



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

TESIS

**APLICACIÓN DE TIC'S PARA GESTIONAR EL PROGRAMA DE
APOYO NUTRICIONAL**

QUE PARA OBTENER EL TÍTULO DE
INGENIERO EN COMPUTACIÓN

PRESENTA:

MARTÍNEZ RODRÍGUEZ MARCOS

M. en A. MARÍA DE LOURDES CAMPOS LUNA

CIUDAD UNIVERSITARIA 21/01/2016.





Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

DEDICATORIA

Dedico esta tesis a mis padres porque si no fuera por ellos nunca hubiese logrado hacer lo que hoy puedo presentar, gracias a ellos y a su incansable esfuerzo cuento con la formación como persona y como gran profesional que aspiro a ser; considero que no hay palabras para expresar todo el agradecimiento y amor que siento por ellos pero sé que al leer estas líneas, aunque breves, les vendrán a la mente muchas experiencias, algunas amargas y otras muy dulces pero todas ellas apreciables etapas de nuestra vida.

A la M. en A. Lourdes Campos que ha tenido una paciencia increíble para conmigo en el desarrollo de este trabajo, agradezco infinitamente los consejos, las enseñanzas y el esfuerzo que me ha obsequiado en el transcurso de este tiempo.

A mi hermano, mi cuñada, mis amigos y colegas por su apoyo, consejos, regaños y demás.

A mis profesores que han dejado grandes enseñanzas con las que ahora puedo lograr muchos objetivos y convertirme en alguien mejor.

A usted que lee estas líneas, ahora también es parte de esta aventura.

A mí también aunque suene un tanto gracioso, completar este trabajo ha sido un gran reto y se ha convertido en el símbolo de una transformación total de mi ser.

GRACIAS.

ÍNDICE

1.	Introducción.....	1
1.1.	Planteamiento del problema	1
1.2.	Objetivo	1
1.3.	Justificación.....	2
1.4.	Propuesta de solución.	2
2.	Marco teórico.....	4
2.1.	Arquitectura cliente servidor	4
2.2.	Elementos para construcción de las vistas o páginas web de un sistema informático.	4
2.2.1.	HTML.....	5
2.2.2.	JavaScript	8
2.2.3.	CSS.....	10
2.2.4.	Herramientas para el desarrollo web	12
2.3.	Bases de datos.....	14
2.3.1.	Definición de base de datos.....	14
2.3.2.	Bases de datos relacionales	14
2.3.3.	Relaciones entre entidades.....	15
2.3.4.	Manejo de la información dentro de la base de datos.	16
2.4.	Lenguajes de programación.....	17
2.4.1.	PHP	17
2.4.2.	ASP.NET	18
2.4.3.	Python.....	19
2.4.4.	Ruby.....	19
2.4.5.	Java.....	19
2.5.	Transporte y almacenamiento de datos.....	20
2.5.1.	XML.....	20
2.5.2.	JSON.....	21
3.	Análisis del problema.....	22
3.1.	Descripción de la situación actual	22
3.1.1.	Registro de candidatos al Programa de Apoyo Nutricional	22
3.1.2.	Consumos de los beneficiarios	23
3.1.3.	Encuestas de satisfacción	23
3.2.	Propuesta de solución al problema.....	23
3.2.1.	Definición de requerimientos del sistema.....	24

4.	Diseño del sistema.....	26
4.1.	Diseño de la base de datos.....	26
4.1.1.	Modelo de datos.....	27
4.1.2.	Modelo lógico: Diagrama entidad – relación.....	35
4.1.3.	Modelo físico: esquema de la base de datos.....	36
4.2.	Diseño del sistema.....	40
4.2.1.	Registro del sistema.....	40
4.2.2.	Conteo de consumos.....	41
4.2.3.	Encuestas de servicios.....	42
4.2.4.	Módulo de administración.....	42
4.2.5.	Estructura de archivos del sistema.....	44
5.	Desarrollo del sistema.....	45
5.1.	Registro de solicitudes.....	46
5.2.	Conteo de consumos.....	53
5.3.	Encuestas.....	55
5.4.	Administración.....	59
5.4.1.	Listados de solicitudes.....	60
5.4.2.	Datos de solicitudes.....	61
5.4.3.	Reporte de encuestas.....	64
6.	Instalación del sistema.....	66
6.1.	Configuración recomendada.....	66
6.1.1.	Requisitos de sistema.....	66
6.1.2.	Plataforma recomendada.....	66
7.	Conclusiones y futuro del trabajo.....	68
7.1.	Pruebas y resultados.....	68
7.2.	Conclusiones.....	69
7.3.	Contribuciones y mejoras.....	70
7.4.	Limitaciones.....	70
7.5.	Líneas futuras.....	71
8.	Bibliografía.....	72

1. INTRODUCCIÓN

1.1. PLANTEAMIENTO DEL PROBLEMA

El Programa de Apoyo Nutricional que se otorga a través de Fundación UNAM a gran cantidad de alumnos en la universidad tiene como objetivo brindar una comida diaria a personas de bajos recursos económicos con un perfil académico que no les permite acceder a la mayoría de las becas ofrecidas ya sea por tener un promedio inferior al requerido o bien, no posee el avance de créditos de acuerdo al tiempo que ha estado inscrito en el nivel licenciatura.

En la operación del Programa de Apoyo Nutricional se genera gran cantidad de información. Al solicitar el apoyo se requieren varios documentos en forma digital y diversos datos personales ya que se realizan auditorías por parte de Fundación UNAM. Una vez que se otorga el apoyo y registra su huella digital en un sistema propietario que poseen las cafeterías, el becario puede ir a la cafetería asignada y tener una comida o desayuno, dependiendo de su elección, generando de esta manera un registro de consumo diario por alumno. Además de lo anterior, con el propósito de evaluar el desempeño del programa, corregir posibles problemas y mejorar la atención a los becarios se realizan dos encuestas en el periodo de apoyo.

El Programa de Apoyo Nutricional requiere tener la posibilidad de informar el estado del mismo, por lo que se generan listas de aspirantes y de beneficiarios; una copia de los archivos proporcionados por los becarios, reportes semanales y mensuales de consumos y reportes de las encuestas, incluyendo gráficas para facilitar su análisis e interpretación.

Debido a la gran cantidad de información, tener una gran cantidad de becarios en el programa se convierte en una ardua labor para procesar los datos del programa en forma manual, de esta manera ha sido notoria la necesidad de crear un sistema que automatice el procesamiento de la información, agilizando en gran medida la operación del programa.

Es por todo lo anterior que se pretende crear un sistema para facilitar la operación del programa, generar registros por los interesados, manejar consumos y generar reportes en un formato fácilmente manejable en paquetería ofimática.

1.2. OBJETIVO

Crear un sistema informático que facilite la operación del Programa de Apoyo Nutricional al automatizar algunas tareas rutinarias con grandes cantidades de datos que se generan día a día, que permita a los beneficiarios proporcionar información de ellos mismos y de la calidad del servicio para generar reportes con

Introducción

la información obtenida sobre la calidad del servicio ofrecido, la operación misma del programa y lograr mejoras.

1.3.JUSTIFICACIÓN.

Debido a la limitación de los recursos tanto tecnológicos, de tiempo y humanos se requiere un sistema que se encargue del procesamiento del grueso de los datos generados en la operación del programa para mantener la información del mismo actualizada y disponible en caso de que se requiera alguna auditoría por parte de los donantes, obtener montos a pagar a las cafeterías entre otras cuestiones. Este sistema debe ser bastante flexible y requerir el mínimo de labores de administración debido a las dimensiones del programa y al crecimiento que puede tener.

1.4.PROPUESTA DE SOLUCIÓN.

El sistema propuesto permitirá a los alumnos interesados en ser beneficiarios del programa de apoyo nutricional realizar su solicitud de ingreso o renovación al Programa de Apoyo Nutricional a través de una interfaz web requiriendo para ello contar con los datos personales solicitados por el programa y los archivos digitalizados en formato PDF de acuerdo a como lo especifica la convocatoria del mismo. Al término del registro el sistema permitirá a los aspirantes descargar su comprobante de registro y un formato de solicitud para que sea firmado y entregado en el área encargada.

También permitirá a los beneficiarios del programa contestar un cuestionario sobre ciertos temas del servicio, los alimentos y una comparación con respecto al mes anterior para saber si ha mejorado o empeorado la calidad del servicio.

Además permitirá la carga de archivos de hojas de cálculo en formato compatible con el formato de archivo Office Open XML SpreadsheetML de Microsoft Office Excel aparecido en su versión 2007 o superior. Estos archivos contendrán los datos de los consumos realizados por los beneficiarios del programa y son generados por sistemas propietarios de la cafetería que brinda el servicio a la escuela o facultad; dichos datos se procesarán automáticamente para brindar información de los consumos realizados por el beneficiario en cierto tiempo con lo que podrán iniciarse procesos administrativos del programa.

El sistema tendrá un módulo para la persona encargada de la operación del Programa de Apoyo Nutricional en la escuela o facultad; dentro del módulo tendrá la posibilidad de descargar archivos compatibles con Microsoft Office Excel en el formato especificado anteriormente donde se indican los datos que especificó el aspirante al momento de registrar la solicitud de ingreso o renovación al programa, de igual manera podrá consultar los datos de algún aspirante o beneficiario en

Introducción

particular, se tendrá una funcionalidad adicional que permitirá ver los resultados de las encuestas empleando gráficas para facilitar su interpretación.

2. MARCO TEÓRICO.

En este capítulo se describen herramientas, tecnologías y paradigmas que constituyen el sustento teórico y metodológico para el desarrollo del trabajo.

2.1. ARQUITECTURA CLIENTE SERVIDOR

Para que un sistema de cómputo pueda operar es necesario contar con elementos que entreguen información a los usuarios, esto se logra a través de los llamados servidores y sus servicios. Un servidor es software que se encarga de atender las necesidades que los usuarios del sistema tienen al interactuar con el mismo.

El software servidor usualmente está acompañado de software que permite la gestión de bases de datos para almacenar información permitiendo así la existencia de un sistema web escrito en algún lenguaje de programación que funciona del lado del servidor.

El software del usuario es el llamado cliente y permite recibir el servicio requerido por los usuarios, dicho servicio puede ser un navegador consultado una página web, una aplicación que requiere información a través de un servicio web o, inclusive otro servidor que requiere consumir los datos del otro para continuar con sus tareas.

2.2. ELEMENTOS PARA CONSTRUCCIÓN DE LAS VISTAS O PÁGINAS WEB DE UN SISTEMA INFORMÁTICO.

En el desarrollo web existen varios conceptos básicos y confluyen diversas tecnologías que permiten la creación de los sistemas y páginas web como se conocen actualmente. Generalmente se busca tener una separación de la funcionalidad, la parte estética y la estructura del sitio o sistema web.

Para la estructura de los sitios o sistemas web se usa HTML o alguna de sus variantes que permiten maquetar el documento y definir los elementos que contendrá la página o vista del sistema.

En el caso de la funcionalidad de una página web o de una vista de un sistema, generalmente se usa el lenguaje de programación JavaScript ya sea sólo o con la ayuda de algún framework como JQuery, Dojo o Prototype que permite el desarrollo más rápido y sencillo.

Al hacer la parte correspondiente al diseño y estética de la página o vista generalmente se emplea hojas de estilo en cascada (CSS) ya sea de forma nativa o automatizada con herramientas como son LESS y SASS.

2.2.1. HTML

El término HTML significa Hyper Text Markup Language en inglés, al ser un lenguaje de marcado, lo que permite es definir la estructura y elementos que se pueden tener dentro de la página, por mencionar algunos ejemplos de elementos, tenemos video, audio, imágenes, párrafos, títulos.

La versión más reciente de este lenguaje es la 5, que incorpora múltiples mejoras y elementos que constituyen un gran avance frente a su antecesor HTML 4 y su variante, combinada con XML, XHTML. Esta última versión aceptada de HTML puede decirse que ha sido de gran ayuda para la evolución de la web puesto que se ha permitido incorporar los elementos de manera semántica y adaptable a las distintas características de los dispositivos que se conectan a Internet.

El lenguaje HTML se compone de etiquetas jerárquicas que permiten especificar por lo menos tres bloques:

- ✓ html que compone al documento y es la raíz del mismo.
- ✓ head es la sección donde se especifican enlaces y datos para establecer parámetros de la página y traer hojas de estilo.
- ✓ body es el cuerpo tal cual del documento, aquí se incorporan los elementos visuales y controles que contendrá el sistema.

La sección head del documento permite establecer parámetros como se ha comentado anteriormente, para hacerlo, se usan las etiquetas meta; un parámetro importante es el conjunto de caracteres o character encoding que permite establecer cómo es que están guardados los archivos para que al leerlos se desplieguen correctamente y no aparezcan símbolos en vez de letras como ñ, á, é. La etiqueta es la siguiente para usar una codificación UTF8 ampliamente aceptado:

Para HTML 4:

```
<meta http-equiv="Content-Type" content="text/html;charset=UTF-8">
```

Para HTML 5:

```
<meta charset="UTF-8">
```

Otro parámetro que ha incrementado su uso para convertirse en algo casi requerido es el siguiente que permite leer los datos específicos del viewport del dispositivo en el que se está consultando la vista o página, esto brinda la oportunidad de hacer cambios en la manera que se muestra el sistema o la página a través de lo que se conoce como *diseño responsivo*. La siguiente línea muestra cómo se ajusta este parámetro para que sea desplegada una vista o página en sus dimensiones correctas:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

Marco teórico

Otros aspectos importantes dentro de la sección head son la incorporación de las hojas de estilo y en algunos casos los archivos JavaScript aunque esta práctica ya es desaconsejada para los últimos y se prefiere su inclusión al final del cuerpo de la página por motivos de rendimiento.

La instrucción para incluir una hoja de estilo es la siguiente:

```
<link rel="stylesheet" href="ruta_al_archivo.css" />
```

El código necesario para incluir un archivo JavaScript es el siguiente:

```
<script src="ruta_al_archivo.js"></script>
```

En la sección *body* del documento HTML hay diversas etiquetas que permiten incluir contenido de diferentes naturalezas, desde párrafos hasta audio y video en HTML 5, por ejemplo, para incluir un párrafo podemos usar la siguiente etiqueta:

```
<p>Esto es un párrafo.</p>
```

En el caso de las imágenes, se usa una etiqueta como la siguiente:

```

```

En algunos casos hay una pequeña distinción entre HTML y XHTML que corresponde a la adición de una barra diagonal para indicar el fin de la etiqueta en elementos que no tienen precisamente un cierre, por ejemplo, la etiqueta párrafo abre con `<p>` y cierra con `</p>`. Un ejemplo de esta diferencia para XHTML podría ser a la etiqueta de imágenes presentada anteriormente:

```

```

Una de las características de HTML que hacen posible construir los sitios web y sistemas como los conocemos en la actualidad es el uso de atributos en las etiquetas. Retomando la etiqueta anterior, podemos observar que existe una pequeña secuencia de letras predefinida: `alt="Hola"`; la función de esto es permitir especificar texto alternativo en caso de no poder cargar la imagen o estar deshabilitada la carga de imágenes en el navegador.

Existen atributos de suma importancia que permiten darle nombre a elementos y establecer parámetros para indicar cómo debe funcionar o presentarse algún elemento en particular.

El atributo *id* permite hacer identificar elementos dentro de una página web para hacer más fácil el acceso a alguna sección del contenido, establecer la presentación que debe tener fácilmente o vía programación alterarlo.

El atributo *name* es de suma importancia cuando se trabaja con formularios por ejemplo dado que permite identificar los controles involucrados en el ingreso de información.

Marco teórico

El atributo *class* permite establecer la presentación del elemento junto a otros que contienen los mismos estilos y también se ocupa para alterar los elementos vía programación.

En algunas ocasiones las herramientas para agilizar el desarrollo requieren el empleo de atributos específicos con los que modifican el comportamiento o la presentación de uno o más elementos, podría tomarse como ejemplo el framework Foundation, puede consultar la documentación del mismo en la dirección electrónica <http://foundation.zurb.com/sites/docs/>, en este caso, se cuenta con una herramienta para conseguir que elementos de diferentes alturas con un mismo elemento padre sean desplegadas con la misma altura, a continuación se presenta una imagen tomada de la documentación del framework.

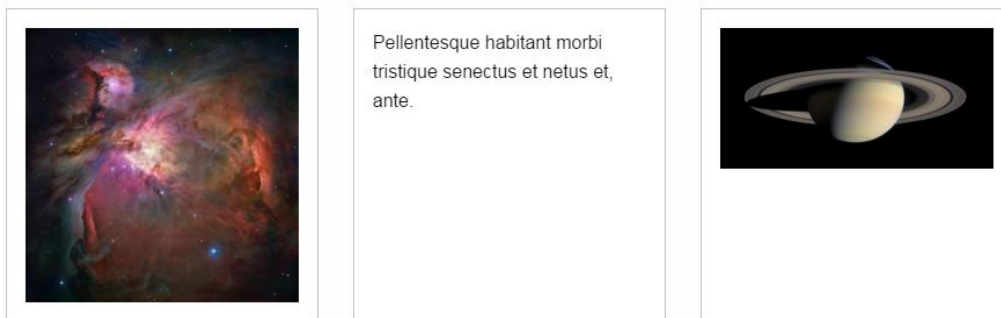


ILUSTRACIÓN 1- FOUNDATION, EFECTOS ACTIVOS

En la siguiente imagen se puede apreciar el mismo contenido sin usar la funcionalidad mencionada, como podrá ver, los elementos han cambiado de tamaño y se muestran muy distintos en cuanto a su altura se refiere si los comparamos con los de la imagen anterior.

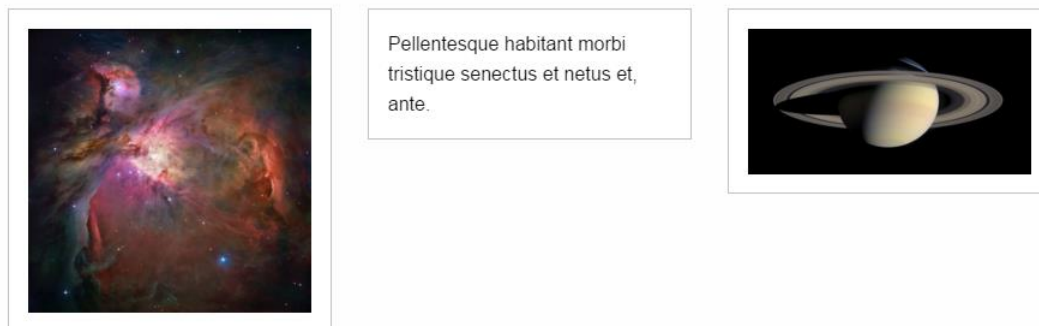


ILUSTRACIÓN 2 - FOUNDATION, EFECTOS ANULADOS

Una parte importante de la programación son los comentarios puesto que permiten documentar el trabajo, en HTML se identifican con las secuencias `<!--` para abrir y `-->` para cerrarlos. Por ejemplo, lo siguiente es un comentario:

Marco teórico

`<!-- Hola, esto no se despliega en el navegador, es un comentario -->`

Debido a la evolución de los estándares y la aceptación de los mismos dependiendo de cada navegador, se emplea un elemento llamado *doctype* cuya función es indicar qué tipo de documento se trata y las características del mismo, esta indicación debe ser la primera línea del documento, para HTML 5 se especifica como sigue:

```
<!DOCTYPE html>
```

En el caso de HTML 4 hay varias maneras de hacerlo, una de ellas, usando el *doctype* estricto que no permite usar elementos de presentación ni obsoletos se muestra a continuación:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
```

El siguiente código es un ejemplo básico de un documento HTML, los archivos tienen extensión *.html*.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>A tiny document</title>
  </head>
  <body>
    <h1>Este es el título</h1>
    <p>Este es un párrafo</p>
  </body>
</html>
```

2.2.2. JAVASCRIPT

JavaScript es un lenguaje interpretado que permite la creación de aplicaciones web del lado del cliente y mejorar notablemente la experiencia de usuario de los sitios web o sistemas.

JavaScript puede incluirse en sitios web por dos maneras, la primera es ingresar el código en el documento HTML a través de la etiqueta `<script>`; la segunda opción es usando un archivo con extensión *.js*.

A continuación se presenta un ejemplo de inclusión de código dentro del documento, usando una función muy simple que únicamente muestra un mensaje:

```
<script>
  alert('Mensaje de bienvenida');
</script>
```

Marco teórico

El resultado de este código es el siguiente:

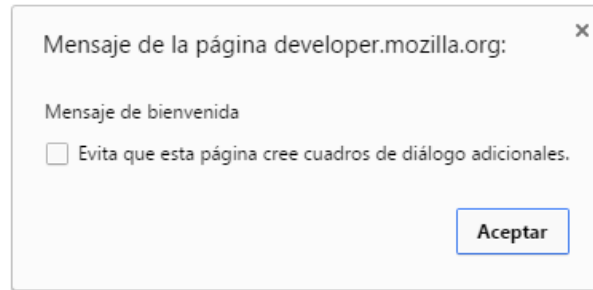


ILUSTRACIÓN 3- EJEMPLO DE ALERTA JAVASCRIPT

Para incluir código JavaScript en un archivo separado, se ocupa la siguiente etiqueta:

```
<script src="js/jquery-1.9.1.js"></script>
```

Dentro del archivo externo no hace falta hacer indicaciones con etiquetas o algo similar, sino que simplemente se inserta el código. También es de hacer notar que existen comentarios de bloques que inician con la secuencia /* y cierran con */ o bien, podría emplearse dos diagonales // indicando un comentario de una línea.

