

ACADEMIA JOURNALS



OPUS PRO SCIENTIA ET STUDIUM

Humanidades, Ciencia, Tecnología e Innovación en Puebla

ISSN 2644-0903 online

Vol. 4. No. 1, 2022

www.academiajournals.com

TRABAJO DE INVESTIGACIÓN AUSPICIADO POR EL
CONVENIO CONCYTEP-ACADEMIA JOURNALS



Gobierno de Puebla

Hacer historia. Hacer futuro.



Secretaría
de Educación
Gobierno de Puebla

CONCYTEP
Consejo de Ciencia
y Tecnología del Estado
de Puebla

Brandom Baruch González Cervantes

Sistema de Encuestas para Tutorías del Programa "Yo No Abandono" en el Área de Tutorías en Instituciones de Educación Media Superior

Tecnológico Nacional de México
Campus San Martín Texmelucan

Asesor: L.C.C. Iván Rafael Sánchez Juárez

Revisor: Mtra. Vianney Morales Zamora

Revisor: Mtra. Petra Paredes Xochihua



**TECNOLÓGICO NACIONAL DE MÉXICO
CAMPUS SAN MARTÍN TEXMELUCAN
DIVISIÓN DE INGENIERÍA EN SISTEMAS
COMPUTACIONALES**

**SISTEMA DE ENCUESTAS PARA TUTORÍAS DEL PROGRAMA “YO NO
ABANDONO” EN EL ÁREA DE TUTORÍAS EN INSTITUCIONES DE
EDUCACIÓN MEDIA SUPERIOR**

TESIS

QUE PARA OBTENER EL GRADO DE:

LICENCIADO EN INGENIERÍA EN SISTEMAS COMPUTACIONALES

PRESENTA:

**Brandom Baruch González Cervantes
16240011**

**ASESOR: L.C.C. Iván Rafael Sánchez Juárez
REVISOR: Mtra. Vianney Morales Zamora
REVISOR: Mtra. Petra Paredes Xochihua**

SAN MARTIN TEXMELUCAN, PUE. JULIO 2021

SISTEMA DE ENCUESTAS PARA TUTORÍAS DEL PROGRAMA “YO NO ABANDONO” EN EL ÁREA DE TUTORÍAS EN INSTITUCIONES DE EDUCACIÓN MEDIA SUPERIOR

Autor: Brandom Baruch González Cervantes

Asesor: L.C.C. Iván Rafael Sánchez Juárez

Tecnológico Nacional de México Campus San Martín Texmelucan

División de Ingeniería en Sistemas Computacionales

RESUMEN

El proyecto consiste en realizar un sistema de encuestas para la materia de tutorías utilizando las encuestas del programa “Yo no Abandono” creada por la SEP. El programa consiste en reducir el nivel de deserción para el nivel de educación media superior. El sistema web puede registrar y gestionar información obtenida, el tutor podrá dar seguimiento individual a los alumnos y los padres podrán contestar las encuestas del programa “Yo no Abandono”. Como objetivo general es desarrollar un sistema web de encuestas apoyando la labor tutorial, gestionando información de alumnos, docentes y padres reduciendo las deserciones escolares. Para realizar el objetivo principal, se implementó el módulo de registros para los docentes, padres de familia, alumnos, materias, grupos, puestos de los docentes y domicilios, el módulo de entrevistas frescas para los alumnos y padres, diseñar los reportes en pdf de las encuestas contestadas para dar seguimiento individual a cada alumno, gestionar toda la información de los alumnos, docentes, padres, etc, también, se realizó un buscador para los alumnos y padres para actualizar los datos de cada uno y el módulo de inicio de sesión para los diferentes usuarios del sistema (alumnos, docentes y padres). Como metodologías se utilizó scrum para dar seguimiento al desarrollo del sistema web, la cuantitativa y cualitativa se puede observar en las encuestas y el tutor podrá sacar conclusiones si es necesario dar seguimiento a un alumno. En los resultados se comprobó la hipótesis a través de pruebas realizadas por alumnos, padres y el director teniendo resultados la comparación de tiempos en contestar encuestas, entrevista, test y tener una mejor administración de datos. Como conclusiones se cumplió el objetivo general, se demostró la hipótesis a bases de pruebas con los docentes, alumnos y padres de familia y se aprendió nuevas herramientas para el desarrollo del sistema web.

Palabras clave: Programa “Yo no abandono”, educación media superior, gestión de información, encuestas, sistema web, reducir la deserción escolar.

	Nombre del documento: Dictamen para Titulación Integral	Código: ITSSMT-AC-NOR-01-FO-04	
	Referencia del Documento: Lineamiento para la Titulación Integral	Revisión: 1 Página 1 de 1	

San Martín Texmelucan, Pue., a 28 de abril de 2021
ASUNTO: Dictamen para Titulación Integral

C. BRANDOM BARUCH GONZÁLEZ CERVANTES
PASANTE DE LA CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES
P R E S E N T E

En respuesta a su solicitud de titulación integral con el proyecto "**Sistema de encuestas para tutorías del programa "Yo no abandono" en el área de tutorías en instituciones de educación media superior.**", me es grato informarle que fue **aceptado**, en modalidad **TESIS**, y se confirma como asesor el **Mtro. Iván Rafael Sánchez Juárez**.

Por lo que le solicito ponerse en contacto con su asesor, en caso de ser necesario.



Además, le informo que deberá pasar al Departamento de Servicios Escolares, a que le revisen su documentación.

ATENTAMENTE

Formación Tecnológica de Vanguardia para el Desarrollo Regional

ALTAGRACIA BERRUECOS XICHTENCATL
JEFA DE LA DIVISIÓN DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

c.c.p. Subdirección de Estudios Profesionales. - PSC
 Departamento de Servicios Escolares. - PSC
 Expediente

	Nombre del documento: Autorización de entrega de informe de Titulación.	Código: ITSSMT-AC-NOR-01-FO-08	
	Referencia del Documento: Lineamiento para la Titulación Integral	Revisión: 1	

San Martín Texmelucan, Puebla a 21 de julio de 2021

C. BRANDOM BARUCH GONZÁLEZ CERVANTES
PASANTE DE LA CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES
P R E S E N T E

De acuerdo a la normatividad vigente de nuestro instituto y habiendo cumplido con todas las indicaciones que el comité revisor le hizo respecto a su informe de titulación integral titulado: "Sistema de encuestas para tutorías del programa "Yo no abandono" en el área de tutorías en instituciones de educación media superior.", comunico a Usted que:

HA SIDO LIBERADO Y SE LE AUTORIZA PARA QUE PROCEDA A LA FORMALIZACION DEL MISMO

Para lo cual deberá entregar su documento digital a la Subdirección de Estudios Profesionales, en un plazo de diez días hábiles contados a partir de la fecha del presente.

Es importante mencionar que usted deberá elegir la manera en que conservará su trabajo de titulación (libro, disco, etc.).



ATENTAMENTE

Formación Tecnológica de Vanguardia para el Desarrollo Regional



Mtra. Altagracia Berruecos Xicohténcatl
Jefa de la División de la Ingeniería en Sistemas Computacionales

c.c.p. Expediente

	Nombre del documento: Liberación de Proyecto para la Titulación Integral	Código: ITSSMT-AC-NOR-01-FO-03	
	Referencia del Documento: Lineamiento para la Titulación Integral	Revisión: 1 Página 1 de 1	

San Martín Texmelucan, Pue., a 14 de julio de 2021
 Asunto: Liberación de proyecto para la titulación integral.

**C. ING. DAMIAN HUERTA GARCIA
 SUBDIRECTOR DE ESTUDIOS PROFESIONALES DEL
 INSTITUTO TECNOLÓGICO SUPERIOR DE SAN MARTÍN TEXMELUCAN
 PRESENTE**

Por este medio informo que ha sido liberado el siguiente proyecto para la titulación integral:

Nombre del/la estudiante y/o egresado/a:	Brandom Baruch González Cervantes
Carrera:	Ingeniería en Sistemas Computacionales
No. de control:	16240011
Nombre del proyecto:	Sistema de encuestas para tutorías del programa "Yo no abandono" en el área de tutorías en instituciones de educación media superior.
Producto:	Tesis

Agradezco de antemano su valioso apoyo en esta importante actividad para la formación profesional de nuestros/as egresados/as.

ATENTAMENTE

Formación Tecnológica de Vanguardia para el Desarrollo Regional



**MTRA. ALTAGRACIA BERRUECOS XICOHTÉNCATL
 JEFA DE LA DIVISIÓN DE INGENIERÍA EN SISTEMAS COMPUTACIONALES**

Nombre y firma del asesor	Nombre y firma del revisora*	Nombre y firma del revisor*
 Lic. Iván Rafael Sánchez Juárez	 Mtra. Vianney Morales Zamora	 Mtra. María Petra Paredes Xochihua

* Solo aplica para el caso de tesis o tesina.

c.c.p.- Expediente

Agradecimientos

Dedico de manera específica a mis padres Fernando González Barrientos y Elia Cervantes Estrada, ellos fueron el principal cimiento para la construcción de mi vida profesional, sentaron en mi la base de responsabilidad, honestidad y deseos de superación, en ellos tengo el espejo en el cual me quiero reflejar ya que sus virtudes y su gran corazón me llevan a admirarlos cada día más. A mis hermanos y familiares que son personas que han confiado en mí y me han brindado de su apoyo en cualquier momento que los necesité.

En memoria a Cristel Salas Ramírez que en paz descansa por su confianza, su amor y brindarme el tiempo necesario para superarme. No fue sencillo culminar con éxito este proyecto, sin embargo, fuiste y serás mi motivación para seguir adelante. Igualmente quiero dar gracias a mis abuelos que están en el cielo y mi abuela Cruz Barrientos Cervantes aquí presente, gracias por creer en mí, por los consejos, por el amor que me dieron y por el apoyo incondicional en mi vida.

Índice general

1. CAPÍTULO I INTRODUCCIÓN	3
1.1. Problema de Investigación	5
1.2. Justificación	6
1.3. Objetivos del Estudio	7
1.3.1. Objetivo General.....	7
1.3.2. Objetivos específicos.....	7
1.4. Alcances y Limitaciones	7
1.4.1. Alcances.....	7
1.4.2. Limitaciones.....	8
1.5. Planteamiento de los Supuestos	8
1.6. Hipótesis	9
2. CAPÍTULO II REVISIÓN DE LA LITERATURA.....	10
2.1. Diseño e implementación de una plataforma digital para la gestión de tutorías y su impacto en la deserción de estudiantes de nivel superior	11
2.2. Sistema de automatización para el programa “Yo no abandono” del CBTIS No. 61.....	11
2.3. Estudio evaluativo del Sistema Institucional de Tutorías para el seguimiento de trayectorias escolares	12
2.4. La tutoría a través de una plataforma web	12
2.5. Sistema web: Herramienta para la gestión del programa de tutorías.....	13
3. CAPÍTULO III MARCO TEÓRICO	14
3.1. Metodología	15
3.1.1. Metodología SCRUM.....	15
3.2. Laravel	16
3.2.1. Modelo Vista Controlador (MVC).....	16
3.2.2. Estructura de un proyecto.....	17
3.3. Composer	18
3.4. DomPDF	19
3.5. PHP.....	19
3.6. Wampserver.....	19
3.7. DigitalOcean	19
3.8. Mailgun	20

3.9. HTML 5	20
3.10. SQL Power Architect	20
3.11. Bootstrap.....	21
3.12. Balsamiq	21
3.13. Sublimet Text 3	21
3.14. GitHub.....	22
3.15. Programa “Yo no abandono”	22
3.16. Cmder	22
4. CAPÍTULO IV METODOLOGÍA APLICADA	23
4.1. Descripción de la Empresa u Organización.....	24
4.2. Organigrama y Lay Out	25
4.2.1. Organigrama.....	25
4.2.2. Lay Out.....	26
4.3. Descripción del puesto o área de trabajo en la que participó el/la estudiante.	27
4.4. Fase de análisis	28
4.4.1. Requerimientos del hardware.....	28
4.4.2. Requerimientos de Software.....	28
4.4.3. Requerimientos Funcionales.....	29
4.4.4. Requerimientos No Funcionales.....	29
4.5. Modelo de la base de datos	30
4.5.1. Diccionario de datos.....	32
4.6. Planeación del proyecto	43
4.7. Desarrollo del proyecto	43
4.7. Fase de pruebas	94
5. CAPÍTULO V RESULTADOS	100
5.1. Análisis de Tiempo de Captura.....	101
5.2. Análisis de las actividades	102
5.2.1. Actividad 1: Responder encuestas de cada alumno.....	102
5.2.2. Actividad 2: Llenar entrevistas fresca alumnos.....	104
5.2.3. Actividad 3: Contestar entrevista fresca padres.....	106
5.2.4. Actividades del director.....	108
5.3. Evaluación del sistema web	109

Conclusiones.....	111
Recomendaciones para el uso del sistema.....	112
Trabajos futuros.....	112
Referencias bibliográficas.....	113
ANEXO A1 MANUAL DE USUARIO.....	118
ANEXO A2 MANUAL TÉCNICO.....	119
GLOSARIO.....	120

Índice de figuras

Figura 3.1 Logotipo de Laravel.....	16
Figura 3.2 Logotipo Composer	18
Figura 3.3 Logotipo PHP	19
Figura 3.4 Logotipo WampServer	19
Figura 3.5 Logotipo DigitalOcean	20
Figura 3.6 Logotipo Mailgun	20
Figura 3.7 Logotipo HTML5.....	20
Figura 3.8 Logotipo SQL Power Architect.....	20
Figura 3.9 Logotipo Bootstrap	21
Figura 3.10 Logotipo Balsamiq.....	21
Figura 3.11 Logotipo Sublime Text 3.....	21
Figura 3.12 Logotipo GitHub	22
Figura 3.13 Logotipo Cmder	22
Figura 4.1 Organigrama del TECNM Campus San Martín Texmelucan	25
Figura 4.2 Esquema del edificio C planta baja	26
Figura 4.3 Esquema del edificio C planta alta	26
Figura 4.4 Modelo de la base de datos parte 1.....	30
Figura 4.5 Modelo de la base de datos parte 2.....	31
Figura 4.6 Modelo de la base de datos parte 3.....	32
Figura 4.7 Pantalla de inicio (Pantalla de baja)	45
Figura 4.8 Pantalla de inicio (Pantalla de alta)	45
Figura 4.9 Diseño del Login (Pantalla de baja)	48
Figura 4.10 Diseño del Login Alumno (Pantalla de alta)	48
Figura 4.11 Diseño del Login Docente (Pantalla de alta).....	48
Figura 4.12 Diseño del Login Padres (Pantalla de alta).....	49
Figura 4.13 Pantalla de listado de alumnos. (Pantalla de baja)	52
Figura 4.14 Pantalla de listado de alumnos. (Pantalla de alta)	53
Figura 4.15 Registro de alumno. (Pantalla de alta).....	53
Figura 4.16 Buscador de alumno(s). (Pantalla de alta).....	54
Figura 4.17 Pantalla de registro padre. (Pantalla de baja)	57
Figura 4.18 Pantalla de registro padre. (Pantalla de alta)	57
Figura 4.19 Listado de padres. (Pantalla de alta)	58
Figura 4.20 Diseño de panel de control. (Pantalla de baja)	59
Figura 4.21 Diseño de panel de control. (Pantalla de alta)	59
Figura 4.22 Pantalla de búsqueda. (Pantalla de alta)	61
Figura 4.23 Pantalla de datos generales padre. (Pantalla de alta)	62
Figura 4.24 Asignación de parentesco del director. (Pantalla de alta)	64
Figura 4.25 Asignación de parentesco del alumno. (Pantalla de alta)	64
Figura 4.26 Asignación de parentesco del padre. (Pantalla de alta)	65
Figura 4.27 Pantalla de registro de docente. (Pantalla de baja)	67

