

UNIVERSIDAD TECNOLÓGICA DE AGUASCALIENTES



SISTEMA DE TRÁMITES EN LÍNEA

REPORTE DE ESTADÍA PROFESIONAL

QUE PARA OBTENER EL TÍTULO DE

TECNICO SUPERIOR UNIVERSITARIO EN TECNOLOGÍAS DE LA
INFORMACIÓN Y COMUNICACIÓN ÁREA DESARROLLO DE SOFTWARE
MULTIPLATAFORMA

PRESENTA:

RICARDO ANGELES RAMÍREZ

AGUASCALIENTES, AGS.

AGOSTO, 2020

PROYECTO REALIZADO EN SOFTWARE SIN LÍMITES SC

ASESOR:

ENRIQUE DE LEÓN DOMÍNGUEZ

TUTOR:

ALEJANDRO RODRÍGUEZ RAMÍREZ

Agradecimientos o Dedicatorias

Optativo

Tabla de contenido

Introducción	1
Capítulo I Generalidades de la empresa	2
Datos generales	3
Antecedentes Históricos.....	3
Misión	3
Visión	3
Valores.....	3
Productos o servicios que ofrece.....	3
Organigrama	3
Descripción del departamento donde se realizó la estadía	3
Capítulo II: Planteamiento de la Estadía Profesional	4
Descripción de la Problemática	5
Justificación	6
Viabilidad	7
Resultados esperados	7
Metodología a aplicar	8
Capítulo III. Desarrollo de la estadía profesional.	9
Requisitos para el desarrollo	10
Dependencias del API Restful.....	10
Roles del proyecto.....	11
Inicio de la estadía.....	12
Investigación Laravel	12
Investigación Bref/serverless	13
Cambio de <i>framework</i> Laravel -> React.....	14
Desarrollo del proyecto	15
Base de datos.....	15
Vista de títulos de la ley de ingresos	15
Vista de capítulos de la ley de ingresos	17
Vista de secciones de la ley de ingresos.....	18
Vista de artículos de la ley de ingresos	19
Vista de descuentos de la ley de ingresos	20
Vista de recargos de la ley de ingresos.....	21
Vista de conceptos de la ley de ingresos	22
Vista de año fiscal de ley de ingresos	23
Capítulo IV. Resultados y conclusiones	24

Descripción de los resultados obtenidos conforme a los objetivos planteados en el proyecto:	25
Conclusiones.....	26
Recomendaciones.....	27
Anexos	28
Cronograma de actividades.....	28
Carta de Liberación en hoja membretada.....	29
Referencias Bibliográficas	30

Introducción

En el documento se presentan los datos generales y específicos de la estadía cuyo objetivo es el diseñar, crear y probar una plataforma para el control de trámites para el municipio de Jesús María.

Se eligió a la empresa ya que había interés en conocer las herramientas que usa esta empresa, como lo son la programación usando el framework laravel que es una herramienta de la que no conozco mucho y sería muy interesante aprenderla, además del hecho de que se trabajaría con la metodología de Scrum, además de que se usa tecnología Serverless por parte de los servicios que provee AWS.

En el reporte se presentarán los datos generales del desarrollo del proyecto, primero se presentarán los datos de la empresa en la que se realiza la estadía y en que consiste el proyecto.

Se anexarán las actividades e investigaciones realizadas durante la elaboración del proyecto, después se presentarán las conclusiones y los resultados obtenidos al finalizar la estadía.

Capítulo I Generalidades de la empresa

ENCABEZADO

Datos generales

La empresa se encuentra ubicada en el Mar caspio 631 en el fraccionamiento colinas del rio, Aguascalientes, Ags. Actualmente cuenta con 11 empleados, la empresa se dedica a Información de software y comunicación de medios masivos.

Antecedentes Históricos

Misión

La empresa no cuenta actualmente con misión.

Visión

La empresa no cuenta actualmente con visión.

Valores

La empresa no cuenta actualmente con valores.

Productos o servicios que ofrece

Información de software y comunicación en medios masivos.

Organigrama

La empresa no cuenta con un organigrama

Descripción del departamento donde se realizó la estadía

Debido a contingencia se estuvo trabajando desde casa.

Capítulo II: Planteamiento de la Estadía Profesional

Descripción de la Problemática

Principalmente, la Secretaria de Finanzas del Municipio de Jesús María ya cuenta con un sistema para realizar los cobros de los impuestos y tramites del municipio, pero así mismo, este sistema es muy básico y desactualizado, prácticamente ellos solo tienen una pantalla donde escriben el nombre de la persona en un campo de texto y una descripción, en la cual se hace referencia lo que se va a pagar en un segundo campo de texto, es decir, cuando alguien se ve en la situación de realizar un pago, se guarda el registro del mismo, pero de manera como si estuviese contando una historia.

Es decir, “Ricardo Angeles Ramírez”, “Realizo un pago para solicitar/mantener mi licencia para el funcionamiento de mi negocio de casino, esta licencia es para el año del 2020 y por un monto de 5000 pesos”, por ejemplo, entonces, esos datos los tienen registrados de esta manera, el problema reside en el hecho de que no es fácil para la Secretaria de Finanzas no es fácil encontrar la información de las personas.

Por ejemplo, si la Secretaria tuviera la necesidad de saber cuántos pagos de licencia un ciudadano tiene hecho, tendrían que bajar toda la información del historial de pagos realizados, después, en base a búsquedas de textos, buscar las coincidencias donde el nombre de la persona sea el mismo de la persona cuya investigación están realizando, y entonces en base a la descripción del pago realizado, ellos tendrán que buscar la información dependiendo de lo que necesiten saber, además, si dicha persona que está siendo investigada, ha pagado algún otro tipo de trámite que no está relacionado con el pago de una licencia, ya sea predial, o multas, o incluso un recibo del agua, la Secretaria de Finanzas deberá buscar entre toda esa información, lo que necesitan saber.

Además, para realizar reportes de la información del sistema que en este momento manejan, es necesario hacer la suma de los datos de forma manual, y si lo quieren ordenar por dependencia, también conlleva la misma situación; hay un universo de información en bruto que no les ayuda a tomar decisiones ni saber cuántos ingresos capta el municipio ni tampoco por dependencia.