En la siguiente imagen se presentan las líneas iniciales del código de la biblioteca JQuery, se pueden apreciar comentarios de bloques, después como inicia el código fuente de la biblioteca y algunos comentarios de una sola línea.

```
1  /*!  
2  * jQuery JavaScript Library v1.9.1  
3  * http://jquery.com/  
4  *  
5  * Includes Sizzle.js  
6  * http://sizzlejs.com/  
7  *  
8  * Copyright 2005, 2012 jQuery Foundation, Inc. and other contributors  
9  * Released under the MIT license  
10 * http://jquery.org/license  
11 *  
12 * Date: 2013-2-4  
13 */  
14 (function( window, undefined ) {  
15  
16 // Can't do this because several apps including ASP.NET trace  
17 // the stack via arguments.caller.callee and Firefox dies if  
18 // you try to trace through "use strict" call chains. (#13335)  
19 //  
20 // ...  
21 }  
22 )  
23 )  
24 )
```

ILUSTRACIÓN 4 - EJEMPLO DE CÓDIGO JAVASCRIPT

2.2.3. CSS

CSS es la abreviación de Cascade Style Sheets u hojas de estilo en cascada, se trata de una tecnología que permite crear efectos y mejorar notablemente la presentación de la página o sistema que se está creando, por ejemplo, se presenta una pequeña sección sin estilos de la documentación del framework Foundation 5 que puede consultarse en la dirección electrónica <http://foundation.zurb.com/sites/docs/v/5.5.3/>.

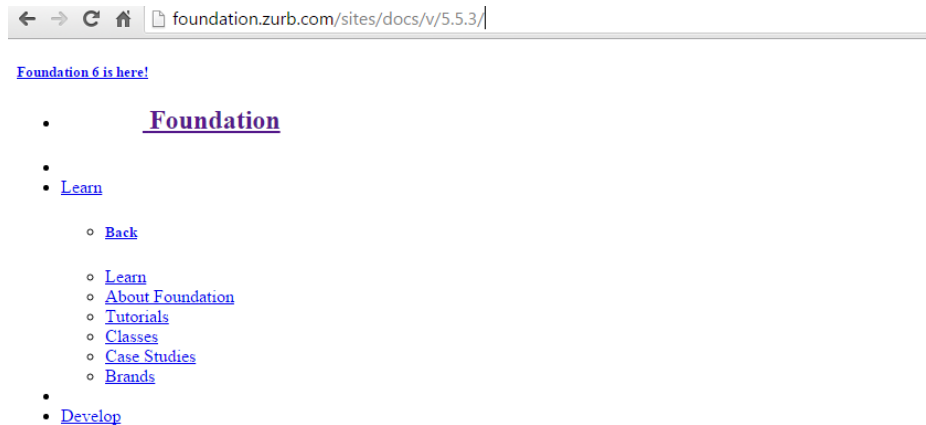


ILUSTRACIÓN 5 - PÁGINA WEB SIN ESTILOS

Puede observarse que la estructura de la página contiene la información organizada jerárquicamente pero no es fácil digerirla puesto que se requiere leer un documento de longitud considerable para encontrar lo que se busca además de ser bastante monótono e incómodo a la vista, a continuación se presenta la versión que aparece normalmente cuando se tienen habilitados los estilos correspondientes:

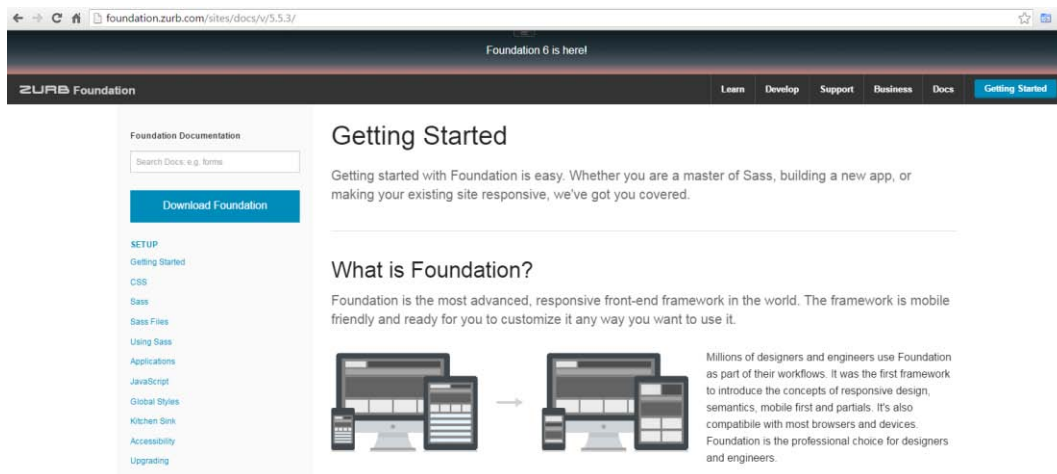


ILUSTRACIÓN 6 - PÁGINA WEB CON ESTILOS

Marco teórico

Se puede notar un drástico cambio con respecto a la imagen anterior, con ayuda de los estilos se ha reorganizado la posición de todos los elementos de la página y se agregaron colores, espacios e imágenes de fondo que permiten los efectos visuales presentados.

El código CSS puede insertarse en los documentos HTML de tres maneras, como un atributo de un elemento HTML, dentro de la estructura del documento mismo o como un archivo independiente, la última opción es la recomendada debido a que permite la reutilización del código y separa la presentación de la estructura del documento.

Cuando se desea insertar código CSS en un elemento HTML de manera rápida, puede emplearse el atributo *style* de los elementos, a continuación se presenta un ejemplo que centra el texto del párrafo:

```
<p style="text-align: center;">Texto centrado</p>
```

El código CSS se inserta en el documento HTML usado la etiqueta *<style>* en la sección *head* del documento, a continuación se presenta un pequeño ejemplo de esta práctica:

```
<style>
  img{
    max-width: 290px;
    max-height: 150px;
    margin: 0px 20px;
    border: none;
  }
</style>
```

En caso de requerir incluir el código CSS dentro de un archivo separado se emplea la etiqueta *<link>* para indicar donde se encuentra el archivo. A continuación se presenta un ejemplo.

```
<link rel="stylesheet" href="lib/js/jquery-ui-1.10.3.custom/css/ui-lightness/jquery-ui-1.10.3.custom.css">
```

CSS emplea el concepto de selector con lo que se puede especificar los estilos aplicables a un elemento, dando prioridad a lo particular frente a lo general y en caso de tener la misma prioridad lo que aparezca después en el archivo de código fuente.

Algunos de los selectores más usados en CSS son las clases que inician con un punto ".", los nombres de elementos HTML que se escriben tal cual y los identificadores de elementos HTML correspondientes al atributo id del mismo que

Marco teórico

inician con el símbolo de numeral “#”. A continuación se presenta un ejemplo las reglas aplicadas a elementos con clase `.ui-widget-header`.

```
.ui-widget-header {  
  border: 1px solid #e78f08;  
  background: #f6a828 url(images/ui-bg_gloss-wave_35_f6a828_500x100.png) 50% 50%  
  repeat-x;  
  color: #ffffff;  
  font-weight: bold;  
}
```

Cada regla se compone del nombre de la misma seguida de dos puntos, el valor de la propiedad y un punto y coma “;” indicando el fin de la regla. Al igual que en diversos lenguajes, se pueden incorporar comentarios; se inician con la secuencia `/*` y se cierran con `*/` pudiendo abarcar más de una línea de código.

2.2.4. HERRAMIENTAS PARA EL DESARROLLO WEB

Existen diversas herramientas que permiten agregar interactividad o simplifican en gran medida el desarrollo de sitios web y sistemas debido a que ofrecen complementos o efectos para incorporarlos fácilmente donde sean necesarios. A continuación se describen dos bibliotecas de gran utilidad para cualquier desarrollo.

2.2.4.1. JQUERY

jQuery es una biblioteca escrita en lenguaje JavaScript que permite escribir código de acuerdo a nuestras necesidades como desarrolladores empleando mucho menos esfuerzo y más compacto entre otras mejoras.

La biblioteca jQuery se distribuye bajo licencia MIT lo que la hace bastante fácil de obtener y trabajar con ella.

jQuery simplifica el trabajo al proveer funciones o metodologías para diferentes rubros:

- ✓ Eventos. Permite la asociación de funciones para reaccionar a la interacción del usuario con la página web que se consulta.
- ✓ Efectos. Permite hacer animaciones fácilmente sobre los elementos HTML para aparezcan, se oculten y se cambien los estilos por ejemplo.
- ✓ Peticiones a servidores. Permite hacer consultas a través de internet sin la necesidad de recargar la página por parte del usuario logrando así una mejor experiencia de usuario cuando así se requiere.

Marco teórico

Cabe mencionar que se pueden crear plugins que permiten ampliar la funcionalidad de la biblioteca sin muchas complicaciones.

La biblioteca puede consultarse y obtener una copia en la dirección electrónica <http://jquery.com/>.

2.2.4.2. JQUERY UI

Es una biblioteca que permite agregarle componentes e interacción a las páginas web o aplicaciones web, está desarrollada empleando jQuery.

jQuery UI presenta funciones interactivas, entre las que destacan están:

- ✓ Selección: permite seleccionar uno o varios elementos de un grupo.
- ✓ Arrastrar: permite la reubicación de elementos con el mouse.
- ✓ Redimensionar: permite cambiar el tamaño de un elemento con el mouse.

Además de interactividad, la biblioteca provee componentes que permiten ingresar o mostrar información de manera cómoda, algunos componentes o widgets son:

- ✓ Acordeón: permite mostrar información en un espacio reducido al expandir y contraer el espacio destinado a la presentación de la misma.
- ✓ Seleccionador de fechas: despliega un calendario del que puede escogerse una fecha, de esta manera se simplifica el ingreso de fechas y problemas por formatos erróneos.
- ✓ Ventanas de diálogo: permite la aparición de una pequeña ventana simulada que despliega contenido.
- ✓ Barra de progreso: permite mostrar visualmente el avance de un proceso en una barra.
- ✓ Pestañas: permite la organización de contenido organizado en pestañas.

El uso de la biblioteca es bastante sencillo, por ejemplo para el widget date picker o seleccionador de fechas, se emplea una caja de texto con un identificador, por ejemplo, podría usarse el siguiente código:

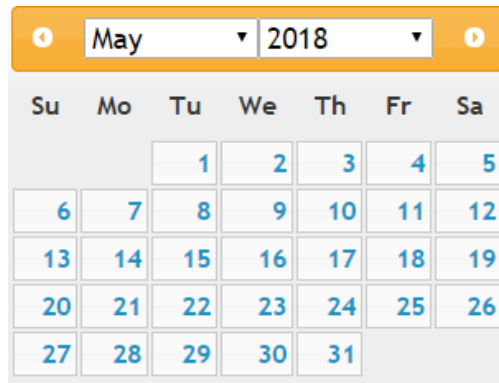
```
<input type="text" name="fecha_inicio" id="fecha_inicio" >
```

Para proveer la funcionalidad, se emplea un pequeño código que inicializa y activa el control indicando parámetros de configuración opcionalmente:

```
$("#fecha_inicio").datepicker({  
  yearRange: "-3:+3",  
  dateFormat: "yy-mm-dd"  
});
```

Marco teórico

El resultado de la ejecución del código anterior es un calendario similar al que se muestra en la siguiente imagen:



The image shows a jQuery calendar widget for the month of May 2018. At the top, there is a header with a left arrow, the text 'May', a dropdown arrow, the text '2018', another dropdown arrow, and a right arrow. Below the header is a grid of days. The days of the week are labeled as Su, Mo, Tu, We, Th, Fr, Sa. The dates are arranged in a grid starting from the first day of the month (1) on a Tuesday. The dates 1 through 31 are displayed in a grid format.

Su	Mo	Tu	We	Th	Fr	Sa
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

ILUSTRACIÓN 7 - CALENDARIO JQUERY

2.3. BASES DE DATOS

Una empresa, evento, programa social o cualquier entidad que se conozca requiere que se almacene información de su razón de ser, clientes o beneficiarios por lo que se vuelve importante tener una manera eficiente de hacerlo al mismo tiempo que se mantiene organizada y lista para usar en los procedimientos correspondientes. Las bases de datos cumplen estas funciones, permiten almacenar de forma organizada, confiable y con la mayor seguridad posible.

2.3.1. DEFINICIÓN DE BASE DE DATOS.

En manera simple, una base de datos es una herramienta con la que podemos almacenar información de manera estructurada, organizada y eficaz que permite la relación entre la misma generando múltiples posibilidades de uso de manera confiable.

Una base de datos puede contener cualquier tipo de información, desde una lista de nombres de usuario y contraseñas hasta el patrón electoral de un país, los proveedores de una empresa, su historial de transacciones bancarias entre otras cuestiones.

2.3.2. BASES DE DATOS RELACIONALES

Existen diversos tipos de bases de datos, uno de ellos, bastante aceptado, es el de modelo de bases de datos relacionales que permite la organización de la información en tablas usando el concepto de entidad.

Marco teórico

Una entidad en una base de datos relacional es una abstracción de un objeto o ente del que se desea almacenar información, llamado atributo. Como se mencionó en el párrafo anterior, una entidad tiene una representación de tabla, donde las columnas son los atributos o campos y las filas representan un registro.

Los atributos de una entidad se representan en diversos tipos de datos para su almacenaje y tratamiento cuando sea necesario, entre otros, están:

- ✓ Enteros, permiten almacenar valores enteros como su nombre lo indica
- ✓ De coma flotante, permiten almacenar valores fraccionarios
- ✓ Carácter, permite almacenar símbolos alfanuméricos
- ✓ Cadenas y texto, series de caracteres de longitud específica o variable

2.3.3. RELACIONES ENTRE ENTIDADES

La información por sí sola no aporta valor a una empresa o institución sino que el tratamiento de la misma y su interpretación son lo importante; para lograr este objetivo, contamos con las relaciones que nos permiten establecer vínculos, por decirlo sencillamente, entre las entidades.

Podría tomarse como ejemplo un caso hipotético de una escuela en la que se tiene una lista de los estudiantes con mejor desempeño académico en un curso, a continuación puede verse las primeras dos filas:

TABLA 1- EJEMPLO DE LISTA

Lugar de aprovechamiento	Nombre	Grupo	Calificación
1	Marcos	101	10
2	Juan	101	9.99

Podríamos trasladar la tabla anterior a una relación entre al menos dos entidades, por un lado la entidad alumno que describe a los alumnos inscritos en general con sus datos particulares y, por otro lado una entidad curso que describe la calificación de los alumnos entre otros datos.

Las relaciones se realizan empleando conceptos llamados llaves primarias y llaves foráneas que permiten establecer los vínculos entre las entidades. Las llaves primarias fundamentalmente identifican al registro en cuestión y las llaves foráneas tienen la función de hacer referencia a una llave primaria de otra entidad. En el ejemplo, podría ser una llave primaria la matrícula del estudiante para la entidad alumno y convertirse en llave foránea en la entidad curso.

La importancia de las relaciones y contar con un buen diseño de ellas radica en su empleo para realizar consultas a la base de datos, estas consultas son las responsables del aprovechamiento de los datos puesto que con ellas se puede

Marco teórico

obtener información de las tablas, agregar y modificar registros entre otras operaciones.

2.3.4. MANEJO DE LA INFORMACIÓN DENTRO DE LA BASE DE DATOS.

2.3.4.1. SISTEMAS DE ADMINISTRACIÓN DE BASES DE DATOS

Debido a la naturaleza de las bases de datos, es necesario contar con una herramienta que se encargue de manejar la base de datos desde su creación, proporcione seguridad, acceso a la información y permita realizar acciones administrativas a la base de datos propiamente. Esta herramienta es conocida como sistema manejador de base de datos (DBMS por sus siglas en inglés). A continuación se presenta una lista de algunos DBMS bastante aceptados.

- ✓ MySQL . Es un manejador ampliamente utilizado distribuido por Oracle en diversas ediciones cada una enfocada a las diversas necesidades de las instituciones, desde una página web sencilla hasta aplicaciones empresariales con gran cantidad de operaciones.
- ✓ PostgreSQL. Es un DBMS de código abierto desarrollado por la comunidad con gran aceptación y tiene más de 15 años de desarrollo, tiene reputación de ser muy confiable.
- ✓ DB2. Es un DBMS distribuido por IBM dirigido hacia aplicaciones empresariales.

2.3.4.2. SQL

SQL es un lenguaje de programación que permite la manipulación de la base de datos a través de un DBMs. Este lenguaje permite la definición de la base de datos, tablas y gran variedad de elementos que puede contener así como la manipulación de la información para extraerla, modificarla o registrarla de acuerdo a las necesidades del usuario.

Este lenguaje es comúnmente utilizado dentro de programas o sistemas que requieren información de una base de datos, en algunas ocasiones se integran extensiones a los programas o sistemas que pueden interactuar con el DBMS y de esta manera tener acceso a los datos.

A continuación se presenta un ejemplo de definición de una tabla:

Marco teórico

```
CREATE TABLE admins
(
  usuario character varying(15) NOT NULL,
  contra character varying(15) NOT NULL,
  CONSTRAINT usuario_pk PRIMARY KEY (usuario , contra )
)
```

La inserción de datos en una base de datos puede hacerse de manera sencilla con este lenguaje de programación, a continuación se presenta un ejemplo:

```
INSERT INTO `test`.`prueba` (`id`, `fecha`) VALUES (NULL, CURRENT_DATE());
```

Otra de las operaciones bastante usuales tiene que ver con la extracción de información de la base de datos, frecuentemente puede volverse muy compleja una consulta debido a la cantidad de tablas requeridas o temas avanzados como son las subqueries (dicho de manera simple, una consulta dentro de otra consulta), a continuación se presenta un ejemplo de este tipo de consulta:

```
SELECT * FROM `prueba`;
```

En algunas ocasiones es necesario realizar más de una operación para almacenar información sobre diferentes temas pero es necesario hacerlo en una sola operación, cuando esto sucede se emplean las llamadas transacciones. Las transacciones permiten agrupar varios comandos en una sola operación o ninguno si hay algún error o así son las reglas del negocio.

2.4. LENGUAJES DE PROGRAMACIÓN

Existen diversos lenguajes de programación que permiten el desarrollo web en el lado del servidor permitiendo la creación del contenido y las acciones necesarias para que un sistema de cómputo pueda existir. A continuación se presenta una breve descripción de algunos de los lenguajes de programación más comunes.

2.4.1. PHP

PHP es un lenguaje de programación interpretado que ha sido pensado para el desarrollo de sistemas web aunque tiene capacidad de funcionar en diversas maneras ya sea como scripts para una línea de comandos o como un programa de escritorio con interfaz gráfica como los que se usan en la vida diaria.

Marco teórico

Este lenguaje es muy fácil de aprender lo que permite que gran cantidad de personas estén interesadas en trabajar con él, se puede hacer uso de características avanzadas por lo que es ampliamente aceptado también por profesionales.

Uno de los datos interesantes sobre el lenguaje es que en la actualidad es uno de los lenguajes más populares la creación de aplicaciones web y páginas web además de ser soportado por diversas marcas y compañías proveedoras de alojamiento en línea.

La creación de documentos web con este lenguaje es bastante simple, únicamente se requiere insertar el código en los documentos HTML dentro de las etiquetas `<?php` y `?>` para poder ejecutarlo.

Por ejemplo, el siguiente código genera un documento HTML:

```
<html>
  <head>
    <title>Ejemplo de uso de PHP</title>
  </head>
  <body>
    <?php
      echo "<p> Este es un documento que genera contenido dinámico<p>";
    ?>
  </body>
</html>
```

2.4.2. ASP.NET

ASP.NET es un entorno de trabajo del lado del servidor distribuido por Microsoft que permite la creación de diversos sistemas de cómputo, para poder usar el lenguaje se requiere contar con el servidor IIS que también provee Microsoft.

El entorno puede usarse en diferentes maneras dependiendo de las habilidades y estilo de trabajo del desarrollador, estas son:

- ✓ Web Pages donde puede insertarse el código del lenguaje en una plantilla HTML
- ✓ Web Forms que permite usar un modelo orientado a eventos en el que el navegador envía los formularios y recibe como respuesta un documento completo, esto facilita el manejo de sesiones y de estados en algún procedimiento.

Marco teórico

- ✓ MVC que permite una programación basada en patrones

Cabe mencionar que el entorno de trabajo está basado en la plataforma .NET de Microsoft por lo que los elementos comparten características y no son completamente independientes.

Puede usarse lenguajes de programación como C#, VB.Net, Jscript y J# para crear las aplicaciones web más fácilmente.

2.4.3. PYTHON

Es un lenguaje de programación de alto nivel de propósito general, interactivo, interpretado y orientado a objetos creado por Guido van Rossum a principios de los 90s. El lenguaje es reconocido por ser fácil de aprender y bastantes personas lo recomiendan como primer lenguaje de programación.

Python tiene gran aceptación y puede enfocarse al desarrollo web gracias a los entornos de trabajo que se han creado para el lenguaje, uno de ellos es Django, que es libre y de código abierto.

La idea detrás de Django es que sea muy rápido el desarrollo de aplicaciones, proteja frente a los ataques informáticos más frecuentes, tenga una excelente escalabilidad y contenga gran cantidad de funcionalidades listas para usar.

2.4.4. RUBY

Ruby es un lenguaje orientado a objetos de código abierto creado por Yukihiro Matsumoto, también conocido como "Matz". El lenguaje fue liberado en 1995 y está influido por los lenguajes Perl, Smalltalk, Eiffel, Ada y Lisp.

Ruby tiene un popular framework llamado Rails que está diseñado para que el desarrollador escriba la menor cantidad de código posible y está enfocado en tener una alta productividad.

Un detalle interesante del lenguaje es que permite el manejo de hilos independientemente de si el sistema operativo lo permite.

2.4.5. JAVA

Java es una plataforma de desarrollo y un lenguaje de programación de propósito general que diseñado para funcionar sobre diversas plataformas al ser un lenguaje interpretado.

El lenguaje ha sido aceptado ampliamente y se ha llevado a diversas categorías de dispositivos, desde teléfonos inteligentes hasta las aplicaciones web y nativas empresariales más demandantes.