Figura 4.28 Pantalla de registro de docente. (Pantalla de alta)	67
Figura 4.29 Pantalla de listado de docentes. (Pantalla de alta)	68
Figura 4.30 Pantalla de datos generales. (Pantalla de alta)	69
Figura 4.31 Panel de control del Docente. (Pantalla de baja)	70
Figura 4.32 Panel de control del Docente. (Pantalla de alta)	70
Figura 4.33 Pantalla de listado de puestos. (Pantalla de alta)	73
Figura 4.34 Pantalla para registrar puesto. (Pantalla de alta)	73
Figura 4.35 Pantalla de asignar un puesto. (Pantalla de alta)	73
Figura 4.36 Pantalla de listado de domicilios. (Pantalla de alta)	76
Figura 4.37 Pantalla para registrar un domicilio. (Pantalla de alta)	76
Figura 4.38 Pantalla de listado de grupos. (Pantalla de alta)	79
Figura 4.39 Pantalla de registrar un grupo. (Pantalla de alta)	79
Figura 4.40 Pantalla de alumnos que pertenecen al grupo. (Pantalla de alta)	80
Figura 4.41 Pantalla de materias que pertenecen al grupo. (Pantalla de alta)	80
Figura 4.42 Pantalla de listado de materias. (Pantalla de alta)	83
Figura 4.43 Pantalla de registro de materia. (Pantalla de alta)	83
Figura 4.44 Panel de encuestas. (Pantalla de baja)	86
Figura 4.45 Panel de encuestas. (Pantalla de alta)	86
Figura 4.46 Panel de entrevistas. (Pantalla de alta)	87
Figura 4.47 Pantalla de listado de encuestas alumno. (Pantalla de alta)	87
Figura 4.48 Pantalla de sección "Entrevista fresca". (Pantalla de alta)	87
Figura 4.49 Pantalla de sección "Cuestionarios anexos". (Pantalla de alta)	88
Figura 4.50 Pantalla de sección "Test sobre estilos de aprendizaje". (Pantalla de alta)	88
Figura 4.51 Pantalla de panel de encuestas. (Pantalla de alta)	91
Figura 4.52 Pantalla para ver resultados. (Pantalla de alta)	92
Figura 4.53 Pantalla del PDF. (Pantalla de alta)	93
Figura 4.54 Error al ejecutar la primera migración	94
Figura 4.55 Archivo AppServicesProvider.php	95
Figura 4.56 Error de relación de modelos	95
Figura 4.57 Método padres	95
Figura 4.58 Relación de muchos a muchos	96
Figura 4.59 Agregando columnas para el método padres	96
Figura 4.60 Visualización del panel del alumno	96
Figura 4.61 Agregando withPivot y withTimesTamps al método padres	97
Figura 4.62 Datos demasiado largos para la columna	97
Figura 4.63 Ejecución del comando php artisan migrate:rollback	98
Figura 4.64 Ejecución del comando php artisan migrate	98
Figura 4.65 Columna "descripción" modificada	98
Figura 4.66 Reglas de validación	98
Figura 4.67 Ejecutando la regla de validación max	99
Figura 5.1 Tiempo en finalizar la entrevista fresca	102
Figura 5.2 Grafica de tiempo en contestar encuestas y test	104
Figura 5.3 Grafica de tiempo en contestar la entrevista fresca alumnos	106

Figura 5.4 Grafica de tiempo en contestar la entrevista fresca padres107
Figura 5.5 Grafica de tiempo de captura de datos109

Índice de tablas

Tabla 4.1 Requerimientos del Hardware	28
Tabla 4.2 Requerimientos del software	28
Tabla 4.3 Requerimientos Funcionales	29
Tabla 4.4 Requerimientos no funcionales.....	29
Tabla 4.5 Diccionario de usuarios	42
Tabla 4.6 Lista de tareas	43
Tabla 4.7 HU - pantalla de principal	43
Tabla 4.8 Tareas específicas Iteración 1	44
Tabla 4.9 Codificación. Iteración 1	46
Tabla 4.10 HU - Inicio de sesión	46
Tabla 4.11 Tareas específicas iteración 2	47
Tabla 4.12 Codificación y conexión de base de datos. Iteración 2	51
Tabla 4.13 HU - Registro de alumnos	51
Tabla 4.14 Tareas específicas iteración 3	52
Tabla 4.15 Codificación y conexión de base de datos. Iteración 3	55
Tabla 4.16 HU - Registro de padres de familia	56
Tabla 4.17 Tareas específicas iteración 4	56
Tabla 4.18 Codificación y conexión de base de datos. Iteración 4	60
Tabla 4.19 Motor de búsqueda.....	60
Tabla 4.20 Tareas específicas iteración 5	61
Tabla 4.21 Codificación y conexión de base de datos. Iteración 5	63
Tabla 4.22 HU - Asignación de parentesco	63
Tabla 4.23 Tareas específicas iteración 6	64
Tabla 4.24 Codificación y conexión de base de datos. Iteración 6	66
Tabla 4.25 HU - Registro de docentes	66
Tabla 4.26 Tareas específicas iteración 7	66
Tabla 4.27 Codificación y conexión de base de datos. Iteración 7	71
Tabla 4.28 Asignar puestos.....	72
Tabla 4.29 Tareas específicas iteración 8	72
Tabla 4.30 Codificación y conexión de base de datos. Iteración 8	75
Tabla 4.31 Registro de domicilios.....	75
Tabla 4.32 Tareas específicas iteración 9	75
Tabla 4.33 Codificación y conexión de base de datos. Iteración 9	77
Tabla 4.34 HU - Registro grupos	78
Tabla 4.35 Tareas específicas iteración 10	78
Tabla 4.36 Codificación y conexión de base de datos. Iteración 10	82
Tabla 4.37 Registro de materias	82
Tabla 4.38 Tareas específicas iteración 11	82
Tabla 4.39 Codificación y conexión de base de datos. Iteración 11	84
Tabla 4.40 HU - Apartado de entrevistas.....	85
Tabla 4.41 Tareas específicas iteración 12	85

Tabla 4.42 Codificación y conexión de base de datos. Iteración 1289

Tabla 4.43 HU - Generar PDF89

Tabla 4.44 Tareas específicas iteración 1390

Tabla 4.45 Codificación y conexión de base de datos. Iteración 1394

Tabla 5.1 Tiempo de registro.....101

Tabla 5.2 Tiempo en responder capturas y test.....103

Tabla 5.3 Tiempo en contestar la entrevista fresca alumnos105

Tabla 5.4 Tiempo en contestar entrevista fresca padres107

Tabla 5.5 Tiempo de captura de datos108

Tabla 5.6 Rubrica para evaluar el sistema web110

CAPÍTULO I

INTRODUCCIÓN

INTRODUCCIÓN

En esta tesis se describe el diseño y el desarrollo para realizar el proyecto. El proyecto consiste en realizar un sistema de encuestas para la materia de tutorías utilizando las encuestas del programa “Yo no Abandono” creada por la Secretaria de Educación Pública (SEP). El programa consiste en reducir el nivel de deserción para el nivel de educación media superior.

El sistema cuenta dos roles muy importantes el director y tutor. El administrador (puesto de director) podrá registrar, editar y eliminar alumnos, docentes, padres de familia, materias, puestos, etc. El administrador podrá asignar puestos como: tutor, director, docente, etc, los puestos más importantes para el sistema es el tutor y el director. El tutor tendrá un panel donde podrá observar los alumnos que tiene asignado el grupo y saber que alumno ha realizado las encuestas, si el alumno ha realizado las encuestas el tutor podrá descargar la información en formato PDF.

El alumno tendrá su panel donde observará su información (podrá editar dicha información), asignar un parentesco con un padre de familia, agregar su domicilio y saber que materias tiene asignada en su grupo, también, cuenta con las encuestas que debe de contestar. El padre de familia tiene parecido su panel con el del alumno, sin embargo, las encuestas que debe de realizar dependerán de los parentescos del alumno, por ejemplo, si tiene 2 parentescos tendrá que hacer 2 encuestas para cada alumno.

1.1. Problema de Investigación

La Dirección General de Bachilleratos (DGB) de la Secretaría de Educación Pública (SEP) ha creado el programa “Yo no abandono”, menciona que “El Movimiento contra el Abandono Escolar es una estrategia integral de carácter nacional que involucra la participación conjunta y coordinada de autoridades educativas, federales y estatales, directivos de planteles, docentes, padres de familia, estudiantes y sociedad en general, para lograr mayores índices de acceso, permanencia y conclusión exitosa de los estudios de nivel medio superior.” (Dirección General del Bachillerato, 2014). Dicho programa contiene una serie de 12 manuales que permiten a los trabajadores en Educación Media Superior (EMS), apoyar en la disminución de la deserción de estudiantes.

De acuerdo con los manuales del programa “Yo no abandono” (Secretaría de Educación Pública, 2015) el principal problema que enfrenta la educación media superior en la actualidad es el abandono escolar de miles de los jóvenes que se matriculan en este nivel educativo. Ante esta realidad, los esfuerzos para elevar la calidad de la enseñanza llegan tan sólo a una fracción mermada de los jóvenes en edad de cursar la educación media superior.

Actualmente en los bachilleratos generales del estado de Puebla se lleva el registro de alumnos, tutores, padres de familia y docentes en hojas de papel o en software de paquetería. Cuando un alumno ingresa a primer semestre, el director le solicita información como domicilio, con quien vive, edad, fecha de nacimiento, entre otras cosas. Una vez obtenido los datos, el director va llenando a mano todos los datos proporcionados por los estudiantes. Cuando se le asigna a un tutor, el tutor nuevamente debe solicitar esa información ocasionando pérdida de tiempo, desperdicio de papel y que en algunas ocasiones la información proporcionada cambie. Al transcurrir los semestres, el tutor les realiza diferentes encuestas con información que ya han respondido en otras ocasiones.

Estas encuestas son parte del programa “Yo no abandono”, sin embargo, cuando necesitan ver la información de encuestas anteriores, deben buscar en varios archivos físicos o buscar en documentos electrónicos las encuestas, ya que no tienen una administración adecuada a la hora de buscar dicha información.

1.2. Justificación

Se pretende desarrollar un sistema web de apoyo para prevenir los problemas de deserción y rezago de los alumnos que cursan el nivel medio superior. El sistema registrará los datos requeridos tanto el de los alumnos, padres de familia y docentes. La información obtenida será administrada por un administrador (docente con puesto de director).

El alumno podrá responder a los diferentes cuestionarios o encuestas que estén disponibles en el sistema, al igual que el padre de familia. En base a la información que arrojen, el tutor podrá generar reportes y hacer seguimientos individuales para tratar de reducir el abandono escolar en este nivel educacional. Por lo general, los alumnos no contestan correctamente los cuestionarios o son mal interpretados, el sistema ayudara en esta problemática tratando de dar los resultados esperados de forma oportuna y así evitar una toma de decisiones desafortunada e inoportuna. Todo esto evitando el famoso papeleo y la pérdida de tiempo que conlleva, actuando de manera rápida ante un caso especial que lleguen a presentar.

Las encuestas se recomienda realizarlas al principio, mediados y al final de cada ciclo escolar. La dirección puede tomar medidas correctivas de acuerdo con la retroalimentación recibida por el tutor, por ejemplo, para mejorar el programa de estudios, el sistema de enseñanza o resolver las dificultades académicas que enfrentan los estudiantes. Dicho lo anterior, al implementar el sistema de encuestas tiene varios factores positivos que pueden ayudar, por ejemplo:

- Son prácticos: puede dirigirse a un público específico y administrarla de diferente manera, es una manera de recopilar grandes cantidades de datos sobre cualquier tema.
- Rápidos resultados: es rápido y fácil de obtener respuestas y ya no tendría que esperar el tutor de verificar las encuestas de cada alumno, sino que estos datos serán en forma de tabla y podrá observar las estadísticas de cada pregunta en menor tiempo posible.

- Abarcar todos los aspectos de un tema: una de las mayores ventajas de los cuestionarios es poder hacer tantas preguntas como quiera, sin embargo, es tener en cuenta que cada cuestionario debe de ser corto porque si son cuestionarios extensos puede ser molesto y cansado.

1.3. Objetivos del Estudio

1.3.1. Objetivo General

Desarrollar un sistema web de encuestas apoyando la labor tutorial, gestionando información de alumnos, docentes y padres reduciendo las deserciones escolares.

1.3.2. Objetivos específicos

- Implementar el módulo de registros para los docentes, padres de familia, alumnos, materias, grupos, puestos de los docentes y domicilios.
- Implementar el módulo de entrevistas frescas para los alumnos y padres de familia.
- Diseñar los reportes en formato pdf de las encuestas contestadas para dar seguimiento individual a cada alumno.
- Implementar el módulo para gestionar la información de los docentes, padres de familia, alumnos, materias, grupos y puestos de los docentes.
- Implementar el módulo de búsqueda y actualización de información para los alumnos y padres de familia.
- Implementar el módulo de inicio de sesión para los diferentes usuarios del sistema (alumnos, docentes y padres de familia).

1.4. Alcances y Limitaciones

1.4.1. Alcances

- Tener un mejor control de datos requeridos para la supervisión escolar.
- Brindar un mejor seguimiento a los tutorados.

- Brindar interfaces amigables y de fácil uso a la hora de visitar el sitio web.
- El sistema administrará la información de los alumnos, docentes, materias, padres de familia, puestos, grupos y domicilios.

1.4.2. Limitaciones

- Falta de recursos y herramientas para el desarrollo del sistema.
- Incremento de información.
- Se requiere servicio de internet para acceder al sistema.
- El personal que opere el sistema deberá de tener conocimientos básicos en computación.
- El sistema web es de idioma monótono (español solamente).
- El bachillerato no cuenta con equipos aptos para implementar el sistema.
- Mantenimiento constante en el servidor.

1.5. Planteamiento de los Supuestos

La realización del sistema pretende apoyar en la problemática en el nivel medio superior en cuanto al porcentaje de deserción escolar en este nivel. Para ello se va a trabajar en la parte tutorial, el cual se encarga de llevar un seguimiento individualizado y personalizado a los alumnos durante su estancia académica. Para facilitar la gestión de información en el sistema se hará registro de los alumnos, docentes, padres de familia, domicilios, etc. También se creará un apartado para las encuestas que son necesarias para la acción tutorial que van dirigidas para los alumnos y padres de familia. Se les asignaran a los padres de familia los parentescos que lo relacionen con sus respectivos hijos, y se les asignara a los docentes los grupos en los cuales trabajaran como tutores. Para un buen manejo de información de las encuestas y entrevistas se generarán reportes en formato PDF que servirán para llenar informes. En el mismo sistema se asignarán las materias que los docentes impartirán durante los ciclos escolares. Todo esto se debe de generar de manera óptima y eficazmente mediante la implementación

de diversos formularios en los cuales proporcionen control y fiabilidad de la realización de estos procesos.

1.6. Hipótesis

Con el uso del sitio web se administrará la información obtenida de docentes, alumnos, padres, materias, grupos, puestos del docente y domicilios y dar seguimiento al programa "yo no abandono", reduciendo el uso del papel y brindar apoyo individual a los alumnos en un 50%.