ENCABEZADO

Entonces, recapitulando, el hecho de no poder administrar ni organizar la información en lo que respecta a los trámites realizados causa una gran pérdida de tiempo y recursos en cuanto al saber que personas deben o no al estado por parte de negocios, lo cual limita la cantidad de decisiones que la Secretaría puede tomar por el tiempo que toma el averiguar la más mínima cosa respecto a los ciudadanos y la información en general de dicho sistema.

Objetivos

El objetivo principal es el lograr la convergencia de los más de 4000 conceptos que se tiene en la ley de ingresos de Jesús María, de esta manera se podrán realizar la documentación y generar pagos de cada uno de los permisos a los que una persona puede tener acceso, todo en una sola aplicación.

Justificación

Se busca realizar un sistema para la Secretaría de Finanzas del municipio de Jesús María el cual facilite la organización, control y cobro de los distintos trámites que pueden ser realizados por parte de los ciudadanos, se plantea el poder hacer usos de los más de 4000 conceptos que se tienen descritos y estipulados por parte del gobierno para los distintos permisos que se deben de tener para que un negocio pueda funcionar de manera legal y correcta.

Así mismo, se tiene planeado el dividir y organizar la información en tablas de base de datos para facilitar la realización de reportes tanto de personas físicas, como por parte de áreas o dependencias específicas que son parte del sistema, de esta manera, mejorar el control y la toma de decisiones en cuanto a dicha información, además de mantener un control mucho mejor respecto a las personas cuyos pagos no han sido realizados, que es un problema latente en el sistema actual.

ENCABEZADO

También se pretende hacer uso de la tecnología serverless de AWS con el objetivo de minimizar la cantidad de recursos que son necesarios para mantener la aplicación y el servicio de la misma en línea, ya que es una de las mejores tecnologías actualmente en el mercado para sustentar de manera constante el servicio de una aplicación, además de lo barato que resulta para proyectos de bastante movimiento de información.

Viabilidad

La viabilidad del proyecto es claramente un proyecto robusto con distintos puntos que deben de verse a profundidad para encontrar soluciones, sin embargo la empresa de Software sin Limites tiene los recursos más que necesarios para poder planear, desarrollar y probar la plataforma sin muchos problemas, el avance tan rápido de los integrantes que conforman su equipo es eficiente, además de que esta es la segunda empresa del dueño, así que tiene los recursos necesarios para desarrollar el proyecto ya que están apoyados por la primera empresa.

Resultados esperados

Al concluir la estadía, lo que se espera es lograr tener una aplicación principal basada en React, que haga funcionalidad respecto a las vistas de la aplicación, que se encargue del movimiento y control de la aplicación.

También se espera una aplicación laravel, la cual asociada con el servicio serverless y Lambda de AWS, permitirá el desarrollo de una API para el control y uso de la información de una base de datos relacional SQL también montada sobre el mismo servicio de AWS.

De esta manera es lo que se espera lograr en la finalización de la estadía, además del familiarizarse a fondo con las tecnologías que se están utilizando.

ENCABEZADO

Metodología a aplicar

Scrum, ya que se busca tener comunicación constante con el cliente y se busca que todos los miembros del equipo tengan control en el proyecto.

Capítulo III. Desarrollo de la estadía profesional.

Requisitos para el desarrollo

Para iniciar a trabajar en el sistema fue necesaria la instalación de algunos componentes:

Node Js para Windows, para instalar Node JS en Windows se necesita descargar el instalador desde la página oficial de Node <https://nodejs.org/es/download/>.

Descargar e Instalar la línea de comandos de aws del siguiente link: https://github.com/aws-labs/aws-sam-cli/releases/latest/download/AWS_SAM_CLI_64_PY3.msi.

Instalar serverless con el administrador de paquetes de Node escribiendo el comando 'npm install -g serverless' en la terminal, una vez instalado lo configuramos con el comando 'serverless config' esto nos pedirá que ingresemos una key y una secret key, estas credenciales las obtenemos creando una cuenta en AWS, después crear un usuario en el apartado de IAM de la consola de aws, esto nos generara las credenciales necesarias para la configuración de serverless más información para este paso puede ser consultada en <https://serverless.com/framework/docs/providers/aws/guide/credentials/>, la línea que debemos ingresar es la siguiente:

'serverless config credentials --provider aws --key (Aquí va tu Access keyid) --secret (Aquí va tu secret key)'

Dependencias del API Restful

El api está utilizando sequelize que es un framework ORM de Node Js para el soporte de bases de datos SQL, la instalación de sequelize requiere los siguientes comandos:

```
npm init -y
```

```
npm i --save-dev serverless-offline
```

```
npm i --save mysql2 sequelize
```

ENCABEZADO

Uso de plantilla fuse React Js, esta plantilla ofrece estructura y componentes de Material UI para realizar interfaces de usuario, uso de AWS para el hospedar sistema y la base de datos para así obtener todos los benéficos de trabajar con arquitecturas serverless.

Los requerimientos se obtienen a través de reuniones con los Key-user (usuarios clave), en estas reuniones se estudia la problemática del cliente, además de las necesidades que los usuarios principales saben que hacen falta para su labor del día a día, de estas reuniones se realiza el análisis necesario y se entrega una propuesta de solución que cubra con sus necesidades, esta reunión se basa en la colaboración de todos los integrantes del equipo que de forma individual responderán a las preguntas, ¿Qué hice?, ¿Qué voy a hacer hoy? y ¿Qué impedimentos se tienen para realizar el trabajo?.

Utilizamos las siguientes herramientas para la gestión del proyecto:

Trello donde se manejará el backlog de las historias de usuario, Clockify para medir los tiempos que le toma a cada desarrollador completar la historia de usuario seleccionada.

