads	On
include_path	:/opt/alt/php70/usr/share/pear
log_errors	On
mail.force_extra_parameters	no value
max_execution_time	30
max_input_time	-1
memory_limit	128M
open_basedir	no value
post_max_size	64M
session.save_path	/tmp
short_open_tag	On
upload_max_filesize	64M

Figura 35 Configuración de parámetros de PHP

Con éste procedimiento concluye el proceso de configuración y carga del proyecto en el servidor.



## ***CAPÍTULO VI CONCLUSIONES***

### ***6.1 Conclusiones***

El uso de herramientas que permitan evaluar el desempeño de las actividades de las empresas es una práctica común, la cual ha incrementado con la intención de mejorar los ingresos financieros (en relación a las instituciones privadas) y la atención del servicio prestado (enfocado a las instituciones públicas).

La empresa Workmeter quien provee una herramienta que ayuda a la concepción y manejo de indicadores de productividad para las empresas establece en sus casos de éxito las reacciones y mejoras que han tenido sus clientes con la implantación de estos sistemas de gestión de productividad. Sin embargo, este tipo de suceso no solo aparece en empresas privadas, sino, que también existen en empresas públicas, como ejemplo se tiene lo sucedido en 2016 donde Vázquez & Estrada demostraron que bajo un programa de análisis de productividad se pudieron incrementar los ingresos que obtienen.

En el rubro de la aplicación de modelos digitales para el manejo judicial y de justicia, se encuentran diversos ejemplos, como lo son aquellos presentados por Flores (2011) el cual demostraba como los sistemas judiciales de Colombia mejoraron en la atención y resolución de procesos judiciales con la aplicación de la justicia en línea; en Brasil, Silva expone como la implantación de un sistema de tramitación electrónica de procesos judiciales permite brindar atención a zonas de bajo alcance y que además permitía tener un seguimiento de los avances del mismo.

No obstante, pese a toda lo anterior, la implementación de un sistema de gestión de indicadores de productividad en una institución sin motivos financieros como lo es el Tribunal de Justicia Administrativa del Estado de Colima, no cuenta con antecedentes relacionados, lo que implica un área de oportunidad a crecimiento en relación a los complementos que se pueden anexar para mejorar su funcionamiento.

Finalmente, en respuesta a la hipótesis que se planteó para este trabajo se tiene que la implementación de un sistema de información que permita visualizar los indicadores de productividad particulares (como se consideran las variantes obtenidas por periodos de actividad, por individuo y por esquema general) si permite al Tribunal de Justicia Administrativa del Estado de Colima mejorar tanto los procesos de atención de expedientes, como la impartición de justicia; esto mediante el monitoreo constante de los indicadores de productividad, lo que establece una herramienta para la toma de decisiones que puede influir en el personal y en el manejo de la organización, ello en el marco de lo establecido en la ley de Justicia Administrativa implementada en el estado.

## Bibliografía

- Abad, J. H. (06 de 04 de 2005). *TIPOS DE PRUEBAS DE SOFTWARE*. Obtenido de <http://ing-sw.blogspot.com/2005/04/tipos-de-pruebas-de-software.html>
- Álvarez, C., Mateos, J., & Moreno, M. (2005). *Metodología de reingeniería de software para la remodelación de aplicaciones científicas heredadas*. Recuperado el 09 de 2018, de [https://www.researchgate.net/publication/39698262\\_Metodologia\\_de\\_reingenieria\\_del\\_software\\_para\\_la\\_remodelacion\\_de\\_aplicaciones\\_cientificas\\_hereditadas](https://www.researchgate.net/publication/39698262_Metodologia_de_reingenieria_del_software_para_la_remodelacion_de_aplicaciones_cientificas_hereditadas)
- Ambler, S. W. (2014). *The Agile Unified Process (UAP)*. Obtenido de Ambysoft: [www.ambysoft.com/unifiedprocess/agileUP.html](http://www.ambysoft.com/unifiedprocess/agileUP.html)
- Bergmann, S. (2019). *PHPUnit: Programmer-oriented testing framework for PHP*. Obtenido de <https://phpunit.de/index.html>
- Bootstrap. (2015). *About Bootstrap*. Obtenido de <http://www.getbootstrap.com/about>
- Eguiluz, J. (2015). *Introducción a CSS*. Obtenido de Librosweb.es: [http://www.librosweb.es/libro/css/capitulo\\_1/breve\\_historia\\_de\\_css.html](http://www.librosweb.es/libro/css/capitulo_1/breve_historia_de_css.html)
- Espinoza, R. (03 de 2017). *Indicadores de Gestión: ¿Qué es KPI?* Recuperado el 09 de 2018, de <https://robertoespinoza.es/2016/09/08/indicadores-de-gestion-que-es-kpi/>
- Fernández, J. M., & Moreno, A. (2017). Diseño e Implementación de un sistema de información para el seguimiento de proyectos de sentencias en el tribunal de lo contencioso administrativo. *3C TIC, Vol 6(N 2º)*, pag. 1-27. doi:<http://dx.doi.org/10.17993/3ctic.2017.56.13-27>
- Flórez, O. (2011). Uso de las tecnologías de la Información y las comunicaciones (TIC) en los sistemas judiciales. *Sistemas Judiciales, 16*, 56-63. Obtenido de <http://www.sistemasjudiciales.org/content/jud/archivos/notaarchivo/911.pdf>
- Galvis, E. A., González, M. P., & Hernán, V. P. (Diciembre de 2011). Tecnologías de información para acercar al ciudadano a los servicios de justicia en Colombia: el caso del mapa de oferta de justicia. *Ciencia e Ingeniería Neograciense, 21((2))*, p. 131-151. doi:<https://doi.org/10.18359/rcin.263>
- GECKOBOARD. (2017). *DAU/MAU ratio*. Obtenido de <https://www.geckoboard.com/learn/kpi-examples/startup-kpis/dau-mau-ratio/#.WiZBI0qWblU>
- Gómez, J. M. (2017). *¿Qué es un KPI en marketing?, definición, como hacerlos y ejemplos*. Recuperado el 09 de 2018, de <https://laculturadelmarketing.com/que-es-un-kpi-en-marketing/>
- Gutiérrez, J. J. (2008). *¿Qué es un Framework WEB?* Obtenido de [http://www.lsi.us.es/~javierj/investigacion\\_ficheros/Framework.pdf](http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf)
- H. Congreso del Estado de Colima. (16 de 06 de 2018). *Ley de Justicia Administrativa del Estdo de Colima*. Obtenido de Última Reforma Decreto 133: [http://congresocol.gob.mx/web/Sistema/uploads/LegislacionEstatal/LeyesEsatales/Contencioso\\_y\\_administrativo\\_2NOV2016.pdf](http://congresocol.gob.mx/web/Sistema/uploads/LegislacionEstatal/LeyesEsatales/Contencioso_y_administrativo_2NOV2016.pdf)<http://congresocol.gob>