CAPÍTULO II

REVISIÓN DE LA

LITERATURA

2.1. Diseño e implementación de una plataforma digital para la gestión de tutorías y su impacto en la deserción de estudiantes de nivel superior

Se describe la implementación de una plataforma digital web para realizar la gestión tutorial en instituciones de educación superior, y muestra los resultados preliminares de su aplicación en estudiantes del Instituto Tecnológico de Morelia. La plataforma ha sido desarrollada con el objeto de contribuir a la disminución de los altos índices de reprobación y deserción, así como ampliar y facilitar el campo tutorial del docente, e incrementar la calidad y competitividad de los programas educativos. La implementación de esta herramienta digital de gestión tutorial pretende contribuir a mejorar la relación entre el docente y el estudiante a través de chats personalizados y públicos, de facilitar la obtención de información académica del estudiante para tener un uso eficiente de la misma, y lograr que el estudiante tenga una mejor experiencia educativa que lo formará no sólo como profesional altamente calificado, sino que también tendrá una asertividad para participar de manera activa, organizada y responsable en la dinámica que le establezca la sociedad. (Miriam Zulma Sánchez, Yaneth Vega Flores, Christian Nivardy Marín, J. Omar Hernández Esquivel, 2017)

2.2. Sistema de automatización para el programa “Yo no abandono” del CBTIS No. 61

El abandono escolar es una problemática muy importante en el nivel Medio Superior, a nivel Nacional se implementó el programa “Yo No Abandono” el cual consiste en la aplicación de una serie de cuestionarios para detectar problemas que propician el abandono escolar, sin embargo, esta tarea toma demasiado tiempo y con ello la toma de decisiones para evitar la deserción no es oportuna. En esta investigación se desarrolla e implementar un sistema web para automatizar los procesos de dicho programa, con el fin de mejorar el manejo de la información por parte de los tutores y orientador educativo y así lograr una toma de decisiones de manera oportuna, dicha propuesta se desarrolló usando software libre y se implementó en el CBTIS No. 61 en Huamantla Tlaxcala. (Norma Mendoza, Patrick Hernández, Carolina Rocío, Marva Angélica Mora, María del Rocío y Aydee, julio de 2016)

2.3. Estudio evaluativo del Sistema Institucional de Tutorías para el seguimiento de trayectorias escolares

La Unacar, ubicado en el sureste de México, ha dispuesto una aplicación web llamada Sistema Institucional de Tutorías (SIT) para el acceso a información académica y personal de los estudiantes que tiene asignados el tutor. El SIT permite el registro del plan de acción tutorial, la identificación de tutorados, el registro de sesiones individuales y grupal con los tutorados, las canalizaciones a otros servicios estudiantiles o de apoyo y consultar el historial académico entre otras cosas. Actualmente, la institución no proporciona una plataforma de información que aporte datos suficientes para detectar el rezago, los problemas de rendimiento escolar o alumnos en riesgo de deserción, por lo que el trabajo de tutoría se complica.

El propósito del presente estudio fue evaluar y reunir información sobre la experiencia y necesidades de los profesores que participan en el Programa Institucional de Tutoría (PIT) y que dan seguimiento y apoyo en las trayectorias escolares de estudiantes de nivel superior. La investigación fue de corte cuantitativo con un alcance descriptivo. El instrumento utilizado fue una encuesta en línea que se aplicó a 114 tutores que están adscritos a las diversas dependencias de educación superior de la Universidad Autónoma del Carmen, donde 43.86 % tiene una experiencia superior a los 10 años como tutor. Entre los hallazgos destaca que el Sistema Institucional de Tutorías (SIT) no cuenta con los indicadores suficientes para realizar un seguimiento de trayectoria escolar eficiente y que 42.11 % de los tutores realiza manualmente un seguimiento individual de acuerdo con su experiencia y necesidades de información. A partir de este estudio se identificaron los indicadores de trayectoria escolar que requieren los tutores para dar un seguimiento exitoso. (Fernando E, Ricardo A y Ana Canepa, 2020)

2.4. La tutoría a través de una plataforma web

Es un reporte parcial de investigación sobre “La Tutoría a través de una Plataforma Web” en la Escuela Normal “Miguel F. Martínez” Centenaria y Benemérita. El objetivo es implementar una plataforma web diseñada especialmente para sistematizar el proceso de tutoría. Es una investigación cualitativa-sociocrítica, estudia el diseño y la

incorporación de un software, también es una investigación aplicada que busca la solución de un problema que aqueja a la institución; por lo cual el Cuerpo Académico Lux, Pax, Vis: Modelos y procesos en la formación docente, se dio a la tarea de indagar al respecto. Asimismo, es una investigación descriptiva porque pretende puntualizar el cómo se desarrolla el programa. Los procedimientos utilizados son la entrevista en profundidad, la observación participante y una escala tipo Likert. Los temas principales son conceptualización, funciones, modelos de tutorías y las plataformas web.

Se diseñó e implementó una plataforma web llamada SIDI-Tutorías, encontrándose como bondades las mencionadas por los docentes que participaron en esta fase, la facilidad de acceso a la misma, su practicidad, que permite ir calendarizando las reuniones; que orientan las entrevistas a los tutorados y facilitan la conducción de las mismas. (Flor del Carmen Montes Rodríguez, Guadalupe Minerva Castillo, Sandra Nelly Martínez Cantú, 2017)

2.5. Sistema web: Herramienta para la gestión del programa de tutorías

El proyecto se origina por los problemas que enfrenta el Instituto Tecnológico de Tehuacán debido a los altos índices de reprobación, deserción escolar, la forma de proceder para dar solución a estos problemas y la falta de gestión del programa de tutorías que se ejerce. El desarrollo de un sistema Web para tutorías resulta de gran utilidad, ya que facilitará la gestión y seguimiento de alumnos para detectar oportunamente factores que puedan contribuir a un bajo desempeño escolar, la detección oportuna de estos problemas permitirá canalizar al alumno a las instancias correspondientes y recibir el apoyo pertinente al disponer de una herramienta que proporciona un canal de comunicación abierto. Sin embargo, es necesario disponer de un sistema Web que proporcione beneficios como, reducción de errores provocados por las personas, velocidad en el procesamiento de datos, posibilidad de realizar tediosos análisis sobre los mismos, reducción de espacio físico destinado a su almacenamiento, agilidad al momento de buscar algún dato en particular, con el sistema web el tutor será capaz de orientar, asesorar, diagnosticar y acompañar al estudiante en su proceso de enseñanza- aprendizaje, incidiendo de esta forma en el cumplimiento de los objetivos del programa institucional de tutorías . (Felipe Rodríguez Ramírez, Irene García Ortega, Saira Antonieta Vásquez Gamboa, 2016)

CAPÍTULO III

MARCO TEÓRICO

3.1. Metodología

La metodología hace referencia al conjunto de procedimientos basados en principios lógicos, utilizados para alcanzar una gama de objetivos que rigen en una investigación científica o en una exposición doctrinal. Existen dos tipos de metodologías: Metodología Ágil y Metodología Clásica. Entre las metodologías clásicas encontramos a la metodología en cascada, incremental, y evolutiva. Del lado de las metodologías Ágiles encontraremos la metodología SCRUM, XP, Lean y Kanban. 6

3.1.1. Metodología SCRUM

La metodología SCRUM es un marco de trabajo que se utiliza dentro de equipos que manejan proyectos complejos. Es decir, se trata de una metodología de trabajo ágil que tiene como finalidad la entrega de valor en períodos cortos de tiempo y para ello se basa en tres pilares: la transparencia, inspección y adaptación. (EquiposyTalento, 2018)

A continuación, se explican los grandes rasgos de la metodología a ocupar para el proyecto:

- **Requerimientos del producto o Backlog:** Entre las ventajas de contar con un product backlog destaca la eficacia, ya que hace visible para todo el equipo el trabajo a realizar de forma priorizada, en otras palabras, aporta una visión panorámica de la importancia y del valor desde el punto de vista de negocio de cada requisito o funcionalidad. Se trata de una herramienta abierta y en constante cambio, que solo puede ser actualizada por el Product Owner, que será el encargado de completar y mantener siempre completo el backlog. (Irene Cañete, 2018)
- **Requerimientos de la iteración o Sprint Backlog:** Cada iteración se inicia con una reunión de planificación. En ella, el propietario del producto tiene una conversación con el equipo y procede al establecimiento de prioridades. Determinado el orden idóneo de actuación, es el momento de materializar lo conceptual y convertir cada propuesta en una tarea. (José Miguel Granda, 2019)
- **SCRUM Daily:** El objetivo de esta reunión es facilitar la transferencia de información y la colaboración entre los miembros del equipo para aumentar

su productividad, al poner de manifiesto puntos en que se pueden ayudar unos a otros. (Xavier Albaladejo, s.f)

- Reunión y Revisión por etapas: Completada cada fase, el equipo presenta su trabajo al propietario del producto. Éste decide si el resultado queda aceptado o es desechado. Todas las iniciativas incompletas deben ser rechazadas. (Anna Pérez, 2014)
- Incremento del producto: En este encuentro se valoran los puntos fuertes y las debilidades. Se habla de lo que salió bien, lo que no se llevó a cabo y se extraen conclusiones acerca de lo que se puede mejorar en la siguiente etapa, uno de los puntos fuertes del método Scrum. Miembros del equipo, propietario del producto y Scrum Master están presentes en este momento, que permite que todo el equipo se concentre en su desempeño general y se identifiquen de forma efectiva estrategias para optimizar los resultados. El papel del Scrum Master es crucial, ya que sólo a él le corresponde observar impedimentos comunes y trabajar para resolverlos. (Anna Pérez, 2014)

3.2. Laravel

Laravel es un framework de código abierto para el desarrollo de aplicaciones web en PHP 5 que posee una sintaxis simple, expresiva y elegante. Fue creado en 2011 por Taylor Otwell, inspirándose en Ruby on Rails y Symfony, de los cuales ha adoptado sus principales ventajas. Laravel facilita el desarrollo simplificando el trabajo con tareas comunes como la autenticación, el enrutamiento, gestión sesiones, el almacenamiento en caché, etc. (Antonio Javier Gallego, 2020)



Figura 3.1 Logotipo de Laravel

Fuente: <https://laravel.com>

3.2.1. Modelo Vista Controlador (MVC)

El modelo–vista–controlador (MVC) es un patrón de arquitectura de software que separa los datos y la lógica de negocio de una aplicación de la interfaz de usuario y el módulo

encargado de gestionar los eventos y las comunicaciones. Este patrón de arquitectura de software se basa en las ideas de reutilización de código y la separación de conceptos, características que buscan facilitar la tarea de desarrollo de aplicaciones y su posterior mantenimiento. De manera general, los componentes del MVC se definen de la siguiente manera: (Antonio Javier Gallego, 2020)

- **Modelo:** Es la representación de la información con la cual el sistema opera, gestiona todos los accesos a dicha información, tantas consultas como actualizaciones. Las peticiones de acceso o manipulación de información llegan al modelo a través del controlador.
- **Controlador:** Responde a eventos (usualmente acciones del usuario) e invoca peticiones al 'modelo' cuando se hace alguna solicitud de información (editar un documento o un registro en una base de datos). El controlador hace de intermediario entre la vista y el modelo.
- **Vista:** El usuario podrá interactuar con la vista y realizar otras peticiones que se enviarán al controlador. (Antonio Javier Gallego, 2020)

3.2.2. Estructura de un proyecto

A continuación, se describirá las carpetas importantes para el desarrollo del proyecto.

- **App:** En esta carpeta los archivos que son importantes son: los modelos, los controladores y el middleware. Los modelos son las tablas donde podemos interactuar con los datos de las tablas, el controlador es donde se va a programar el funcionamiento del sistema, por ejemplo: registrar, editar, eliminar, consultar, etc. El middleware es parte de la seguridad, aquí toma mucha importancia en la autenticación debido a que nos ayuda a colocar que ruta debe tener para cada usuario.
- **Database:** En esta carpeta se encuentra todo acerca de creación de las tablas. La carpeta cuenta con tres subcarpetas:

- **Factories:** En esta carpeta sirve para crear información falsa para las tablas con el comando: `php artisan make:factory nombre_archivo_factory`.
- **Migrations:** En esta carpeta se colocan las tablas creadas con el comando: `php artisan make:migration create_nombre_tabla_table`.
- **Seeds:** Aquí se coloca cuanta información se va a crear para las tablas, utilizando los archivos factories, existen dos tipos de creación de datos: `make` y `create`. La diferencia de estos dos es que el `make` crea el dato, pero no lo guarda en la tabla, sin embargo, `create` crea el dato y lo guarda en la tabla.
- **Public:** En esta carpeta se encuentra los css, js e imágenes que se vayan a utilizar.
- **Resources:** En esta carpeta cuenta con una subcarpeta que se llama `view`, dicha carpeta contiene las vistas que verán los usuarios a la hora de visitar el sitio web.
- **Routes:** En esta carpeta se encuentra una subcarpeta llamada `web`. Aquí se definen las rutas URL que se van a utilizar en el proyecto y para cada uno de los usuarios.
- **Vendor:** En esta carpeta se alojan todas las librerías y dependencias que conforman el framework de Laravel. Esta carpeta no se modifica.
- **.env:** Almacena los valores de configuración que son propios de la máquina o instalación actual.

3.3. Composer

Composer es un manejador de paquetes para PHP que proporciona un estándar para administrar, descargar e instalar dependencias y librerías. Similar a NPM en Node.js y Bundler en Ruby. (Yair, 2019)



Figura 3.2 Logotipo

Fuente: <https://getcomposer.org>

3.4. DomPDF

La librería DomPDF es una alternativa sencilla para la construcción de PDF en PHP. Ofrece la posibilidad de crear los documentos PDF a partir de código HTML, que puede residir en un archivo, una cadena de texto, etc. A partir de ese código HTML crea un documento PDF que se puede descargar, almacenar en el servidor o cualquier otra operación. (Miguel Ángel Álvarez, 2020)

3.5. PHP

El PHP es un lenguaje de programación de código abierto, se ejecuta del lado del servidor y se utiliza para el desarrollo de aplicaciones y páginas web. Favoreciendo así, la conexión entre los servidores y la interfaz del usuario. (Rocío González, 2021)



Figura 3.3 Logotipo

Fuente: <https://www.php.net/manual/es/intro-whatcando>.