Roles del proyecto

Project Manager: Jorge Adalberto Domínguez Ibarra

Líder técnico: Luis Antonio Cruz Macías

Desarrollador DBA: Ricardo Angeles Ramírez

Desarrollador: Fátima Arámbula

Desarrollador: Erik Daniel Mendoza Olivares

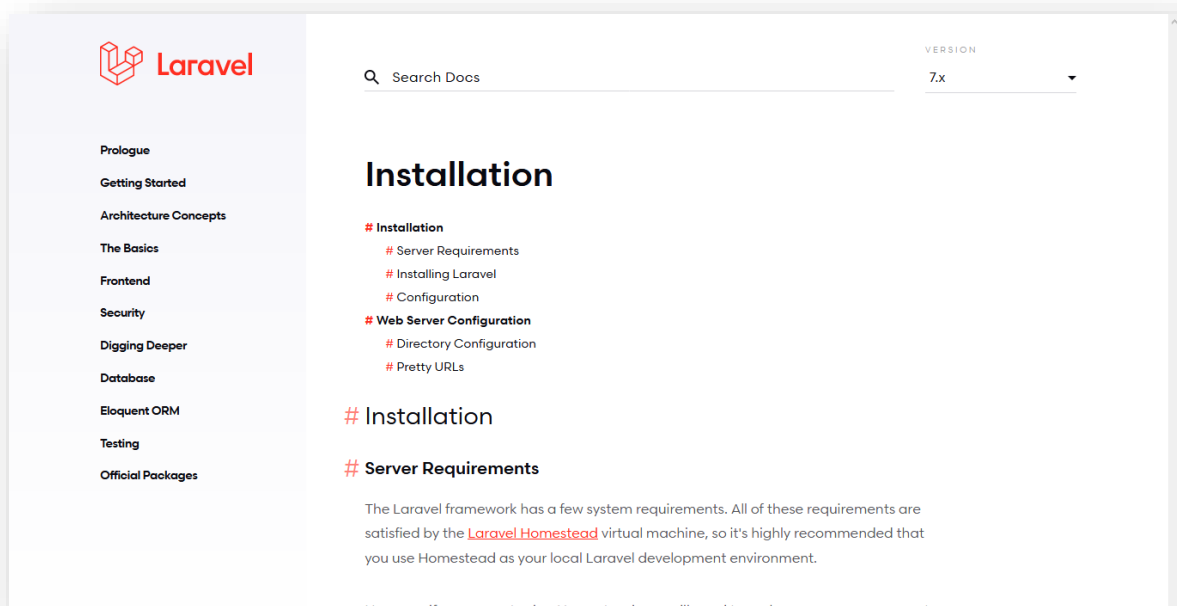
Desarrollador: Alejandro Villalobos

Inicio de la estadía

Investigación Laravel

Al comienzo principalmente mis actividades trataron de investigación acerca del framework basado en php **Laravel**, desde un principio se tenía planteado el uso de Laravel para permitirnos facilitar el desarrollo de la aplicación ya que se tenía contemplado que se desarrollaría la parte del back-end con el servicio de AWS serverless **Bref** ya que permitía una integración de ambas partes de manera sencilla ya que ambos están basados en el lenguaje de php.

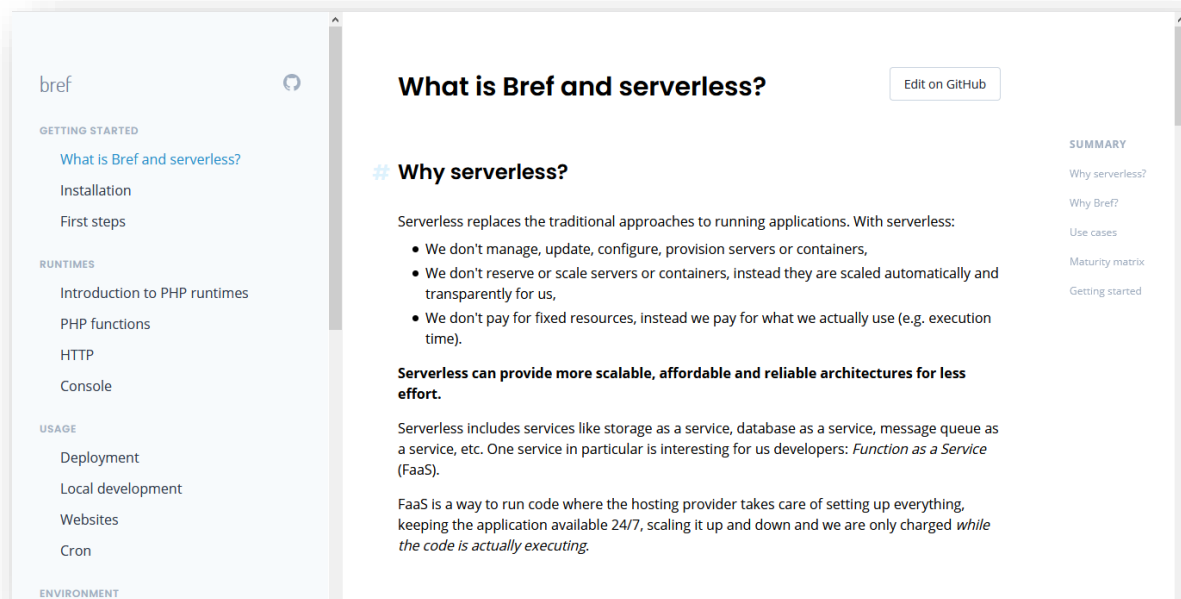
Así que en primera instancia lo que hice fue familiarizarme con el framework de Laravel haciendo uso de su documentación en la página oficial <https://laravel.com/docs/7.x> en la cual lo que en primer lugar realice fueron ejemplos y entender la secuencia de información del mismo framework.



Lo que aprendí principalmente de la investigación fue el desarrollo de una aplicación simple con el framework, la conexión con una base de datos y el sistema de envío de información con el que trabaja php.

Investigación Bref/serverless

Así mismo también me fue asignado el investigar Bref, que es una librería creada a base de php con el objetivo de facilitar el uso del servicio como función de AWS, el sistema de Lambdas que maneja Amazon, ya que el mismo no tiene soporte para el lenguaje de php, así que Bref se puede comparar como un “puente” que permite el uso de php con las Lambdas; para la investigación, en lo principal que me base fue en la documentación oficial <https://bref.sh/docs/>.