- .mx/web/Sistema/uploads/LegislacionEstatal/LeyesEstatales/Justicia\_Administrativa\_16jun2018.pdf
- IEEE. (2004). *Guide to Sweebok*. Obtenido de <https://www.computer.org/education/bodies-of-knowledge/software-engineering>
- INEGI. (2014). Catálogo de Dependencias e Instituciones de Interés Público. Obtenido de CADIIP 2014-2015: [http://www.beta.inegi.org.mx/contenidos/proyectos/enchogares/regulares/enoe/doc/cadiip\\_2014.pdf](http://www.beta.inegi.org.mx/contenidos/proyectos/enchogares/regulares/enoe/doc/cadiip_2014.pdf); Última consulta: 24 de Mayo 2017
- INEGI. (2014). *Catálogo de Dependencias e Instituciones de Interés Público*. Obtenido de [beta.inegi.org.mx](http://www.beta.inegi.org.mx/contenidos/proyectos/enchogares/regulares/enoe/doc/cadiip_2014.pdf): [http://www.beta.inegi.org.mx/contenidos/proyectos/enchogares/regulares/enoe/doc/cadiip\\_2014.pdf](http://www.beta.inegi.org.mx/contenidos/proyectos/enchogares/regulares/enoe/doc/cadiip_2014.pdf)
- Internetyya. (10 de 10 de 2016). *¿Qué es el servicio FTP – File Transfer Protocol?* Obtenido de <https://www.internetya.co/que-es-el-servicio-ftp-file-transfer-protocol/>
- ISDI. (2014). *La herramienta que te permite realizar prototipos de tus proyectos: Balsamiq*. Obtenido de [isdi.education](https://www.isdi.education): <https://www.isdi.education/es/isdigital-now/herramienta-te-permite-realizar-prototipos-de-tus-proyectos-balsamiq>
- Isoftwareunesum. (04 de 2011). *Reingeniería de la Ingeniería del Software*. Obtenido de <https://isoftwareunesum.wordpress.com/2011/04/28/reingenieria-de-la-ingenieria-del-software/>
- Kemp, R. (2009). Current developments in Open Source Software. *Computer Law & Security Review*, 25(6), 569-582.
- M, D. (2010). HTML 5 JavaScript Model. *Focalpress*, 209-240.
- March, J. G., & Sutton, R. I. (1997). Crossroads--Organizational Performance as a Dependent Variable. *Organization Science*, Vol 8(issue 6), pags 698-706. Recuperado el nov de 2017, de <https://econpapers.repec.org/RePEc:inm:orosc:v:8:y:1997:i:6:p:698-706>
- March, J., & Sutton, R. (1997). Organizational Performance as Dependet Variable. *Organization Science*, pp 698-706. Obtenido de <https://pubsonline.informs.org/doi/pdf/10.1287/orsc.8.6.698>
- Matilla, M. M., & Chalmeta, R. (2007). Metodología para la implantacion de un sistema de Medicion de rendimeitno empresarial. *Información Tecnológica*, 18((1)), 119-126. Obtenido de [https://scielo.conicyt.cl/scielo.php?script=sci\\_arttext&pid=S0718-07642007000100016](https://scielo.conicyt.cl/scielo.php?script=sci_arttext&pid=S0718-07642007000100016)
- Mejía, G. (2007). Seguimiento de la Productividad en Obra: Técnicas de Medición de Rendimientos de Mano de Obra. *UIS Ingenierías*, Vol. 6(Issue 2), 45-59. Obtenido de <http://search.ebscohost.com/login.aspx?direct=true&db=zbh&AN=36074334&site=ehost-live>
- Mesbah, A. (2014). Advances in testing JavaScript-Based Web applications. *Advances in computers*, 201-235.
- Monty Programing. (2017). *MariaDB*. Obtenido de <http://www.mariadb.com>

- NeoAttack. (2018). *Filezilla*. Obtenido de <https://neoattack.com/neowiki/filezilla/>
- OpenSource. (marzo de 2017). *Most popular programing languages*. Obtenido de <http://www.opensourceforu.com/2017/03/most-popular-programing-lenguajes>
- Oracle. (2017). *About MySQL*. Obtenido de <http://www.mysql.com/about>
- Pressman, R. S. (2010). *Ingeniería de software: Un enfoque práctico*. México DF: Mc Graw Hill.
- Prokofyeva, N., & Boltunova, V. (2017). Analysis and Practical Application of PHP Frameworks in Development of Web Information Systems. *Procedia Computer Science*, 104, 51-56.
- RAE. (25 de nov de 2017). *Real Academia de la Lengua Española*. Obtenido de Productividad: <http://dle.rae.es/?id=UH8mXZv>
- Rojas, Á. H., Eduardo, E., & Gaona, G. (2001). Modelo de productividad de David Sumanth aplicado a una empresa del sector de maquinaria no eléctrica. *Dialnet*, Vol 6(issue 2), pags 81-87. Obtenido de <https://dialnet.unirioja.es/servlet/articulo?codigo=4797261>
- Rubén, A. (2014). *¿Qué es y para qué sirve el dominio de tu página web?* Obtenido de <https://computerhoy.com>: <https://computerhoy.com/noticias/internet/que-es-que-sirve-dominio-tu-pagina-web-22007>
- Samra, J. (2015). Comparing Performance of Plain PHP and Four of Its Popular Frameworks. *DIVA*.
- Santamaría, P. (2013). *Balsamiq mockup, una muy buen herramienta para esbozar tus futuras apps*. Recuperado el 09 de 2018, de <https://www.applesfera.com/aplicaciones-os-x-1/balsamiq-mockup-una-muy-buen-herramienta-para-esbozar-tus-futuras-apps>
- Shneiderman, B., & Plaisant, C. (2005). *Desingning the User Interface: Strategies for effective Human-computer Interacction*. Pearson.
- Sierra, F., Acosta, J., Ariza, J., & Salas, M. (Octubre de 2013). Estudio y análisis de los framework en php basados en el modelo vista controlador para el desarrollo de software orientado a la web. *Revistas Científicas Universidad Simón Bolívar*.
- Silva, P. C. (2012). O Sistema PJe- Procesos Judicial Electrónico. *Sistemas Judiciales*, 17. Obtenido de <http://sistemasjudiciales.org/content/jud/archivos/notaarchivo/893.pdf>
- Sparxsystems. (2017). *Enterprice Architect*. Obtenido de [sparxsystemos.com](http://www.sparxsystems.com): <http://www.sparxsystems.com.ar/>
- Talend. (2017). *Open Source ETL: Talend Open Studio for data integration*. Obtenido de <https://es.talend.com/products/data-integration/data-integration-open-studio/>
- TAYLOR OTWELL. (2017). *Laravel*. Obtenido de Documentation: <https://laravel.com/docs/5.4>
- TCA del Estado de Colima. (2017). Recuperado el 24 de 05 de 2017, de [tcacolima.gob.mx](http://tcacolima.gob.mx)
- TCAGTO. (07 de 2017). *Servicios Informáticos*. Obtenido de Tribunal de Justicia Administrativa del Estado de Guanajuato: [http://tcagto.gob.mx/?page\\_id=4015](http://tcagto.gob.mx/?page_id=4015)

- TCASIN. (2017). *Juicion en Linea Sinaloa*. Obtenido de Tribunal de lo Contencioso Admnsitrativo del Estado de Sinaloa:  
<https://juicioenlineasinaloa.gob.mx>
- Trent, S., Tatsubori, M., Suzumura, T., Tozawa, A., & Onodera, T. (2008). Performance Comparison of PHP and JSP as Server-Side Scripting Languages. *Middleware 2008*, 164-182.
- Vázquez, E. R., & Estrada, S. (2016). Modelo de medición del desempeño en el proceso de transferencia de conocimiento y tecnología. Casos de estudio. *Gestion y Estrategia*(issue 50), p 67-85. Obtenido de <http://search.ebscohost.com/login.aspx?direct=true&db=zbh&AN=121727407&site=ehost-live>
- Vela, B., Cavero, J. M., & Marcos, E. (2001). *Diseño de Base de Datos Objeto-Relacionales*. JIISIC.
- Wolfgang, J. (2011). *Diseño y Desarrollo del sistema de información judicial para centros judiciales integrados del órgano judicial de El Salvador y su implantacion en el centro judicial integrado de Soyapango*. El Salvador: Gohetals Consultirng corp. Obtenido de [https://www.goethalsconsulting.com/sites/default/files/casos/caso\\_de\\_exito\\_Soyapango\\_El\\_Salvador.pdf](https://www.goethalsconsulting.com/sites/default/files/casos/caso_de_exito_Soyapango_El_Salvador.pdf)
- WorkMeter. (2018). *EffiWork: Programa de eficiencia*. Obtenido de <https://web.workmeter.com/es/effiwork-programa-eficiencia.html>
- Zend Engine. (2015). *PHP*. Obtenido de <http://www.zend.com/en/company/community/php>

## Anexos

### 1. Artículo de publicación Científica

#### REINGENIERÍA DEL SISTEMA DE SEGUIMIENTO DE DEMANDAS DEL TRIBUNAL DE JUSTICIA ADMINISTRATIVO REENGINEERING OF THE SYSTEM FOR FOLLOW-UP OF CLAIMS OF THE COURT OF ADMINISTRATIVE JUSTICE

**Andrés Hernández Jiménez**

Estudiante de Maestría del Tecnológico Nacional de México,  
Instituto Tecnológico de Colima, México.

E-mail: [dreamsroom3@gmail.com](mailto:dreamsroom3@gmail.com) ORCID: <https://orcid.org/0000-0002-5958-2098>

**Ramona Evelia Chávez Valdez**

Profesor adjunto de la División de Estudios de Posgrado del Tecnológico Nacional de México,  
Instituto Tecnológico de Colima, México.