3.6. Wampserver

Wampserver es un entorno de desarrollo web para Windows con el que se puede crear aplicaciones web con Apache, PHP y bases de datos MySQL database. También incluye PHPMyAdmin y SQLLiteManager para manejar tus bases de datos. (Josseph Muñoz, 2017)



Figura 3.4 Logotipo

Fuente: <http://www.wampserver.com/>

3.7. DigitalOcean

DigitalOcean es un proveedor de computación en la nube que ofrece una infraestructura como servicio (IaaS), enfoca solo en las necesidades de los desarrolladores, a diferencia de AWS de Amazon con su enfoque de todo para todas las personas. DigitalOcean se concentra en tres puntos de venta principales para diferenciarse: precios, máquinas virtuales de alto rendimiento y simplicidad. Su modelo de negocio se concentra en brindar

a los desarrolladores una forma de iniciar rápidamente instancias de Linux asequibles llamadas droplets. (Hamid Jabbari, 2017)



Figura 3.5 Logotipo

Fuente: <http://www.digitalocean.com>

3.8. Mailgun

Mailgun es un servicio de correo electrónico de terceros que ofrece hasta 10,000 mensajes de correo electrónico gratuitos por mes. Ofrece una API programática, la retención de registros, la personalización del correo electrónico, estadísticas, validación de correo electrónico y otras opciones. (Google Cloud, 2020)



Figura 3.6 Logotipo

Fuente: <http://www.mailgun.com>

3.9. HTML 5

HTML5 (Hyper Text Markup Language) es un lenguaje markup usado para estructurar y presentar el contenido para la web. (Juan Diego Pérez, 2019)



Figura 3.7 Logotipo

Fuente: <https://www.w3.org/html/logo/>

3.10. SQL Power Architect

SQL Power Architect es una herramienta de modelado. Compara modelos de datos y estructuras de bases de datos e identifica las discrepancias, Drag-and-drop de las tablas origen y las columnas en el área de trabajo. (Luigy, 2016)



Figura 3.8 Logotipo SQL Power

Fuente: <https://macdownload.informer.com/sql-power-architect/>

3.11. Bootstrap

Bootstrap es un kit de herramientas de código abierto para desarrollos web responsive con HTML, CSS y JavaScript. Se puede dar forma a un sitio web a través del uso de sus librerías CSS y JavaScript. Incluye diferentes componentes: ventanas modales, menús, cuadros, botones, formularios. Todos los elementos que se necesita para maquetar una página. (Álvaro Fontela, 2020)



Figura 3.9 Logotipo Bootstrap

Fuente: <https://codingpotions.com/desarrollo-web-bootstrap>

3.12. Balsamiq

Es una pequeña herramienta gráfica para esbozar las interfaces de usuario, para los sitios web y escritorio/aplicaciones móviles web. Se centran en la fase de creación, proporcionando una herramienta de limpieza que permite obtener en el flujo y enfocarte en la estructura, con la cual se crean pantallas de baja fidelidad, pudiendo modificar, reordenar fácilmente los elementos, obteniendo una vista previa de las interfaces. (MartechForum, s.f)



Figura 3.10 Logotipo

Fuente: <https://www.capterra.mx/software/145723/balsamiq-mockups>

3.13. Sublimet Text 3

Sublime text es un editor de texto y código multiplataforma muy ligero y potente con versiones tanto para Mac, Linux y Windows, el mismo fue escrito en lenguaje C++ y los plugin que se instalan están hechos en lenguaje Python. Este editor no es libre, hay que adquirir una licencia, pero se puede descargar de forma gratuita para evaluarlo y funcionará sin ningún problema. (Alberth Bompart, 2018)



Figura 3.11 Logotipo Sublime Text

Fuente: <https://aficionweb.wordpress.com/2018/01/20/sublime-text-full/>

3.14. GitHub

GitHub es un portal creado para alojar el código de las aplicaciones de cualquier desarrollador, y que fue comprada por Microsoft en junio del 2018. Un sistema de gestión de versiones es ese con el que los desarrolladores pueden administrar su proyecto, ordenando el código de cada una de las nuevas versiones que sacan de sus aplicaciones para evitar confusiones. Así, al tener copias de cada una de las versiones de su aplicación, no se perderán los estados anteriores cuando se va a actualizar. (Yúbal Fernández, 2019)



Figura 3.12 Logotipo

Fuente: <https://www.xataka.com/basics/que-github-que-que-le-ofrece-a-desarrolladores>

3.15. Programa “Yo no abandono”

El Movimiento contra el Abandono Escolar es una estrategia integral de carácter nacional que involucra la participación conjunta y coordinada de autoridades educativas, federales y estatales, directivos de planteles, docentes, padres de familia, estudiantes y sociedad en general, para lograr mayores índices de acceso, permanencia y conclusión exitosa de los estudios de nivel medio superior. (Secretaría de Educación Media Superior, 2017)

3.16. Cmder

Es un emulador de consola portátil y gratuito para Windows. Por emulador de consola queremos decir, que puedo envolver CMD y proporcionar muchas más características y experiencia intuitiva. La variante completa viene incluida con Git para Windows. Lo que significa que puede usar fácilmente los comandos Git y algunos comandos Unix en su ordenador también. (PcHardwarePro, s.f)



Figura 3.13 Logotipo

Fuente: <https://alternativas-a.com/cmdr/>

CAPÍTULO IV

METODOLOGÍA

APLICADA

4.1. Descripción de la Empresa u Organización

El proyecto se realizará en el Departamento de Investigación del Tecnológico Nacional de México (TecNM) Campus San Martín Texmelucan ubicado en Camino a Barranca de Pesos S/N San Lucas Atoyatenco San Martín Texmelucan, Puebla, el cual se dedica a ofertar carreras de educación superior. Cuenta con las siguientes carreras: Ingeniería en Sistemas Computacionales, Ingeniería Electromecánica, Ingeniería Industrial, Ingeniería Ambiental, Ingeniería en Tecnologías de la Información y Comunicación, Ingeniería en Gestión Empresarial y Contador Público.

Misión: Formar profesionistas con sentido innovador, cuyo conocimiento se base en la investigación científica y tecnológica, que propongan soluciones creativas a problemas afines a su región, para fortalecer y mejorar la calidad de vida, vinculados al sector productivo y comprometidos con la preservación del medio ambiente.

Visión: Ser la principal opción como Institución de Educación Superior de la región, que ofrezca servicios educativos de calidad, cuyos programas de estudio incorporen tecnología actualizada, que se constituya en el detonador del progreso económico - social, consolidando el desarrollo con creatividad, competitividad y sentido humanístico.

Política: EL ITSSMT proporciona servicios educativos de calidad, a través de un SGI basado en las normas ISO 9001 e ISO 14001, que incluye procesos orientados al cumplimiento de los requisitos de los/as estudiantes, la identificación y tratamiento de los riesgos, control de impactos ambientales y prevención de la contaminación, mediante el modelo educativo basado en formación y desarrollo de competencias profesionales, además de verificar el cumplimiento de los objetivos, requisitos legales, reglamentarios aplicables y la búsqueda de la mejora continua

4.2. Organigrama y Lay Out

En esta sección se presenta el organigrama y Lay Out de la institución.

4.2.1. Organigrama

En la figura 4.1 se representa de manera gráfica la estructura del TECNM Campus San Martín Texmelucan en la cual se muestran las relaciones entre sus diferentes partes mediante el uso de un organigrama.

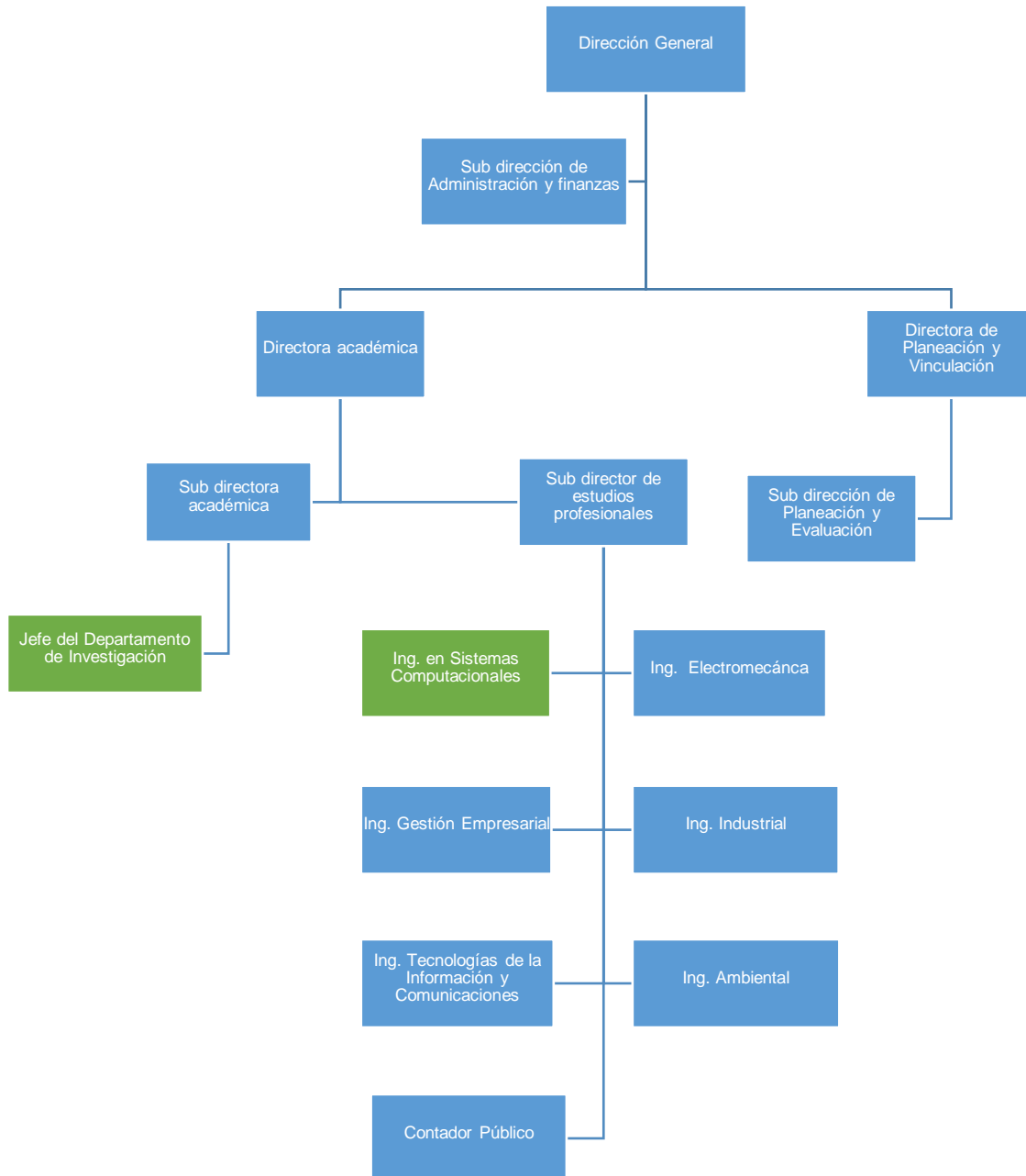


Figura 4.1 Organigrama del TECNM Campus San Martín Texmelucan

Fuente: Elaboración propia

4.2.2. Lay Out

A continuación, en las figuras 4.2 y 4.3 se muestra los esquemas del edificio de la institución resaltando de color rojo el departamento donde se va a realizar el proyecto.

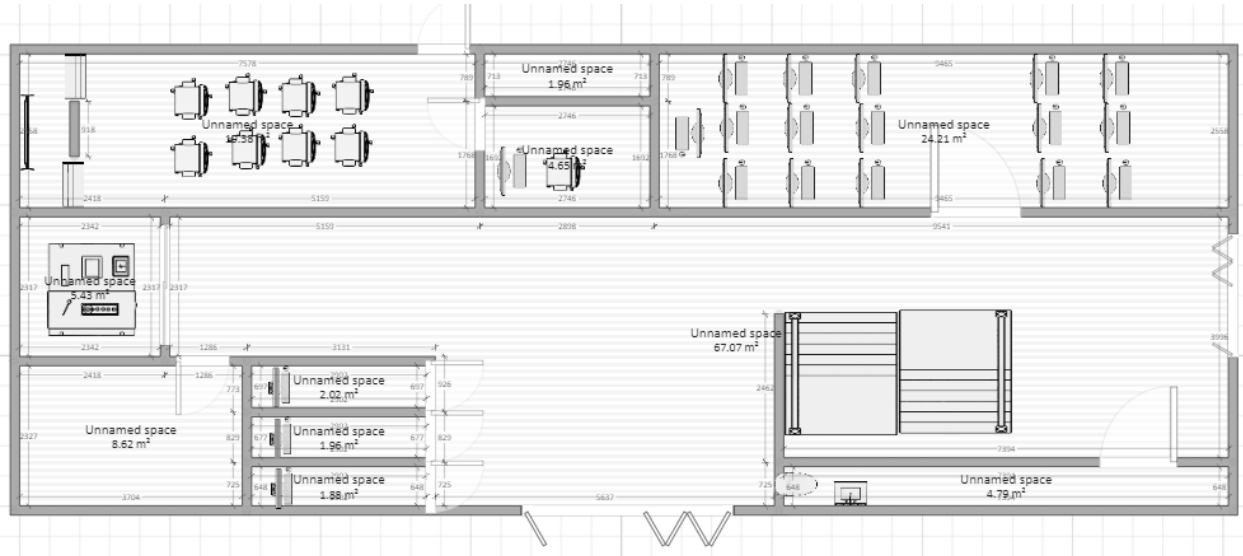


Figura 4.2 Esquema del edificio C planta baja

Fuente: Elaboración propia



Figura 4.3 Esquema del edificio C planta alta

Fuente: Elaboración propia

4.3. Descripción del puesto o área de trabajo en la que participó el/la estudiante.

El área implicada para el desarrollo de este proyecto es el Depto. de Investigación, la institución ha fomentado el trabajo de investigación a través del desarrollo de proyectos que los y las estudiantes y el personal docente generan para dar atención a necesidades regionales. Los principales avances radican en soluciones donde han presentado proyectos integrales, contando algunos de ellos, con el registro de derecho de autor y patente, como son:

- Plataforma web para la sistematización de cultivo aeropónico (módulos: reportes y graficación).
- Proyecto de reactivación económica PROMOITSSMT (Back-end y frontend).
- Sistema administrativo comercial para la organización SOSAPATEX.
- Sistema para estadísticas de tutorías en los bachilleratos oficiales de la región de San Martín Texmelucan.
- Plataforma web para la sistematización de cultivo aeropónico (módulos: autenticación de usuario, registro de cultivo y proceso de nebulización).

Análisis del proyecto

Durante el análisis que se realizó al sistema de encuestas para tutorías del programa “Yo no abandono” se identificaron los tipos de usuario necesarios para la elaboración del sistema, tales como: Alumno, Padre de familia, Docente, y Administrador.

- **Alumno:** Al usuario alumno se le dará los privilegios específicos, ya que no toda la información que se maneja en el sistema no es para uso público.
- **Padre de familia:** Al igual que al usuario alumno, se le asignaran privilegios específicos, este usuario pareciera que no es necesario, pero para la acción tutorial se requiere para el apoyo en la orientación educativa.
- **Docente:** Un docente puede tener el rol de **tutor** para hacer registros en áreas específicas del sistema, este usuario genera informes y reportes de tutorías y seguimiento de los alumnos de bajo rendimiento o atenciones especiales.

- **Administrador:** este rol se le asignara al encargado de la casa de estudios (el director), este podrá gestionar de manera rápida y sencilla la información almacenada en el sistema, por tener el rol más privilegiado, este registrara la información de los demás usuarios para que puedan ingresar de manera correcta.

4.4. Fase de análisis

4.4.1. Requerimientos del hardware

En la tabla 4.1 se observa las características del equipo de cómputo utilizado para el desarrollo del proyecto.

Tipo de Hardware	Nombre	Descripción	Sistema Operativo	Justificación	Provee
Laptop	HP	8 GB RAM 500 GB Disco Duro	Windows 10 Home Single Language	Equipo de desarrollo, para el entorno de programación.	Practicante

Tabla 4.1 Requerimientos del Hardware

Fuente: Elaboración propia

4.4.2. Requerimientos de Software

En la tabla 4.2 se muestra el software utilizado para el desarrollo del sistema.