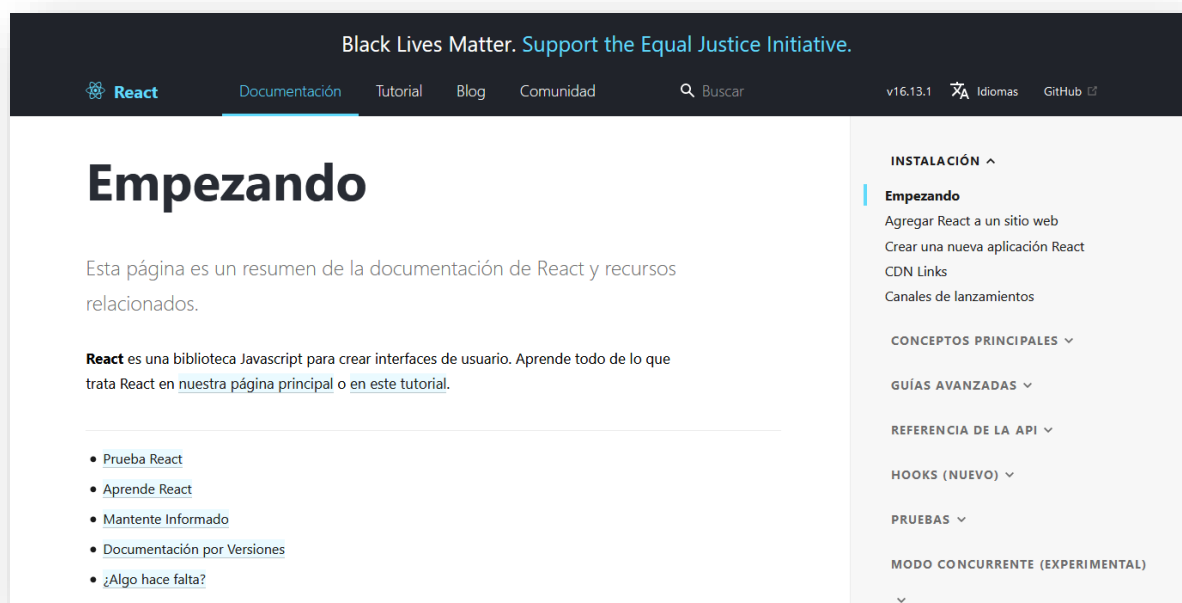


Lo que aprendí respecto a serverless y Bref fue una de las cosas más interesantes del proyecto, aprendí que serverless probablemente es la manera a futuro en que el back-end en aplicaciones web se trabajará, principalmente por el hecho de la cantidad de dinero que una persona u organización debe proveer por el servicio, las Lambdas es un sistema que permite llamar una función específica para un trabajo específico y Amazon solo cobra por el tiempo o las peticiones que hagan uso de la misma, por ende es mucho más factible en términos de inversión que el hecho de tener tu propio servidor para una aplicación web o el rentar un servidor por el hecho de que muchas veces se pierde dinero porque se desaprovecha la capacidad de trabajo máximo del mismo.

Cambio de *framework* Laravel -> React

Después de unos días en cuanto a la investigación se decidió por parte del líder técnico y del encargado del proyecto que trabajaríamos con el *framework* React, que está basado en JavaScript, por ende, me fue encomendada de nuevo la tarea de investigar, pero esta vez con un objetivo, realizar un CRUD simple utilizando React, lo cual al principio fue un desafío por la manera tan diferente que trabaja la comunicación de información el framework.

Lo primero que hice fue los mismos pasos que la investigación de Laravel, fue el revisar la documentación oficial de React en la página <https://es.reactjs.org/docs/getting-started.html>.



Lo primero que aprendí de React es el hecho de que se basa en componentes para crear una interfaz visual para el usuario, permite el crear componentes personalizados con el objetivo de permitir reutilizar código de manera sencilla, además de que, para compartir información entre componentes, solo es posible si el componente “padre” le proporciona la información al “hijo”, pero para mandar información de manera contraria es más complicado.

ENCABEZADO

Así que utilizando estos conceptos y siguiendo unos diseños que me proporciono el encargado del proyecto, realice un CRUD utilizando React, una base de datos en MySQL y usando archivos php en un servidor Apache para realizar la conexión entre la aplicación y las peticiones hacia la base de datos.