Marco teórico

La capacidad de la plataforma es tan grande que puede usarse de diversas formas, una de ellas es a través las llamadas Java Web Application que generan páginas web y contenido dinámico, están compuestas por tecnologías como JavaServer Pages (JSP), servlets y JavaBeans permitiendo almacenar y procesar información temporalmente, interactuar con otros sistemas vía servicios web. Se han creado varios entornos de trabajo con el fin de hacer el desarrollo de aplicaciones más sencillo, uno de ellos es JavaServer Faces.

La plataforma empresarial de Java es ampliamente empleada en la industria debido a su robustez y por la facilidad que otorga para el desarrollo de sistemas de cómputo de gran escala.

2.5. TRANSPORTE Y ALMACENAMIENTO DE DATOS.

Existen ocasiones en que es necesario tener en cuenta que los sistemas requieren intercomunicación de unos con otros y también que aunque algunas las bibliotecas usadas dentro del desarrollo de un sistema son generalmente escritas en el lenguaje requieren un formato especial para la adquisición de datos e inicialización de parámetros, es por ello que entran en juego dos grandes formatos ampliamente aceptados XML y JSON.

2.5.1. XML

El Extensible Markup Language (XML) es un lenguaje de marcado que permite almacenar y transportar datos independientemente del lenguaje y sistema operativo usado, el lenguaje está pensado para ser descriptivo, no tiene etiquetas predefinidas, es estrictamente jerárquico, se enfoca en la descripción del contenido y está diseñado para poder ser leído por humanos o máquinas.

La sintaxis básica del lenguaje es simple:

- ✓ Debe tener un elemento padre
- ✓ Todos los elementos deben tener una etiqueta de cierre, por ejemplo
<carro>Rojo</carro>
- ✓ Los elementos deben estar bien agrupados, haciendo caso a su nivel jerárquico.

Un ejemplo de un documento XML es el siguiente:

```
<auto>
  <marca>Ford</marca>
  <color>Rojo</color>
</auto>
```

2.5.2. JSON

JSON (JavaScript Object Notation) Es un formato de transporte de datos fácil de generar e interpretar por humanos y máquinas, es independiente del lenguaje de programación usado así como sistema operativo.

El formato se compone de dos formas:

Objetos: son una serie de pares nombre-valor separados por dos comas (,), estos pares se componen del nombre entre comillas, dos puntos (:) y el valor. El nombre y valor deben estar entre comillas. Ejemplo:

```
{"nombre":"Marcos", "apellido":"Martinez"}
```

Arrays: Los arrays son series de valores separados por comas (,). Por ejemplo tenemos:

```
["uno", "dos", "tres"]
```

3. ANÁLISIS DEL PROBLEMA

3.1. DESCRIPCIÓN DE LA SITUACIÓN ACTUAL

En la actualidad no hay un sistema de cómputo general que permita la operación del programa, hay casos, como el de la Facultad de Ingeniería y la Facultad de Química, donde se cuenta con sistemas adaptados a las necesidades de las mismas.

En las siguientes secciones se presentan los procesos para la operación del Programa de Apoyo Nutricional.

3.1.1. REGISTRO DE CANDIDATOS AL PROGRAMA DE APOYO NUTRICIONAL

El proceso para solicitar el acceso al Programa de Apoyo Nutricional se realiza semestralmente para primer ingreso así como para renovación en las fechas que estipula la convocatoria correspondiente al semestre, para ello se requiere diversos documentos probatorios que muestren en primera instancia el cumplimiento de los requisitos de la beca y la identificación de la persona que solicita el apoyo.

Los documentos requeridos para aplicar a la beca son:

- ✓ Comprobante de inscripción
- ✓ Credencial de la UNAM vigente
- ✓ Historial académico actualizado
- ✓ CURP
- ✓ Carta de autorización de uso de datos personales
- ✓ Comprobante de domicilio
- ✓ Comprobante de ingresos o algún documento donde se indique a cuánto ascienden los ingresos mensuales
- ✓ Fotografía reciente en tamaño infantil
- ✓ Carta de exposición de motivos para solicitar el apoyo
- ✓ Consumo de alimentos por parte de los beneficiarios

Una vez se han registrado los aspirantes a la beca entregando la documentación requerida, se realiza un proceso de selección en el cual trabajan los comités de cada escuela o facultad y Fundación UNAM, una vez que se concluye este proceso se requiere se digitalicen los documentos entregados en papel y se entregue en formato digital a la Fundación UNAM para que ellos tengan documentación que justifique el uso de recursos a los donantes que permiten la operación del programa

Análisis del problema

3.1.2. CONSUMOS DE LOS BENEFICIARIOS

El Programa de Apoyo Nutricional ofrece a sus beneficiarios una comida al día, ya sea desayuno o comida, lo que en el mejor de los casos genera un registro por día hábil. Este registro se hace de manera automática en un sistema local que tiene la cafetería donde se presta el servicio, se identifica al estudiante con su huella digital previamente cargada en dicho sistema para que se le otorgue el beneficio del día correspondiente.

Al final de la semana se requiere que se tenga un reporte de los consumos que han generado los beneficiarios con el fin de establecer la asistencia a los servicios ya que es un requisito para continuar con el apoyo además de tener elementos en caso de tener una auditoría por parte de Fundación UNAM.

Al llegar a fin de mes también se hace un concentrado de los consumos ocurridos durante el transcurso del mismo por las mismas razones que se comentan en el párrafo anterior.

3.1.3. ENCUESTAS DE SATISFACCIÓN

Cada cierto tiempo, definido de acuerdo a la planeación del programa por parte de Fundación UNAM, se realiza una encuesta de satisfacción a los beneficiarios con el fin de obtener realimentación acerca del funcionamiento del programa así como la calidad del servicio y alimentos que se ofrecen en las cafeterías prestadoras del servicio.

La respuesta en tiempo y forma a las encuestas de satisfacción son un requisito indispensable para la permanencia en el apoyo por parte de los alumnos por lo que usualmente se tiene una gran cantidad de participación en las mismas.

Al concluir el periodo de recepción de respuestas a la encuesta, los responsables de cada escuela o facultad generan reportes con la información obtenida en el proceso concentrándola y elaborando gráficas que permiten una interpretación de los resultados más fácil.

3.2. PROPUESTA DE SOLUCIÓN AL PROBLEMA

Debido a las amplias necesidades detectadas y descritas en la sección anterior se propone desarrollar un sistema informático que permite la gestión y automatización de la mayoría de las tareas que se realizan manualmente en la actualidad implicando una gran inversión de esfuerzo y tiempo para completarlas.

Análisis del problema

3.2.1. DEFINICIÓN DE REQUERIMIENTOS DEL SISTEMA

El sistema debe permitir el registro de estudiantes que aspiren a obtener la beca, solicitando información de los mismos como es:

- ✓ Nombre
- ✓ Apellidos paterno y materno
- ✓ Teléfono fijo y celular
- ✓ Correos electrónicos
- ✓ Estado civil
- ✓ Sexo
- ✓ Fecha de nacimiento
- ✓ CURP
- ✓ Domicilio (calle y número, colonia, municipio o delegación, estado, código postal)
- ✓ Carrera
- ✓ Si trabaja al momento de solicitar la beca y aporta al ingreso familiar
- ✓ Las personas con quien vive
- ✓ Quién es el principal sostén económico familiar
- ✓ A cuánto ascienden los ingresos mensuales
- ✓ Si la casa donde vive es propia, rentada o de algún familiar
- ✓ El tiempo promedio de trayecto hacia la facultad
- ✓ Documentación en formato PDF (historial académico, comprobante de inscripción con horario, copia de la credencial UNAM vigente, carta de exposición de motivos, comprobante de percepciones, comprobante de domicilio, fotografía en formato JPG del rostro)

Al cumplir con el requisito de registrarse y procesarse correctamente, el sistema debe generar un par de documentos que permitirán al aspirante comprobar el cumplimiento de la etapa indicada así como un documento que se entrega a los responsables del programa en la facultad de adscripción para la creación del expediente.

El sistema debe contar con funciones que permitan simplificar el conteo de consumos para aliviar la carga laboral que implica contar y procesar los consumos de alimentos, se debe reportar la cantidad de desayunos, comidas y total de consumos que se tienen en algún lapso de tiempo.

Se debe contar con una encuesta que permita obtener las opiniones de los beneficiarios en cuanto a diversos aspectos:

- ✓ Variedad de los alimentos, ingredientes, si se ofrece alguna bebida acompañando los alimentos, si se ofrecen los complementos adecuados para comida y desayuno.
- ✓ Calidad de platillos ofrecidos en desayuno y comida
- ✓ Respeto al horario de servicio

Análisis del problema

- ✓ Tiempos de espera para recibir el servicio
- ✓ Limpieza y calidad del servicio ofrecido por el personal de la cafetería
- ✓ Breve comparación respecto al mes anterior

El sistema debe contar con una sección administrativa que permitirá facilitar la operación del sistema y ofrecerá los reportes necesarios tanto para el responsable del programa así como para Fundación UNAM.

El sistema debe tener integración con Microsoft Office Excel para la generación de reportes con el fin de facilitar el manejo de la información por cualquier persona, esté o no familiarizada con el sistema de cómputo.

Debe permitir verificar los datos de una solicitud en particular, visualizar los documentos entregados en el registro, permitir la actualización del estado de solicitud y generar la lista de los candidatos compatible con software de ofimática para su validación y la eventual aceptación de aspirantes al Programa de Apoyo Nutricional.

El sistema debe contar con una sección para reportar los resultados de la encuesta en forma amigable empleando gráficas de pie que permitan simplificar la interpretación de los resultados y limitar la cantidad de esfuerzo requerido para completar la tarea.

Los resultados de la encuesta son requeridos de acuerdo a parámetros como fecha de inicio y final de la evaluación permitiendo así conocer la información independientemente del periodo de servicios evaluado, así mismo se tiene la posibilidad de limitar el reporte a una sola persona para verificar las respuestas enviadas y de esta manera saber si se ha contestado la encuesta en algún tiempo particular.

4. DISEÑO DEL SISTEMA

Dada la posibilidad de tener diversos perfiles de usuarios, casos especiales en el futuro para alguna escuela o facultad y los recursos con que se cuente para operar el sistema propuesto, se ha decidido adoptar un diseño modular del sistema permitiendo de esta manera tener simplemente la funcionalidad que soporte: encuesta, el registro de solicitudes para obtener una beca, el conteo de consumos automatizado y/o la parte administrativa del sistema.

Para lograr lo anterior se hace uso de la arquitectura modelo-vista-controlador (MVC) con apoyo en la programación orientada a objetos (POO) lo que permite que el sistema sea más simple para darle mantenimiento, se pueda actualizar y sea confiable.

El sistema emplea una base de datos relacional para brindar una estructura a los datos y aprovechar los beneficios de este tipo de base de datos, en particular se usó el sistema manejador de bases de datos MySQL ya que se puede usar la versión “MySQL Community Edition” que se distribuye bajo la licencia GPL por lo que puede usarse libremente. Se ha elegido al sistema MySQL por varias razones, entre ellas, su popularidad, compatibilidad con estándares y herramientas que permiten interactuar con el servidor.

Para la atención a las peticiones del lado del servidor se eligió el lenguaje de programación PHP que nos permite tener entornos de programación con recursos limitados, tiene extensa documentación y una comunidad bastante grande que permite encontrar desarrolladores así como entornos de producción con relativa facilidad.

En el entorno de ejecución del cliente se deben tener en cuenta diversos aspectos funcionales que permiten una interacción más simple entre el usuario y el sistema como son calendarios para ingresar la fecha, notificaciones como ventanas emergentes (dentro de la misma página) entre otros elementos; de esta manera es que se ha procurado tener una experiencia de usuario más amigable y permitir que el trabajo diario con el sistema sea más fácil. Estos efectos, elementos y acciones se desarrollan con los estándares *de facto* hojas de estilo (CSS en inglés) para cambiar la presentación del sistema y el lenguaje de programación JavaScript además del lenguaje de marcado HTML en su versión 5 que nos permite crear la estructura para presentar el sistema.

4.1.DISEÑO DE LA BASE DE DATOS

Se ha mencionado que el sistema debe ser modular por lo que la base de datos cobra un papel protagonista dentro del funcionamiento al proveer toda la información requerida para trabajar, también es importante hacer notar que al requerirse que el sistema tenga la menor cantidad de mantenimiento posible sin

afectar su funcionamiento, es necesario tener una base de datos robusta, que tenga un buen rendimiento por diseño y sobre todo permita al sistema ser escalable.

4.1.1. MODELO DE DATOS

Column Name	Data Type	Constraints
nombre	VARCHAR(100)	
ape_materno	VARCHAR(100)	
ape_paterno	VARCHAR(100)	
cuenta	INT(11)	PRIMARY
telefono	VARCHAR(10)	
celular	VARCHAR(13)	
email	VARCHAR(50)	
email2	VARCHAR(50)	
carrera_id	INT(11)	FK
edo_civil	SMALLINT(6)	FK
trabaja	CHAR(2)	
aporta	CHAR(2)	
compania_alumno_id	INT(11)	FK
sosten_economico	INT(11)	FK
ingreso_mensual	INT(11)	FK
tipo_casa	INT(11)	FK
tiempo_traslado	INT(11)	FK
curp	CHAR(18)	
sexo	CHAR(1)	
fecha_nacimiento	DATE	
fecha_registro	DATE	
estado_beca_id	INT(11)	FK

Index Name	Index Type
PRIMARY	PRIMARY
cuenta	PRIMARY
fk_alumno_sosten_economico_idx	FK
fk_alumno_carrera1_idx	FK
fk_alumno_compania_alumno1_idx	FK
fk_alumno_edo_civil1_idx	FK
fk_alumno_estado_beca1_idx	FK
fk_alumno_ingreso_mensual1_idx	FK
fk_alumno_tipo_casa1_idx	FK
fk_alumno_tiempo_traslado1_idx	FK

ILUSTRACIÓN 8 - TABLA ALUMNO

Para almacenar la información de manera estructurada, confiable, no redundante, consistente e íntegra se definieron diversas tablas, en las siguientes secciones se presenta la descripción del diseño y una breve explicación de su razón de ser.

4.1.1.1. DICCIONARIO DE DATOS

La base de datos se compone de 22 tablas enfocadas a guardar datos de los alumnos que realizan el registro y contestan las encuestas. La base de datos se diseñó de tal manera que sea posible limitar el acceso a la información de acuerdo a las necesidades del programa y requisitos de seguridad teniendo en cuenta la sensibilidad de los datos personales que se manejan en el Programa de Apoyo Nutricional.

A continuación se explica más a detalle los elementos que conforman la base de datos:

4.1.1.1.1. TABLA ALUMNO

Esta tabla es la más importante puesto que contiene la información personal de los solicitantes por lo que se utiliza para identificar a los aspirantes o becarios dependiendo de su estado en las diversas etapas dentro del período del Programa de Apoyo Nutricional.

La llave primaria de esta tabla es el campo *cuenta* que almacena como su nombre lo indica el número de cuenta de los alumnos, asimismo tiene varias claves foráneas que permiten mantener la integridad referencial de la información.

La tabla alumno es referida por las tablas documento, domicilio y enc_respuestas donde es necesaria información almacenada en esta tabla.

La tabla alumno refiere algunos campos de las tablas sosten_economico, carrera, compañía_alumno, edo_civil, estado_beca, ingreso_mensual, tiempo_traslado y tipo_casa. La razón por la que estos datos están fuera de la tabla alumno es para

Diseño del sistema

tener la posibilidad de hacer más eficientes las búsquedas y extraer estadísticas de la información, al mismo tiempo que se tiene una cantidad limitada de parámetros haciendo más fácil la manipulación de la información por parte del encargado de la ejecución del Programa de Apoyo Nutricional en la escuela o facultad.

La tabla alumno cuenta con varios índices que mejoran el rendimiento de la base de datos, estos índices corresponden a los campos que confirman las llaves primarias y foráneas como se puede apreciar en la imagen.

4.1.1.1.2. TABLA DOCUMENTO

Esta tabla es importante puesto que tiene la información de los documentos que se han entregado y se han validado, es necesario que se validen los archivos puesto que son evidencias que pudiesen requerirse en una auditoría.

La llave primaria de la tabla es el campo *documento_id* y tiene una llave foránea que es el campo *tipo_doc*.

La tabla documento es referenciada desde la tabla alumno para complementar la información del registro que se tiene del aspirante o becario.

La tabla documento refiere a la tabla tipo_documento para identificar el contenido del documento que se registra.

Esta tabla, tiene como índices los correspondientes a la llave primaria y foránea.

Column	Field	Index
documento_id	INT(11)	PRIMARY
tipo_doc	INT(11)	
entregado	CHAR(1)	
validado	CHAR(1)	
uri	VARCHAR(200)	
no_cuenta	INT(11)	

Indexes

- PRIMARY
- fk_documento_alumno1_idx
- fk_documento_tipo_documento1_idx

ILUSTRACIÓN 9 - TABLA DOCUMENTO

4.1.1.1.3. TABLA DOMICILIO

La tabla domicilio, como su nombre lo indica, almacena la información del domicilio en la etapa de registro de aspirantes al Programa de Apoyo Nutricional.

La tabla domicilio tiene una llave primaria formada por el campo *cuenta* y una llave foránea que referencia a la tabla estados_republica.

Esta tabla también cuenta con los índices correspondientes a las llaves primaria y foránea.

Column	Field	Index
estado	INT(11)	
municipio_delegacion	VARCHAR(255)	
colonia	VARCHAR(255)	
calle_num	VARCHAR(255)	
codigo_postal	INT(11)	
cuenta	INT(11)	PRIMARY

Indexes

- PRIMARY
- cuenta
- estado

ILUSTRACIÓN 10 - TABLA DOMICILIO

4.1.1.1.4. TABLA TIPO_DOCUMENTO

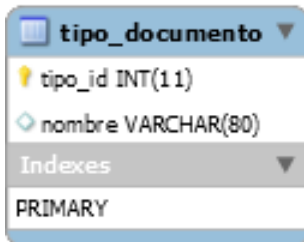


ILUSTRACIÓN 11 - TABLA TIPO DE DOCUMENTO

La tabla tipo_documento es un catálogo que permite diferenciar el contenido de los archivos que se han entregado al momento de solicitar una beca.

La tabla tipo_documento tiene una llave primaria formada por el campo tipo_id y no tiene llaves foráneas.

La tabla tiene sólo un índice correspondiente a la llave primaria.

4.1.1.1.5. TABLA SOSTÉN ECONÓMICO

La tabla sosten_economico es un catálogo que permite especificar quién es el principal sostén económico en la familia del aspirante.

La tabla sosten_economico tiene una llave primaria formada por el campo opcion_id y no tiene llaves foráneas.

La tabla tiene sólo un índice correspondiente a la llave primaria.

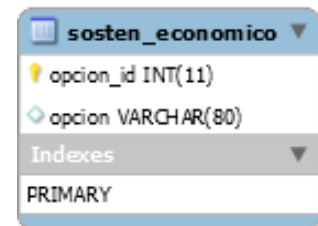


ILUSTRACIÓN 12 - TABLA SOSTÉN ECONÓMICO

4.1.1.1.6. TABLA CARRERA

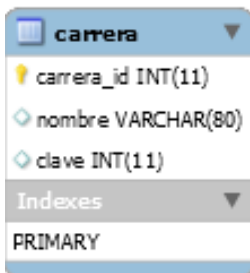


ILUSTRACIÓN 13 - TABLA CARRERA

La tabla carrera es un catálogo que permite especificar la carrera del aspirante al solicitar una beca.

La tabla carrera tiene una llave primaria formada por el campo carrera_id y no tiene llaves foráneas.

La tabla tiene sólo un índice correspondiente a la llave primaria.

4.1.1.1.7. TABLA COMPAÑÍA DEL ALUMNO

La tabla compania_alumno es un catálogo que permite especificar al aspirante con quién vive al solicitar una beca.

La tabla compania_alumno tiene una llave primaria formada por el campo opcion_id y no tiene llaves foráneas.

La tabla tiene sólo un índice correspondiente a la llave primaria.

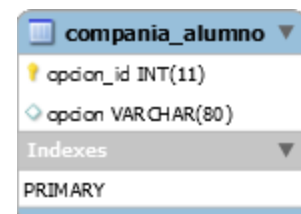


ILUSTRACIÓN 14 - TABLA COMPAÑÍA ALUMNO

4.1.1.1.8. TABLA ESTADO CIVIL

La tabla `edo_civil` es un catálogo que permite especificar al aspirante su estado civil al solicitar una beca.

La tabla `edo_civil` tiene una llave primaria formada por el campo `opcion_id` y no tiene llaves foráneas.

La tabla tiene sólo un índice correspondiente a la llave primaria.

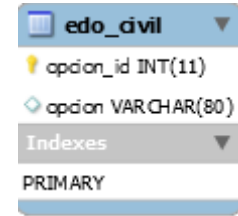


ILUSTRACIÓN 15 -
TABLA EDO. CIVIL

4.1.1.1.9. TABLA INGRESO MENSUAL

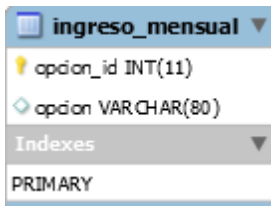


ILUSTRACIÓN 16 -
TABLA INGRESO
MENSUAL

La tabla `ingreso_mensual` es un catálogo que permite especificar al aspirante el monto aproximado de ingreso mensual de su familia al solicitar una beca.

La tabla `ingreso_mensual` tiene una llave primaria formada por el campo `opcion_id` y no tiene llaves foráneas.

La tabla tiene sólo un índice correspondiente a la llave primaria de la tabla.

4.1.1.1.10. TABLA TIEMPO DE TRASLADO

La tabla `tiempo_traslado` es un catálogo que permite especificar al aspirante el tiempo promedio que requiere para trasladarse de la facultad o escuela a su hogar o viceversa.