E-mail: [echavez@itcolima.edu.mx](mailto:echavez@itcolima.edu.mx) ORCID: <https://orcid.org/0000-0002-5697-6825>

**J. Reyes Benavides Delgado**

Profesor adjunto de la División de Estudios de Posgrado del Tecnológico Nacional de México,  
Instituto Tecnológico de Colima, México.

E-mail: [rbenavides@itcolima.edu.mx](mailto:rbenavides@itcolima.edu.mx) ORCID: <https://orcid.org/0000-0002-6190-5933>

**Patricia Elizabeth Figueroa Millán**

Profesor adjunto de la División de Estudios de Posgrado del Tecnológico Nacional de México,  
Instituto Tecnológico de Colima, México.

E-mail: [patricia.figueroa@itcolima.edu.mx](mailto:patricia.figueroa@itcolima.edu.mx) ORCID: <https://orcid.org/0000-0001-7562-7578>

**Recepción:** 06/02/19 **Aceptación:** 06/06/19 **Publicación:** 14/06/19

#### **Citación sugerida:**

Hernández Jiménez, A., Chávez Valdez, R. E., Benavides Delgado, J. R. y Figueroa Millán, P. E. (2019). Reingeniería del sistema de seguimiento de Demandas del Tribunal de Justicia Administrativa. *3C Tecnología. Glosas de innovación aplicadas a la pyme*, 8(2), pp. 12-35. doi: <http://dx.doi.org/10.17993/3ctecno/2019.v8n2e30.12-35>

## RESUMEN

En esta investigación se realizó el proceso de reingeniería para corregir fallos y aplicar mejoras al Sistema de Seguimiento de Demandas (SiSeDe) implantado en el Tribunal de Justicia Administrativa del estado de Colima, México. Se trata de un sistema de información con características web que facilita el seguimiento a los proyectos de sentencias, enfocándose en un módulo de indicadores de productividad para mejorar los tiempos de atención a los juicios de carácter administrativo. Para controlar el proceso de ingeniería de software se empleó el Proceso Unificado Ágil, por sus características para manejar proyectos pequeños, cambiantes y de cierto riesgo. Los resultados residen en un software que presenta con elementos estadísticos el tiempo de atención a la demanda, considerando métricas de evaluación de procesos por tiempo además de las establecidas por la Ley Administrativa del estado de Colima, México. Los actores responsables de la impartición de justicia administrativa, con apoyo de las señales de semáforos, pueden visualizar en el sistema la productividad de los actores que intervienen en la demanda, identificando cuántas demandas han atendido y cuántas se tienen en prevención o en rezago. Estos datos permiten a la dependencia incorporar estrategias para mejorar la atención de los procesos de justicia.

## ABSTRACT

*In this research, the process of reengineering was carried out to correct faults and apply improvements to the Follow-up System of Demands (SiSeDe) implanted in the Administrative Court of the State of Colima, Mexico. It is an information system with web features that facilitates the follow-up of sentencing projects, focusing on a module of productivity indicators to improve the time of attention to administrative trials. To control the software engineering process, the Agile Unified Process was used, due to its characteristics to handle small, changing and risky projects. The results reside in a software that presents the time of attention to the demand with statistical elements, considering metrics of evaluation of processes by time in addition to those established by the Administrative Law of the State of Colima, Mexico. The actors responsible for administering administrative justice, with the support of signals like traffic lights, can visualize in the system the productivity of the actors that intervene in the demand, identifying how many demands they have attended and how many are in prevention or in lag of attendance. These data allow the administrative personal to incorporate strategies to improve the attention of the justice processes.*

## PALABRAS CLAVE

Indicadores, Juicios, Productividad, Reingeniería, Seguimiento.

## KEYWORDS

Follow-up, Indicators, Productivity, Reengineering, Trials.

## 1. INTRODUCCIÓN

Cuando se obtiene un producto de tecnología se asumen riesgos en garantía del mismo, uno de los puntos que más se observan en el desarrollo del software es la obsolescencia del mismo, es decir, comienzan las fallas, existen errores que tardan en repararse y la solución vigente deja de ser la solución actual para la problemática para la que se adquirió (Sommerville, 2005). Por ejemplo, un software personalizado es un elemento del que no tan fácilmente se dispone o se puede dejar de utilizar, por lo tanto, es necesario reconstruirlo, con la finalidad de crear un producto con una funcionalidad nueva, que presente un mejor rendimiento y fiabilidad, que tenga un mantenimiento mejorado, esto es a lo que se conoce como reingeniería (Sicilia & De la Morena, 2009).

Un software personalizado es un elemento del que no tan fácilmente se dispone o se puede dejar de utilizar, por lo tanto, es necesario reconstruirlo, con la finalidad de crear un producto con una funcionalidad nueva.

El principal enfoque que se aplica a la reingeniería es aquel que está dirigido a las actividades de mantenimiento: El entendimiento (predecir), la reparación (corregir), mejoramiento (perfeccionar) y la evolución (adaptar) (Olsem, Sittenauer, Dawood, & Rasmussen, 1995). De la misma forma, Sicilia (2009) establece que dentro de los beneficios que aporta la reingeniería a un producto se encuentran:

- Reducir riesgos evolutivos en la organización.
- Ayudar a recuperar las inversiones en software.
- Hacer que el software sea más fácilmente modificable.
- Catalizar la automatización del mantenimiento del software.

Para la implementación de mejoras en un sistema se deben considerar cuatro preguntas esenciales : ¿Qué transformar?, ¿Por qué transformar?, '¿Cómo transformar?' y ¿Sobre qué transformar?; la primer pregunta se rige sobre los elementos que definen el proceso realizar con ayuda de los diferentes procesos de reingeniería ( ingeniería inversa, re documentación, estructuración y modernización), la segunda se refiere al objetivo por el que se hace la mejora, la tercera se enfoca a las técnicas utilizadas para realizar la transformación, y la final se refiere al producto sobre el cual se aplicarán las mejoras (Jürgen, Franz, Volker, & Andreas, 2005). Teniendo esto en cuenta, se puede estimar que el proceso de reingeniería de un producto de software atrae beneficios a la organización, ya que puede incrementar la funcionalidad del sistema.

Por otro lado, otro aspecto que define a las empresas es su productividad, entendida como aquella que establece la relación entre lo producido y los medios empleados, tales como la mano de obra, materiales, energía, etc. (RAE, 2017); en cambio, en el sentido de una empresa, March & Sutton (1997) determinan que el desempeño de toda organización se presenta en forma de un constructo clave en la dirección



estratégica de la misma, por lo que se puede establecer que en toda empresa u organización uno de los principales fines es el conducirse hacia la eficiencia organizacional; de igual forma, Mejía (2007) menciona que este factor puede contabilizarse con la ayuda de indicadores que sirvan de referencia para identificar puntos clave que requieran modificarse, con un enfoque en los factores que incidan en el desempeño de procesos, tales como la calidad, seguridad, costos, tiempos, la planeación y el control, esto con el fin de alcanzar el desarrollo óptimo de la misma. Considerando lo establecido en la página Workmeter (2012) los indicadores de productividad son aquellas variables que nos ayudan a identificar algún defecto o imperfección que exista en el proceso en el que elaboramos un producto o se ofrece un servicio, lo que refleja la eficiencia en el uso de los recursos generales y recursos humanos de la empresa, estos pueden ser cualitativos o cuantitativos, la fórmula del cálculo de la productividad es el cociente entre la producción y el consumo, todo acorde al tipo de empresa a la que se está refiriendo.

Los aspectos anteriores fundamentan las razones por las cuales se establece la necesidad de aplicar la reingeniería al producto desarrollado por Fernández & Moreno (2017), el Sistema de Información para el Seguimiento de las Demandas (SiSeDe) con el que el Tribunal de Justicia Administrativa del Estado de Colima (TJA) antes conocido como Tribunal de lo Contencioso Administrativo del Estado de Colima, México, realiza sus actividades, esto con la finalidad de mejorar la sistematización de los procesos implementados y medir la productividad del organismo en la impartición de justicia. Esta dependencia se encarga de gestionar los juicios administrativos que interponen los ciudadanos por la no conformidad con la prestación de algún servicio asignado a dependencias gubernamentales con ámbito local o estatal.