Nombre	Versión	Sistema Operativo	Equipo	Justificación
Balsamiq Mockups	3.5.16	Windows 10	HP	Pantallas de Baja Fidelidad.
SQL Power Architect	1.0.8	Windows 10	HP	Diseño de Base de Datos.
WampServer	3.2.3	Windows 10	HP	Base de Datos.
SublimeText	3.2	Windows 10	HP	IDE de Desarrollo.
Office	365 ProPlus 2019 365 ProPlus 2019	Windows 10	HP	Documentación.
		Windows 10		Documentación.
Microsoft Edge	90.0.818.56	Windows 10	HP	Navegador.
Laravel	5.5.*	Windows 10	HP	Framework

Tabla 4.2 Requerimientos del software

Fuente: Elaboración propia

4.4.3. Requerimientos Funcionales

En la tabla 4.3 muestra las declaraciones de los servicios solicitados del sistema.

Nº REQ	Descripción	Prioridad	Modulo
RF 01	Diseño de Interfaz	Alta	Diseño
RF 02	Diseño de Base de Datos	Alta	Diseño
RF 03	Recuperación de Contraseña	Alta	Usuario
RF 04	Encuestas	Alta	Programación
RF 05	Registro de alumnos	Alta	Registro
RF 06	Registro de docentes	Alta	Registro
RF 07	Formulario para la entrevista fresca	Alta	Programación
RF 08	Registro de padres de familia	Alta	Registro
RF 09	Programación de Login	Alta	Programación
RF 10	Programación de encuestas	Alta	Programación
RF 11	Generación de informes	Alta	Programación
RF 12	Asignación de puestos	Alta	Programación
RF 13	Motor de búsqueda	Alta	Programación

Tabla 4.3 Requerimientos Funcionales

Fuente: Elaboración propia

4.4.4. Requerimientos No Funcionales

En la tabla 4.4 muestra los requerimientos que no afecta en el sistema, sin embargo, están presentes.

Nº REQ	Descripción
Paleta de colores	Colores solicitados
Tipo de letra	Arial número necesario según los módulos
Derechos de autor	Personas que realizaron el proyecto
Formato para datos obligatorios	Texto color rojo
Distribución de la pantalla	Lugar de cada botón/información
Estándares de error	Letreros de error por cada dato mal ingresado
Tiempos de respuesta	Tiempo que tarde en responder el sistema
Lenguaje	Español, inglés.

Tabla 4.4 Requerimientos no funcionales

Fuente: Elaboración propia

4.5. Modelo de la base de datos

En las figuras 4.4, 4.5 y 4.6 muestra la estructura lógica de la base de datos incluyendo las relaciones y limitaciones que determinan el cómo se almacenan y acceden los datos en este sistema. Las tablas utilizadas están indicadas de color rojo. Utilizando el software SQL Power Architect.

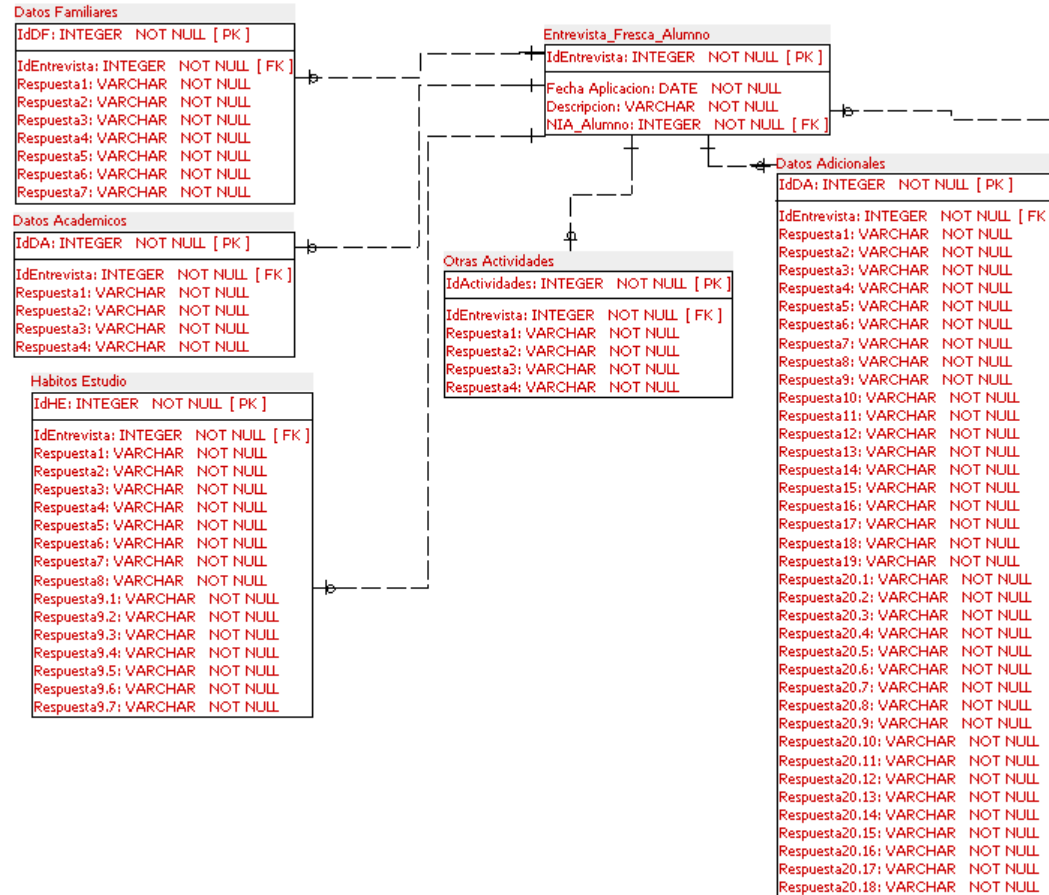


Figura 4.4 Modelo de la base de datos parte 1

Fuente: Elaboración propia

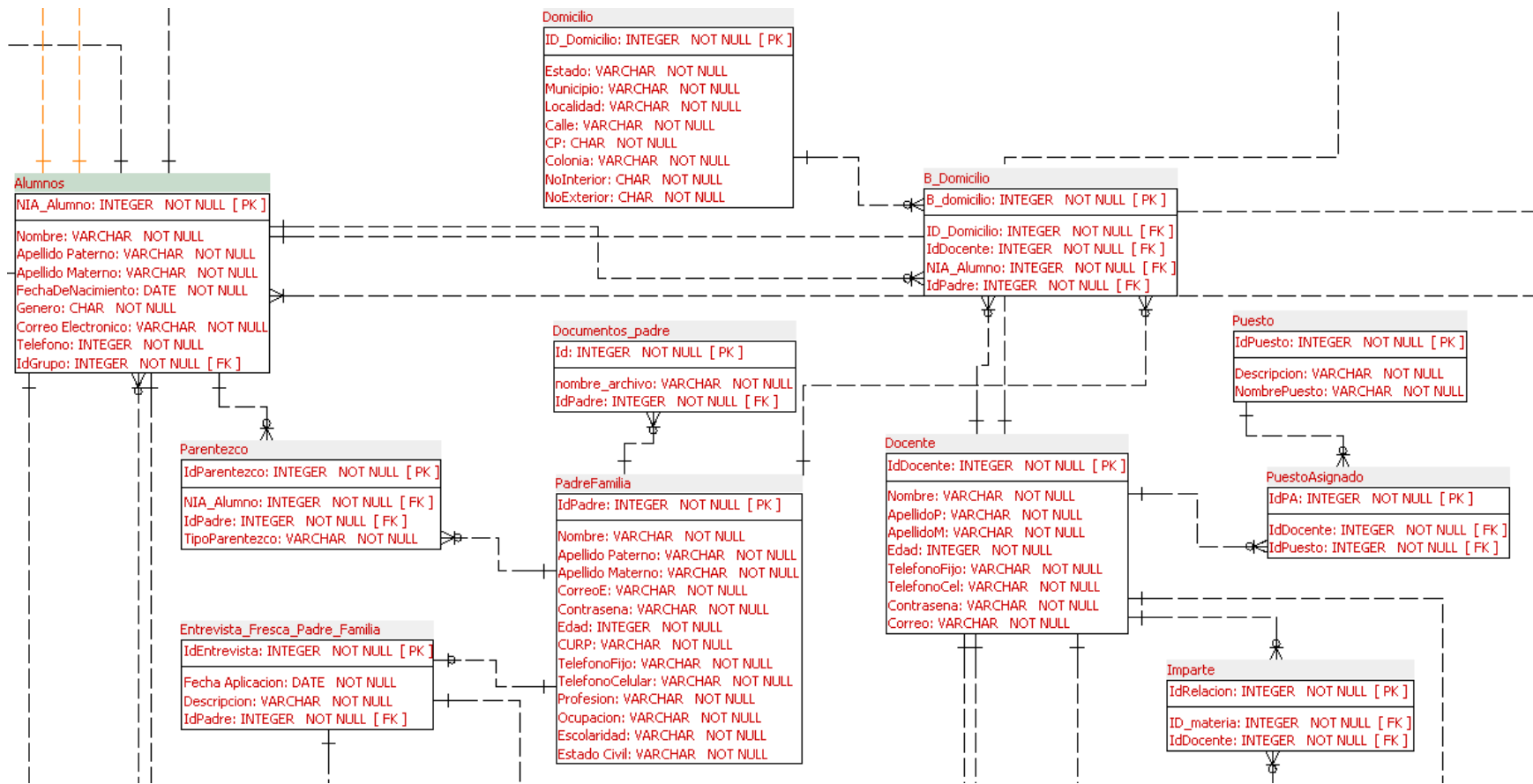


Figura 4.5 Modelo de la base de datos parte 2

Fuente: Elaboración propia

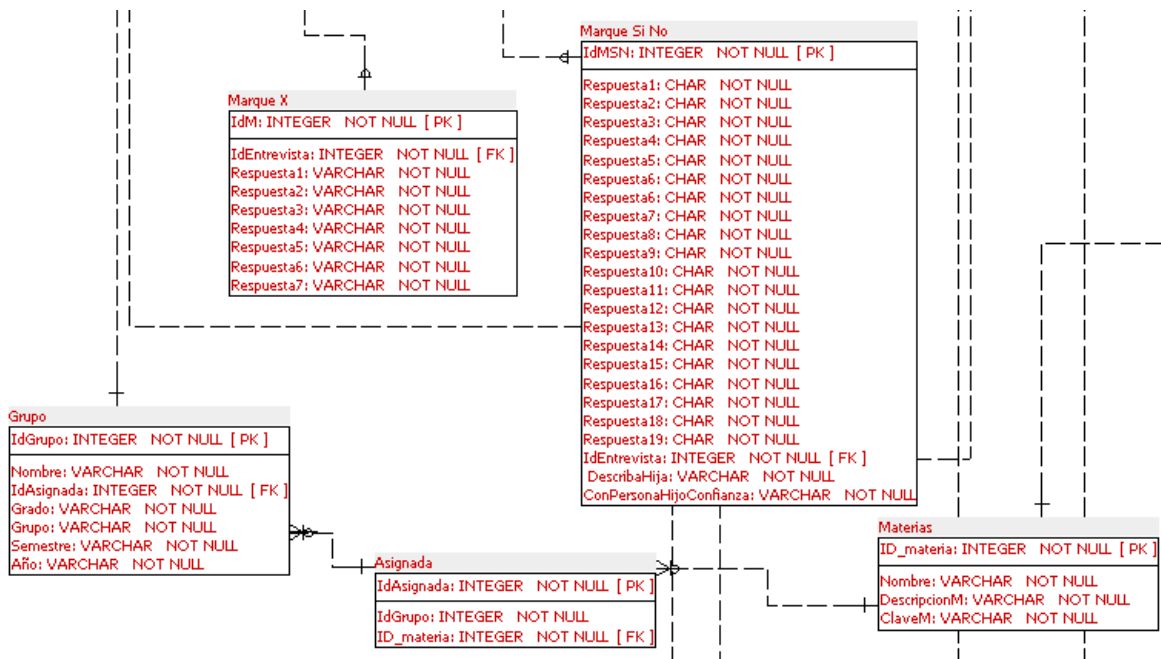


Figura 4.6 Modelo de la base de datos parte 3

Fuente: Elaboración propia

4.5.1. Diccionario de datos

Al terminar el diagrama de clases se creó el diccionario de datos de cada una de las tablas de la base de datos.

ALUMNOS							
Atributos	Tipo de dato	Tamaño	PK	FK	UNI	Null	Extra
nia	int	11	x			No	
name	varchar	50				No	
edad	tinyint	4				No	
apellidoP	varchar	50				No	
apellidoM	varchar	50				No	
fechaN	date					No	
genero	char	2				No	
phone	varchar	14				No	
email	varchar	30			x	No	
password	varchar	60				No	
grupo_id	Int unsigned	10		x		No	

ENTREVISTA FRESCA ALUMNO							
Atributos	Tipo de dato	Tamaño	PK	FK	UNI	Null	Extra
id	int unsigned	10	x			No	auto_increment
fecha_aplicacion	date					Si	
descripcion	varchar	30				No	
alumno_id	int unsigned	10		x		No	
DATOS FAMILIARES							
Atributos	Tipo de dato	Tamaño	PK	FK	UNI	Null	Extra
id	int unsigned	10	x			No	auto_increment
entrevista_id	int unsigned	10		x		No	
respuesta1	varchar	191				No	
r1	varchar	191				No	
respuesta2	varchar	191				No	
respuesta3	varchar	191				No	
r3	varchar	191				No	
respuesta4	varchar	191				No	
respuesta5	varchar	191				No	
respuesta6	varchar	191				No	
respuesta7	varchar	191				No	
DATOS ACADEMICOS							
Atributos	Tipo de dato	Tamaño	PK	FK	UNI	Null	Extra
id	int unsigned	10	x			No	auto_increment
Entrevista_id	int unsigned	10		x		No	
Respuesta1	varchar	191				No	
Respuesta2	varchar	191				No	
R2	varchar	191				No	
Respuesta3	varchar	191				No	
Respuesta4	varchar	191				No	
R4	varchar	191				No	
HABITOS ESTUDIO							

Atributos	Tipo de dato	Tamaño	PK	FK	UNI	Null	Extra
id	int unsigned	10	x			No	auto_increment
entrevista_id	int unsigned	10		x		No	
respuesta1	varchar	191				No	
respuesta2	varchar	191				No	
respuesta3	varchar	191				No	
respuesta4	varchar	191				No	
respuesta5	varchar	191				No	
r5	varchar	191				Si	
respuesta6	varchar	191				No	
r6	varchar	191				Si	
respuesta7	varchar	191				No	
r7	varchar	191				Si	
respuesta8	varchar	191				No	
r8	varchar	191				Si	
r9_1	varchar	191				No	
r9_2	varchar	191				No	
r9_3	varchar	191				No	
r9_4	varchar	191				No	
r9_5	varchar	191				No	
r9_6	varchar	191				No	
r9_7	varchar	191				No	