Desarrollo del proyecto

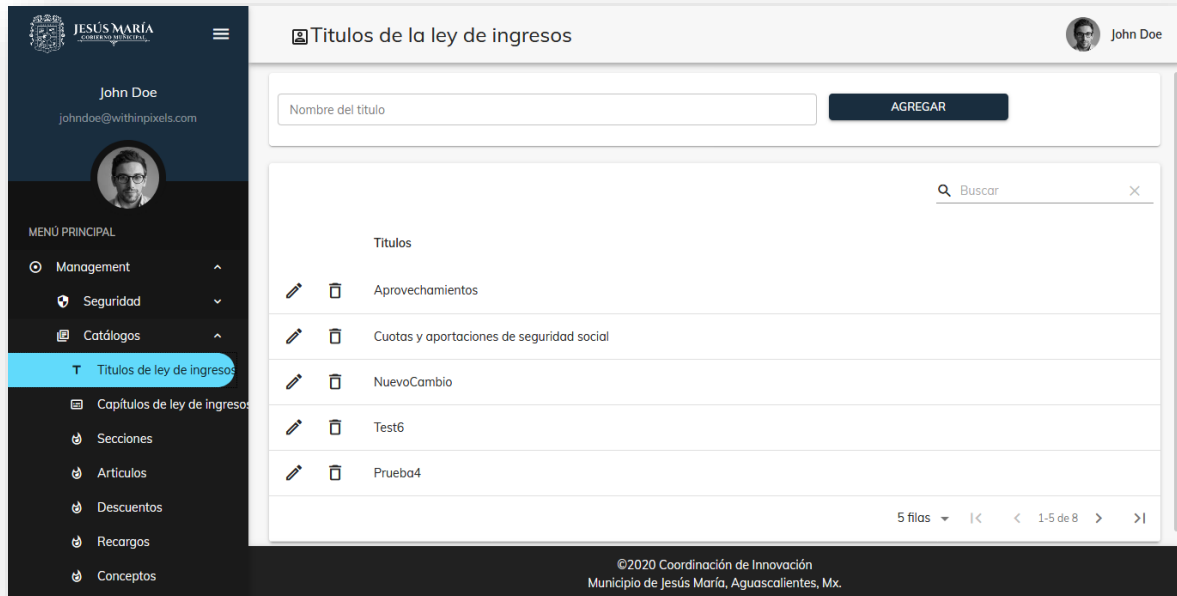
Base de datos

Mis primeras actividades respecto al desarrollo de la aplicación fue el encargarme de generar el script de la base de datos de un modelo de 50 tablas para la base de datos <https://n9.cl/d80x> además de que se me asigno la responsabilidad de seguir los cambios que cada uno de los integrantes del equipo realizara al respecto de la base de datos y de esta manera mantener un control o almenos una persona que pudiera ser capaz de mantener la base de datos alojada en el servidor de Amazon lo más “limpia” posible.

Vista de títulos de la ley de ingresos

El proyecto fue dividido en tres secciones diferentes, “Áreas”, “Ley de ingresos” y “Contribuyentes”, dichas secciones fueron asignadas a cada integrante de desarrollo, la que me fue encomendada fue la ley de ingresos, la cual consiste de 9 vistas, en primer lugar, la de “Títulos”, ya que la ley de ingresos de Jesús María está dividida en distintas partes, uno de los componentes importantes son los títulos de las leyes.

ENCABEZADO

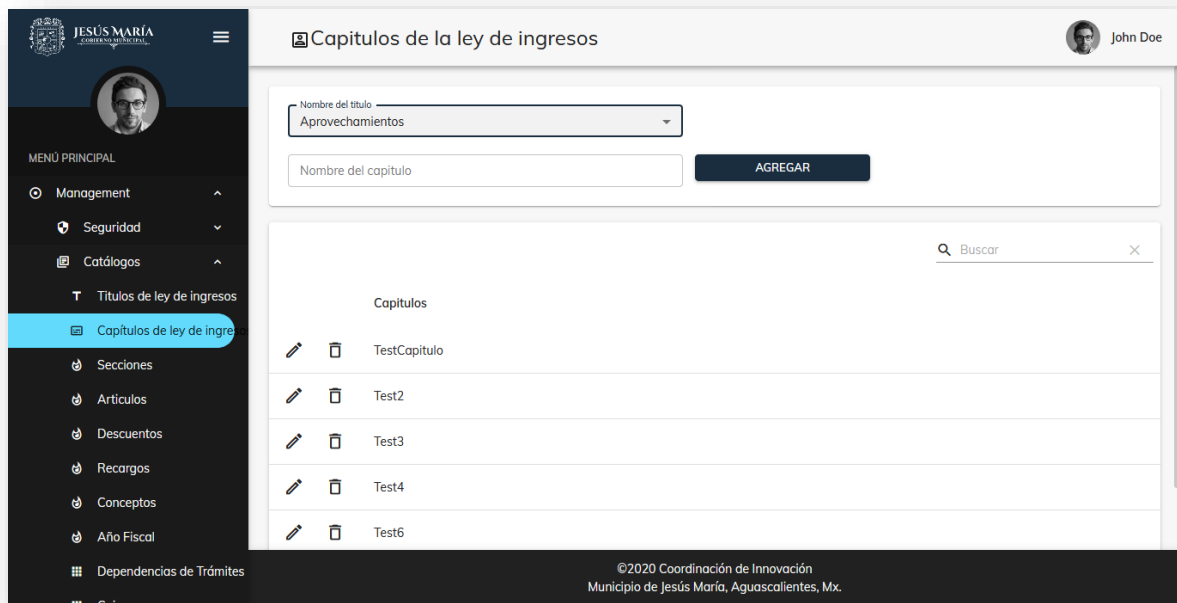


Lo que hice fue en primer lugar escribir una función *ir* con la cual pude definir que al leer la url en la que se encontrara el usuario, poder modificar el título de la página dependiendo de la vista que se mostrara.

Utilicé un componente llamado *textfield* en cual cumpliría la función de obtener la cadena de texto que después con el uso del botón “Agregar” permitiría mandar una petición a la API Restful hosteada en un servidor de AWS, el cual crearía el nuevo registro en la tabla `Cat_Titulos` de la base de datos; además hicimos uso de una librería proporcionada por Material-UI llamada “material-table” cuyo componente nos permitiría buscar registros específicos dentro de la tabla, además de facilitar el poder usar funciones como los botones de borrado y de edición para modificar la información en la tabla y por ende mandar peticiones para hacer los cambios pertinentes en la base de datos.

La tabla fue un desafío el lograr hacer que realizará una actualización cada vez que se creara un nuevo registro en la vista, haciendo uso de *hooks*, que es una funcionalidad relativamente reciente de React, la cual permite el realizar llamadas asíncronas a la API al momento que se detecta un cambio en una variable.

Vista de capítulos de la ley de ingresos



En la siguiente vista, lo que hice fue que a base de uso de *hooks*, al inicio de la renderización de la vista se realiza una llamada a la API, mandando a llamar los registros de los títulos en la base de datos, con dicha información, lo que hice fue el mostrar el nombre de los títulos en el componente *select*, en la parte superior de la vista, de esta manera, al seleccionar un título en específico, cuando se escribe el nuevo nombre del capítulo que se quiere agregar, este automáticamente estará asociado con el título que fue previamente seleccionado.

Además, haciendo uso de la misma información del *select*, lo que hice fue el enviar el id del título a la tabla, y de esta manera, haciendo uso de uno de los *endpoints* de la API, pude mandar a llamar a los capítulos que ya estuvieran relacionado con el título seleccionado, haciendo más fácil la visualización de la información ya que no mostraría todos los registros de los capítulos, y ya que la ley de ingresos tiene más de 4000 conceptos, el mantener la visualización de la información de manera limpia, es esencial.