Para la reestructuración del SiSeDe se utilizaron tecnologías de información que facilitarían la reingeniería, orientadas al trabajo de migración de datos, de asistencia y de diseño, de pruebas y de validación del sistema, así como la integración de indicadores que miden tiempos de proceso para crear un módulo que evalué la productividad de los actores que intervienen en el seguimiento de la demanda.

Para la integración de indicadores se establecieron aquellos que evalúan tiempos de proceso como lo es la métrica de rango DAU/MAU (por sus siglas en inglés de Monthly Active Users y Daily Active Users), las cuales se definen como la Proporción de Usuarios Activos Diarios (DAU) a Usuarios Activos Mensuales (MAU) y que mide la rigidez de su producto, es decir, la frecuencia con la que la gente interactúa con un producto (GECKOBOARD, 2017), otra de ellas es la que se establece en la Ley de Justicia Administrativa del Estado de Colima, México (H. Congreso del Estado de Colima, 2018) y finalmente aquellas establecidas por los propios elementos del tribunal.

En referencia a los procesos de migración y ordenamiento de datos, para los cuales existe el término ETL, entendido como el proceso de Extraction, Transformation and Load (Extracción, Transformación y Carga) (Microsoft, 2018), permite obtener los antecedentes de una o más fuentes de almacenamiento, procesarlos, reformatearlos y/o limpiarlos y almacenarlos nuevamente en una base de datos;

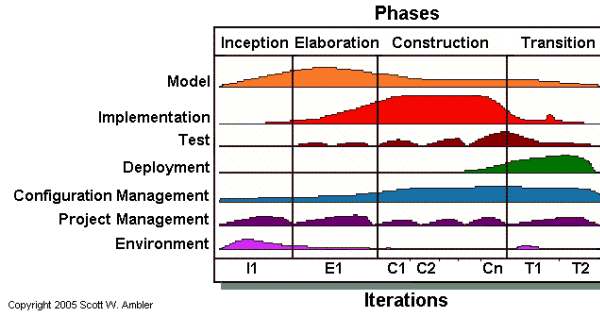
dicho proceso fue asistido por la herramienta Talend Open Studio (Talend, 2018) , esta última es una herramienta de código abierto desarrollada bajo el lenguaje de programación Java® , que permite realizar los procesos ETL. Durante las primeras etapas de diseño de una nueva interface se utilizó una versión de prueba de un software de ayuda de diseño denominada Balsamiq Muckups, ya que ofrece la misma velocidad y la misma sensación que dibujar con un lápiz, pero con la ventaja del medio digital: arrastrar y soltar para cambiar el tamaño y reorganizar elementos, hacer cambios sin volver a empezar desde cero (Balsamiq Studios ,LLC, 2018). El uso de esta herramienta permite el desarrollo de trazos o plantillas que sirven como referencia para plasmar las ideas de diseño tanto de diseñadores o de usuarios, las cuales funcionan como reseña para que el programador transforme esas ideas y bocetos a una interface que estará cercana a la realidad de lo que los clientes esperan. Para finalizar, las pruebas que se realizaron al producto de reingeniería se basaron en aquellas aplicadas en el proceso de desarrollo de software, denominadas pruebas de caja blanca y caja negra, Sommerville (2005) define que las pruebas en un software realizadas sobre las funciones internas de un módulo son aquellas conocidas como pruebas de caja blanca, en cambio, aquellas que se enfocan en los requisitos funcionales desde el exterior del módulo o a un conjunto mayor de módulos en lo que se considera la integración final del sistema, para esto se utilizó una herramienta incluida en el framework de programación denominada PHPUnit (Bergmann, 2018) en su versión 4.8 que permitió realizar todas las pruebas unitarias y de integración del sistema.

## 2. METODOLOGÍA

El proceso de reingeniería se basó en la metodología del Proceso Unificado Ágil (PUA), la cual es una versión simplificada de la metodología de Proceso Racional Unificado, en donde se describe un enfoque simple y fácil de entender para el desarrollo de software de aplicaciones de negocio utilizando técnicas y conceptos ágiles, las cuales incluyen el desarrollo guiado por pruebas, el modelado ágil , la gestión de cambio ágil y la refactorización de base de datos para mejorar la productividad del producto informático (Ambler, 2014).

Acorde a lo establecido por (Ambler, 2014), la metodología PUA que presenta la figura 1, consta de cuatro fases que el proyecto atiende de forma iterativa e incremental:

1. Iniciación: se identifica el alcance inicial del proyecto, una arquitectura potencial para el sistema y obtener, si procede, financiación para el proyecto y la aceptación por parte de los promotores del sistema.
2. Elaboración: esta fase permite identificar y validar la arquitectura del sistema.
3. Construcción: el objetivo de esta fase consiste en construir software desde un punto de vista incremental basado en las prioridades de los participantes.
4. Transición: en esta fase se valida y despliega el sistema en el entorno de producción.



**Figura 1** Ciclo de vida de la metodología PUA. **Fuente:** <http://www.ambysoft.com/unifiedprocess/agileUP.html>, 2005.

A lo largo de las cuatro fases, se desarrollan actividades relativas a siete disciplinas de manera iterativa:

- **Modelado:** su objeto es entender la lógica de negocio de la aplicación, el dominio del problema del proyecto e identificar una solución viable para el dominio del problema.
- **Implementación:** transformar los modelos en código ejecutable y realizar pruebas básicas, en particular pruebas unitarias.
- **Pruebas:** realizar una evaluación de los objetivos para asegurar la calidad; incluye encontrar defectos, validar que el sistema funciona como fue diseñado y verificar que los requisitos se cumplan.
- **Despliegue:** planear la entrega del sistema y ejecutar el plan para hacer que el sistema quede disponible para los usuarios finales.
- **Gestión de la configuración:** gestionar el acceso a los artefactos del proyecto. Esto incluye, además de la traza de versiones de los artefactos, el control de cambios y la gestión de los mismos.
- **Gestión del proyecto:** dirige las actividades que tienen lugar dentro del proyecto, incluyendo gestión de riesgos, dirección del personal y coordinación.
- **Entorno:** apoyar el resto del esfuerzo asegurando que los procesos, métodos y herramientas están disponibles para el equipo cuando los necesitan.

Esta metodología permitió controlar el proceso de ingeniería de software, así en la fase de iniciación se determinó el alcance y la arquitectura del sistema mediante la recopilación de los requerimientos, se establecieron además las prioridades en los casos de uso que determinan las funcionalidades requeridas en el sistema; en la fase de elaboración se realizaron los modelados de clases, datos e interfaces, que corresponden a los procesos de reingeniería del sistema y otros a la inclusión de los indicadores de productividad del proceso de seguimiento de demanda, de acuerdo a las iteraciones definidas previamente; los modelos elaborados fueron implementados y probados en la fase de construcción donde se realizó la codificación necesaria para cumplir con los requerimientos, tanto actuales como anteriores del sistema, realizando pruebas de funcionalidad a cada una de las actividades programadas. Finalmente, en la fase de transición se desplegaron los resultados en un medio local para posteriormente ser colocados en el servidor de producción.

La metodología PUA permitió controlar el proceso de ingeniería de software, desde la fase de iniciación hasta la fase de transición.

### 3. RESULTADOS

#### 3.1. El proceso de Reingeniería

Para realizar el proceso de reingeniería se consideró la normatividad del TJA, la documentación del Sistema de Seguimiento de Demandas (SiSeDe) – reporte técnico, manual de usuario y código - a disposición del TJA. La principal razón para realizar el proceso de reingeniería fue el cambio estructural que recibió el TJA en razón a la estructura orgánica recientemente aprobada, la cual pasó de ser un organismo con estructura simple ( un magistrado para legislar y gestionar la legalidad y jurisprudencia en la dependencia) a una estructura compuesta (múltiples magistrados que rigen y gestionan la legalidad y jurisprudencia en la dependencia) lo cual hizo necesaria la reingeniería del sistema de gestión SiSeDe, incorporando un módulo de indicadores de productividad dirigido a los magistrados para facilitar su trabajo.

Otros de los aspectos que justifican la reingeniería son los encontrados con base al análisis del SiSeDe desarrollado por Fernández y Moreno (2017), mismos que reflejan la estructura del proceso de demanda, que implica el cambio del modelo organizacional de la institución, la liberación del proceso de seguimiento de demandas en línea en el estado de Colima, así como validaciones de los procesos de almacenamiento de datos, y finalmente la petición propia de los interesados de aplicar una mejora visual a las interfaces para hacer el sistema más intuitivo e interactivo al usuario.