La tabla `tiempo_traslado` tiene una llave primaria formada por el campo `opcion_id` y no tiene llaves foráneas.

La tabla tiene sólo un índice correspondiente a la llave primaria de la tabla.

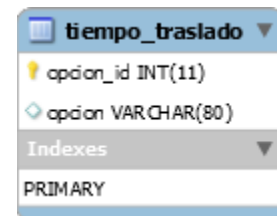


ILUSTRACIÓN 17 -
TABLA TIEMPO DE
TRASLADO

4.1.1.1.11. TABLA TIPO DE CASA

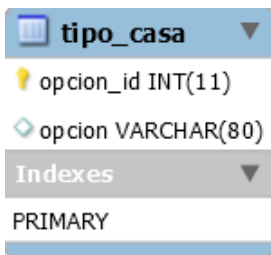


ILUSTRACIÓN 18 -
TABLA TIPO DE CASA

La tabla `tipo_casa` es un catálogo que permite especificar al aspirante quién es el propietario de la casa donde habita, es decir, si es rentada, de un familiar o propia.

La tabla `tipo_casa` tiene una llave primaria formada por el campo `opcion_id` y no tiene llaves foráneas.

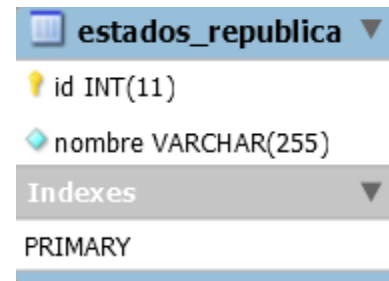
La tabla tiene sólo un índice correspondiente a la llave primaria de la tabla.

4.1.1.1.12. TABLA ESTADOS DE LA REPÚBLICA

La tabla `estados_republica` es un catálogo que contiene los nombres de los estados de la república, permitiendo de esta manera extraer estadísticas y un análisis de la información más fácil de realizar.

La tabla `estados_republica` tiene una llave primaria formada por el campo `id` y no tiene llaves foráneas. Esta tabla es referenciada por la tabla `domicilio` descrita anteriormente.

La tabla tiene sólo un índice correspondiente a la llave primaria de la tabla.

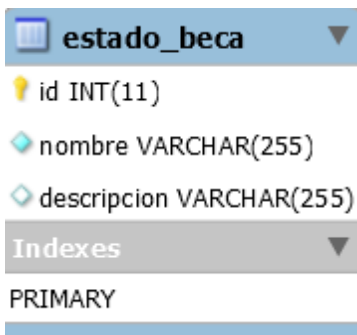


Column	Type
id	INT(11)
nombre	VARCHAR(255)

Indexes

PRIMARY

ILUSTRACIÓN 19 - TABLA ESTADOS DE LA REPÚBLICA



Column	Type
id	INT(11)
nombre	VARCHAR(255)
descripcion	VARCHAR(255)

Indexes

PRIMARY

ILUSTRACIÓN 20 - TABLA ESTADO DE LA BECA

4.1.1.1.13. TABLA ESTADO DE LA BECA

La tabla `estado_beca` es un catálogo que contiene los posibles estados de una solicitud de ingreso/permanencia en el programa, permitiendo de esta manera extraer estadísticas y un análisis de la información más fácil de realizar.

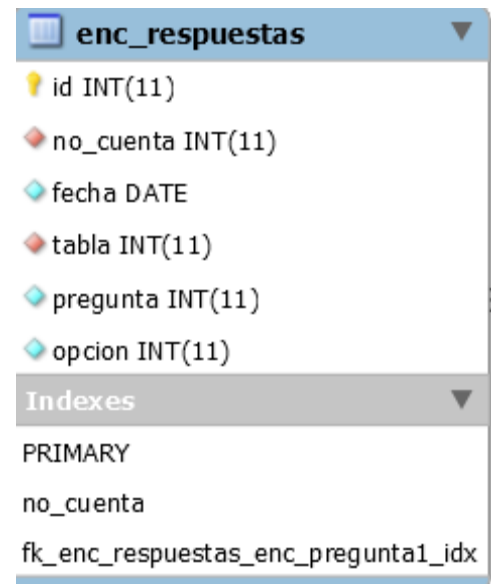
La tabla `estado_beca` tiene una llave primaria formada por el campo `id` y no tiene llaves foráneas. Esta tabla es referenciada por la tabla `alumno` descrita anteriormente.

La tabla tiene sólo un índice correspondiente a la llave primaria de la tabla.

Hasta ahora se han descrito las tablas que se encargan de almacenar datos provenientes del registro y para dar seguimiento al proceso de selección de beneficiarios; las siguientes tablas son empleadas para almacenar la información de las encuestas que se realizan en el transcurso del programa.

4.1.1.1.14. TABLA PARA ALMACENAR LAS RESPUESTAS A LAS ENCUESTAS

La tabla `enc_respuestas` permite almacenar las respuestas de las encuestas de forma organizada de acuerdo a índices que indican el tema, la pregunta indicada y quién respondió.



Column	Type
id	INT(11)
no_cuenta	INT(11)
fecha	DATE
tabla	INT(11)
pregunta	INT(11)
opcion	INT(11)

Indexes

PRIMARY

no_cuenta

fk_enc_respuestas_enc_pregunta1_idx

ILUSTRACIÓN 21 - TABLA RESPUESTAS DE ENCUESTAS

Diseño del sistema

La tabla estado_beca tiene una llave primaria formada por el campo id y tiene dos llaves foráneas que hacen referencia a las tablas alumno y a la tabla de preguntas.

La tabla tiene los índices correspondientes a la llave primaria de la tabla y a las llaves foráneas que se mencionaron anteriormente.

4.1.1.1.15. TABLA PARA ALMACENAR LAS PREGUNTAS Y POSIBLES RESPUESTAS PARA LAS MISMAS

La tabla enc_pregunta permite identificar las tablas de las encuestas de acuerdo al tema, la pregunta tiene un número único que la identifica y las posibles respuestas a la pregunta, mismas que se usan para desplegarlas en la interfaz del usuario final.

La tabla enc_pregunta tiene una llave primaria formada por el campo id y no tiene llaves foráneas. Esta tabla es referenciada por la tabla de respuestas descrita en la sección anterior.

La tabla tiene únicamente el índice correspondiente a la llave primaria de la tabla.

enc_pregunta	
id	INT(11)
tabla	VARCHAR(256)
opcion_1	VARCHAR(256)
opcion_2	VARCHAR(256)
opcion_3	VARCHAR(256)
opcion_4	VARCHAR(256)
opcion_5	VARCHAR(256)
Indexes	
PRIMARY	

ILUSTRACIÓN 22 - TABLA PREGUNTAS DE ENCUESTAS

4.1.1.1.16. TABLA PARA ALMACENAR LAS PREGUNTAS CON RESPECTO A LA CALIDAD DE LOS ALIMENTOS

enc_calidad	
id	INT(11)
pregunta	VARCHAR(256)
Indexes	
PRIMARY	

ILUSTRACIÓN 23 - TABLA CALIDAD

La tabla enc_calidad es un catálogo con las preguntas respecto a la calidad de los alimentos ofrecidos por la cafetería.

La tabla enc_calidad tiene una llave primaria formada por el campo id y no tiene llaves foráneas.

La tabla enc_calidad tiene únicamente el índice correspondiente a la llave primaria de la tabla.

4.1.1.1.17. TABLA PARA ALMACENAR LAS PREGUNTAS CON RESPECTO A LOS INGREDIENTES DE LOS ALIMENTOS EN LA COMIDA

La tabla enc_comida es un catálogo con las preguntas respecto a la calidad e ingredientes con que se preparan los alimentos ofrecidos por la cafetería, esto con el fin de tener certeza en la calidad nutricional de los alimentos ofrecidos.

enc_comida	
id	INT(11)
pregunta	VARCHAR(256)
Indexes	
PRIMARY	

ILUSTRACIÓN 24 - TABLA COMIDA

Diseño del sistema

La tabla enc_comida tiene una llave primaria formada por el campo id y no tiene llaves foráneas.

La tabla enc_comida tiene únicamente el índice correspondiente a la llave primaria de la tabla.

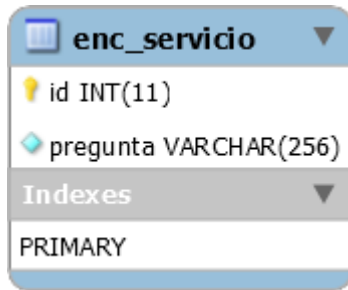


ILUSTRACIÓN 25 - TABLA SERVICIO

4.1.1.1.18. TABLA PARA ALMACENAR LAS PREGUNTAS CON RESPECTO AL SERVICIO OTORGADO POR LA CAFETERÍA

La tabla enc_servicio es un catálogo con las preguntas respecto a la calidad del servicio al cliente con que se atiende a los beneficiarios del Programa de Apoyo Nutricional.

La tabla enc_servicio tiene una llave primaria formada por el campo id y no tiene llaves foráneas.

La tabla enc_servicio tiene únicamente el índice correspondiente a la llave primaria de la tabla.

4.1.1.1.19. TABLA PARA ALMACENAR LAS PREGUNTAS CON RESPECTO AL TIEMPO DE ESPERA PARA RECIBIR EL SERVICIO

La tabla enc_tiempo_espera es un catálogo con las preguntas respecto al tiempo necesario para ser atendido en las cafeterías.

La tabla enc_tiempo_espera tiene una llave primaria formada por el campo id y no tiene llaves foráneas.

La tabla enc_tiempo_espera tiene únicamente el índice correspondiente a la llave primaria de la tabla.

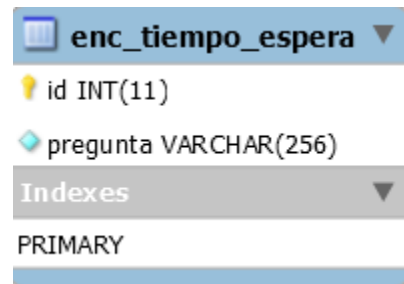


ILUSTRACIÓN 26 - TABLA TIEMPO DE ESPERA

4.1.1.1.20. TABLA PARA ALMACENAR LAS PREGUNTAS CON RESPECTO AL DESAYUNO

La tabla enc_desayuno es un catálogo con las preguntas respecto a la calidad de los alimentos que se sirven en el desayuno, la calidad de los ingredientes así como la variedad de los mismos.

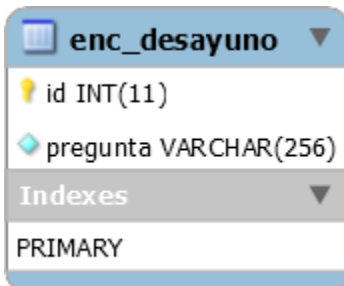


ILUSTRACIÓN 27 - TABLA DESAYUNO

La tabla enc_desayuno tiene una llave primaria formada por el campo id y no tiene llaves foráneas.

La tabla enc_desayuno tiene únicamente el índice

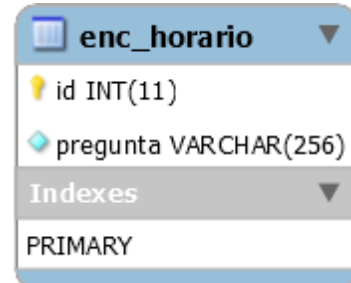
correspondiente a la llave primaria de la tabla.

4.1.1.1.21. TABLA PARA ALMACENAR LAS PREGUNTAS CON RESPECTO AL HORARIO DE SERVICIO DE LA CAFETERÍA

La tabla enc_horario es un catálogo con las preguntas respecto al horario en que se brinda atención por parte de las caferías que forman parte del programa.

La tabla enc_horario tiene una llave primaria formada por el campo id y no tiene llaves foráneas.

La tabla enc_horario tiene únicamente el índice correspondiente a la llave primaria de la tabla.

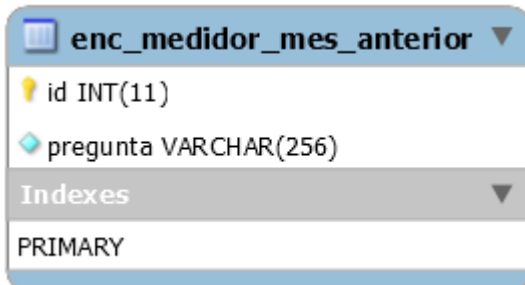


enc_horario	
id	INT(11)
pregunta	VARCHAR(256)
Indexes	
PRIMARY	

ILUSTRACIÓN 28 - TABLA HORARIO

4.1.1.1.22. TABLA PARA ALMACENAR LAS PREGUNTAS DE COMPARACIÓN

La tabla enc_medidor_mes_anterior es un catálogo con las preguntas para obtener la opinión de los beneficiarios acerca de la mejora o las áreas de oportunidad que presenta el servicio de las cafeterías con respecto al periodo de evaluación anterior.



enc_medidor_mes_anterior	
id	INT(11)
pregunta	VARCHAR(256)
Indexes	
PRIMARY	

ILUSTRACIÓN 29 - TABLA COMPARACIÓN CON MES ANTERIOR

La tabla enc_medidor_mes_anterior tiene una llave primaria formada por el campo id y no tiene llaves foráneas.

La tabla enc_medidor_mes_anterior tiene únicamente el índice correspondiente a la

llave primaria de la tabla.

4.1.2. MODELO LÓGICO: DIAGRAMA ENTIDAD - RELACIÓN

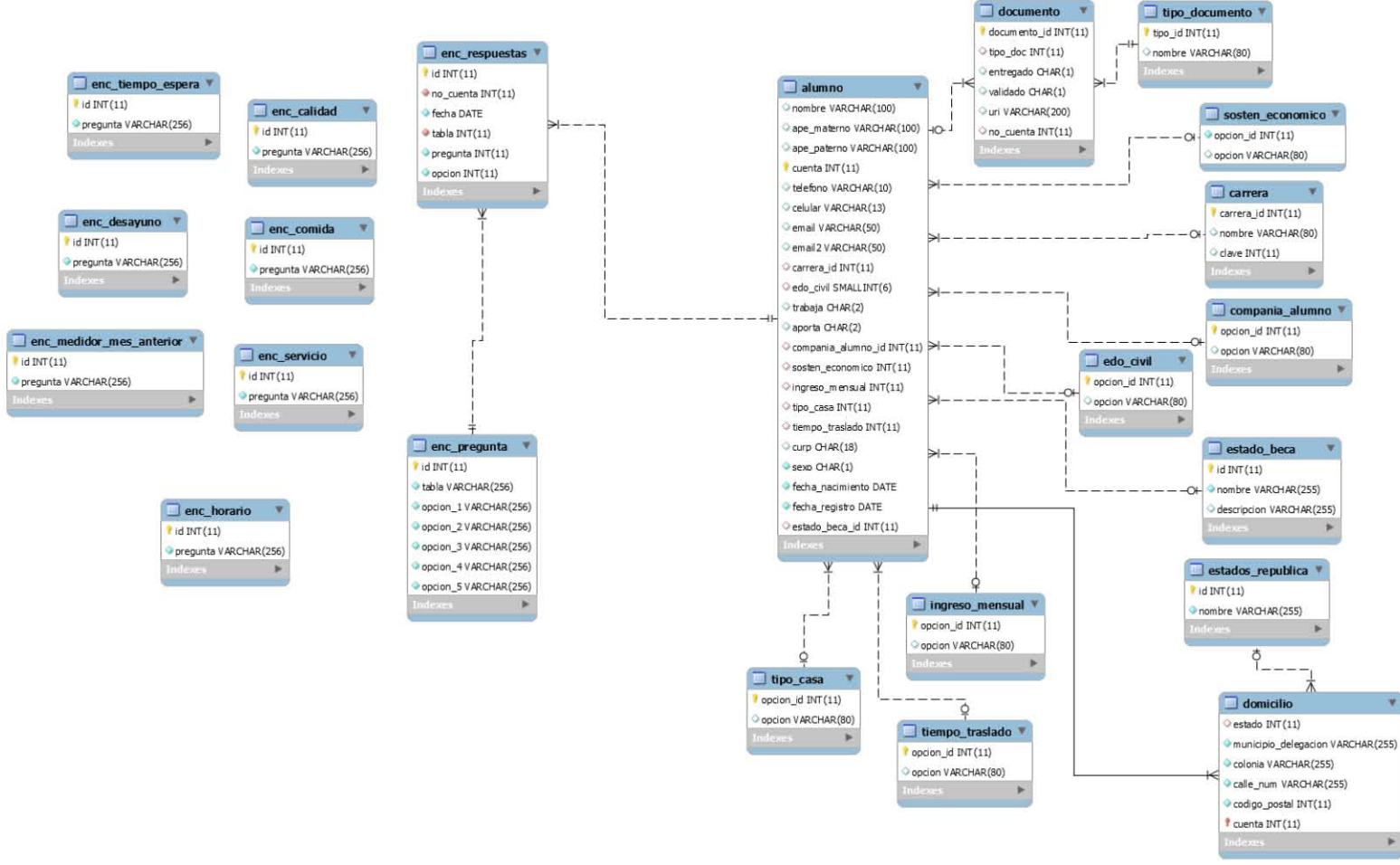


ILUSTRACIÓN 30 - DIAGRAMA ENTIDAD-RELACION

4.1.3. MODELO FÍSICO: ESQUEMA DE LA BASE DE DATOS

Se omitieron las llaves foráneas para mejorar la compatibilidad con distintos motores de almacenamiento dentro del MySQL debido a los diferentes escenarios y proveedores que existen. Las revisiones de integridad se hacen en la capa de aplicación por lo que esta adaptación no afecta la confiabilidad del sistema.

```
-----  
-- Tabla `sosten_economico`  
-----  
CREATE TABLE `sosten_economico` (  
  `opcion_id` INT(11) NOT NULL  
  AUTO_INCREMENT,  
  `opcion` VARCHAR(80) NULL DEFAULT  
  NULL,  
  PRIMARY KEY (`opcion_id`))  
DEFAULT CHARACTER SET = utf8;  
  
-----  
-- Tabla `carrera`  
-----  
CREATE TABLE `carrera` (  
  `carrera_id` INT(11) NOT NULL  
  AUTO_INCREMENT,  
  `nombre` VARCHAR(80) NULL DEFAULT  
  NULL,  
  `clave` INT(11) NULL DEFAULT NULL,  
  PRIMARY KEY (`carrera_id`))  
DEFAULT CHARACTER SET = utf8;  
  
-----  
-- Tabla `compania_alumno`  
-----  
CREATE TABLE `compania_alumno` (  
  `opcion_id` INT(11) NOT NULL  
  AUTO_INCREMENT,  
  `opcion` VARCHAR(80) NULL DEFAULT  
  NULL,  
  PRIMARY KEY (`opcion_id`))  
DEFAULT CHARACTER SET = utf8;  
  
-----  
-- Tabla `edo_civil`  
-----  
CREATE TABLE `edo_civil` (  
  `opcion_id` INT(11) NOT NULL  
  AUTO_INCREMENT,  
  `opcion` VARCHAR(80) NULL DEFAULT  
  NULL,  
  PRIMARY KEY (`opcion_id`))  
DEFAULT CHARACTER SET = utf8;  
  
-----  
-- Tabla `estado_beca`  
-----  
CREATE TABLE `estado_beca` (  
  `id` INT(11) NOT NULL  
  AUTO_INCREMENT,  
  `nombre` VARCHAR(255) NOT NULL,  
  `descripcion` VARCHAR(255) NULL  
  DEFAULT NULL,  
  PRIMARY KEY (`id`))  
DEFAULT CHARACTER SET = utf8;  
  
-----  
-- Tabla `ingreso_mensual`  
-----  
CREATE TABLE `ingreso_mensual` (  
  `opcion_id` INT(11) NOT NULL  
  AUTO_INCREMENT,  
  `opcion` VARCHAR(80) NULL DEFAULT  
  NULL,  
  PRIMARY KEY (`opcion_id`))  
DEFAULT CHARACTER SET = utf8;  
  
-----  
-- Tabla `tipo_casa`  
-----  
CREATE TABLE `tipo_casa` (  
  `id` INT(11) NOT NULL  
  AUTO_INCREMENT,  
  `nombre` VARCHAR(255) NOT NULL,  
  `descripcion` VARCHAR(255) NULL  
  DEFAULT NULL,  
  PRIMARY KEY (`id`))  
DEFAULT CHARACTER SET = utf8;
```