Vista de secciones de la ley de ingresos

The screenshot shows a web application interface for 'Secciones de la ley de ingresos'. On the left is a dark sidebar with a logo for 'JESÚS MARÍA' and a user profile. The sidebar menu includes 'MENÚ PRINCIPAL' with options like 'Management', 'Seguridad', 'Catálogos', 'Títulos de ley de ingresos', 'Capítulos de ley de ingresos', 'Secciones' (highlighted), 'Artículos', 'Descuentos', 'Recargos', 'Conceptos', 'Año Fiscal', 'Dependencias de Trámites', and 'Gastos'. The main content area has a header 'Secciones de la ley de ingresos' and a user profile 'John Doe'. Below the header are two dropdown menus for 'Nombre del título' and 'Nombre del capítulo', followed by a 'Seccion *' input field and an 'AGREGAR' button. A search bar with the text 'Buscar' is also present. The main content area displays the word 'Secciones' and a message 'No se encuentran datos para mostrar'. The footer contains copyright information: '©2020 Coordinación de Innovación Municipio de Jesús María, Aguascalientes, Mx.'

En la vista de secciones lo que hice fue el hacer que el *select* de títulos y capítulos estuviera relacionados, por decirlo de una manera; al momento de elegir un título, modificaría la información que el *select* de capítulos de permitiría ver, algo muy parecido a la interacción entre el *select* de la vista anterior y la tabla, pero en este caso es para poder minimizar la cantidad de información que se mostrara en el campo de capítulos, y relacionar cada capítulo con sus secciones correspondientes.

La manera en que funciona la tabla es muy similar a la anterior, ya que también hace uso de *hooks* además de estar constantemente atenta si es que hay un cambio en la variable que actualiza la tabla cuando hay un nuevo registro generado desde la vista.

Vista de artículos de la ley de ingresos

En este caso, se puede ver la similitud con la vista anterior ya que hace uso de la misma dinámica de control de información, tres diferentes *selects* que permiten el filtrar información dependiendo del título, capítulo y sección que se elijan en cada campo.

Sin embargo, hay una diferencia clara aquí ya que se hace uso de dos *textfields* ya que son los dos campos particulares que tienen los artículos, en el nombre del artículo hice uso de **regex** que es una manera que permite el filtrar la información que el campo tiene permitido aceptar, en este caso solo lo adapte para que permitiera escribir números y no cadenas de texto, ya que los artículos son definidos por un número, y evitar que el usuario que haga uso de esta vista, tenga la posibilidad de escribir un nombre incorrecto, ya sea de forma deliberada o por error.

Vista de descuentos de la ley de ingresos

The screenshot shows a web application interface for managing income tax discount rules. On the left is a dark sidebar with the user profile 'John Doe' and a main menu. The main content area is titled 'Reglas de descuento de la ley de ingresos'. It contains several form fields: a 'Descripción' field with a placeholder 'Regla *'; a 'Calcular por' section with a dropdown set to 'Monto' and a 'Monto' field showing '\$ 0.0'; an 'Aplicable a' section with dropdowns for 'Estado civil', 'Situación laboral', and 'Edad'; and an 'Aplicable en' section with a checkbox for 'Todo el año' and a grid of checkboxes for each month of the year. A blue 'AGREGAR' button is at the bottom right. The footer includes copyright information for 2020.

John Doe
johndoe@withinpixels.com

MENÚ PRINCIPAL

- Management
- Seguridad
- Catálogos
- Titulos de ley de ingresos
- Capítulos de ley de ingresos
- Secciones
- Artículos
- Descuentos**
- Recargos
- Conceptos

Reglas de descuento de la ley de ingresos

John Doe

Descripción Regla *

Calcular por Calcular por Monto Monto \$ 0.0

Aplicable a Estado civil Situación laboral Edad

Aplicable en ☐ Todo el año

☐ Enero ☐ Febrero ☐ Marzo ☐ Abril ☐ Mayo ☐ Junio
☐ Julio ☐ Agosto ☐ Septiembre ☐ Octubre ☐ Noviembre ☐ Diciembre

AGREGAR

©2020 Coordinación de Innovación
Municipio de Jesús María, Aguascalientes, Mx.

En esta vista tuve que trabajar diferentes tipos de componentes para lograr la funcionalidad necesaria para generar los registros de manera correcta en la base de datos, ya que debido a la interacción de tantas tablas en una sola vista. En primer lugar, la descripción es el campo básico del descuento que describe de manera simple que clase de descuento es, después el campo de “Monto” permite decidir entre tres diferentes maneras de calcular cuánto valor tiene dicha regla de descuento, que podía ser un monto fijo, que es el que se ve la imagen, un porcentaje, en el cual, en caso de estar seleccionado, lo que mostraría otro *textfield* el cual estaría adaptado con un símbolo de porcentaje para dejar claro la diferencia entre ellos, y por ultimo UMA que es un valor fijo proporcionado por el INEGI.

Después, en los campos de estado civil, situación laboral y condición lógica, funcionan de manera muy similar a las vistas anteriores, ya que es necesario llamar información desde la API Restful, ya que cada campo tenía que llamar información de tres tablas diferentes, y mostrar la misma en cada uno de los campos.

ENCABEZADO

Además, la parte más desafiante fue el hecho de generar un proceso almacenado en la base de datos que me permitiera hacer los registros necesarios por cada mes que fuese aplicable dicha regla de descuento, ya que cada regla de recargo puede llegar a poder ser aplicada tanto en todos los meses, como solo en unos en específicos, en el modelo de la base de datos se tiene una tabla que hace función de relación, la cual tiene un registro por cada mes en que una regla de descuento puede ser aplicada, así que tuve que crear un proceso almacenado dentro de la base de datos, el cual sería llamado por parte de la API.