Siguiendo los procesos de la reingeniería, para ser más específico, con ayuda de la ingeniería inversa, con base al cambio de requerimientos y de la estructura orgánica del TJA, así como la implementación de mejoras, presentamos la propuesta descrita en este artículo.

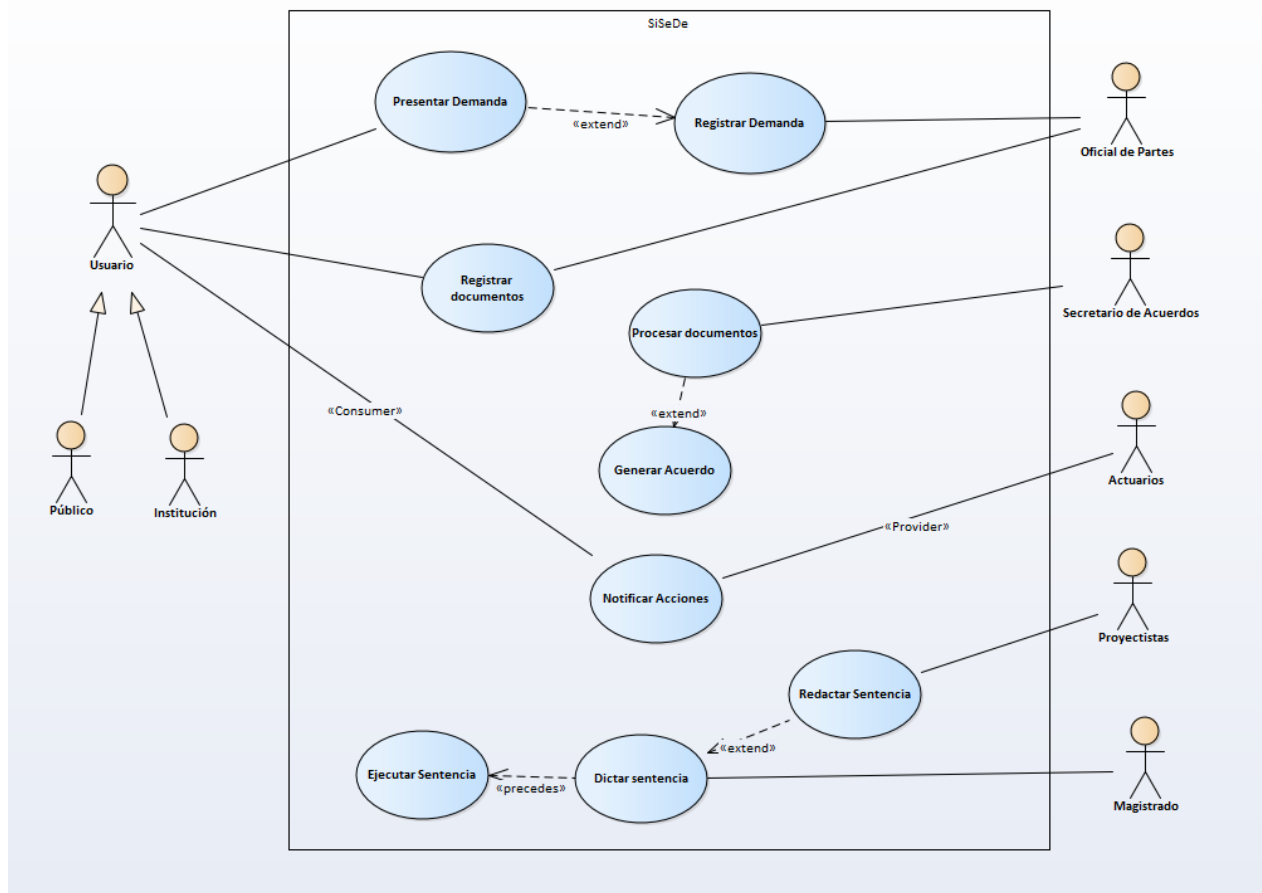
#### 3.2. Reingeniería de procesos

Apegados a la metodología, la fase de iniciación incluyó la reingeniería de procesos, donde se tomaron en cuenta los requerimientos establecidos en el sistema anterior, estos se compararon con el levantamiento de requisitos actuales, a fin de definir el alcance que permitiera elaborar los casos de uso que representan el comportamiento del sistema.

Aunado a esto, se recopiló información con ayuda de entrevistas individuales, asesoramiento, observación y la lectura de la documentación que se obtuvo del TJA, lo cual dio lugar a establecer estas premisas:

- El sistema debe permitir la correcta distribución de cargas de trabajo, esto está justificado en el cambio estructural del TJA, en el cual se cuenta con más personal del considerado en la versión anterior.

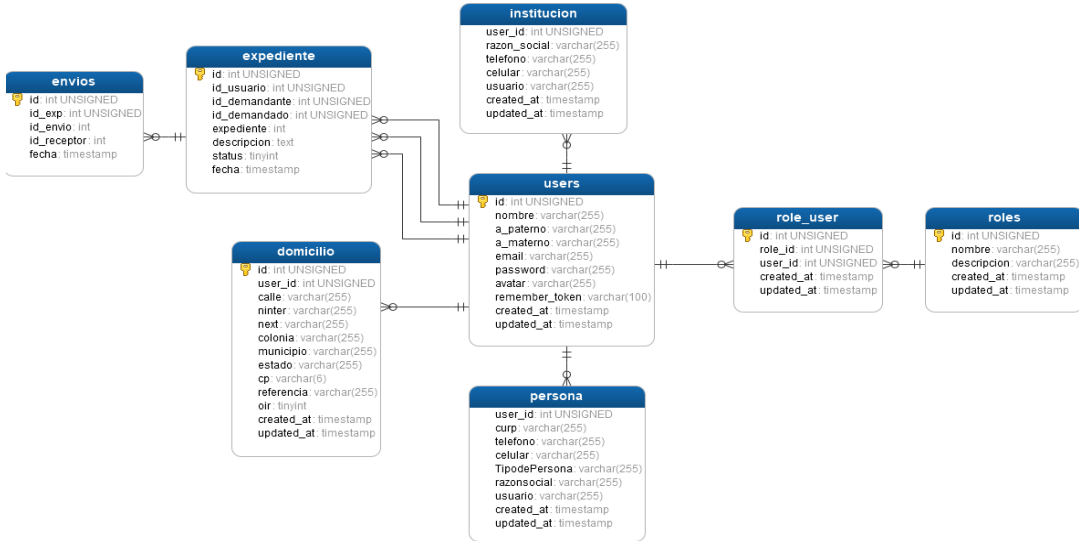
- El sistema debe simplificarse, evitar el uso excesivo de clics y que las interfaces sean más intuitivas al usuario, referenciando los medios de registro, navegación, y uso general del sistema.
- Se hace necesario regularizar y habilitar el sistema para la impartición de justicia en línea, permitiendo ingresar al sistema tanto a los trabajadores como a los usuarios interesados en el proceso de demanda, asegurándose que el uso del programa sea conforme a los procedimientos del proceso presencial en el TJA.
- Se debe considerar una nueva clasificación en las demandas en la estructura de registro del sistema, aquellas de proceso simple como por ejemplo las resoluciones de multas de tránsito, y aquellas de proceso complejo que requieren mayor investigación, tiempo y recursos para solventarlas, lo que permite distribuir de forma correcta el punto anterior.
- EL sistema debe permitir que las notificaciones y resoluciones de cada uno de los procesos puedan realizarse ya sea por medio físico o por el medio digital.
- Estas premisas están modeladas en el diagrama de casos de uso que se muestra en la figura 2, como se observa se presentan el escenario general del funcionamiento del SiSeDe, donde el usuario puede ser público (demandante) o institución, quien presenta la demanda y la registra en el sistema; por su parte el Oficial de partes realiza la verificación de documentos y valida dicho registro. Una vez registrada la demanda el Secretario de Acuerdos revisa los documentos, los analiza y fundamenta la creación de un acuerdo, éste es notificado por el Actuario a los usuarios demandantes. Una vez que se cumplan los requisitos establecidos en los acuerdos se procede a generar un dictamen de sentencia, en el intervienen los Proyectistas con apoyo de los Magistrados, siendo éstos últimos los que tienen la decisión final sobre el veredicto de la sentencia, procediendo así a la ejecución de la misma.