Diseño del sistema

```
`opcion_id` INT(11) NOT NULL
AUTO_INCREMENT,
`opcion` VARCHAR(80) NULL DEFAULT
NULL,
PRIMARY KEY (`opcion_id`))
DEFAULT CHARACTER SET = utf8;

-----
-- Tabla `tiempo_traslado`
-----
CREATE TABLE `tiempo_traslado` (
  `opcion_id` INT(11) NOT NULL
  AUTO_INCREMENT,
  `opcion` VARCHAR(80) NULL DEFAULT
  NULL,
  PRIMARY KEY (`opcion_id`))
  DEFAULT CHARACTER SET = utf8;

-----
-- Tabla `alumno`
-----
CREATE TABLE `alumno` (
  `nombre` VARCHAR(100) NULL DEFAULT
  NULL,
  `ape_materno` VARCHAR(100) NULL
  DEFAULT NULL,
  `ape_paterno` VARCHAR(100) NULL
  DEFAULT NULL,
  `cuenta` INT(11) NOT NULL,
  `telefono` VARCHAR(10) NULL DEFAULT
  NULL,
  `celular` VARCHAR(13) NULL DEFAULT
  NULL,
  `email` VARCHAR(50) NULL DEFAULT
  NULL,
  `email2` VARCHAR(50) NULL DEFAULT
  NULL,
  `carrera_id` INT(11) NULL DEFAULT
  NULL,
  `edo_civil` SMALLINT(6) NULL DEFAULT
  NULL,
  `trabaja` CHAR(2) NULL DEFAULT NULL,
  `aporta` CHAR(2) NULL DEFAULT NULL,
  `compania_alumno_id` INT(11) NULL
  DEFAULT NULL,
```

```
`sosten_economico` INT(11) NULL
DEFAULT NULL,
`ingreso_mensual` INT(11) NULL DEFAULT
NULL,
`tipo_casa` INT(11) NULL DEFAULT NULL,
`tiempo_traslado` INT(11) NULL DEFAULT
NULL,
`curp` CHAR(18) NULL DEFAULT "",
`sexo` CHAR(1) NOT NULL,
`fecha_nacimiento` DATE NOT NULL,
`fecha_registro` DATE NOT NULL,
`estado_beca_id` INT(11) NULL,
PRIMARY KEY (`cuenta`),
INDEX `cuenta` (`cuenta` ASC),
INDEX `fk_alumno_sosten_economico_idx`
(`sosten_economico` ASC),
INDEX `fk_alumno_carrera1_idx`
(`carrera_id` ASC),
INDEX `fk_alumno_compania_alumno1_idx`
(`compania_alumno_id` ASC),
INDEX `fk_alumno_edo_civil1_idx`
(`edo_civil` ASC),
INDEX `fk_alumno_estado_beca1_idx`
(`estado_beca_id` ASC),
INDEX `fk_alumno_ingreso_mensual1_idx`
(`ingreso_mensual` ASC),
INDEX `fk_alumno_tipo_casa1_idx`
(`tipo_casa` ASC),
INDEX `fk_alumno_tiempo_traslado1_idx`
(`tiempo_traslado` ASC))
DEFAULT CHARACTER SET = utf8;

-----
-- Tabla `tipo_documento`
-----
CREATE TABLE `tipo_documento` (
  `tipo_id` INT(11) NOT NULL
  AUTO_INCREMENT,
  `nombre` VARCHAR(80) NULL DEFAULT
  NULL,
  PRIMARY KEY (`tipo_id`))
  DEFAULT CHARACTER SET = utf8;

-----
-- Tabla `documento`
```

Diseño del sistema

```
-----  
CREATE TABLE `documento` (  
  `documento_id` INT(11) NOT NULL  
  AUTO_INCREMENT,  
  `tipo_doc` INT(11) NULL DEFAULT NULL,  
  `entregado` CHAR(1) NULL DEFAULT  
  NULL,  
  `validado` CHAR(1) NULL DEFAULT NULL,  
  `uri` VARCHAR(200) NULL DEFAULT  
  NULL,  
  `no_cuenta` INT(11) NULL,  
  PRIMARY KEY (`documento_id`),  
  INDEX `fk_documento_alumno1_idx`  
  (`no_cuenta` ASC),  
  INDEX  
  `fk_documento_tipo_documento1_idx`  
  (`tipo_doc` ASC))  
DEFAULT CHARACTER SET = utf8;
```

```
-----  
-- Tabla `estados_republica`  
-----
```

```
CREATE TABLE `estados_republica` (  
  `id` INT(11) NOT NULL  
  AUTO_INCREMENT,  
  `nombre` VARCHAR(255) NOT NULL,  
  PRIMARY KEY (`id`))  
DEFAULT CHARACTER SET = utf8;
```

```
-----  
-- Tabla `domicilio`  
-----
```

```
CREATE TABLE `domicilio` (  
  `estado` INT(11) NULL,  
  `municipio_delegacion` VARCHAR(255)  
  NOT NULL,  
  `colonia` VARCHAR(255) NOT NULL,  
  `calle_num` VARCHAR(255) NOT NULL,  
  `codigo_postal` INT(11) NOT NULL,  
  `cuenta` INT(11) NOT NULL,  
  PRIMARY KEY (`cuenta`),  
  INDEX `cuenta` (`cuenta` ASC),  
  INDEX `estado` (`estado` ASC))  
DEFAULT CHARACTER SET = utf8;
```

```
-----  
-- Tabla `enc_calidad`  
-----
```

```
CREATE TABLE `enc_calidad` (  
  `id` INT(11) NOT NULL  
  AUTO_INCREMENT,  
  `pregunta` VARCHAR(256) NOT NULL,  
  PRIMARY KEY (`id`))  
DEFAULT CHARACTER SET = utf8;
```

```
-----  
-- Tabla `enc_comida`  
-----
```

```
CREATE TABLE `enc_comida` (  
  `id` INT(11) NOT NULL  
  AUTO_INCREMENT,  
  `pregunta` VARCHAR(256) NOT NULL,  
  PRIMARY KEY (`id`))  
DEFAULT CHARACTER SET = utf8;
```

```
-----  
-- Tabla `enc_desayuno`  
-----
```

```
CREATE TABLE `enc_desayuno` (  
  `id` INT(11) NOT NULL  
  AUTO_INCREMENT,  
  `pregunta` VARCHAR(256) NOT NULL,  
  PRIMARY KEY (`id`))  
DEFAULT CHARACTER SET = utf8;
```

```
-----  
-- Tabla `enc_horario`  
-----
```

```
CREATE TABLE `enc_horario` (  
  `id` INT(11) NOT NULL  
  AUTO_INCREMENT,  
  `pregunta` VARCHAR(256) NOT NULL,  
  PRIMARY KEY (`id`))  
DEFAULT CHARACTER SET = utf8;
```

```
-----  
-- Tabla `enc_medidor_mes_anterior`  
-----
```

Diseño del sistema

```
CREATE TABLE `enc_medidor_mes_anterior`  
(  
  `id` INT(11) NOT NULL  
  AUTO_INCREMENT,  
  `pregunta` VARCHAR(256) NOT NULL,  
  PRIMARY KEY (`id`))  
DEFAULT CHARACTER SET = utf8;
```

-- Tabla `enc_pregunta`

```
CREATE TABLE `enc_pregunta` (  
  `id` INT(11) NOT NULL  
  AUTO_INCREMENT,  
  `tabla` VARCHAR(256) NOT NULL,  
  `opcion_1` VARCHAR(256) NOT NULL,  
  `opcion_2` VARCHAR(256) NOT NULL,  
  `opcion_3` VARCHAR(256) NOT NULL,  
  `opcion_4` VARCHAR(256) NOT NULL,  
  `opcion_5` VARCHAR(256) NOT NULL,  
  PRIMARY KEY (`id`))  
DEFAULT CHARACTER SET = utf8;
```

-- Tabla `enc_respuestas`

```
CREATE TABLE `enc_respuestas` (  
  `id` INT(11) NOT NULL  
  AUTO_INCREMENT,  
  `no_cuenta` INT(11) NOT NULL,  
  `fecha` DATE NOT NULL,  
  `tabla` INT(11) NOT NULL,  
  `pregunta` INT(11) NOT NULL,  
  `opcion` INT(11) NOT NULL,  
  PRIMARY KEY (`id`),  
  INDEX `no_cuenta` (`no_cuenta` ASC,  
  `fecha` ASC, `tabla` ASC, `pregunta` ASC,  
  `opcion` ASC),  
  INDEX  
  `fk_enc_respuestas_enc_pregunta1_idx`  
  (`tabla` ASC))  
DEFAULT CHARACTER SET = utf8;
```

-- Tabla `enc_servicio`

```
-----  
CREATE TABLE `enc_servicio` (  
  `id` INT(11) NOT NULL  
  AUTO_INCREMENT,  
  `pregunta` VARCHAR(256) NOT NULL,  
  PRIMARY KEY (`id`))  
DEFAULT CHARACTER SET = utf8;
```

-- Tabla `enc_tiempo_espera`

```
CREATE TABLE `enc_tiempo_espera` (  
  `id` INT(11) NOT NULL  
  AUTO_INCREMENT,  
  `pregunta` VARCHAR(256) NOT NULL,  
  PRIMARY KEY (`id`))  
DEFAULT CHARACTER SET = utf8;
```

4.2.DISEÑO DEL SISTEMA

El sistema tiene una estructura modular lo que permite tener distintos entornos de ejecución y perfiles de acuerdo a los recursos y las necesidades de la escuela o facultad donde se está trabajando con el sistema.

4.2.1. REGISTRO DEL SISTEMA

El siguiente diagrama muestra de manera muy simple el proceso de registro automatizado empleando sistema propuesto.

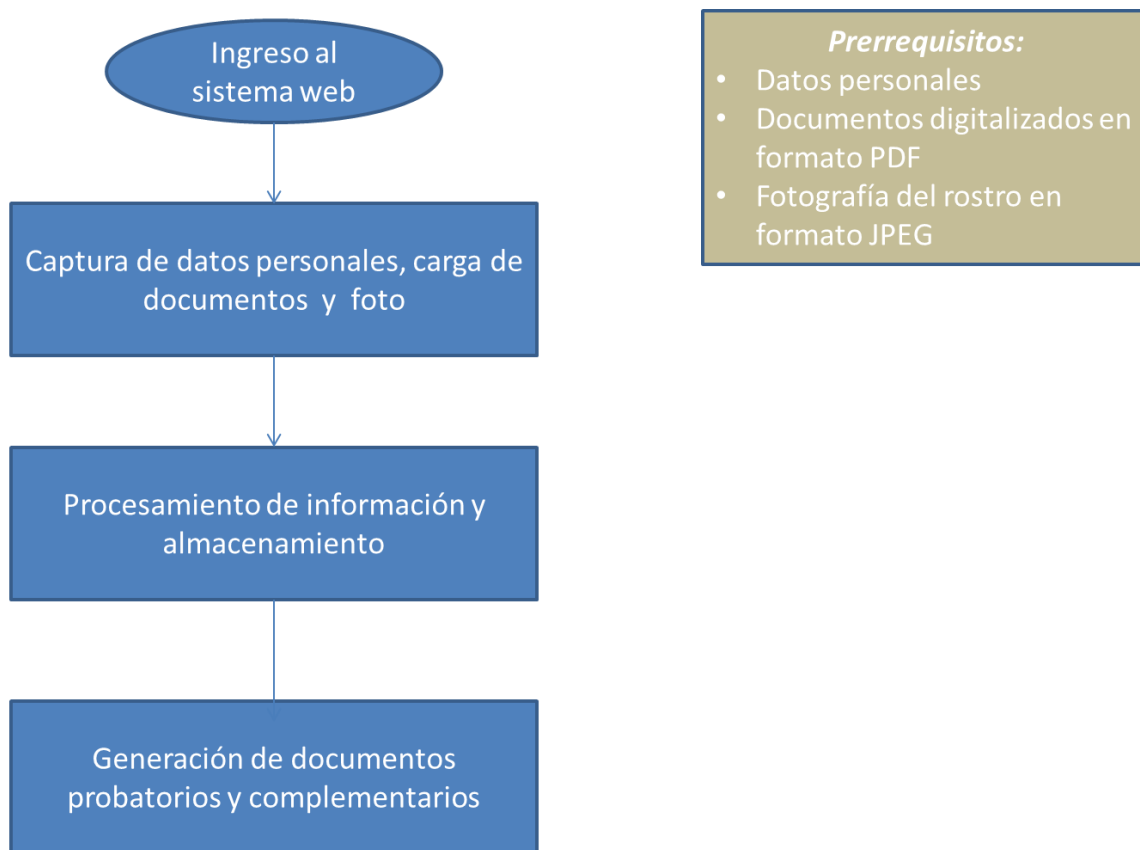


ILUSTRACIÓN 31 - ESQUEMA DEL REGISTRO DE SOLICITUDES DE BECA

El proceso de registro es sencillo en cuanto a su diseño, ya que se requiere formularios para capturar la información en el llamado *front end*; después se necesita algún programa en el *back end* para procesar la información, almacenarla en la base de datos y registrar los archivos cargados dependiendo de ciertas

Diseño del sistema

condiciones como es que los archivos tengan la extensión o el tipo MIME de acuerdo a los requerimientos, en este caso ser imágenes o archivos en formato PDF; una vez se ha registrado la información y se tienen los archivos cargados, se puede generar la constancia de registro y demás documentación necesaria para continuar con las etapas posteriores al registro de la solicitud como parte del Programa de Apoyo Nutricional.

4.2.2. CONTEO DE CONSUMOS

El conteo de consumos se puede realizar más fácilmente con esta herramienta debido a que ya no es necesario contar manualmente los consumos uno por uno como solía hacerse sino que se tienen ahora los totales de consumo por desayuno o comida al procesar los datos de consumo que se generan por el sistema de software propietario que se encuentra instalado en las cafeterías.

El siguiente diagrama muestra el proceso simplificado.

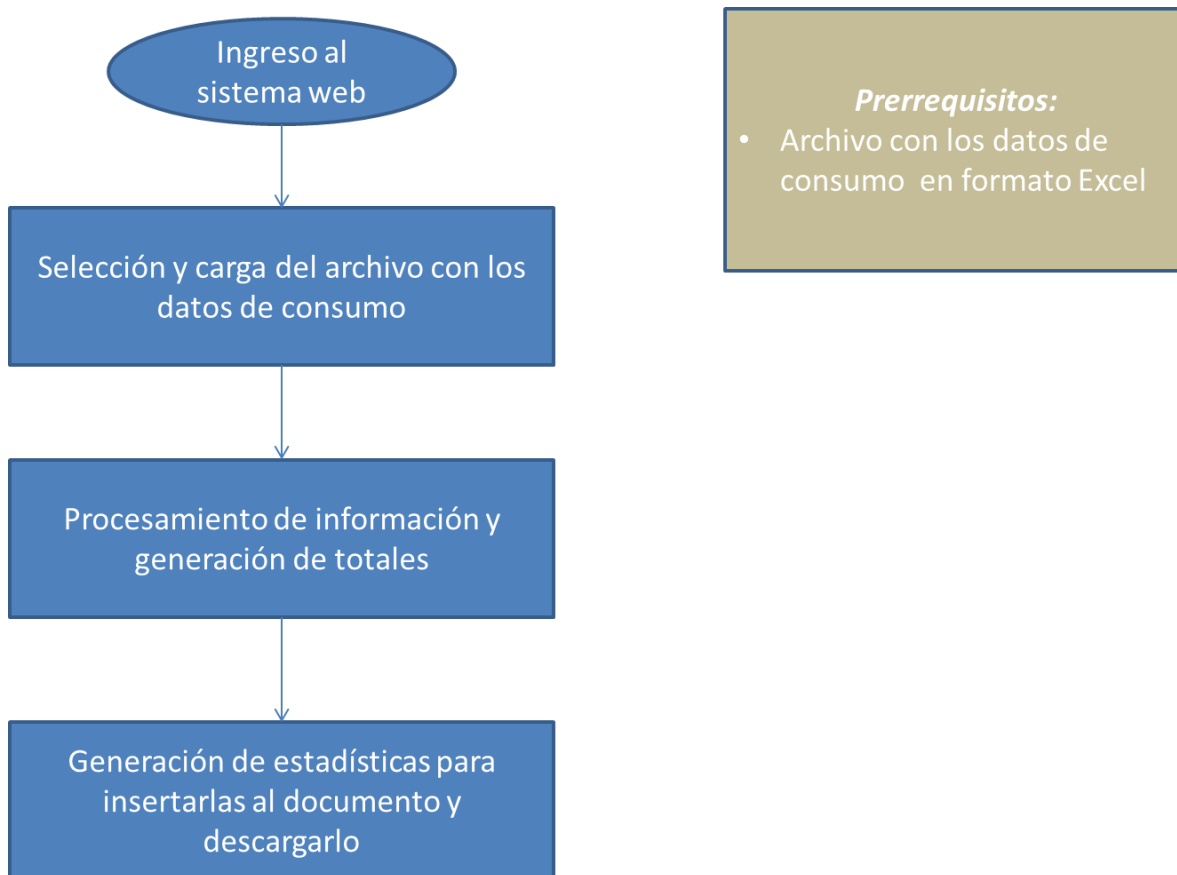


ILUSTRACIÓN 32 - DIAGRAMA DEL CONTEO DE CONSUMOS USANDO EL SISTEMA

4.2.3. ENCUESTAS DE SERVICIOS

El uso del sistema para automatizar la realización de las encuestas permite a los beneficiarios disponer de su tiempo al no tener que trasladarse hacia las oficinas de los responsables del programa en la institución o viceversa para llevar a cabo ese proceso sino que puede accederse vía web y permitir la generación de reportes con gráficas con mínimo esfuerzo.

A continuación se presenta un diagrama del proceso en forma simple.

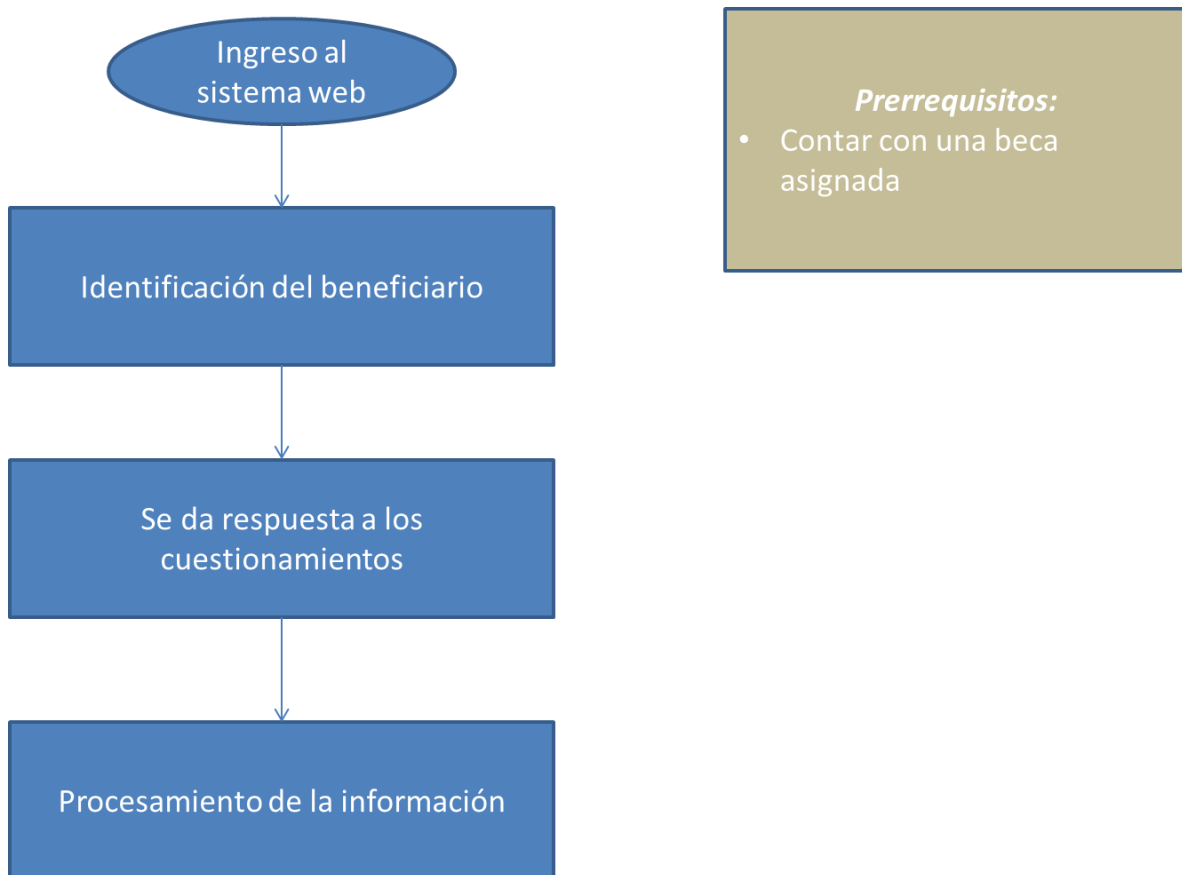


ILUSTRACIÓN 33 - DIAGRAMA DE RESPUESTA A ENCUESTA

4.2.4. MÓDULO DE ADMINISTRACIÓN

El módulo de administración tiene como objetivo brindar herramientas a los responsables del programa para poder realizar sus labores de la mejor manera posible con un esfuerzo razonable.

Diseño del sistema

Un subcomponente del módulo permite por una parte descargar los listados de los beneficiarios de acuerdo a como se requiere por parte de la Fundación UNAM. Es importante resaltar que la información queda disponible en archivos con formato fácil de editar para entregarlos en los formatos solicitados en Fundación UNAM.

Otro subcomponente del módulo permite verificar la información que el alumno proporcionó y actualizar el estado de su beca, por ejemplo, podría aceptar o rechazar la solicitud de la beca.

El subcomponente restante pero no menos importante permite la generación de un reporte basado en gráficas de pie acerca de los resultados de las encuestas, este reporte se puede hacer de manera colectiva dentro de un periodo o bien hacerlo de manera individual para conocer las preguntas o saber si se contestó la encuesta en cierto periodo.

A continuación se muestra un diagrama simplificado del proceso.