Vista de recargos de la ley de ingresos

The screenshot shows a web application interface for managing income tax surcharge rules. On the left is a dark sidebar with the user's name 'John Doe' and email 'johndoe@withinpixels.com'. Below this is a 'MENÚ PRINCIPAL' (Main Menu) with items: Management, Seguridad, Catálogos, Titulos de ley de ingresos, Capítulos de ley de ingresos, Secciones, Artículos, Descuentos, Recargos (highlighted in blue), and Conceptos. The main content area is titled 'Reglas de recargo de la ley de ingresos'. It contains a form with the following fields: 'Descripción' (labeled 'Regla *'), 'Calcular por' (a dropdown menu set to 'Monto'), 'Monto' (a text input field showing '\$ 0.0'), 'Aplicable a' (a dropdown menu set to 'Estado civil'), 'Situación laboral' (a dropdown menu), and 'Edad' (a dropdown menu). Below these are checkboxes for 'Aplicable en' with a 'Todo el año' option and a grid of 12 checkboxes for each month from Enero to Diciembre. An 'AGREGAR' button is at the bottom right of the form. The footer of the application shows the copyright '©2020 Coordinación de Innovación' and the location 'Municipio de Jesús María, Aguascalientes, Mx.'

La vista de regla de recargos es prácticamente un espejo de la vista de descuentos, así que gracias a esto me fue muy sencillo aplicar las mismas soluciones que realice en la vista anterior, y aplicarlos en esta vista, tanto funcionalidad relacionada con la base de datos, como el proceso almacenado, hasta las llamadas de la API para los *selects* cuya información tenía que ser llamada desde la base de datos.

El desarrollo de estas vistas tomo mucho más tiempo de lo esperado, debido a la cantidad de información y la organización que tuve que tomar en comparación al resto de las vistas, ya que en las anteriores se manejaba muy poca información en comparación, así que este fue uno de los desafíos más grandes que tuve en la estadía,

ENCABEZADO

lograr permitir que la vista fuera capaz de realizar registros nuevos y que lo hiciera con la secuencia de información correcta.

Vista de conceptos de la ley de ingresos

The screenshot displays a web application interface for managing income law concepts. On the left is a dark sidebar with a 'MENÚ PRINCIPAL' (Main Menu) containing icons and labels for 'Management', 'Seguridad' (Security), 'Catálogos' (Catalogs), 'Títulos de ley de ingresos' (Income law titles), 'Capítulos de ley de ingresos' (Income law chapters), 'Secciones' (Sections), 'Artículos' (Articles), 'Descuentos' (Discounts), 'Recargos' (Surcharges), 'Conceptos' (Concepts - highlighted in blue), 'Año Fiscal' (Fiscal Year), and 'Dependencias de Trámites' (Dependencies of Procedures). The main content area is titled 'Conceptos de la ley de ingresos' and features a user profile 'John Doe' in the top right. The form includes several input fields: 'Área *' (Area), 'Título *' (Title), 'Capítulo *' (Chapter), 'Sección *' (Section), and 'Artículo *' (Article). Below these are text boxes for 'Descripción del artículo' (Article description), 'Fracción' (Fraction), and 'Concepto *' (Concept). A toggle switch for 'Importe fijo' (Fixed amount) is present. The 'Importe *' (Amount) field shows '0.0 %', 'Importe mínimo' (Minimum amount) shows '\$ 0.0', and 'Periodicidad *' (Periodicity) is a dropdown menu. At the bottom, there is a section for 'Reglas de descuento posibles aplicables' (Possible applicable discount rules) with an 'AGREGAR DESCUENTO' (Add discount) button. The footer contains the text '©2020 Coordinación de Innovación, Municipio de Jesús María, Aguascalientes, Mx.'

Esta vista también fue algo complicada de lograr el movimiento correcto de la información de los conceptos que se manejaban por parte de la ley de ingresos, ya que esta vista en particular hace uso de información con la que no tuve interacción hasta el momento de estar trabajando en esta vista, ya que tiene relación con las áreas, que era una sección con la cual una compañera de la estadía, fue la que estaba trabajando.

A diferencia de la anterior no fue extremadamente complicado hallar la manera de dar las soluciones necesarias para realizar los registros de los conceptos gracias a la experiencia que obtuve con la vista anterior, además de que en este caso fuimos dos personas las que nos encargamos de lograr la funcionalidad completa de la vista, dividiendo el esfuerzo y avanzando el doble de rápido.

ENCABEZADO

Vista de año fiscal de ley de ingresos

The screenshot shows a web application interface for configuring fiscal years. On the left is a dark sidebar with the logo of 'JESÚS MARÍA AGUASCALIENTES' and a user profile picture. Below the profile is a 'MENÚ PRINCIPAL' (Main Menu) with the following items: Management, Seguridad, Catálogos, Titulos de ley de ingresos, Capítulos de ley de ingresos, Secciones, Artículos, Descuentos, Recargos, Conceptos, Año Fiscal (highlighted in blue), and Dependencias de Trámites. The main content area is titled 'Nuevo ejercicio fiscal - Configuración de Ley de Ingresos'. It contains two input fields: 'Ejercicio fiscal año' (empty) and 'Ejercicio fiscal base' (set to '2021' with a dropdown arrow). To the right of these fields is a button labeled 'CREAR NUEVO EJERCICIO'. At the bottom of the page, there is a footer with the text: '©2020 Coordinación de Innovación Municipio de Jesús María, Aguascalientes, Mx.'

Esta vista es en extremo simple en comparación con las anteriores, pero la diferencia es que la misma solo llamaba un proceso almacenado en la base de datos, el verdadero reto fue el generar dicho proceso almacenado, ya que el mismo hacia una copia de los conceptos, títulos, secciones, capítulos, etc., de una año fiscal específico, y generaba uno nuevo en base al año fiscal base.

Capítulo IV. Resultados y conclusiones

Descripción de los resultados obtenidos conforme a los objetivos planteados en el proyecto:

Lo que se obtuvo al final de la estadía fue una aplicación web capaz de mantener un registro constante de los diferentes conceptos de la ley de ingresos, además de permitir el cobro de diferentes trámites para el municipio de Jesús María.

Se logró el objetivo de permitir facilitar y mejorar la obtención de la información necesaria, ya que a base de *triggers* en la base de datos, se mejoró el control de las distintas áreas que se trabajan en el sistema de Jesús María, de esta manera ayuda en gran medida la toma de decisiones por parte de esa secretaria.