**Figura 2** Diagrama de casos de uso del flujo de la demanda en el sistema SiSeDe. **Fuente:** elaboración propia.

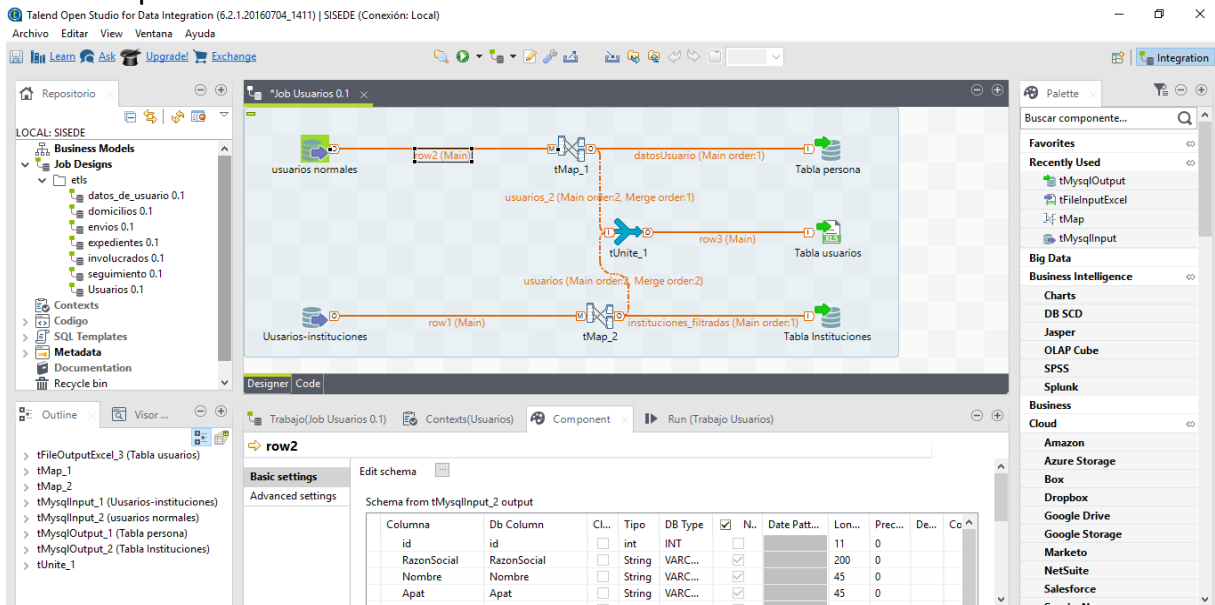
### 3.3. Reingeniería en el modelo de datos

La reingeniería en el modelo de datos implicó ajustes considerando las modificaciones estructurales de la organización y los procedimientos que marca la ley (H. Congreso del Estado de Colima, 2018) , se establecieron las nuevas especificaciones en la base de datos; la figura 3 muestra un fragmento del nuevo modelo, en donde sobresale la mejora en la estructura, ya que se añaden los procedimientos y funciones que conforman los indicadores de tiempo estipulados para medir la productividad del seguimiento de las demandas, esto se observa desde las tablas expediente, personas e instituciones, donde al incluir marcas de tiempo se hace posible la implementación del módulo de indicadores para determinar la semaforización de actividades de cada actor. Así, una vez colocada la demanda, se podrá medir cuanto tiempo estuvo en atención del oficial de parte, del secretario de acuerdos, del proyectista, entre otros actores. Dicho tiempo se contrastará con lo indicado por la ley y se emitirá el semáforo con su color correspondiente; esto dará lugar a crear estrategias para que la demanda avance a su siguiente estado.



**Figura 3** Fragmento del modelo de base de datos después de la Reingeniería **Fuente:** elaboración propia.

Por otra parte, se tuvo especial cuidado en la migración de los datos, ya que al realizar una reestructuración de un sistema es necesario visualizar la necesidad del mantenimiento al mismo, esto incluye la información existente en él. El proceso de transformación de datos se separó en trabajos enfocados a secciones específicas de nuestro diseño de base de información como se puede ver en la figura 4, la cual representa la transformación de la información de los usuarios registrados en el sistema anterior para acoplarse al nuevo modelo con el apoyo de la herramienta Talend Open Studio en su versión 6.2.1.\*.

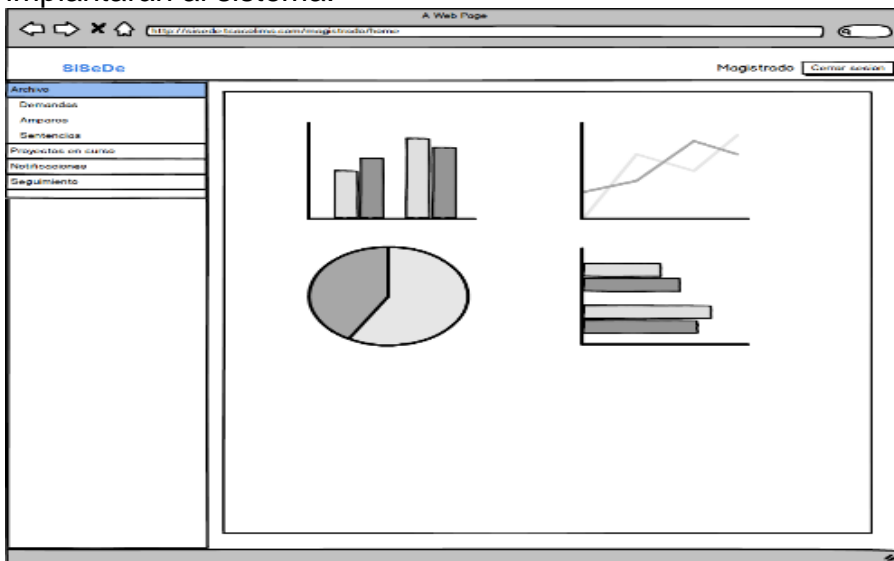


**Figura 4** Trabajo ETL de integración de los datos de la tabla Usuarios. **Fuente:** elaboración propia.

### 3.4. Reingeniería de las interfaces de Usuario (GUI) –

Para definir la nueva estructura de las interfaces de usuario se tomaron en consideración los requerimientos y solicitudes de los usuarios, los cuales

establecieron sus preferencias hacia una reestructuración general de la visualización, que permita navegar en la aplicación de una forma más intuitiva, que sea amigable y a la vez responsiva, lo que permita el despliegue de la plataforma en múltiples dispositivos, para ello se utilizó el software Balsamic Muckups, y resultado de esto en la Figura 5 se observa el boceto de cómo el administrador general podrá visualizar los resultados estadísticos de los indicadores que se implantarán al sistema.