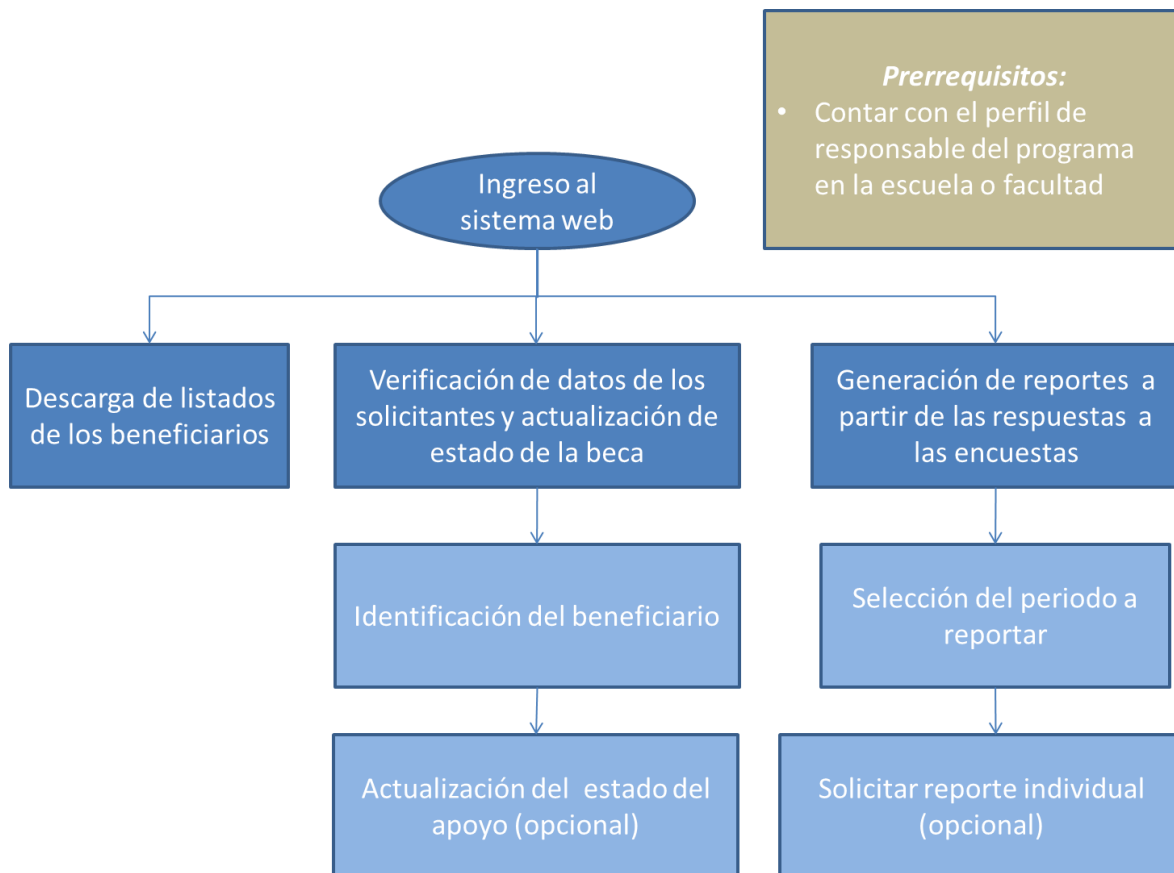


ILUSTRACIÓN 34 - DIAGRAMA ADMINISTRACIÓN DE LA INFORMACIÓN

4.2.5. ESTRUCTURA DE ARCHIVOS DEL SISTEMA

Una buena estructura de archivos en un sistema permite que se tenga un desarrollo más organizado y facilita el mantenimiento del mismo por lo que se ha dado especial importancia en la separación de componentes del sistema.

En la estructura de archivos se cuenta con varias carpetas y archivos que componen al sistema, en el directorio raíz tenemos:

- ✓ *config.php* – este archivo contiene la configuración del sistema lo que permite cambiar el comportamiento del mismo modificando parámetros.
- ✓ *index.php* – este archivo tiene como objetivo ser la página que recibe al usuario por primera vez cuando entra al sistema.
- ✓ *php.ini* – este es un archivo opcional que permite realizar configuraciones al servidor en caso de ser necesario; por ejemplo, si la configuración del servidor lo permite, podría incrementarse la memoria RAM o el tiempo de ejecución antes de terminar el programa por parte del motor de PHP.
- ✓ */administración/* – este directorio contiene el módulo de administración que se introdujo anteriormente.
- ✓ */conteo/* – este directorio contiene el módulo para realizar el conteo de consumos que se introdujo anteriormente.
- ✓ */encuesta/* – este directorio contiene el módulo para automatizar la realización de las encuestas mencionadas anteriormente.
- ✓ */imgs /* – este directorio contiene las imágenes requeridas para dar mejor presentación al sistema.
- ✓ */lib/* – este directorio contiene las bibliotecas requeridas para cumplir con diversos propósitos como son brindar la compatibilidad con software de ofimática, generación de documentos en formato PDF, entre otras.
- ✓ */registro/* – este directorio contiene el módulo para automatizar el registro de las solicitudes de ingreso y renovación del apoyo.
- ✓ */reportes/* – en este directorio se almacenan los archivos que funcionan como plantilla para generar los reportes que provee el sistema.

5. DESARROLLO DEL SISTEMA.

En el desarrollo del sistema se adoptó la programación orientada a objetos por las ventajas y oportunidades que ofrece, entre ellas la encapsulación y la reutilización de código que hacen al sistema más fácil de mantener y simplifican tanto el diseño como el desarrollo del mismo.

La siguiente imagen se presenta la pantalla de la página inicial que encuentra un usuario con acceso total al ingresar al sistema.



ILUSTRACIÓN 35 - PÁGINA PRINCIPAL DEL SISTEMA

Un primer punto a notar acerca de la estructura del sistema es que se ha adoptado un diseño responsivo que permite tener en cuenta a los usuarios de dispositivos móviles mejorando increíblemente la experiencia de usuario que tienen al interactuar con el sistema, la imagen anterior puede verse de con una pantalla mediana y la siguiente se obtiene al tener acceso con un teléfono inteligente o un dispositivo similar con pantalla pequeña y soporte para navegación web.



ILUSTRACIÓN 36 - PÁGINA PRINCIPAL DEL SISTEMA VISTA CON UN DISPOSITIVO DE PANTALLA PEQUEÑA

Desarrollo del sistema

Se puede notar a simple vista que hay una mejor distribución de los elementos al compararla con la versión para dispositivos con pantallas más grandes y por lo que es posible para el usuario que tener acceso a lo que requiere fácilmente, evitando de esta manera la necesidad de cambiar el nivel de zoom para consultar el contenido apropiadamente; esta configuración permite también que haya más espacio entre los enlaces evitando molestos errores de navegación.

Las mejoras señaladas en párrafos anteriores se dan gracias al aprovechamiento de diversas características nuevas que presenta el estándar HTML 5 y CSS3, en particular la capacidad de distinción del *viewport* y la disponibilidad de las *media queries* para cambiar los estilos necesarios de acuerdo a ciertas condiciones.

La detección del viewport con el que está presentándose la información se ejecuta con la siguiente etiqueta:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

Los estilos siguientes son una muestra de los empleados para la vista en pantallas medianas y grandes:

```
.opcion_sis {  
    float: left;  
    margin: 40px;  
    width: 310px;  
    height: 190px;  
}
```

Estos estilos son modificados por una media query que permite cambiar la presentación para equipos cuyo viewport sea de máximo 481 pixeles de ancho. Los cambios existentes aparecen resaltados a continuación:

```
@media screen and (max-width: 480px) {  
    .opcion_sis {  
        float: none;  
        margin: auto;  
        width: 310px;  
        height: 190px;  
    }  
}
```

5.1.REGISTRO DE SOLICITUDES.

Este módulo se encarga de registrar las solicitudes de los alumnos que aspiran a obtener una beca al obtener, procesar y almacenar los datos y documentos digitales requeridos por el Programa de Apoyo Nutricional para agilizar el proceso

Desarrollo del sistema

y además nos permite eliminar tiempos de traslado así como reducir la cantidad de recursos humanos necesarios para realizar esta actividad dentro del programa.

El primer paso para solicitar el apoyo consiste en ingresar el número de cuenta del solicitante en el campo indicado, a continuación se muestra la pantalla que corresponde a esta solicitud.



The screenshot shows a web form titled "Registro al Programa de Apoyo Nutricional". Below the title, there is a prompt: "Por favor, completa todos los campos con tu información". A text input field is labeled "Número de cuenta (sólo números)" and contains the value "306180390". Below the input field is a button labeled "Siguiete".

ILUSTRACIÓN 37 - INICIO DE REGISTRO DE SOLICITUDES DE BECA

Como se puede observar la interfaz del sistema es muy sencilla e intuitiva, una vez que se ha proporcionado el dato, éste se valida por diferentes cuestiones:

- ✓ El texto ingresado corresponda a un número de cuenta válido en la UNAM – esto se verifica con una expresión regular que permite identificar sólo dígitos por lo que se presenta un mensaje de error en caso de escribir un número no válido, a continuación se muestra una imagen del caso:



The screenshot shows the same web form as in the previous illustration, but with an error message. A red banner with diagonal stripes contains the text "No. de cuenta no válido". Below this banner, the text input field is empty, and the "Siguiete" button is still present.

ILUSTRACIÓN 38 - MENSAJES DE ERROR DEL SISTEMA AL REGISTRAR SOLICITUDES

Desarrollo del sistema

- ✓ El alumno ya ha completado su registro con anterioridad, de esta manera sólo se requiere que tenga acceso a los comprobantes que el sistema ofrece en caso de que los necesite el alumno nuevamente, estos comprobantes pueden ser descargados sin mayor problema, a continuación se muestra una imagen que tendría el solicitante.

Registro al Programa de Apoyo Nutricional

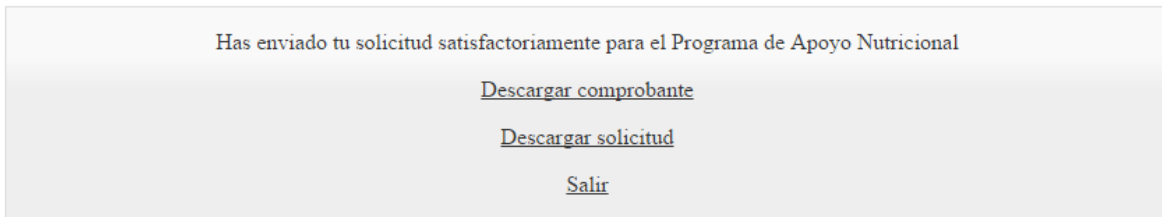


ILUSTRACIÓN 39 - DESCARGA DE COMPROBANTES AL TERMINAR EL REGISTRO DE SOLICITUD DE BECA

- ✓ Se requiera tener una nueva solicitud de beca por lo que se inicia el proceso de registro solicitando los datos y documentos necesarios para cumplir con este requisito. A continuación se muestra la imagen de la pantalla a la que se llegaría.

Registro al Programa de Apoyo Nutricional

I: DATOS PERSONALES Y ACADÉMICOS

Número de cuenta: 306180399

Nombre(s)

A

Ap

Teléfono

Teléfono c

Correo electrónico 1:

Correo electrónico 2:

Estado civil:

Sexo:

Selecciona tu fecha de nacimiento (aaaa-mm-dd):

Instrucciones

Instrucciones: Contesta con veracidad la siguiente información.

ILUSTRACIÓN 40 - INSTRUCCIONES PARA EL REGISTRO

Desarrollo del sistema

Se puede observar que se muestra una ventana modal pidiendo se contesten los datos con veracidad, se solicitan varios datos que se listan a continuación:

- ✓ Nombre
- ✓ Apellidos paterno y materno
- ✓ Teléfono fijo y celular
- ✓ Correos electrónicos
- ✓ Estado civil
- ✓ Sexo
- ✓ Fecha de nacimiento
- ✓ CURP
- ✓ Domicilio (calle y número, colonia, municipio o delegación, estado, código postal)
- ✓ Carrera
- ✓ Si trabaja al momento de solicitar la beca y aporta al ingreso familiar
- ✓ Las personas con quien vive
- ✓ Quién es el principal sostén económico familiar
- ✓ A cuánto ascienden los ingresos mensuales
- ✓ Si la casa donde vive es propia, rentada o de algún familiar
- ✓ El tiempo promedio de trayecto hacia la facultad
- ✓ Documentación en formato PDF (historial académico, comprobante de inscripción con horario, copia de la credencial UNAM vigente, carta de exposición de motivos, comprobante de percepciones, comprobante de domicilio, fotografía en formato JPG del rostro)

En caso de faltar algún dato o que algún documento que tenga el formato requerido, PDF o JPG para el caso de la fotografía, el sistema despliega un mensaje de error indicando estas cuestiones.



ILUSTRACIÓN 41 - MENSAJES DE ERROR AL SOLICITAR UN BECA

Desarrollo del sistema

Dentro del sistema se usan elementos de JQuery IU que permiten el ingreso de la información de manera confiable y permiten también evitar complicaciones para el usuario al tener que apegarse a un formato o leer muchas instrucciones, se ha visto que usualmente las personas no leen todas las instrucciones al interactuar con un sistema provocando que a veces se requiera soporte técnico que haría al proyecto bastante difícil de mantener. Un par de ejemplos de estos controles son la ventana modal que se presentó con anterioridad y el calendario empleado para ingresar la fecha de nacimiento en un formato que facilite su procesamiento y a la vez sea fácil de manejar, a continuación se muestra la imagen de este último:

The image shows a web form with several input fields and a date picker. The date picker is open, showing the month of June 1990. The date 16th is selected. The form fields include:

- Sexo: Masculino
- Selecciona tu fecha de nacimiento (aaaa-mm-dd): 1990-06-16
- Ingresa tu CURP: AAAA999999AAA
- Datos de tu domicilio
- Calle y número: Calle Número
- Colonia: Colonia
- Municipio o delegación: Ciudad De México

Do	Lu	Ma	Mi	Ju	Vi	Sa
						1
						2
	3	4	5	6	7	8
	9	10	11	12	13	14
	15	16	17	18	19	20
	21	22	23	24	25	26
	27	28	29	30		

ILUSTRACIÓN 42 - HERRAMIENTAS AUXILIARES PARA RECABAR DATOS

La biblioteca JQuery UI puede consultarse y descargarse en la dirección electrónica <https://jqueryui.com>, contiene diversos widgets que permiten la fácil interacción del usuario con el sistema.

Una vez se ha completado el registro de los datos y los documentos en formato digital, se permite descargar documentos que comprueban la conclusión de este trámite, la pantalla es muy similar a la descrita anteriormente cuando el solicitante desea volver a descargar o imprimir sus comprobantes generados en formato PDF.

The image shows a confirmation message in a light gray box. The text reads:

Registro al Programa de Apoyo Nutricional

Marcos has enviado tu solicitud satisfactoriamente para el Programa de Apoyo Nutricional

23 de Noviembre de 2015

[Descargar comprobante](#)

[Descargar solicitud](#)

[Salir](#)

ILUSTRACIÓN 43 - DESCARGA DE COMPROBANTES

Desarrollo del sistema

Con el objetivo de implementar al generación de los documentos en formato PDF se empleó una biblioteca llamada FPDF escrita en el lenguaje de programación PHP con mínimos requerimientos y libre de uso, se pueden consultar detalles y descargar una copia en la dirección electrónica <http://www.fpdf.org>.

A continuación se presenta la imagen de un documento que comprueba haber solicitado una beca:

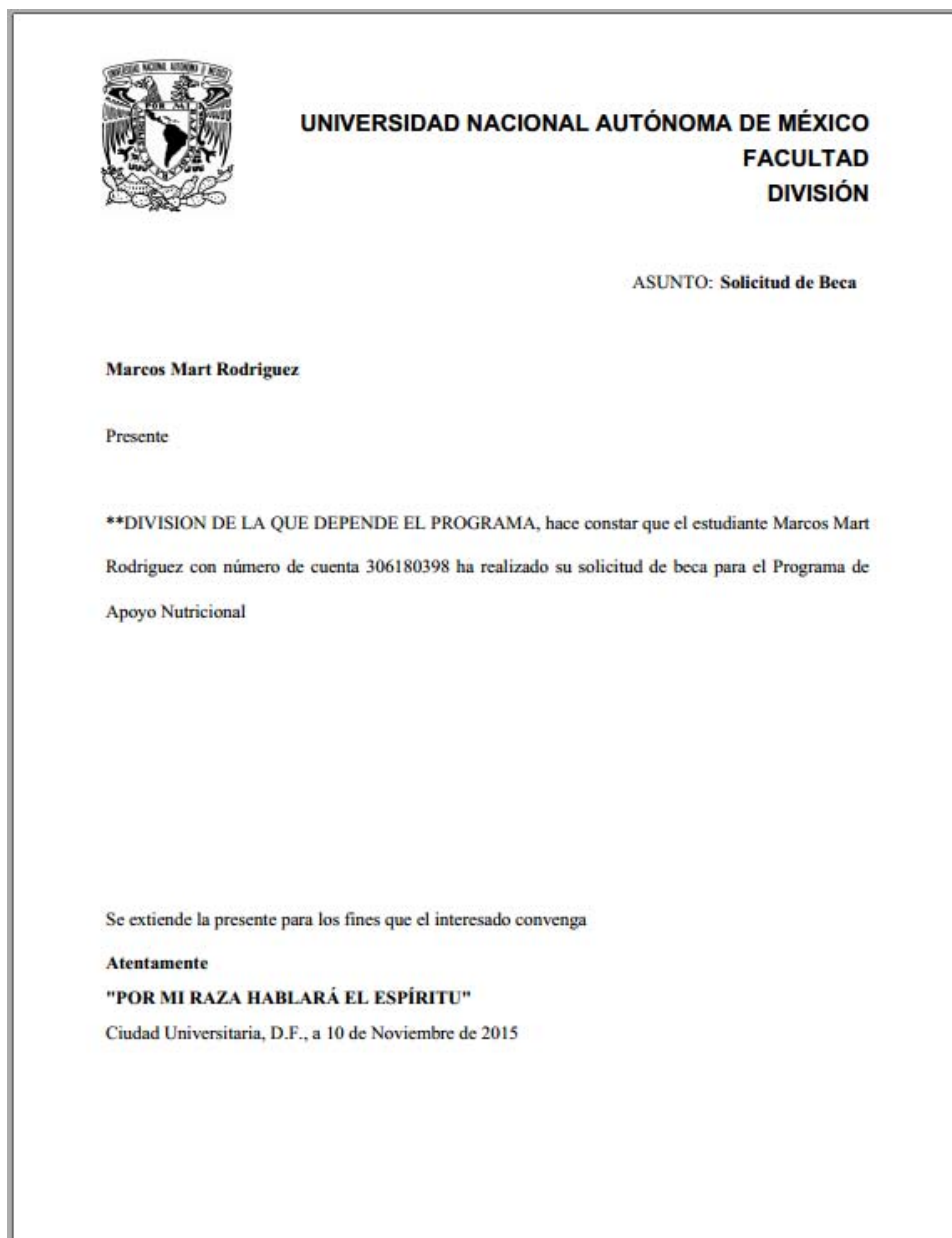


ILUSTRACIÓN 44 - COMPROBANTE DE SOLICITUD DE BECA

El otro documento que se genera es el documento de solicitud de beca que se presenta a continuación, dicho documento permite contar con un

Desarrollo del sistema

expediente de los alumnos a los responsables del programa en la escuela o facultad.


	FACULTAD DE INGENIERÍA UNAM* SECRETARÍA DE APOYO A LA DOCENCIA* COORDINACION DE PROGRAMAS DE ATENCIÓN DIFERENCIADA PARA ALUMNOS (COPADI)* SEMESTRE 2013-2*		
	Formato de solicitud para el Programa de Apoyo Nutricional		
I. DATOS PERSONALES Y ACADÉMICOS			
Nombre:	Edo. civil:	Fecha:	
Marcos Mari Rodríguez	Soltera (o)	10 de Noviembre de 2015	
Carrera:	No. Cuenta:	Teléfono:	Correo Electrónico:
Ingeniería en Computación	306180398	5555555555 044555555555	marcos@example.com otro@example.com
II. DIVERSOS			
¿Actualmente trabajas?	si		
¿Aportas al ingreso familiar?	si		
¿Vives con tus padres?	Ambos		
¿Ingreso total familiar mensual?	menos de 3000		
La casa donde habitas es:	otro		
Tiempo de traslado de tu casa a la Facultad	Entre dos y tres horas		
III. DOCUMENTOS			
Situación escolar (Llenado exclusivo por el programa)	Regular () Irregular () Artículo 22 () Otro:		
Historial académico	OK	Carta de exposición de motivos	OK
Comprobante de inscripción con horario	OK	Fotografía	OK
Copia de credencial UNAM	OK		
¿Es candidato para ingresar al programa? (Uso exclusivo de los responsables del programa)	Si () No ()		
Observaciones:			
Evaluó:			
Bajo protesta de decir verdad manifiesto que los datos en la presente solicitud son fidedignos Acepto la decisión que se emita respecto a mi solicitud y que el fallo, a la misma, será inapelable			
Firma del solicitante: _____			

ILUSTRACIÓN 45 - DOCUMENTO PARA SOLICITAR BECA

5.2. CONTEO DE CONSUMOS

Este módulo se encarga de realizar el conteo de los registros producidos por los consumos de alimentos diarios de cada beneficiario del Programa de Apoyo Nutricional lo que permite destinar menor cantidad de recursos a esta tarea y reduce el margen de error al procesarse automáticamente y tener un procedimiento más simple de operar.

Al otorgarse una beca, los alumnos adquieren obligaciones que deben cumplir para continuar con el apoyo, una de ellas es contar con al menos el 80 % de consumos durante el periodo de apoyo por lo que es importante su identificación y registro de consumo de alimentos dentro del apoyo es por eso que las cafeterías que ofrecen servicios de alimentos dentro del Programa de Apoyo Nutricional debe contar indispensablemente con un sistema local que permite identificar a los beneficiarios con su huella digital que evidentemente se registra junto con sus datos personales al iniciar las fechas de consumos.

El sistema que poseen las cafeterías es software propietario que operan ellas mismas y generan reportes de consumos que contienen el número de cuenta del alumno, la hora y la fecha junto con otros datos que no son relevantes para el sistema en desarrollo; dicho archivo de consumos puede ser extraído por el propio personal encargado del Programa de Apoyo Nutricional en la facultad o escuela a través de la red mediante programas de acceso remoto como es TeamViewer, este programa permite el control de la máquina remoto y de esta forma se agiliza la generación de reportes, un punto a considerar es que este software puede usarse gratuitamente

mientras sea uso privado, para uso comercial aplican ciertas restricciones, para mayor información puede consultar la dirección electrónica

www.teamviewer.com. A continuación se presenta una muestra de un documento similar a los que se generan en algunas cafeterías que ofrecen sus servicios en el Programa de Apoyo Nutricional.