DATOS ADICIONALES

Atributos	Tipo de dato	Tamaño	PK	FK	UNI	Null	Extra
id	int unsigned	10	x			No	auto_increment
entrevista_id	int unsigned	10		x		No	
respuesta1	varchar	191				No	
r1	varchar	191				Si	
respuesta2	varchar	191				No	
r2	varchar	191				Si	

respuesta3	varchar	191				No	
r3	varchar	191				Si	
r3_2	varchar	191				Si	
respuesta4	varchar	191				No	
r4	varchar	191				Si	
respuesta5	varchar	191				No	
r5	varchar	191				Si	
respuesta6	varchar	191				No	
respuesta7	varchar	191				No	
respuesta8	varchar	191				No	
respuesta9	varchar	191				No	
respuesta10	varchar	191				No	
respuesta11	varchar	191				No	
respuesta12	varchar	191				No	
respuesta13	varchar	191				No	
respuesta14	varchar	191				No	
r14	varchar	191				Si	
respuesta15	varchar	191				No	
r15	varchar	191				No	
respuesta16	varchar	191				No	
r16	varchar	191				Si	
respuesta17	varchar	191				No	
r17	varchar	191				Si	
r17_2	varchar	191				Si	
respuesta18	varchar	191				No	
respuesta19	varchar	191				No	
r20_1	varchar	191				No	
r20_2	varchar	191				No	
r20_3	varchar	191				No	
r20_4	varchar	191				No	

r20_5	varchar	191				No	
r20_6	varchar	191				No	
r20_7	varchar	191				No	
r20_8	varchar	191				No	
r20_9	varchar	191				No	
r20_10	varchar	191				No	
r20_11	varchar	191				No	
r20_12	varchar	191				No	
r20_13	varchar	191				No	
r20_14	varchar	191				No	
r20_15	varchar	191				No	
r20_16	varchar	191				No	
r20_17	varchar	191				No	
r20_18	varchar	191				No	

OTRAS ACTIVIDADES

Atributos	Tipo de dato	Tamaño	PK	FK	UNI	Null	Extra
id	int unsigned	10	x			No	auto_increment
entrevista_id	int unsigned	10		x		No	
respuesta1	varchar	191				No	
respuesta2	varchar	191				No	
respuesta3	varchar	191				No	
respuesta4	varchar	191				No	

PARENTEZCOS

Atributos	Tipo de dato	Tamaño	PK	FK	UNI	Null	Extra
id	int unsigned	10	x			No	auto_increment
parentezco	varchar	191				No	
alumno_id	int unsigned	10		x		No	
padre_id	int unsigned	10		x		No	

PADRE DE FAMILIA

Atributos	Tipo de dato	Tamaño	PK	FK	UNI	Null	Extra
-----------	--------------	--------	----	----	-----	------	-------

id	int unsigned	10	x			No	auto_increment
name	varchar	50				No	
apellidoP	varchar	20				No	
apellidoM	varchar	20				No	
email	varchar	30			x	No	
password	varchar	60				No	
edad	tinyint	3				No	
curp	varchar	20			x	No	
telefono_fijo	varchar	20				No	
telefono_cel	varchar	20				No	
profesion	varchar	30				No	
ocupacion	varchar	50				No	
escolaridad	varchar	20				No	
estado_civil	varchar	10				No	

DOCUMENTOS PADRE

Atributos	Tipo de dato	Tamaño	PK	FK	UNI	Null	Extra
id	int unsigned	10	x			No	auto_increment
nombre_archivo	varchar	191				No	
padre_id	int unsigned	10		x		No	

ENTREVISTA FRESCA PADRE

Atributos	Tipo de dato	Tamaño	PK	FK	UNI	Null	Extra
id	int unsigned	10	x			No	auto_increment
fecha_aplicacion	date					Si	
descripcion	varchar	30				No	
alumno_id	int	11		x		No	
padre_id	int unsigned	10					

MARCA X

Atributos	Tipo de dato	Tamaño	PK	FK	UNI	Null	Extra
id	int unsigned	10	x				auto_increment
entrevista_id	int unsigned	10		x		No	

r1	varchar	3				No	
r1_2	varchar	5				No	
r1_3	varchar	50				No	
r2	varchar	3				No	
r2_2	varchar	5				No	
r2_3	varchar	50				No	
r3	varchar	3				No	
r3_2	varchar	5				No	
r3_3	varchar	50				No	
r4	varchar	3				No	
r4_2	varchar	5				No	
r4_3	varchar	50				No	
r5	varchar	3				No	
r5_2	varchar	5				No	
r5_3	varchar	50				No	
r6	varchar	3				No	
r6_2	varchar	5				No	
r6_3	varchar	50				No	
r7	varchar	3				No	
r7_2	varchar	5				No	
r7_3	varchar	50				No	
r8	varchar	3				No	
r8_2	varchar	5				No	
r8_3	varchar	50				No	
r9	varchar	3				No	
r9_2	varchar	5				No	
r9_3	varchar	50				No	
r10	varchar	3				No	
r10_2	varchar	5				No	
r10_3	varchar	50				No	

r11	varchar	3				No	
r11_2	varchar	5				No	
r11_3	varchar	50				No	
r12	varchar	3				No	
r12_2	varchar	5				No	
r12_3	varchar	50				No	
r13	varchar	3				No	
r13_2	varchar	5				No	
r13_3	varchar	50				No	
r14	varchar	3				No	
r14_2	varchar	5				No	
r14_3	varchar	50				No	
r15	varchar	3				No	
r15_2	varchar	5				No	
r15_3	varchar	50				No	
r16	varchar	3				No	
r16_2	varchar	5				No	
r16_3	varchar	50				No	
r17	varchar	3				No	
r17_2	varchar	5				No	
r17_3	varchar	50				No	
r18	varchar	3				No	
r18_2	varchar	5				No	
r18_3	varchar	50				No	
r19	varchar	3				No	
r19_2	varchar	5				No	
r19_3	varchar	50				No	
r20	varchar	3				No	
r20_2	varchar	5				No	
r20_3	varchar	50				No	

21	varchar	3				No	
r21_2	varchar	5				No	
r21_3	varchar	50				No	
MACA SI NO							
Atributos	Tipo de dato	Tamaño	PK	FK	UNI	Null	Extra
id	int unsigned	10	x				auto_increment
entrevista_id	int unsigned	10		x		No	
respuesta1	varchar	3				No	
respuesta2	varchar	3				No	
respuesta3	varchar	3				No	
respuesta4	varchar	3				No	
respuesta5	varchar	3				No	
respuesta6	varchar	3				No	
respuesta7	varchar	3				No	
respuesta8	varchar	3				No	
respuesta9	varchar	3				No	
respuesta10	varchar	3				No	
respuesta11	varchar	3				No	
respuesta12	varchar	3				No	
respuesta13	varchar	3				No	
respuesta14	varchar	3				No	
respuesta15	varchar	3				No	
respuesta16	varchar	3				No	
respuesta17	varchar	3				No	
respuesta18	varchar	3				No	
respuesta19	varchar	250				No	
respuesta20	varchar	250				No	
GRUPOS							
Atributos	Tipo de dato	Tamaño	PK	FK	UNI	Null	Extra
id	int unsigned	10	x			No	auto_increment

name	varchar	191				No	
grado	varchar	100				No	
grupo	char	2				No	
semestre	varchar	100				No	
year	year	4				No	
ASIGNADAS							
Atributos	Tipo de dato	Tamaño	PK	FK	UNI	Null	Extra
id	int unsigned	10	x			No	auto_increment
grupo_id	int unsigned	10		x		No	
materia_id	int unsigned	10		x		No	
MATERIAS							
Atributos	Tipo de dato	Tamaño	PK	FK	UNI	Null	Extra
id	int unsigned	10	x			No	auto_increment
name	varchar	255				No	
descripcion	varchar	255				No	
clave	varchar	50					
IMPARTE							
Atributos	Tipo de dato	Tamaño	PK	FK	UNI	Null	Extra
id	int unsigned	10	x			No	auto_increment
docente_id	int unsigned	10		x		No	
materia_id	int unsigned	10		x		No	
DOCENTES							
Atributos	Tipo de dato	Tamaño	PK	FK	UNI	Null	Extra
id	int unsigned	10	x			No	auto_increment
name	varchar	50				No	
apellidoP	varchar	20				No	
apellidoM	varchar	20				No	
edad	tinyint	4				No	
email	varchar	50			x	No	
password	varchar	60				No	

telefono_fijo	varchar	20				No	
telefono_cel	varchar	20				No	
PUESTO ASIGNADOS							
Atributos	Tipo de dato	Tamaño	PK	FK	UNI	Null	Extra
id	int unsigned	10	x			No	auto_increment
docente_id	int unsigned	10		x		No	
puesto_id	int unsigned	10		x		No	
PUESTOS							
Atributos	Tipo de dato	Tamaño	PK	FK	UNI	Null	Extra
id	int unsigned	10	x			No	auto_increment
puesto	varchar	255				No	
descripcion	varchar	255				No	
B DOMICILIO							
Atributos	Tipo de dato	Tamaño	PK	FK	UNI	Null	Extra
id	int unsigned	10	x			No	auto_increment
alumno_id	int unsigned	10		x		Si	
docente_id	int unsigned	10		x		Si	
padre_id	int unsigned	10		x		Si	
domicilio_id	int unsigned	10		x		No	
DOMICILIOS							
Atributos	Tipo de dato	Tamaño	PK	FK	UNI	Null	Extra
id	int unsigned	10	x			No	auto_increment
estado	varchar	50				No	
municipio	varchar	100				No	
localidad	varchar	100				No	
calle	varchar	100				No	
cp	char	6				No	
colonia	varchar	30				No	
no_interior	char	3				No	
no_exterior	char	3				No	

Tabla 4.5 Diccionario de usuarios

Fuente: Elaboración propia

4.6. Planeación del proyecto

En la tabla 4.6 se muestra los requerimientos iniciales que determinan las diferentes tareas y/o actividades que se describe en el Sprint Backlog.

Para identificar cada historia de usuario, se muestran los siguientes campos:

- **ID:** Valor único de cada historia de usuario.
- **Nombre:** Descripción de cada historia.
- **Tiempo:** Tiempo inicial para implementar cada historia, determinado en días.
- **Sprint:** Indicación de cada iteración.

ID	Nombre	Tiempo
1	Pantalla principal	3
2	Autenticación de usuario	7
3	Registro de alumnos	7
4	Registro de padres de familia	5
5	Motor de búsqueda para alumnos y padres de familia	9
6	Asignación de parentescos	5
7	Registro de docentes	5
8	Asignación de puestos	9
9	Registro de domicilios	7
10	Registro de grupos	7
11	Registro de materias	7
12	Apartado de entrevistas	6
13	Generar PDF	8

Tabla 4.6 Lista de tareas
Fuente: Elaboración propia

4.7. Desarrollo del proyecto

A continuación, se va a describir cada una de las iteraciones realizadas.

Historia de usuario	
Numero: 1	Usuario: Docente, alumno y padre de familia
Nombre de la historia: pantalla principal (bienvenida)	
Prioridad en negocio: alta	Riegos en desarrollo: baja
Programador responsable: Brandom Baruch González Cervantes	
Descripción: Al visitar el sitio web, como página principal es mostrar información de la escuela y del programa “Yo no abandono”	

Tabla 4.7 HU - pantalla de principal
Fuente: Elaboración propia

Iteración 1

Hace referencia a la primera historia de usuarios sobre la pantalla principal que verán los usuarios.

Sprint Blacklog iteración 1

En la tabla 4.8 muestra las actividades para la realización de esta iteración

ID	Tarea	Inicio	Finalización	Estado
1	Análisis sobre la propuesta del contenido y colores,	20/08/2020	23/08/2020	Completado
2	Asignación de los componentes en la vista del framework.	20/08/2020	23/08/2020	Completado
3	Pruebas de direccionamiento a otras pantallas.	20/08/2020	23/08/2020	Completado

Tabla 4.8 Tareas específicas Iteración

Fuente: Elaboración propia

Diseño de la iteración 1

A continuación, se mostrará la página de inicio que verán los usuarios al ingresar al sitio web.

Pantallas de alta y baja

La pantalla principal podrá ver la información de la escuela (nombre y descripción), tendrá unos apartados donde podrán conocer más acerca de: descripción del programa “Yo no abandono”, guía para los padres y como detectar un posible abandono escolar como se muestra en las figuras 4.7 y 4.8.

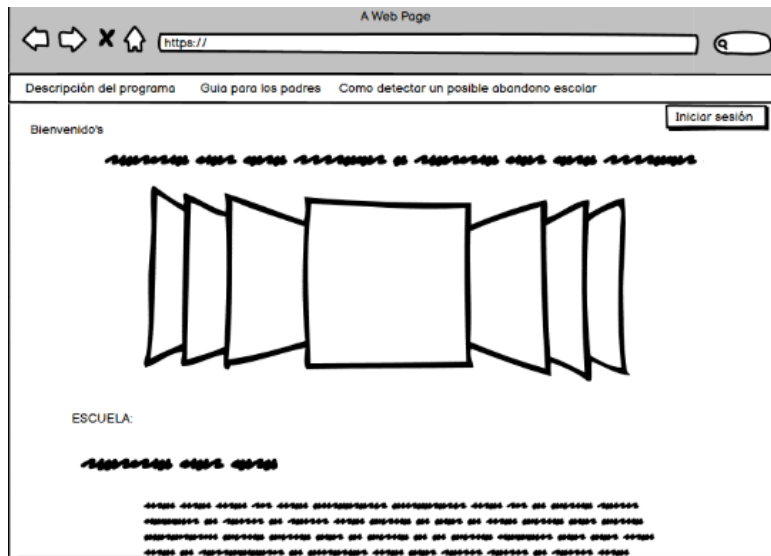


Figura 4.7 Pantalla de inicio (Pantalla de baja)
Fuente: Elaboración propia



Acerca del Bachillerato "Profra. Ignacia Islas"

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

		
<p>Descripción del programa</p> <p><small>Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam.</small></p> <p>SABER MÁS</p>	<p>Guía para padres de familia</p> <p><small>Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam.</small></p> <p>SABER MÁS</p>	<p>Como detectar un posible abandono escolar</p> <p><small>Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam.</small></p> <p>SABER MÁS</p>

Figura 4.8 Pantalla de inicio (Pantalla de alta)
Fuente: Elaboración propia

La tabla 4.9 muestra la estructura de la página principal que verán los usuarios. @extends hereda los estilos css y js de la estructura padre. Ayuda a que sea dinámico y cargue más rápido la página. @sections indica en donde va el código, el primer parámetro es el nombre de la sección donde estará colocado y el segundo parámetro es el contenido.

```

Codificación
<div class="page-header header-filter" data-parallax="true" style=" background-image: url('{{asset('img/identidad-gobierno-federal.jpg')}}') ">
  <div class="container">
    <div class="row">
      <div class="col-md-6 text-center">
        <h1 class="title">Bachillerato "Profra. Ignacia Islas"</h1>
        
      </div>
    </div>
  </div>
</div>
<div class="main main-raised">
  <div class="container">
    <div class="section text-center">
      <div class="row">
        <div class="col-md-9 ml-auto mr-auto">
          <h2 class="title">Acerca del Bachillerato "Profra. Ignacia Islas"</h2>
          <h5 class="description text-justify" style="color: black;">Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmodtempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam,quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodoconsequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.</h5>
        </div>
        <div class="features">
          <div class="row">
            <div class="col-md-4">
              <div class="info">
                <div class="icon icon-info">
                  <i class="material-icons">chat</i>
                </div>
                <h5 class="info-title">Descripción del programa</h5>
                <p style="color: black;" class="text-justify">Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam.</p>
                <button type="button" class="btn btn-info">Saber más</button>
              </div>
            </div>
            <div class="col-md-4">
              <div class="info">
                <div class="icon icon-success">
                  <i class="material-icons">chrome_reader_mode</i>
                </div>
                <h5 class="info-title">Guía para padres de familia</h5>
                <p style="color: black;" class="text-justify">Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam.</p>
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>
</div>

```

Tabla 4.9 Codificación. Iteración 1
Fuente: Elaboración propia

Historia de usuario	
Numero: 2	Usuario: Docente, alumno y padre de familia
Nombre de la historia: Inicio de sesión	
Prioridad en negocio: alta	Riegos en desarrollo: alta
Programador responsable: Brandom Baruch González Cervantes	
Descripción: Cada usuario contara con un Login, el docente y el padre de familia podrán ingresar con el correo electrónico y la contraseña, en caso del alumno debe de ingresar su NIA y contraseña, en caso que no recuerden su contraseña contara con un apartado para reestablecer la contraseña y colocar una nueva contraseña.	