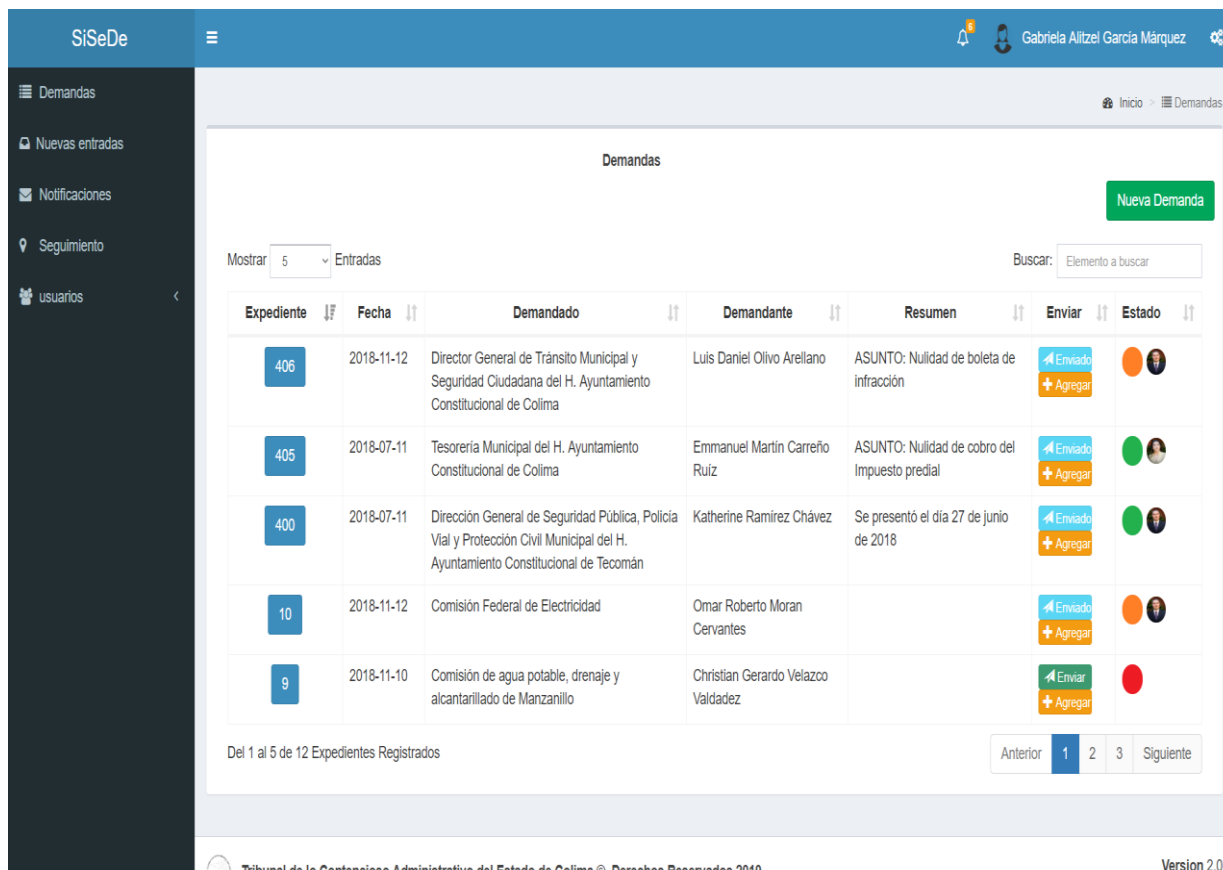


**Figura 5** Boceto-Página de inicio del módulo de Administrador. **Fuente:** Elaboración propia.

### 3.5. Reingeniería en modificación y nueva función del programa.

La reingeniería aplicada al SiSeDe tuvo como alcance la modificación del producto con mejoras en el Registro de nuevas demandas como se muestra en la Figura 6, ahí el oficial de partes puede visualizar el histórico de las demandas, así como el monitoreo del estatus de cada una de los casos en revisión con la implementación de un semáforo de color, la cual indica mediante los colores verde, naranja y rojo, el estado que guarda la atención de la demanda.



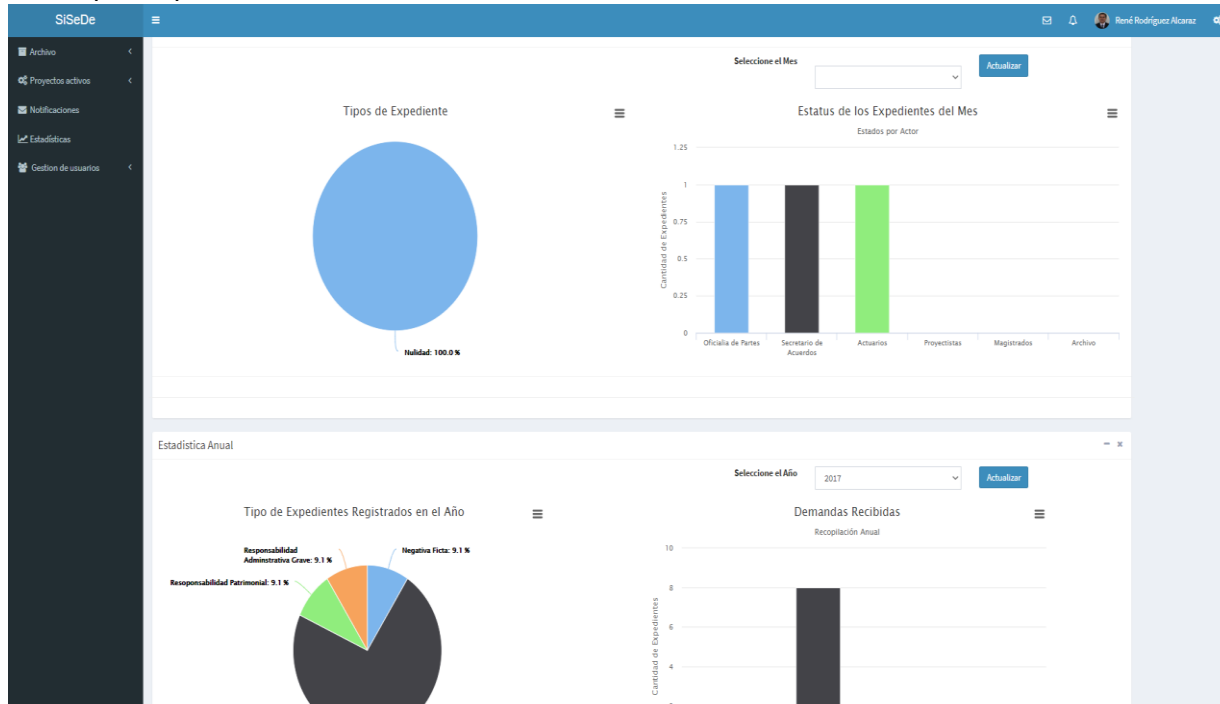


**Figura 6** Página de Gestión de demandas del módulo de oficial de partes. **Fuente:** elaboración propia.

En la Figura 6 se pueden reconocer la nueva estructura para presentar la información al usuario, desde una perspectiva tabular, esto incluye datos como el número de expediente, la fecha de ingreso al sistema, lo participantes de la demanda, el resumen general y como anexo principal los indicadores de semáforo los cuales estipulan de forma visual el estado activo de la demanda, así como el personal que tiene en posesión dicho expediente. Los indicadores de color simbolizan respectivamente el tiempo de proceso para cada folio acorde a lo que establece la ley, teniendo marcadores verdes cuando se encuentra dentro de lo establecido, naranja cuando se exceda el tiempo recomendado y rojo cuando el proceso se encuentre en un periodo de desfase mayor al indicado en la ley.

En la Figura 7 se observa la forma en la que los magistrados visualizan la información estadística que generan los indicadores dentro del proceso de seguimiento de demanda, dentro de estas especificaciones, el administrador de sistema o magistrado puede elegir

entre que visualizar: información por periodo de fecha, por expediente, tipo de expediente o tiempo de proceso de cada uno de los actores.



**Figura 7** Página de inicio del módulo de Magistrados. **Fuente:** elaboración propia.

Un aspecto importante de la nueva funcionalidad es la integración del módulo de medición de productividad, para la cual se utilizaron tres indicadores de medición de tiempo, los cuales son: la métrica de rango DAU/MAU, ésta puede ser fácilmente modificable para medir la cantidad de casos únicos que interactúan en el sistema en un período de 30 días; la segunda métrica se encuentra definida en la Ley de Justicia Administrativa de Colima en sus artículos 52, 60, 64, 67, 71 y 91 establecen las pautas de los tiempos que deberán de tomar los procesos de seguimiento de demanda, estableciendo que los procesos de recepción, verificación de documentos, preparación de acuerdos para desechar o admitir el recurso de reclamación y la notificación de los mismos; finalmente, se tomaron en cuenta aquellas especificaciones que los magistrados interpusieron para la evaluación de productividad de los actores.

En general la reingeniería estuvo enfocada en el módulo de indicadores de productividad, en la actualización del diseño de vistas para todos los usuarios y en la asignación de privilegios por perfiles de usuario; así el nuevo módulo permite dar seguimiento puntual a la atención de las demandas, y el diseño de vistas favoreció la usabilidad del sistema por parte de los actores.

### 3.4. Pruebas Aplicadas al Producto

Acorde a las especificaciones de prueba del sistema, que manejan algunos autores, estas se generalizan en dos, la primera es cuando se conoce la función específica del diseño, a la cual se aplican pruebas que demuestren que cada función es

plenamente operacional y se buscan y localizan los posibles errores de la misma; y la segunda es si se conoce el funcionamiento interno, esto produce pruebas en las que se asegura la correcta integración de cada módulo del sistema asegurando que se cumplan las especificaciones del mismo.