	A	B	C	D	E	F
1	55555555	06/08/2022 11:50	1	0	1	0
2	55555555	07/08/2022 17:55	1	0	1	0
3	55555555	08/08/2022 12:08	1	0	1	0
4	55555555	09/08/2022 12:08	1	0	1	0
5	55555555	13/08/2022 10:16	1	0	1	0
6	55555555	14/08/2022 18:02	1	0	1	0
7	55555555	15/08/2022 11:11	1	0	1	0
8	55555555	16/08/2022 12:14	1	0	1	0
9	44444444	05/08/2022 14:41	1	0	1	0
10	44444444	06/08/2022 15:06	1	0	1	0
11	44444444	07/08/2022 15:14	1	0	1	0
12	44444444	08/08/2022 15:33	1	0	1	0
13	44444444	09/08/2022 16:02	1	0	1	0
14	44444444	12/08/2022 14:47	1	0	1	0
15	44444444	13/08/2022 14:15	1	0	1	0
16	44444444	14/08/2022 14:45	1	0	1	0
17	44444444	15/08/2022 15:48	1	0	1	0
18	44444444	16/08/2022 14:56	1	0	1	0
19						

ILUSTRACIÓN 46 - MUESTRA DE CONSUMOS DE ALIMENTOS

Como se puede observar, el conteo se realizaba de forma manual, en el caso particular de las personas que lo realizaban se hacía un agrupamiento y después

Desarrollo del sistema

se contabilizaban los consumos pero se debía tomar en cuenta la hora puesto que deben separarse comidas de desayunos porque cuestan diferentes montos.

En la siguiente imagen se puede apreciar la pantalla que tendría alguien con acceso a este módulo del sistema; la interfaz es también muy simple, únicamente se tiene un formulario que permite seleccionar un archivo en formato compatible con Microsoft Office Excel (con extensión .xlsx) y enviarlo para que sea procesado, una vez que se termina el procesamiento de los datos, se agrega una hoja de cálculo al libro y se escriben los totales de consumos tanto para desayunos, comidas y la suma de ambas cantidades.

Control de consumos del Programa de Apoyo Nutricional

En el siguiente formulario puede seleccionar un archivo para procesar

Send this file: prueba quimica.xlsx

ILUSTRACIÓN 47 - CARGA DE DATOS DECONSUMO

Al terminar de procesar el archivo con el conteo de consumos, se inicia la descarga del documento actualizado con la información que se ha generado en una nueva hoja de cálculo llamada Resultados, a continuación se muestra una imagen que ilustra este punto.

	A	B	C	D
1				
2	No. Cuenta	Desayuno	Comida	Total
3	555555555	3	5	8
4	444444444	0	10	10
5	Totales:	3	15	18
6				
7				
8				
9				

ILUSTRACIÓN 48 - CONTEO DE CONSUMOS REALIZADO

Cabe destacar que los datos usados para generar estos resultados no se ven afectados, estos aparecen en las otras hojas de cálculo así como cualquier otro contenido del archivo. Un único detalle al manejar el archivo es indicar que cuando se carga un archivo para procesar consumos, los datos de los mismos deben aparecer en la hoja activa, para fines prácticos se puede identificar fácilmente ya que es la hoja que se ve al abrir el archivo dentro de Excel.

Desarrollo del sistema

Al igual que para la generación de documentos en formato PDF, en esta ocasión se usó otra biblioteca llamada PHPEXcel que contiene la licencia LGPL por lo que es fácil de adquirir, la biblioteca se puede consultar y descargar en la dirección electrónica <https://phpexcel.codeplex.com>.

5.3. ENCUESTAS

Al solicitar una beca del Programa de Apoyo Nutricional se les informa a los alumnos que contestar las encuestas de satisfacción es un requisito para continuar recibiendo el apoyo es por ello que cobra importancia tener una herramienta para automatizar la tarea.

En este componente del sistema se puede realizar una encuesta sobre la calidad de los servicios ofrecidos por las cafeterías, lo que permite tener un instrumento para saber qué tan satisfechos se encuentran los becarios, así como documentar sus opiniones.

Para ingresar al módulo, el beneficiario debe contar con un apoyo activo del programa ya que el sistema de cómputo tiene la base de datos actualizada y se comprueba la existencia del apoyo basados en el número de cuenta del alumno. A continuación se muestra la pantalla que recibe a los alumnos para iniciar el proceso, como podrá apreciar es bastante sencilla, simplemente muestra las instrucciones y recordatorios correspondientes al proceso de respuesta a encuestas y solicita el número de cuenta del beneficiario para identificarlo e iniciar el proceso.

Encuesta del Programa de Apoyo Nutricional

Instrucciones: Contesta con veracidad la siguiente información

Encuesta de opinión mensual

Estimado becari@:

A continuación te presentamos una serie de preguntas que nos ayudará a brindarte un mejor servicio, así mismo recabaremos información necesaria para evaluar al proveedor y poder brindarte una mejor atención.

Te pedimos que contestes con veracidad cada una de las preguntas, ya que es una información muy valiosa para ti y para nosotros.

Por último, te recordamos que es tu responsabilidad contestar ésta encuesta mensualmente, para que puedas seguir gozando de éste apoyo que se te ha brindado.

Ingresa tu número de cuenta y pulsa el botón continuar para iniciar con la encuesta.

ILUSTRACIÓN 49 - INSTRUCCIONES PARA CONTESTAR ENCUESTA

Desarrollo del sistema

En caso de no ser beneficiario del programa o de intentar contestar la encuesta más de una vez en el mismo día, el sistema no permite acceder a la sección de preguntas y muestra un mensaje de error indicando la situación.



ILUSTRACIÓN 50 - YA SE CONTESTO LA ENCUESTA EN EL DÍA O NO SE POSEE EL PERFIL PARA HACERLO

Una vez que se ha identificado el alumno, se presenta una serie de preguntas agrupadas en diversos temas:

- ✓ **Desayuno:** En esta sección se presentan cuestionamientos sobre la variedad de los alimentos y la frecuencia con la que son servidos.
- ✓ **Comida:** En esta sección se presentan preguntas muy similares a las de la sección *Desayuno*
- ✓ **Calidad:** En esta sección se requiere la opinión en cuanto a calidad de los alimentos ofrecidos tanto en comida como en desayuno.
- ✓ **Horario:** Se consulta si se ha cumplido el horario de servicio que se ha indicado.
- ✓ **Tiempo de espera:** En esta sección se consulta la cantidad de tiempo a esperar desde que se toma el ticket hasta el momento cuando se sirve la comida.
- ✓ **Servicio:** En esta sección se hacen cuestiones acerca de la calidad del servicio que se ofrece en la cafetería, desde la amabilidad con que son tratados los becarios hasta las prácticas de higiene al manejar los alimentos.
- ✓ **Medidor respecto al mes anterior:** En esta sección de la encuesta se hace una comparativa con respecto al mes anterior para evaluar la posible mejora o retrocesos en la calidad del servicio en general con que se ofrecen los alimentos.

Desarrollo del sistema

A continuación se presenta una imagen de la sección referente al servicio.

Servicio					
	Siempre	Frecuentemente	Regularmente	Casi nunca	Nunca
El servicio de la barra de alimentos es eficiente.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Se te brinda el servicio de comida para llevar.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
El personal de la cafetería te atiende amablemente.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
La cafetería presenta condiciones de limpieza e higiene adecuados en loza, cubiertos y mesas.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
El manejo de los alimentos en la barra es higiénico.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
El área de la barra de alimentos se encuentra limpia.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
El personal de la cafetería usa uniforme, cofia y cubrebocas adecuadamente.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Los lugares destinados para desechos se encuentran en buenas condiciones.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Se observa que los alimentos están preparados con buenas prácticas de higiene.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

ILUSTRACIÓN 51 - CUESTIONAMIENTOS DEL SERVICIO

Una vez se completa el registro de la información, esta se envía, se procesa y almacena en la base de datos del sistema, en caso de no presentarse errores, se presenta un mensaje corroborando la recepción de la información. A continuación se presenta una imagen que finaliza el reporte de la encuesta.



ILUSTRACIÓN 52 - LA INFORMACIÓN SE REGISTRÓ CORRECTAMENTE

Cabe mencionar que las preguntas están organizadas de acuerdo a sus temas en diferentes tablas dentro de la base de datos por lo que se simplifica en gran medida la administración de las mismas; en cuanto se hacen cambios a la base de datos, automáticamente se actualiza los formularios para evitar errores de edición en el código fuente de la aplicación, en caso de requerir una nueva categoría de preguntas sí es necesaria la inserción del código manualmente aunque no es complicado debido a que se siguió un patrón para realizar estos procedimientos con mayor facilidad.

Desarrollo del sistema

A continuación se presenta una muestra del código que genera el formulario para la sección del desayuno dentro de la encuesta.

```
<h2>Desayuno</h2>
<?php
$html = '<table>';
$cambio = false;
$datos = $db->trae_preg_desayuno();
$html .= '<tr><th></th><th>'. $cabeceras['enc_desayuno']['opcion_1'] . '</th>';
$html .= '<th>'. $cabeceras['enc_desayuno']['opcion_2'] . '</th>';
$html .= '<th>'. $cabeceras['enc_desayuno']['opcion_3'] . '</th>';
$html .= '<th>'. $cabeceras['enc_desayuno']['opcion_4'] . '</th>';
$html .= '<th>'. $cabeceras['enc_desayuno']['opcion_5'] . '</th> </tr>';

foreach($datos as $id => $row){
    $html .= '<tr '. (($cambio)?'class="tr_dif":"").'><td style="text-align: left;">'. $row['pregunta'] .
    '</td>';
    $html .= '<td> <input name="desayuno_' . $id . '" id="desayuno_' . $id . '" value="1" type="radio">
    </td>';
    $html .= '<td> <input name="desayuno_' . $id . '" id="desayuno_' . $id . '" value="2" type="radio">
    </td>';
    $html .= '<td> <input name="desayuno_' . $id . '" id="desayuno_' . $id . '" value="3" type="radio">
    </td>';
    $html .= '<td> <input name="desayuno_' . $id . '" id="desayuno_' . $id . '" value="4" type="radio">
    </td>';
    $html .= '<td> <input name="desayuno_' . $id . '" id="desayuno_' . $id . '" value="5" type="radio">
    </td></tr>';
    $cambio = !$cambio;
}

$html .= '</table>';
echo $html;
?>
```

Como se puede observar, el proceso de generación de formularios consiste en tres pasos:

1. Se extraen los datos de la base de datos usando una capa de abstracción diseñada para facilitar el mantenimiento del código y aprovechar las características de la Programación Orientada a Objetos (POO).
2. Una vez se tienen los datos se procede a crear las cabeceras de la tabla y después las opciones con controles *radio buttons* de acuerdo a como lo marca la sintaxis y la semántica del lenguaje de marcado HTML.

3. Al completar la generación del código en HTML, se incorpora a la página y se continúa mandando la estructura de la página faltante hacia el navegador del usuario, en este caso el beneficiario del programa.

5.4. ADMINISTRACIÓN

Este módulo del sistema permite obtener información tanto general como específica de los diferentes módulos descritos anteriormente; para el caso del registro de solicitudes, permite la generación de las listas requeridas por Fundación UNAM y otra con toda la información de los estudiantes con que se cuenta en el sistema; asimismo, el módulo permite la verificación de los datos y documentos que se entregaron al momento de hacer las solicitudes de becas para otorgar, suspender o cancelar el apoyo de algún alumno. Otra función del módulo es la generación de reportes con gráficas con las respuestas a la encuesta, estos reportes contienen gráficas que permiten la visualización e interpretación de la información de manera muy sencilla.

A continuación se presenta la pantalla de inicio que vería alguien con los permisos de acceso respectivos.

Administración del Programa de Apoyo Nutricional

Listados de solicitudes

Al momento hay 4 solicitudes en total

[Listado de solicitudes \(campos FUNAM\)](#)

[Listado de solicitudes \(todos los campos\)](#)

Datos de solicitud

Ingrese el número de cuenta de algún alumno para mostrar sus datos

Número de cuenta (sólo números)

Reportes de encuestas

Ingrese el rango de las fechas para mostrar los reportes, active la casilla para mostrar un reporte personalizado

Seleccione la fecha de inicio del reporte (aaaa-mm-dd):

Seleccione la fecha de corte del reporte (aaaa-mm-dd):

Personalizar el reporte de la encuesta

ILUSTRACIÓN 53 - PANEL DE ADMINISTRACIÓN DEL SISTEMA

5.4.1. LISTADOS DE SOLICITUDES.

El sistema permite generar listados de las solicitudes presentadas para obtener una beca, uno de ellos es el que se requiere por Fundación UNAM para procesar las solicitudes, cuyos campos son los siguientes:

- ✓ Número de cuenta.
- ✓ Nombre completo.
- ✓ Teléfono.
- ✓ Celular.
- ✓ Dos correo electrónicos.
- ✓ Columnas vacías para correo, carrera, promedio, avance, registro.
- ✓ Ingreso total mensual de la familia
- ✓ CURP

Existe otro documento que contiene los datos listados en el párrafo anterior y otros que complementan la información disponible del sistema, dichos datos adicionales son los siguientes:

- ✓ Estado civil
- ✓ Con quién vive (padres, sólo madre, sólo padre, etc.)
- ✓ Datos del domicilio del estudiante
- ✓ Fecha de nacimiento
- ✓ Quién es el propietario de la casa donde vive
- ✓ El tiempo de traslado promedio de su casa a la facultad o viceversa
- ✓ Fecha de registro de la solicitud
- ✓ Si trabaja y aporta al ingreso familiar
- ✓ Quién es el sostén económico en su familia (padre, madre, otros, etc.)
- ✓ Estado de la beca (solicitud de primer ingreso, renovada, otorgada, etc.)

Dada la necesidad de tener una manera de manejar la información por personas con distintos tipos de habilidades, formación y recursos de cómputo, se decidió utilizar el formato de Microsoft Office Excel dada su popularidad y disponibilidad en la mayoría de los equipos de cómputo, se tomó en cuenta que los equipos que no cuenten con esta suite ofimática podrían tener acceso a programas de software libre como son LibreOffice con características similares sin costo.

Desarrollo del sistema

En cuanto a la interfaz de usuario es bastante simple, con sólo dar clic sobre los botones para descargar los listados se tiene acceso a la información como se muestra a continuación:

Listados de solicitudes

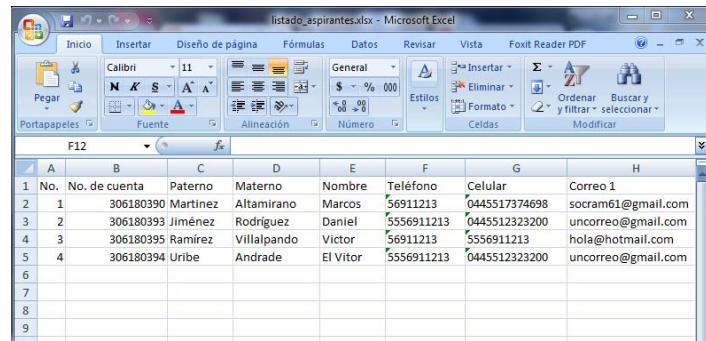
Al momento hay 4 solicitudes en total

[Listado de solicitudes \(campos FUNAM\)](#)

[Listado de solicitudes \(todos los campos\)](#)

ILUSTRACIÓN 54 - BOTONES PARA DESCARGAR LOS LISTADOS DE SOLICITUDES

A continuación se muestra una imagen que muestra cómo son los listados descritos anteriormente.



The screenshot shows a Microsoft Excel spreadsheet with the following data:

No.	No. de cuenta	Paterno	Materno	Nombre	Teléfono	Celular	Correo 1	
1	1	306180390	Martínez	Altamirano	Marcos	56911213	0445517374698	socram61@gmail.com
3	2	306180393	Jiménez	Rodríguez	Daniel	5556911213	0445512323200	uncorreos@gmail.com
4	3	306180395	Ramírez	Villalpando	Victor	56911213	5556911213	hola@hotmail.com
5	4	306180394	Uribe	Andrade	El Vitor	5556911213	0445512323200	uncorreos@gmail.com

ILUSTRACIÓN 55 - EJEMPLO DE LISTADO GENERADO POR EL SISTEMA

5.4.2. DATOS DE SOLICITUDES.

El sistema permite acceder a los datos de las solicitudes individualmente de manera sencilla en caso de necesitarse, a continuación se muestran los controles que lo permiten:

Datos de solicitud

Ingrese el número de cuenta de algún alumno para mostrar sus datos

Número de cuenta (sólo números)

ILUSTRACIÓN 56 - PERSONALIZACIÓN DEL REPORTE DE DATOS

Desarrollo del sistema

Como se puede ver, simplemente se requiere ingresar el número de cuenta del alumno que solicitó el apoyo y dar clic en el botón *siguiente*.

Cabe señalar que en esta vista donde se muestran los datos de la solicitud, es posible cambiar el estado de la misma. Las opciones disponibles son almacenadas en la base de datos lo que simplifica la administración del sistema como tal, al momento de escribir este documento, son:

- ✓ Solicitud primer ingreso - Primera vez en el programa
- ✓ Otorgada - Se ha otorgado el apoyo por primera vez
- ✓ Renovada - Otra vez en el programa
- ✓ Cancelada - Se le quito el apoyo al alumno
- ✓ Rechazada - No se le otorgó la beca al alumno
- ✓ Solicitud renovación - Se solicita la renovación del apoyo

A continuación se muestra la vista donde puede cambiarse el estado de la beca o solicitud que se está analizando.

Sexo	Masculino
Fecha de nacimiento	1990-06-16
Fecha de registro	2013-11-19
Estado de la beca	Renovada (estado actual)
Estado donde vive	Renovada <input type="button" value="Cambiar"/>
Municipio o delegación	Solicitud primer ingreso
Colonia	Otorgada
Calle y número	Renovada
Código postal	Cancelada
	Rechazada
	Solicitud renovación

ILUSTRACIÓN 57 - CAMBIO DEL ESTADO DE LA SOLICITUD DE BECA

Desarrollo del sistema

A continuación se presenta una imagen con parte de la vista mencionada, se muestran los datos de la solicitud y también los controles para cambiar el estado de la beca.

Administración del Programa de Apoyo Nutricional

Información de solicitud

No. de cuenta	306180390
Nombre	Martinez Altamirano Marcos
Teléfono	56911213
Celular	0445517374698
Correo electrónico	socram61@gmail.com
Correo electrónico alternativo	otrocorreo@yahoo.com.mx
Carrera	Ingeniería en Computación
Edo. civil	Casada (o)
Trabaja	si
Aporta al ingreso familiar	si
Con quien vive	Ambos
Sostén económico	Otros
Ingreso mensual	más de 5000
La casa donde vive es	de un familiar
Tiempo traslado	Más de tres horas
CURP	MARM900616HDFRDR01
Sexo	Masculino
Fecha de nacimiento	1990-06-16
Fecha de registro	2013-11-19
Estado de la beca	Renovada (estado actual) <div style="display: flex; align-items: center;"><div style="border: 1px solid #ccc; padding: 2px;">Renovada ▼</div><div style="margin-left: 5px; border: 1px solid #ccc; padding: 2px;">Cambiar</div></div>
Estado donde vive	Distrito Federal
Municipio o delegación	Venustiano
Colonia	Santa Cruz
Calle y número	Calle 45
Código postal	9

ILUSTRACIÓN 58 - INFORMACIÓN DEL ALUMNO Y ESTADO DE LA BECA

También es posible consultar los documentos que se entregaron al momento de solicitar la beca dentro de esta vista, aparece también una tabla con enlaces a los documentos que se cargaron.

Tipo de documento	Enlace
Historial académico	Ver
Comprobante inscripción	Ver
Credencial UNAM	Ver
Carta exposición de motivos	Ver
Fotografía	Ver
Comprobante de percepciones	Ver
Comprobante de domicilio	Ver

ILUSTRACIÓN 59 - DOCUMENTOS DISPONIBLES DEL ALUMNO

5.4.3. REPORTE DE ENCUESTAS

En esta sección del módulo se permite obtener un reporte de las respuestas de los beneficiarios del programa en las encuestas. El funcionamiento es también bastante sencillo, simplemente se ingresa la fecha de inicio y final del periodo a evaluar indicando también el número de cuenta de algún beneficiario si se requiere algún reporte en particular.

A continuación se presenta la imagen de la interfaz que controla esta funcionalidad.

Datos de solicitud

Ingrese el número de cuenta de algún alumno p

Número de cuenta (sólo números)

Reportes de encuestas

Ingrese el rango de las fechas para mostrar los reportes, active la cas

Seleccione la fecha de inicio del reporte (aaaa-mm-dd):

Seleccione la fecha de corte del reporte (aaaa-mm-dd):

Personalizar el reporte de la encuesta

Calendar widget showing Nov 2015. Days 29 and 30 are highlighted.

ILUSTRACIÓN 60 - CONTROLES DE MÓDULO DE ENCUESTAS

Una vez que se indican los datos a trabajar, se presenta una pantalla con los resultados o un mensaje indicando que no se han respondido encuestas en el periodo indicado.

Administración del Programa de Apoyo Nutricional

Existen 3 beneficiarios del programa

Aún no se ha respondido encuestas del 2015-06-01 al 2015-12-31

ILUSTRACIÓN 61 - RESULTADOS DE BÚSQUEDA PARA EL PERÍODO SELECCIONADO

Desarrollo del sistema

En el caso de las gráficas, de forma similar a la generación de documentos en formatos PDF y de Microsoft Office Excel, se ha implementado una biblioteca que permite mostrar la información de forma ágil, interactiva y con bastante presencia estética. La biblioteca empleada se llama Highcharts, permite la creación de varios tipos de gráficos y puede usarse libremente para proyectos sin fines de lucro, esta biblioteca puede consultarse y obtener una copia en la dirección electrónica <http://www.highcharts.com/>.