Tabla 4.10 HU - Inicio de sesión
Fuente: Elaboración propia

Iteración 2

Hace referencia a la segunda historia de usuario sobre la autenticación de usuarios donde muestra el formulario para acceder al panel de cada usuario.

Sprint Backlog iteración 2

En la tabla 4.11 se muestra las tareas específicas para la realización de la iteración 2 y el estado en la que se encuentran.

Tarea asignada a: Brandom Baruch González Cervantes				
ID	Tarea	Inicio	Finalización	Estado
1	Asignación de los componentes en la vista del framework.	24/08/2020	27/08/2020	Completado
2	Configuración del archivo. env para la conexión de la base de datos.	27/08/2020	27/08/2020	Completado
4	Creación de tablas y migración en la base de datos	28/08/2020	28/08/2020	Completado
5	Creación de modelos y controladores para programar	29/08/2020	31/08/2020	Completado
6	Pruebas de direccionamiento a otras pantallas.	31/08/2020	31/08/2020	Completado

Tabla 4.11 Tareas específicas iteración
Fuente: Elaboración propia

Diseño de la iteración 2

A continuación, se muestra los resultados obtenidos de este sprint.

Pantallas de alta y baja

La figura 4.9 muestra la pantalla de ejemplo del Login, las figuras 4.10, 4.11 y 4.12 muestra cada Login para cada usuario. El docente y el padre de familia podrán iniciar sesión con su correo y contraseña, pero el alumno debe de introducir su NIA y contraseña para corresponder al panel correspondiente.



Figura 4.9 Diseño del Login (Pantalla de baja)
Fuente: Elaboración propia

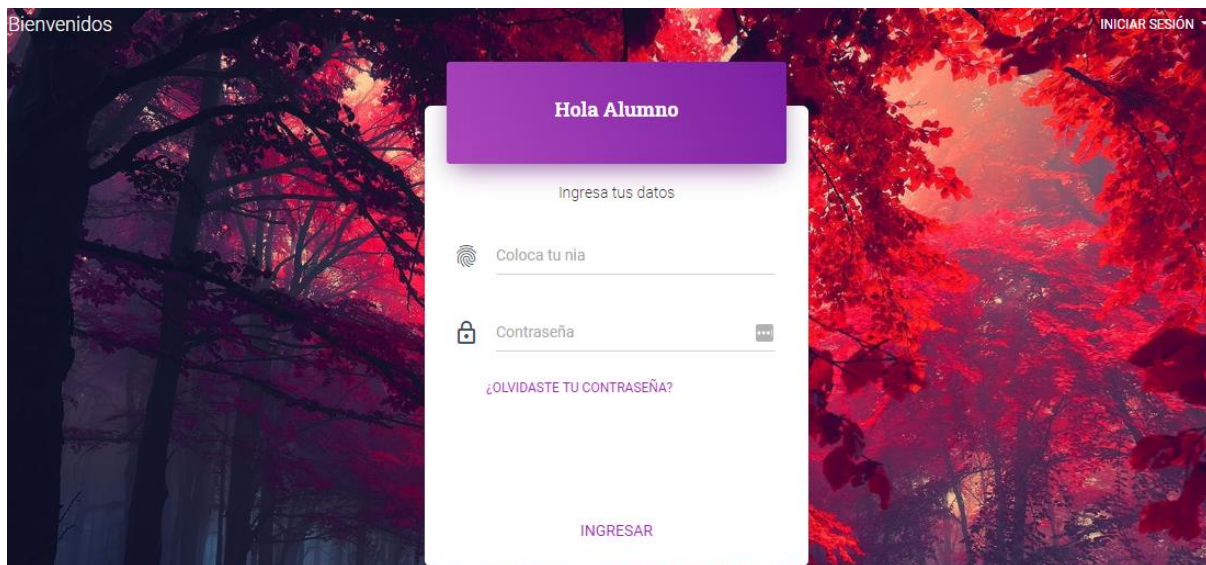


Figura 4.10 Diseño del Login Alumno (Pantalla de baja)
Fuente: Elaboración propia

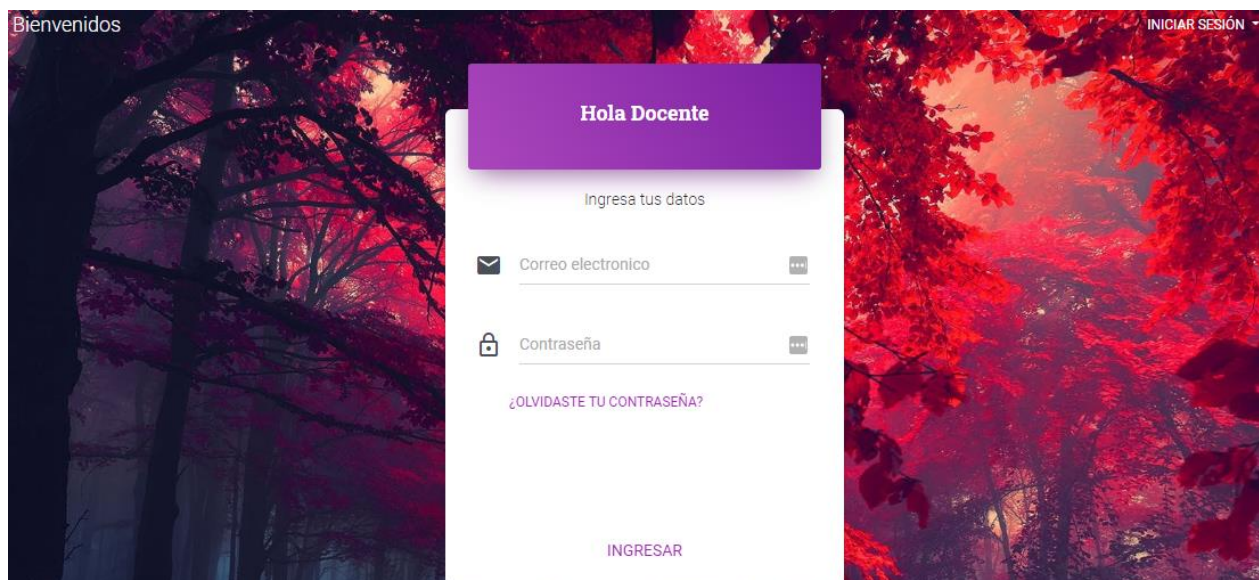


Figura 4.11 Diseño del Login Docente (Pantalla de alta)
Fuente: Elaboración propia



Figura 4.12 Diseño del Login Padres (Pantalla de alta)
Fuente: Elaboración propia

La tabla 4.12 muestra cada controlador para iniciar sesión y cada base de datos de los usuarios, la función Login contiene lo siguiente: dos arreglos que sirve para validar los datos que manda el usuario, la condición indica que si existe un usuario con los datos colocados lo redirecciona a la ruta indicada para cada usuario, en caso contrario muestra un mensaje indicando si el correo electrónico/NIA o contraseña son incorrectos. El docente y padre de familia inician sesión con correo y contraseña, el alumno debe de colocar el NIA y su contraseña.

Codificación Login Alumno

```

1  class AlumnoLoginController extends Controller
2  {
3
4      public function __construct()
5      {
6          $this->middleware('guest:alumno');
7      }
8
9      public function showLoginForm()
10     {
11         return view('auth.alumno.alumno-login');
12     }
13
14     public function login(Request $request)
15     {
16         //Reglas de validacion
17         $rules = [
18             $this->username() => 'required',
19             'password' => 'required',
20         ];
21         $this->username().'.required' => 'Debes de colocar tu NIA',
22         'password.required' => 'Debes de colocar tu contraseña',
23     ];
24         //Validamos las reglas
25         $this->validate($request,$rules,$message);
26
27         //dd($request->all());
28
29         //Si existe un alumno con el NIA y el password
30         if(Auth::guard('alumno')->attempt([$this->username() => $request->nia, 'password' => $request->password]))
31         {
32             //mandarlo a la ruta
33             return redirect('/alumno');
34         }
35         else{
36             $mensaje = 'Nia o contraseña incorrecta, vuelve a interarlo.';
37             //Regresar al login solamente con el nia
38         }
39     }
40 }

```

```

47         return back()
48         ->withInput($request->only($this->username()))
49         ->with(compact('mensaje'));
50     }
51 }
52 }
53 }
54 public function username() // Metodo para indicar que atributo sera iniciar sesion
55 {
56     return 'nia';
57 }

```

Conexión base de datos Alumno

```

Schema::create('alumnos', function (Blueprint $table) {
    $table->integer('nia')->primary(); //El nia sera unico
    $table->string('name')->length(50);
    $table->tinyInteger('edad')->length(3);
    $table->string('apellidoP')->length(20); //ApellidoP
    $table->string('apellidoM')->length(20); //ApellidoM
    $table->date('fechaN'); //Fecha Nacimiento
    $table->char('genero')->length(2); //genero
    $table->string('phone')->length(20);
    $table->string('email')->unique()->length(50);
    $table->string('password')->length(60);
    //FK Relacion entre el grupo
    $table->integer('grupo_id')->unsigned();
    $table->foreign('grupo_id')->references('id')->on('grupos');
    $table->rememberToken(); //Necesario para la autenticación por
    $table->timestamps(); //Crea dos atributos, uno para indicar

```

Codificación Login Docente

```

15 public function __construct()
16 {
17     $this->middleware('guest:docente'); //Indicamos que la vista del login sera para invitados
18 }
19 public function showLoginForm()
20 {
21     return view('auth.docente.docente-login'); //Mostramos la vista
22 }
23 }
24 public function login(Request $request)
25 {
26     //Colocamos las validaciones
27     $rules = [
28         'email' => 'required|email',
29         'password' => 'required',
30     ];
31     $this->validate($request,$rules,$message); //Ejecutamos la validacion si existe un error
32 }
33 //dd($request->all());
34 //Si existe un docente con el correo y contraseña colocadas
35 if(Auth::guard('docente')->attempt(['email' => $request->email, 'password' => $request->password]))
36 {
37     return redirect('/docente'); //Redirige a la ruta docente
38 }else{
39     //Redirige a login con un mensaje y el correo colocado
40     $mensaje = 'Correo o contraseña equivocadas. Intentelo de nuevo';
41     return back()
42     ->withInput($request->only('email'))
43     ->with(compact('mensaje')); //Solo muestra el correo electronico
44 }

```

Conexión base de datos Docente

```

Schema::create('docentes', function (Blueprint $table) {
    $table->increments('id');
    $table->string('name')->length(50);
    $table->string('apellidoP')->length(20);
    $table->string('apellidoM')->length(20);
    $table->tinyInteger('edad')->length(3);
    $table->string('email')->unique()->length(50); //El
    $table->string('password')->length(60);
    $table->string('telefono_fijo')->length(20);
    $table->string('telefono_cel')->length(20);
    $table->rememberToken();
    $table->timestamps();

```

Codificación Login Padre de familia

```

9 class PadreLoginController extends Controller
10 {
11     public function __construct()
12     {
13         $this->middleware('guest:padre');
14     }
15 }
16 public function showLoginForm()
17 {
18     return view('auth.padre.padre-login');
19 }
20 }

```

```

21 public function login(Request $request)
22 {
23     //validation
24     $rules = [
25         'email' => 'required|email',
26         'password' => 'required',
27     ];
28
29     $message = [
30         'email.required' => 'Debes de colocar tu correo electrónico.',
31         'email.email' => 'Este campo acepta solo correo electrónico',
32         'password.required' => 'Debes de colocar tu contraseña.'.
33     ];
34
35     $this->validate($request,$rules,$message);
36
37     //dd($request->all());
38
39     if(Auth::guard('padre')->attempt(['email' => $request->email, 'password' => $request->password]))
40     {
41         return redirect('/padre_familia');
42     }else{
43         $mensaje = 'Correo electrónico o contraseña incorrecta, vuelve a intentarlo.';
44         return back()
45             ->withInput($request->only('email'))
46             ->with(compact('mensaje'));
47     }

```

Conexión base de datos Padre de familia

```

Schema::create('padre_familias', function (Blueprint $table) {
    $table->increments('id');
    $table->string('name')->length(50); //brandom baruch
    $table->string('apellidoP')->length(20);
    $table->string('apellidoM')->length(20);
    $table->string('email')->unique()->length(50);
    $table->string('password')->length(60);
    $table->tinyInteger('edad')->length(3);
    $table->string('curp')->unique()->length(20);
    $table->string('telefono_fijo')->length(20);
    $table->string('telefono_cel')->length(20);
    $table->string('profesion')->length(30);
    $table->string('ocupacion')->length(50);
    $table->string('escolaridad')->length(20);
    $table->string('estado_civil')->length(10); //soltero casado
    $table->rememberToken();
    $table->timestamps();

```

Tabla 4.12 Codificación y conexión de base de datos.
Fuente: Elaboración propia

Historia de usuario	
Numero: 3	Usuario: Docente (puesto de director)
Nombre de la historia: Registro de alumnos	
Prioridad en negocio: alta	Riegos en desarrollo: alta
Programador responsable: Brandom Baruch González Cervantes	
Descripción: El docente con puesto de director podrá buscar, registrar, editar y eliminar alumnos.	

Tabla 4.13 HU - Registro de alumnos
Fuente: Elaboración propia

Iteración 3

Hace referencia a la tercera historia de usuario. El registro de alumnos puede acceder solamente el administrador.

Sprint Backlog iteración 3

En la tabla 4.14 se muestran las tareas que se hicieron durante la iteración 3.

Tarea asignada a: Brandom Baruch González Cervantes				
ID	Tarea	Inicio	Finalización	Estado
1	Asignación de los componentes en la vista del framework.	02/09/2020	02/09/2020	Completado
2	Creación de los métodos del controlador para programar	04/09/2020	06/09/2020	Completado
3	Creación de vistas	06/09/2020	07/09/2020	Completado
4	Pruebas de direccionamiento a otras pantallas.	07/09/2020	07/09/2020	Completado

Tabla 4.14 Tareas específicas iteración
Fuente: Elaboración propia

Diseño iteración 3

A continuación, se muestra el diseño inicial y el resultado obtenido para este sprint.

Pantallas de alta y baja

La figura 4.13 muestra el prototipo de la página web, la figura 4.14 muestra el diseño final donde muestra el listado de los alumnos, cada alumno tiene las funciones de editar, eliminar y ver a detalle la información del alumno, la figura 4.15 muestra el formulario para registrar un alumno y por último la figura 4.16 muestra los resultados obtenidos del buscador (en este caso se buscó un alumno con el nombre que tuviera **br**).

Figura 4.13 Pantalla de listado de alumnos. (Pantalla de baja)

Fuente: Elaboración propia



Listado de Alumnos

[AGREGAR NUEVO ALUMNO](#)

¿Qué alumno buscas?