Para todo esto, con la ayuda de la herramienta PHPUnit se diseñaron e implementaron pruebas tanto unitarias enfocadas en acreditar el correcto funcionamiento de un módulo de código, clase o funcionalidad programada como se observa en la Figura 8, en donde se realiza la revisión de la clase que genera los correos electrónicos; en cambio cuando se requiere realizar una prueba de cómo funciona un conjunto de elementos unitarios para verificar que las partes del software funcionan correctamente, por ejemplo el evaluar los eventos de entrega de documento o avance dentro del proceso de seguimiento de la demanda como se observa en la Figura 9.

```
C:\xampp\htdocs\laravel>phpunit --group message
PHPUnit 4.8.16 by Sebastian Bergmann and contributors.
.
Time: 2.12 seconds, Memory: 8.75Mb
OK (1 test, 2 assertions)
C:\xampp\htdocs\laravel>
```

Figura 8 Ejemplo de prueba unitaria. Fuente: elaboración propia.

```
C:\xampp\htdocs\DocumentaQro>phpunit --group paquetes
PHPUnit 4.8.16 by Sebastian Bergmann and contributors.
..
Time: 4.12 seconds, Memory: 15.75Mb
OK (2 tests, 8 assertions)
C:\xampp\htdocs\DocumentaQro>
```

Figura 9 Ejemplo de prueba de integración. Fuente: elaboración propia.

Dada la extensión del proyecto todas las pruebas unitarias y de integración se agruparon conforme a sus funcionalidades como se puede observar en la Tabla 1.

Tabla 1 Grupos de pruebas del sistema.

Grupos de Pruebas	
Unitarias	De Integración
Visualización de formularios	Escritura de información
Visualización de menús	Subida de documentos
Carga de vistas	Llenado de formularios
Lectura de información	Validación de datos
Generación de Correos de aviso	Envío de notificaciones
	Cálculo de Indicadores

#### 4. CONCLUSIONES

Con el presente desarrollo basado en la reingeniería del SiSeDe se cubrieron los nuevos requerimientos de la institución, se entrega una herramienta actualizada, desarrollada conforme lo establecido en las normativas de ley del TJA del Estado de Colima, México. Este sistema tiene la capacidad de dar atención integral al proceso de demanda, de manera interna al TJA, mientras se libera el juicio en línea. A diferencia de la primera versión del SiSeDe, este sistema otorga más funcionalidades que tienen que ver con el incremento de usuarios en un mismo rol, claro ejemplo de los magistrados; de igual manera, las funcionalidades de cada uno de los usuarios se vieron incrementadas, tanto para la gestión de sus labores dentro del proceso de seguimiento como para manejo y adaptación del sistema, entre estas la notificación automática vía correo electrónico y el envío del expediente electrónico entre los actores que atienden el proceso de la demanda. Por otro lado, la base de datos original, después de aplicar procesos de normalización, se redujo de 25 tablas a 19, permitiendo que los aspectos de eficiencia, flexibilidad de almacenamiento, recuperación ante errores, así como el escalamiento de alcances del sistema en un futuro inmediato se vean incrementados. Respecto a la visualización e interfaces de usuario, todas las vistas del sistema se ampliaron y modificaron, tomando en consideración un enfoque minimalista y usable, lo que permite al usuario tenga en primer 32 3C Tecnología. Glosas de innovación aplicadas a la pyme. ISSN: 2254-4143 plano toda la información prioritaria y con un fácil acceso, gracias a la integración de paneles de navegación, notificaciones y alertas que retroalimentan al usuario sobre sus acciones en el sistema. Para el desarrollo de este sistema se planteó la hipótesis que establece que el seguimiento informático a los indicadores de productividad establecidos, permite a la organización realizar evaluaciones periódicas de la eficiencia de sus procesos para mantener la mejora continua de la misma; dicha hipótesis aún se encuentra en proceso de validación, debido que el software se encuentra en su etapa de liberación.

La hipótesis permite a la organización realizar evaluaciones periódicas de la eficiencia de sus procesos para mantener la mejora continua de la misma.

#### Recomendaciones y puntos de mejora

Unos de los puntos de mejora será la necesidad de desarrollar e implementar un sello o firma digital a cada uno de los documentos que conforman los expedientes, esto aumentará el nivel de seguridad de los mismos y además integrarán un respaldo legal, representando una medida de legalidad y legitimidad en cada uno de los documentos. También será importante incorporar los audios de las reuniones del pleno, a fin de mantenerlas en un repositorio propio del SiSeDe. Pese a que se tiene la infraestructura en esta reingeniería del SiSeDe, queda pendiente la habilitación del juicio en línea, porque está sujeto a la liberación de la ley que lo declare válido; una vez habilitado el juicio en línea podría difundirse en redes

sociales, para que los usuarios demandantes tengan presente el medio donde puedan colocar sus demandas.

## REFERENCIAS BIBLIOGRÁFICAS

Ambler, S. W. (2014). The Agile Unified Process. Recuperado de: <http://www.ambysoft.com/unifiedprocess/agileUP.html>

Balsamiq Studios, LLC. (2018). Balsamiq Muckups. Recuperado de: <https://balsamiq.com/products/> Bergmann, S. (2018).

PHPUnit. Recuperado de: <https://phpunit.de/index.html>

Fernández, J. M., y Moreno, A. A. (2017). Diseño e Implementación de un sistema de información para el seguimiento de proyectos de sentencias en el tribunal de lo contencioso administrativo. 3C TIC, 6(2), pp. 1-27. doi: <http://dx.doi.org/10.17993/3ctic.2017.56.13-27>

GECKOBOARD. (2017). DAU/MAU ratio. Recuperado de: <https://www.geckoboard.com/learn/kpiexamples/startup-kpis/dau-mau-ratio/#.WiZBI0qWbIU>

H. Congreso del Estado de Colima. (16 de junio de 2018). Ley de lo Contencioso y Administrativo del Estdo de Colima. Recuperado de: Última Reforma Decreto 133: [http://congresocol.gob.mx/web/Sistema/uploads/LegislacionEstatal/LeyesEstatales/Justicia\\_Administrativa\\_16jun2018.pdf](http://congresocol.gob.mx/web/Sistema/uploads/LegislacionEstatal/LeyesEstatales/Justicia_Administrativa_16jun2018.pdf)

Jürgen, E., Franz, L., Volker, R., y Andreas, W. (2005). Informatik - Forschung und Entwicklung, 19(3), pp. 125-126. doi: <https://doi.org/10.1007/s00450-005-0183-y>

March, J., y Sutton, R. (1997). Organizational Performance as Dependet Variable. Organization Science, pp. 698-706. Recuperado de: <https://pubsonline.informs.org/doi/pdf/10.1287/orsc.8.6.698>

Mejía, G. A. (2007). Seguimiento de la Productividad en Obra: Técnicas de Medición de Rendimientos de Mano de Obra. UIS Ingenierías, 6(2), pp. 45-59. Recuperado de: <http://search.ebscohost.com/login.aspx?direct=true&db=zbh&AN=36074334&site=ehost-live>

Microsoft. (2018). Extract, transform, and load (ETL). Recuperado de: Microsoft Azure: <https://docs.microsoft.com/en-us/azure/architecture/data-guide/relational-data/etl>

Olsem, M. R., Sittenauer, C., Dawood, M., y Rasmussen, K. J. (1995). Reengineering Technology Report, 2. UTHA: Defense Technical Information Center. Recuperado de: <http://www.dtic.mil/dtic/tr/fulltext/u2/a324750.pdf>

Sicilia, M. A., y De la Morena, V. (2009). Técnicas de mantenimiento de Software. Huston, Texas: Rice University. Recuperado de: [https://cnx.org/contents/RunP\\_m1t@6.1:jXj8TA20@3/Qu-esReingeniera-del-Software](https://cnx.org/contents/RunP_m1t@6.1:jXj8TA20@3/Qu-esReingeniera-del-Software)

Sommerville, I. (2005). Ingeniería de Software. Ámsterdam: Pearson Education.

Talend. (2018). Open Source Integration Software. Recuperado de: Talend Open Studio: <https://www.talend.com/products/talend-open-studio/>

Workmeter. (12 de junio de 2012). Indicadores de productividad, ¿Qué son y cómo analizarlos? Recuperado de: El blog de WorkMeter; Consejos para hacer crecer tu negocio: <https://es.workmeter.com/blog/bid/172634/indicadores-de-productividad-qu-son-y-c-mo-analizarlos>