En caso de que existan datos, se presenta la información resumida en gráficas como se ha descrito anteriormente, las gráficas están agrupadas de acuerdo a las preguntas que se formularon en sus respectivas categorías, en la siguiente imagen puede verse un ejemplo de los resultados sobre la calidad de los alimentos que se ofrecen en la cafetería.

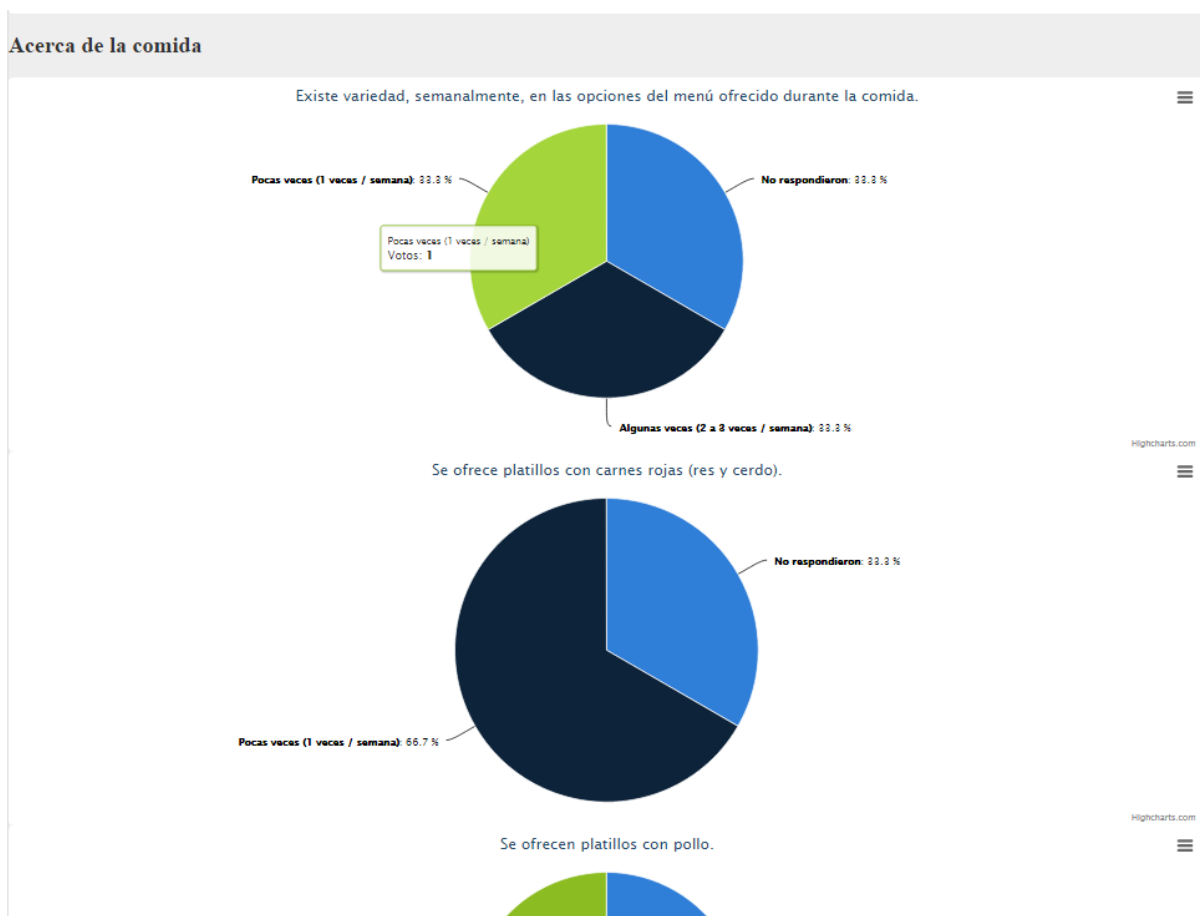


ILUSTRACIÓN 62 - RESULTADOS DE LAS ENCUESTAS

6. INSTALACIÓN DEL SISTEMA

6.1. CONFIGURACIÓN RECOMENDADA.

La instalación del sistema tiene requerimientos en cuanto a software se refiere que permiten la creación de un entorno de producción con relativa facilidad ya que emplea software libre y con gran aceptación, actualmente se ha puesto en marcha únicamente un módulo del sistema para el conteo de consumos en la Facultad de Química sobre plataforma Windows.

Al ser un sistema basado en web es posible, incluso necesario en algunas circunstancias, instalarlo en un sitio público para tener acceso vía internet por lo que se requerirá tener un sistema capaz de alojarlo; dicho sistema podría ser una plataforma Windows o *nix siempre y cuando se cumpla con los requisitos del sistema.

6.1.1. REQUISITOS DE SISTEMA.

PHP 5.2 con las extensiones php_zip, php_xml, php_gd2, MySQLi. Se recomienda al menos tener 80 MB como memoria máxima para la ejecución del script y ampliar los límites para carga de archivos en tamaño y tiempo así como tiempo máximo de ejecución del script para evitar pérdida de información o errores inesperados debido al entorno de ejecución del sistema.

MySQL 5.5 es recomendado, aunque podría funcionar con 4.1.13+, o bien 5.0.7+ según la documentación de la extensión MySQLi. Es necesario el motor InnoDB debido a que se requiere soporte para transacciones en el módulo de registro.

El sistema está diseñado para funcionar en equipos con pocos recursos disponibles, al ser un sistema basado en web, la demanda de recursos podría aumentar de acuerdo al número de peticiones concurrentes se requiera atender. Se recomienda al menos un equipo con procesador de doble núcleo a una velocidad de 2.0 GHz y 4 GB de memoria RAM.

6.1.2. PLATAFORMA RECOMENDADA.

Debido a las características del sistema, la facilidad para obtener y usar el sistema, las herramientas que ofrece la plataforma y cantidad de recursos humanos disponibles, se recomienda emplear una distribución GNU/Linux en especial Ubuntu para servidores en una versión LTS (Long-term Support) que según la documentación del sistema tiene soporte por 5 años a partir de la fecha en que se libera por parte de Canonical, una empresa que le da soporte a la distribución.

Instalación del sistema

Al momento de escribir este documento, la última versión del sistema operativo Ubuntu Server es 14.04 LTS que cumple con los requerimientos de software recomendados, además cuenta con un gran soporte, aceptación por la comunidad, fabricantes y corre sobre diversas arquitecturas. Se puede obtener mayor detalle de las características de la distribución en la dirección electrónica

<http://www.ubuntu.com/server>.

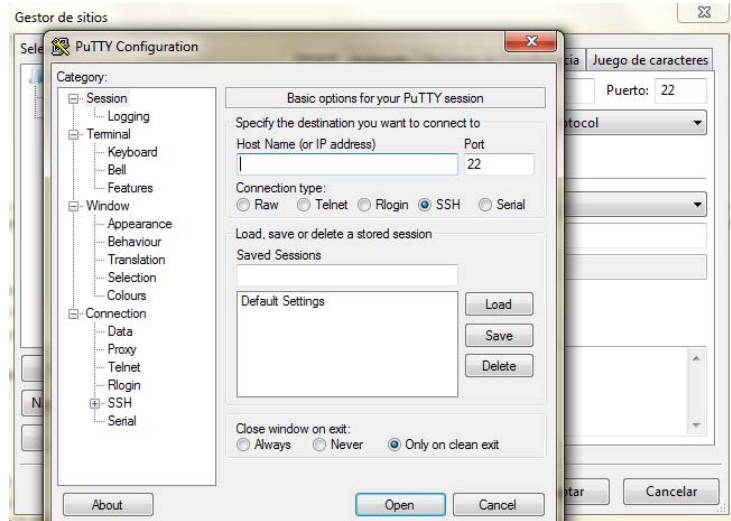


ILUSTRACIÓN 63 - PUTTY (CLIENTE SSH)

Es recomendable que se tengan activos servicios para administrar y monitorear el sistema operativo así como el sistema de cómputo discutido en este documento.

Existen varias opciones que permiten realizar conexiones con el servidor con relativa facilidad, se ha seleccionado SSH porque permite tener un buen nivel de seguridad y es ampliamente aceptado.

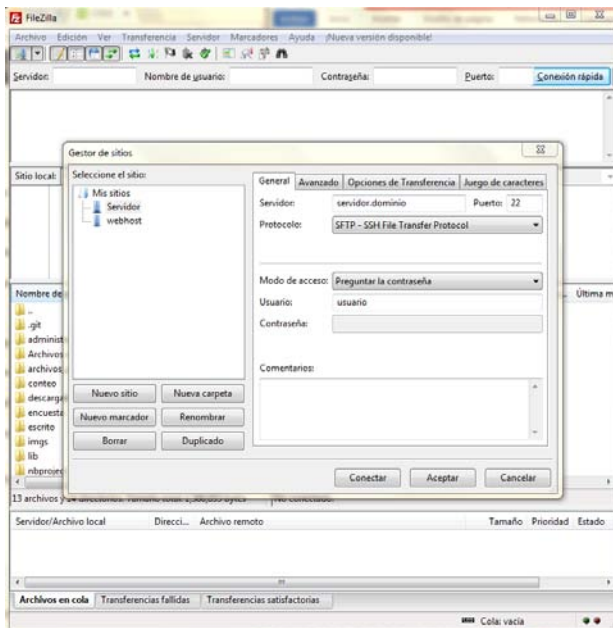


ILUSTRACIÓN 64 - FILEZILLA, CLIENTE FTP

que es bastante fácil de utilizar, ambas herramientas Filezilla y PuTTY son software de código abierto.

En el mercado se encuentran diversas herramientas que permiten la interacción con los servidores tanto para transferir archivos como para trabajar remotamente usando la línea de comandos; en el caso de transferencia de archivos se recomienda usar la herramienta *Filezilla* con el protocolo SFTP, esta herramienta es distribuida libremente bajo licencia GPL, puede consultarse sus características y obtener una copia en la dirección electrónica

<https://filezilla-project.org/index.php>;

para trabajar remotamente con línea de comandos se recomienda usar la herramienta *PuTTY* que es un cliente SSH y Telnet ampliamente utilizado que se distribuye bajo licencia MIT por lo

7. CONCLUSIONES Y FUTURO DEL TRABAJO

7.1. PRUEBAS Y RESULTADOS.

Las pruebas iniciales se hicieron fueron las siguientes:

- ✓ Se validó que la información de las solicitudes de becas se almacenaran correctamente en la base de datos al momento de hacer los registros, poniendo especial atención en problemas con la codificación de caracteres para conservar letras acentuadas o tildadas como es el caso de la letra 'ñ' por citar algunos ejemplos.
- ✓ También se analizaron los diferentes métodos para generar y mantener sesiones dentro del sistema conservando los datos entre las etapas de los procesos que se tienen contemplados en el sistema en general.
- ✓ Se realizaron pruebas minuciosas de integridad y correspondencia entre la información presentada en los reportes generales y los gráficos contra la información que está almacenada en la base de datos.
- ✓ Se verificaron la codificación de los archivos y caracteres en las conexiones del api funcionando entre PHP y MySQL para presentar correctamente la información en los reportes tanto PDF como en formato de Microsoft Office Excel.
- ✓ Se verificaron también la generación de los formularios en el módulo de encuesta y la generación de claves e índices de los controles para asegurar la correspondencia entre la opinión del usuario y los datos asentados en la base de datos.
- ✓ Se realizaron pruebas para corroborar la correcta aplicación de los filtros en el sistema, por ejemplo, al desplegar los datos de una solicitud o hacer un reporte individual de los reportes de encuestas.

Los resultados de las pruebas permitieron confirmar la robustez y confiabilidad del sistema en cuanto al almacenamiento de la información.

Se corrigieron posibles problemas en el futuro con la codificación de los caracteres puesto que en algunas ocasiones aparecen símbolos en vez de las letras requeridas debido a que los archivos se han almacenado en alguna codificación distinta al UTF-8 sin BOM.

Como se mencionó anteriormente, se instaló el módulo del sistema para automatizar el conteo de consumos en la Facultad de Química, lo que ha otorgado una reducción considerable del tiempo dedicado a esas tareas.

Conclusiones y futuro del trabajo

7.2. CONCLUSIONES

El presente trabajo ha representado un gran esfuerzo y una gran oportunidad para practicar, usar nuevas tecnologías y sobre todo, aprender más. Tener la oportunidad de trabajar con personas de diferentes áreas, formación, formas de ser y hacer las cosas siempre tiene un gran premio que es tener nuevas ideas y puntos de vista, cuando se desarrolla una nueva herramienta se crea algo que no estaba, se busca solventar un problema y se emplea toda la creatividad con la que se cuenta para hacerlo de la mejor manera posible pero en algunas ocasiones aparte de conseguir un producto en calidad de operación, se obtienen nuevas experiencias, nuevos objetivos y nuevos caminos que no se habían contemplado, lo cual, al menos para el que escribe, es una de las más grandes satisfacciones.

Es un hecho que los sistemas de cómputo no son eternos, lo que un día sirve y es tendencia al otro día puede verse superado por las circunstancias o por los cambios tecnológicos que suceden tan rápido que siempre hay una nueva tecnología más eficiente, más potente y ya es generalmente aceptada. El presente trabajo describe la primera versión de un sistema que pretende automatizar las tareas monótonas y bastante consumidoras de tiempo para los encargados de operar el Programa de Apoyo Nutricional, sería un orgullo que el sistema desarrollado sea explotado a su máxima capacidad, brinde el apoyo a los interesados y sobre todo se multiplique el esfuerzo realizado en su creación para hacer *“el trabajo pesado”* como se dice coloquialmente.

El desarrollo de sistemas no es una tarea sencilla, involucra un gran compromiso, gran cantidad de conocimiento, una investigación profunda de lo que se requiere y una gran cantidad de creatividad que permita aterrizar las ideas y concentrarlas en un producto, un procedimiento y en resultados significativos en cuanto a mejora de las actividades que viene a cumplir si pudiésemos comparar el rendimiento contra el proceso manual de la información.

Con las herramientas que ofrece el sistema de cómputo, puede obtenerse una gran cantidad de ahorros para todos los involucrados en el Programa de Apoyo Nutricional; ésto se da desde que se elimina o se reduce drásticamente la cantidad de papel requerido para hacer solicitudes con lo que se reduce el impacto ambiental del programa, también se reduce la cantidad de espacio requerido para los expedientes (si existen físicamente, podrían conservarse en formato digital), se eliminan tiempos de traslado hacia las oficinas de responsables o los lugares destinados para el cumplimiento de trámites, por citar algunos ejemplos.

Como resultado del trabajo realizado se cuenta con un sistema de cómputo robusto, confiable y con mínimos requisitos de sistema que puede ayudar en gran medida a todos los responsables de las escuelas o facultades forman parte del Programa de Apoyo Nutricional.

Conclusiones y futuro del trabajo

Se espera que continúe la instalación de los distintos módulos que conforman el sistema de cómputo en las distintas escuelas o facultades participantes del programa, cuando esto suceda podrá obtenerse realimentación para continuar el desarrollo del sistema y conforme madure el proyecto se incorporen nuevas funciones, se mejore las existentes, se adecuen los componentes a requerimientos específicos de algunas instituciones entre otras actividades.

7.3.CONTRIBUCIONES Y MEJORAS

Conforme se use el sistema, los usuarios podrán aportar ideas que permitan mejorar el funcionamiento y agregar funciones para que les sea de mayor utilidad, además, de apoyar con información ante cualquier posible bug que aparezca, recordemos que el sistema está creado por un humano y hay errores ocasionalmente, usualmente es mala idea generalizar pero tal vez no haya software libre de errores.

Existen múltiples mejoras que pudieran realizarse, por ejemplo, podría hacerse paneles de administración para reducir todavía más la necesidad de un administrador de tiempo completo que le de soporte al sistema, de esta manera, no se necesitaría a un programador o un administrador de base de datos sino que el propio responsable del programa en la institución podría llevar a cabo acciones administrativas desde una interfaz gráfica. La razón por la que se ha delimitado hasta este punto el desarrollo es para tener una versión disponible y no alargar el proceso, también para destinar el tiempo a comprobar y obtener realimentación una vez que se implemente el sistema en su totalidad, desde luego se presentarán inconvenientes o ajustes parciales a los requerimientos especiales de ciertas instituciones pero podrá manejarse una mejora continua mientras el sistema crezca y madure junto con el Programa de Apoyo Nutricional.

7.4.LIMITACIONES

La manera en que está construido el sistema permite que se incorporen nuevas funcionalidades sin mayor esfuerzo, es posible incorporar nuevos módulos o perfiles de usuario, como se ha comentado es esperado un crecimiento del mismo de acuerdo al uso que se le dé al sistema.

Uno de los grandes problemas al desarrollar nuevas funciones e incorporarlas al uso diario es la dificultad administrativa que en ocasiones se presenta tanto para obtener información como para tener acceso a servidores y equipo de cómputo requerido para operar el sistema.

Otro de los limitantes del desarrollo del sistema es el tiempo, como el lector sabrá, el desarrollo de sistemas es una actividad que requiere bastante dedicación y conocimiento, a pesar de que el sistema haya sido diseñado para funcionar sobre las herramientas más comunes y aceptadas por la comunidad, incorporar a más

Conclusiones y futuro del trabajo

recursos humanos a un proyecto es una tarea complicada y en muchas ocasiones nos vemos en necesidad de incorporarnos a varios proyectos a la vez que limitan la cantidad de tiempo disponible para mantener un sistema.

7.5.LÍNEAS FUTURAS.

Como se ha comentado en repetidas ocasiones, el sistema es ampliamente escalable, la cantidad de recursos que necesita para operar es mínima y por lo mismo es que se espera haya muy buena aceptación del mismo, aunado a eso, se ha desarrollado para tener una gran cantidad de recursos humanos a disposición mejorando así la viabilidad del mismo a largo plazo.

El empleo de bibliotecas que permiten la creación y lectura de documentos en formatos ampliamente aceptados permite al sistema tener más posibilidades de interconexión con software ya establecido, ampliando de esta manera el campo de acción al que puede llevarse, inclusive, con el uso de APIs o web services podría interactuar con otros sistemas similares y ofrecer información así como consumirla para complementar la funcionalidad de las aplicaciones.

8. BIBLIOGRAFÍA

- ✓ <https://developer.mozilla.org/en-US/docs/Web/HTML> [consultado 10 de diciembre de 2015 a las 13:30 horas]
- ✓ <https://dojotoolkit.org> [consultado 3 de diciembre de 2015 a las 11:30 horas]
- ✓ <http://prototypejs.org/> [consultado 11 de noviembre de 2015 a las 13:30 horas]
- ✓ <http://sass-lang.com/> [consultado 3 de diciembre de 2015 a las 12:40 horas]
- ✓ <http://www.w3.org/Consortium/> [consultado 3 de diciembre de 2015 a las 9:40 horas]
- ✓ <https://developer.mozilla.org/es/> [consultado 4 de diciembre de 2015 a las 9:42 horas]
- ✓ <https://developers.google.com/web/fundamentals/getting-started/your-first-multi-screen-site/responsive?hl=en> [consultado 4 de diciembre de 2015 a las 10:52 horas]
- ✓ <http://foundation.zurb.com/sites/docs/equalizer> [consultado 7 de diciembre de 2015 a las 9:42 horas]
- ✓ <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide> [consultado 8 de diciembre de 2015 a las 8:42 horas]
- ✓ https://www.ibm.com/developerworks/ssa/data/library/tipos_bases_de_datos/ [consultado 6 de diciembre de 2015 a las 10:42 horas]
- ✓ http://www.aulaclie.es/sqlserver/b_1_1_4.htm [consultado 4 de diciembre de 2015 a las 8:53 horas]
- ✓ <http://www.lcc.uma.es/~galvez/ftp/bdst/Tema2.pdf> [consultado 6 de diciembre de 2015 a las 8:42 horas]
- ✓ <https://support.office.com/es-mx/article/Conceptos-b%C3%A1sicos-sobre-bases-de-datos-a849ac16-07c7-4a31-9948-3c8c94a7c204> [consultado 7 de diciembre de 2015 a las 18:42 horas]
- ✓ http://www.tutorialspoint.com/asp.net/asp.net_introduction.htm [consultado 1 de enero de 2016 a las 18:12 horas]
- ✓ <https://www.djangoproject.com/start/overview/> [consultado 1 de enero de 2016 a las 19:42 horas]
- ✓ <http://www.tutorialspoint.com/python> [consultado 1 de enero de 2016 a las 20:15 horas]
- ✓ <https://www.ruby-lang.org/es/about/> [consultado 1 de enero de 2016 a las 20:42 horas]
- ✓ <http://www.w3schools.com/xml> [consultado 9 de diciembre de 2015 a las 18:42 horas]
- ✓ DATE, C. J., *An Introduction to Database Systems* 8a. Edición. Reading, Massachusetts, U.S.A Addison Wesley, 2003

Bibliografía

- ✓ DE MIGUEL MARTÍNEZ, Adoración, PIATTINI , Mario, ESPERANZA, Marcos. *Diseño de bases de datos relacionales*. México Alfaomega, 2000
- ✓ COMER, Douglas E. *Internetworking with TCP/IP Vol. III. Client server Programming and applications*. 3a. edición. USA Prentice Hall, 2000
- ✓ SOMMERVILLE, Ian. *Software Engineering* 7a Edición. Reading, Massachussets, U.S.A Addison Wesley, 2004
- ✓ HANSEN, Gary W., HANSEN, James V. *Diseño y administraciòn de bases de datos* 2ª. Edición. México, Prentice Hall,1997
- ✓ GILMORE W. J. (W Jason), BRYLA Bob. *Beginning PHP and Oracle: from novice to professional*, California, U.S.A., Apress 2007