De la misma manera, permite el control de los distintos contribuyentes que necesiten un permiso o un servicio por parte del municipio, facilitando el hecho de que sean capaces de mejorar su sistema de cobro y no perder rastro de las personas que aún no han pagado la renovación de sus permisos o cualquier otro trámite con relación al gobierno.

Conclusiones.

Me pareció una experiencia muy enriquecedora el poder trabajar con un framework completamente nuevo para mí, además de el tener que aplicarlo de una manera tan intensiva para dar soluciones a los distintos problemas con los que me enfrente.

Me gusta la manera de trabajar del equipo con el que tuve la oportunidad de trabajar, principalmente por el apoyo que se debe en cuanto alguno de los integrantes se enfrentaba con un problema o una limitación que evitaba la posibilidad de avanzar con su carga de trabajo.

Una de las tecnologías que más me llamaron la atención y con la que sé que me quiero familiarizar aún más y fue gracias a la estadía, fue con el sistema *serverless* y las Lambdas que proporciona AWS como servicio, la manera en el que la arquitectura en la que está pensada y creada este sistema es muy llamativo para mí, además de la facilidad que permite a una aplicación hacer el uso de un back-end de manera sencilla y barata.

Además de la experiencia que he adquirido por parte de la empresa Software sin Limites se me ha dado la oportunidad de seguir en el equipo de la empresa, así que puedo concluir que ha sido una experiencia de los más fructífera para mí.

Recomendaciones.

Una recomendación para cualquiera que lea mi reporte es el hecho de que se haga lo mejor que pueda para mantener un control y organización respecto a su trabajo, además de siempre mantener comunicación con tus jefes inmediatos, y principalmente, esforzarse por aprender lo más que pueda, ya que cualquier cosa puede ser útil en el futuro.

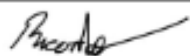
Anexos

Cronograma de actividades.

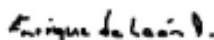
	UNIVERSIDAD TECNOLÓGICA DE AGUASCALIENTES Cronograma de Actividades
---	--

Programa Educativo: TSU EN TECNOLOGÍAS DE LA INFORMACIÓN	Periodo: MAYO - AGOSTO 2020
Nombre del Estudiante: Ricardo Angeles Ramirez	
Nombre de la Empresa o Institución: Software sin Limites SC	
Título de la Estadía Profesional: Sistema de trámites en línea	Fecha de elaboración: 21/05/2020
Objetivo de la Estadía Profesional: Realizar una plataforma para configuración de flujos y aprobaciones de tramites online.	

No.	P/R	Actividad	Semanas														
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	P	Análisis de requerimientos															
	R																
2	P	Diseño del front end															
	R																
3	P	Diseño de la base de datos															
	R																
4	P	Codificación de la base de datos															
	R																
5	P	Codificación del back end															
	R																
6	P	Codificación del front end															
	R																
7	P	Solución de errores/bugs															
	R																
	P																
	R																
	P																
	R																
	P																
	R																
	P																
	R																
	P																
	R																
	P																
	R																
	P																
	R																



Firma del estudiante



Firma del asesor



Firma del tutor

P= Programado
R= Real

F-SA-06
Revisión No. 4
Fecha de Revisión: 24/03/17

Carta de Liberación en hoja membretada

Voyastic

Aguascalientes, Ags, a 11 del mes de agosto, 2020.

A quien corresponda:

Por medio de la presente, se hace constar que el estudiante Ricardo Angeles Ramirez, de la carrera de TSU. en Tecnologías de la Información Área Desarrollo de Software Multiplataforma realizó el proyecto de estadia denominado Sistema de recaudación de Jesús Maria, teniendo como tutor por parte de la empresa a Alejandro Rodriguez Ramirez, y como asesor por parte de la Universidad Tecnológica de Aguascalientes a Enrique de León Dominguez, cumpliendo con un total de 525 horas de trabajo, y con la elaboración y entrega a la empresa del reporte del trabajo de estadia digital, con lo cual se libera de cualquier responsabilidad al estudiante a partir de la fecha señalada, en el entendido de que cumplió satisfactoriamente con los requisitos y objetivos del mismo.

Se extiende la presente como evidencia de la conclusión del proyecto, y para los fines que al interesado convengan.

ATENTAMENTE:

Alejandro Rodriguez Ramirez
DIRECTOR DE OPERACIONES

**THE TECHNOLOGY PARTNER
TO SKYROCKET YOUR COMPANY**



Referencias Bibliográficas

Enbonnet.me. (2020). Construir API Rest Nodejs Serverless Con AWS Lambda Y Aurora Serverless. [online] Available at: <<https://enbonnet.me/article/49/construir-api-rest-nodejs-serverless-con-aws-lambda-y-aurora-serverless>> [Accessed 14 August 2020].

Serverless Framework -AWS Lambda Guide -Installing TheServerless Framework. (s. f.). serverless. Recuperado 13 de agosto de 2020, de <https://www.serverless.com/framework/docs/providers/aws/guide/installation/>

Abellán, E. (2020, 5 marzo). Metodología Scrum: qué es y cómo funciona. WAM. <https://www.wearemarketing.com/es/blog/metodologia-scrum-que-es-y-como-funciona.html#:~:text=Scrum%20es%20una%20metodolog%C3%ADa%20de,equipos%20que%20manejan%20proyectos%20complejos>.

AWS | Cloud Computing -Servicios de informática en la nube. (s. f.). Amazon Web Services, Inc. Recuperado 13 de agosto de 2020, de <https://aws.amazon.com/es/>

Manual | Sequelize. (s. f.). Sequelize. Recuperado 13 de agosto de 2020, de <https://sequelize.org/master/>

Empezando -. (s. f.). React. Recuperado 13 de agosto de 2020, de <https://es.reactjs.org/docs/getting-started.html>Material-UI. (s. f.).

Material-UI: A popular React UI framework. <https://material-ui.com/W>. (s. f.).

Fuse React -Material Design Admin Template. FUSE. <http://react-material.fusetheme.com/documentation/getting-started/introduction>