NIA	Nombre	Apellido Paterno	Apellido Materno	Correo electrónico	Opciones
16240011	Brandom Baruch	González	Cervantes	baruchoo94@gmail.com	  
16240010	Brenda	González	Cervantes	brenda@live.com	  
10101010	Elia	Cervantes	Estrada	elia@live.com	  
159753	Elias	Atonal	Carro	elias@live.com	  
250525	Fernando	Barragan	Santana	barraga@live.com	  

Figura 4.14 Pantalla de listado de alumnos. (Pantalla de alta)
Fuente: Elaboración propia

Registrar nuevo Alumno

NIA	Nombre(s)	Apellido Paterno
Coloca un NIA	Escribe un nombre/s	Escribe un apellido paterno
Apellido Materno	Edad	Correo electrónico
Escribe un apellido materno	Escribe una edad	Escribe un correo electrónico
Grupo	Genero	fecha de nacimiento
Sin grupo	H	dd/mm/aaaa
Teléfono	Contraseña	Confirmar contraseña
Escribe un teléfono celular/fijo

Formato para teléfono celular/fijo : 123-123-1234

La contraseña por defecto es **secret**

[REGISTRAR ALUMNO](#) [CANCELAR](#)

Figura 4.15 Registro de alumno. (Pantalla de alta)
Fuente: Elaboración propia

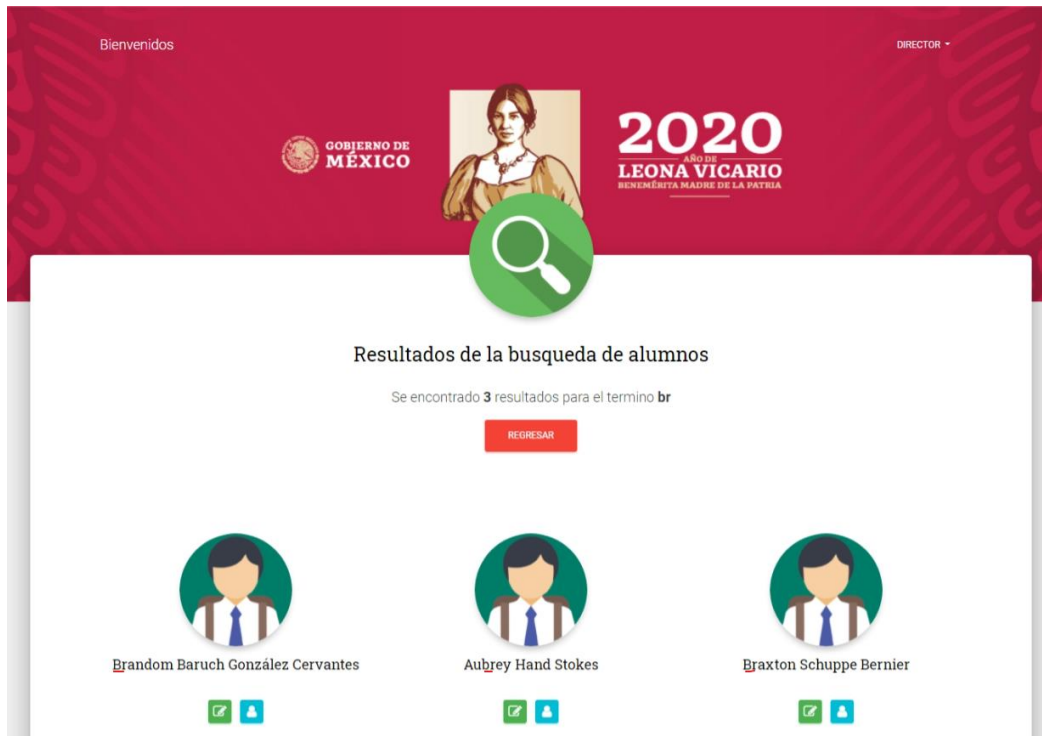


Figura 4.16 Buscador de alumno(s). (Pantalla de Fuente: Elaboración propia)

La tabla 4.15 muestra el controlador del alumno donde contiene varios métodos que son los siguientes:

- Index: Muestra el listado de los alumnos que se han registrado.
- Create: Muestra el formulario para registrar un alumno.
- Store: Captura los datos registrados en el formulario y los registra en la tabla alumnos.
- Edit: Muestra el formulario del alumno que se va a editar.
- Update: Captura los datos registrados en el formulario y actualiza los datos de la tabla alumnos.
- Delete: Elimina el alumno seleccionado.

Codificación

```

15 public function index(Request $request)
16 {
17     $request->user()->autorizarPuestos('Director');
18     $alumnos = Alumno::orderBy('name')->paginate(10);
19     return view('director.alumno.alumno_index')->with(compact('alumnos'));
20 }

```

```

22 public function create(Request $request)
23 {
24     $request->user()->autorizarPuestos('Director');
25     $grupos = Grupo::all();
26     return view('director.alumno.alumno_create')->with(compact('grupos'));
27 }
28 $alumno->nia = $request->input('nia');
29 $alumno->name = $request->input('name');
30 $alumno->edad = $request->input('edad');
31 $alumno->apellidoP = $request->input('apellidoP');
32 $alumno->apellidoM = $request->input('apellidoM');
33 $alumno->fechaN = $request->input('fechaN');
34 $alumno->genero = $request->input('genero');
35 $alumno->phone = $request->input('phone');
36 $alumno->email = $request->input('email');
37 $alumno->password = Hash::make($request->input('password'));
38 $alumno->grupo_id = $request->input('grupo_id');
39 $alumno->remember_token = str_random(100);
40 $alumno->save();
41
42 $mensaje = 'Se ha agregado el alumno ' . $alumno->name . ' exitosamente.';
43 return redirect('director/alumnos/index')->with(compact('mensaje'));
44 }
45
46 public function edit(Request $request,$nia)
47 {
48     $request->user()->autorizarPuestos('Director');
49     $alumno = Alumno::find($nia);
50     $grupo = Grupo::all();
51     return view('director.alumno.alumno_edit')->with(compact('alumno','grupo'));
52 }
53
54 public function update(Request $request, $nia)
55 {
56     //Creamos un objeto de la clase alumno
57     $alumno = Alumno::find($nia);
58     $alumno->nia = $alumno->nia;
59     $alumno->name = $request->input('name');
60     $alumno->edad = $request->input('edad');
61     $alumno->apellidoP = $request->input('apellidoP');
62     $alumno->apellidoM = $request->input('apellidoM');
63     $alumno->fechaN = $request->input('fechaN');
64     $alumno->genero = $request->input('genero');
65     $alumno->phone = $request->input('phone');
66     $alumno->email = $request->input('email');
67     if ($request->password) {
68         $alumno->password = Hash::make($request->input('password'));
69     }
70     $alumno->grupo_id = $request->input('grupo_id');
71     $alumno->remember_token = str_random(100);
72     $alumno->save();
73
74     $mensaje = 'Se ha actualizado el alumno ' . $alumno->name . ' exitosamente.';
75     return redirect('director/alumnos/index')->with(compact('mensaje'));
76 }
77 public function destroy ($nia)
78 {
79     $alumno = Alumno::find($nia);
80     $alumno->delete(); //Eliminamos al docente
81     $eliminado = 'Se ha eliminado el Alumno ' . $alumno->name . ' nia: ' . $alumno->nia;
82     return back()->with(compact('eliminado'));
83 }
84
85 public function show(Request $request, $nia)
86 {
87     $request->user()->autorizarPuestos('Director');
88     $alumno = Alumno::find($nia);
89     $grupo = Grupo::find($alumno->grupo_id);
90     $materia = $grupo->materias()->get();
91     return view('director.alumno.alumno_show')->with(compact('alumno','grupo','materia'));
92 }
93 }

```

Conexión base de datos

```

Schema::create('alumnos', function (Blueprint $table) {
    $table->integer('nia')->primary(); //EL nia sera unico
    $table->string('name')->length(50);
    $table->tinyInteger('edad')->length(3);
    $table->string('apellidoP')->length(20); //ApellidoP
    $table->string('apellidoM')->length(20); //ApellidoM
    $table->date('fechaN'); //Fecha Nacimiento
    $table->char('genero')->length(2); //genero
    $table->string('phone')->length(20);
    $table->string('email')->unique()->length(50);
    $table->string('password')->length(60);
    //FK Relacion entre el grupo
    $table->integer('grupo_id')->unsigned();
    $table->foreign('grupo_id')->references('id')->on('grupos');
    $table->rememberToken(); //Necesario para la autentificación p
    $table->timestamps(); //Crea dos atributos, uno para indicar

```

Tabla 4.15 Codificación y conexión de base de datos. Iteración 3

Fuente: Elaboración propia

Historia de usuario	
Numero: 4	Usuario: Docente (puesto de director)
Nombre de la historia: Registro de padres de familia	
Prioridad en negocio: alta	Riesgos en desarrollo: alta
Programador responsable: Brandom Baruch González Cervantes	
Descripción: El docente con el puesto de director podrá registrar a los padres de familia. El padre de familia tendrá un panel de control con su información general y un apartado donde debe de contestar las entrevistas correspondientes.	

Tabla 4.16 HU - Registro de padres de familia

Fuente: Elaboración propia

Iteración 4

Esta iteración hace referencia a la cuarta historia de usuario. El registro de los padres de familia y la pantalla que accede una vez que ha ingresado al sistema, el padre de familia y el alumno podrán realizar las entrevistas correspondientes.

Sprint Backlog iteración 4

La tabla 4.17 muestra las tareas específicas para la realización de la iteración 4.

Tarea asignada a: Brandom Baruch González Cervantes				
ID	Tarea	Inicio	Finalización	Estado
1	Asignación de los componentes en la vista del framework.	08/09/2020	08/09/2020	Completado
2	Creación del modelo y controlador para programar	09/09/2020	11/09/2020	Completado
3	Creación de tablas y migración en la base de datos	12/09/2020	12/09/2020	Completado
4	Pruebas de direccionamiento a otras pantallas.	12/09/2020	13/09/2020	Completado

Tabla 4.17 Tareas específicas iteración

Fuente: Elaboración propia

Diseño de la iteración 4

A continuación, se presenta el diseño y los resultados de la interacción.

Pantallas de alta y baja

El administrador es un docente con el puesto de director que tendrá acceso a las páginas de registro. En las figuras 4.17 y 4.18 muestra la pagina para registrar un nuevo padre de familia.

A screenshot of a web browser window showing a registration form. The browser's address bar shows 'https://'. The page title is 'A Web Page'. The form is titled 'Datos personales' and contains the following fields: 'Nombre del padre o tutor:', 'Apellido Paterno', 'Apellido Materno', 'NIA', 'Localidad', 'Direccion', 'Codigo postal', and 'Parentesco'. Below these fields is a large rectangular box with a plus icon and the text 'Agregar Padre o tutor'. At the bottom of the form is a 'Guardar' button.

Figura 4.17 Pantalla de registro padre. (Pantalla de baja)
Fuente: Elaboración propia

Registrar nuevo Padre de familia

Nombre(s) Coloca el nombre	Apellido Paterno Coloca el apellido paterno	Apellido Materno Coloca el apellido materno
Edad Ejemplo: 21	Teléfono fijo Formato: 123-123-1234	Teléfono celular Formato: 123-123-1234
Curp COLOCA LA CURP	Estado Civil Casado(a)	Escolaridad Primaria
Ocupación Ejemplo: Artesano	Profesión Ejemplo: Científico	Correo electronico Coloca el correo electrónico
Contraseña	Confirmar contraseña La contraseña por defecto es secret	Formato para teléfono fijo y celular : 123-123-1234 Si solamente tiene un numero telefónico, colócalo en las dos opciones

REGISTRAR PADRE **CANCELAR**

Figura 4.18 Pantalla de registro padre. (Pantalla de alta)
Fuente: Elaboración propia

La figura 4.19 muestra el listado de los padres de familia registrados de 10 en 10. Cada padre de familia tiene las opciones de: editar, eliminar y ver a detalle la información del padre de familia. La página cuenta con un buscador que solamente funciona con el nombre del padre de familia.



Listado de Padres de familia

REGISTRAR NUEVO PADRE
 🔍

#	Nombre	Apellido Paterno	Apellido Materno	Curp	Profesión	Escolaridad	Opciones
1	Alyce	Hansen	Sauer	EIR11KMSP6R9HHITUG	ipsa	Secundaria	👤 ✎ ✖
2	Andreane	Lowe	Kessler	DUP7ZSTVAVLFT4GHI6	nihil	Preparatoria	👤 ✎ ✖
3	Bernadette	O'Connell	Schimmel	KNCSQYQ0EFYDI5S7FF	ut	Universidad	👤 ✎ ✖
4	Cleora	Greenholt	Roob	6JVLFP8TBVJLQYIK8	quos	Preparatoria	👤 ✎ ✖
5	Elia	Cervantes	Estrada	DASDIVBSADKJFKSND	Licenciado en educación	Universidad	👤 ✎ ✖
6	Eric	Jakubowski	Hand	0VA56H91DUKC4EFTCD	in	Secundaria	👤 ✎ ✖
7	Gilda	Runolfsson	Pfeffer	JEXWJJ2FV7QQCZJWK	nobis	Universidad	👤 ✎ ✖
8	Golda	Hodkiewicz	Stoltenberg	3ZFEJNVQUDNP00OZZR	voluptas	Secundaria	👤 ✎ ✖
9	Green	Tremblay	Pagac	FVTCPIVAL3LDRJYJXF	possimus	Secundaria	👤 ✎ ✖
10	Herman	Hoppe	Eichmann	U0C5ZJK94RCWWEFF0X	qui	Primaria	👤 ✎ ✖

1 2 3 ▶

Figura 4.19 Listado de padres. (Pantalla de alta)

Fuente: Elaboración propia

En la figura 4.20 y 4.21 se muestran la interfaz del panel de control del usuario (padre de familia). El usuario podrá ver su información y editarla en caso de ser necesario, su domicilio, asignar un parentesco con un alumno y subir documentos (INE, referencia del domicilio, etc).

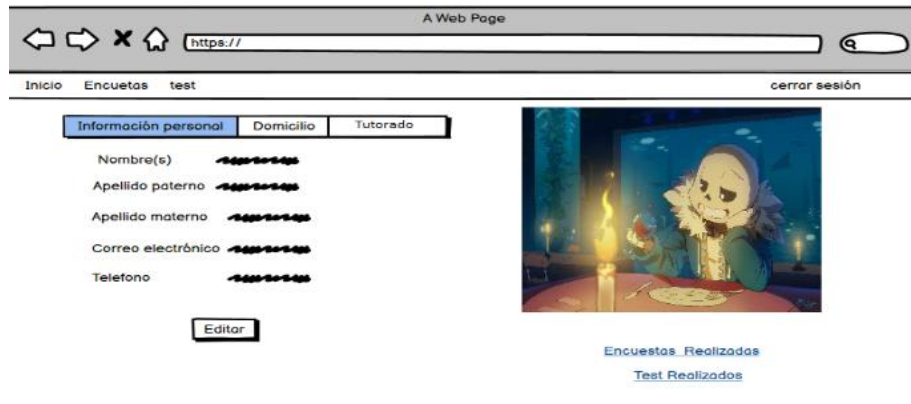


Figura 4.20 Diseño de panel de control. (Pantalla de baja)
Fuente: Elaboración propia

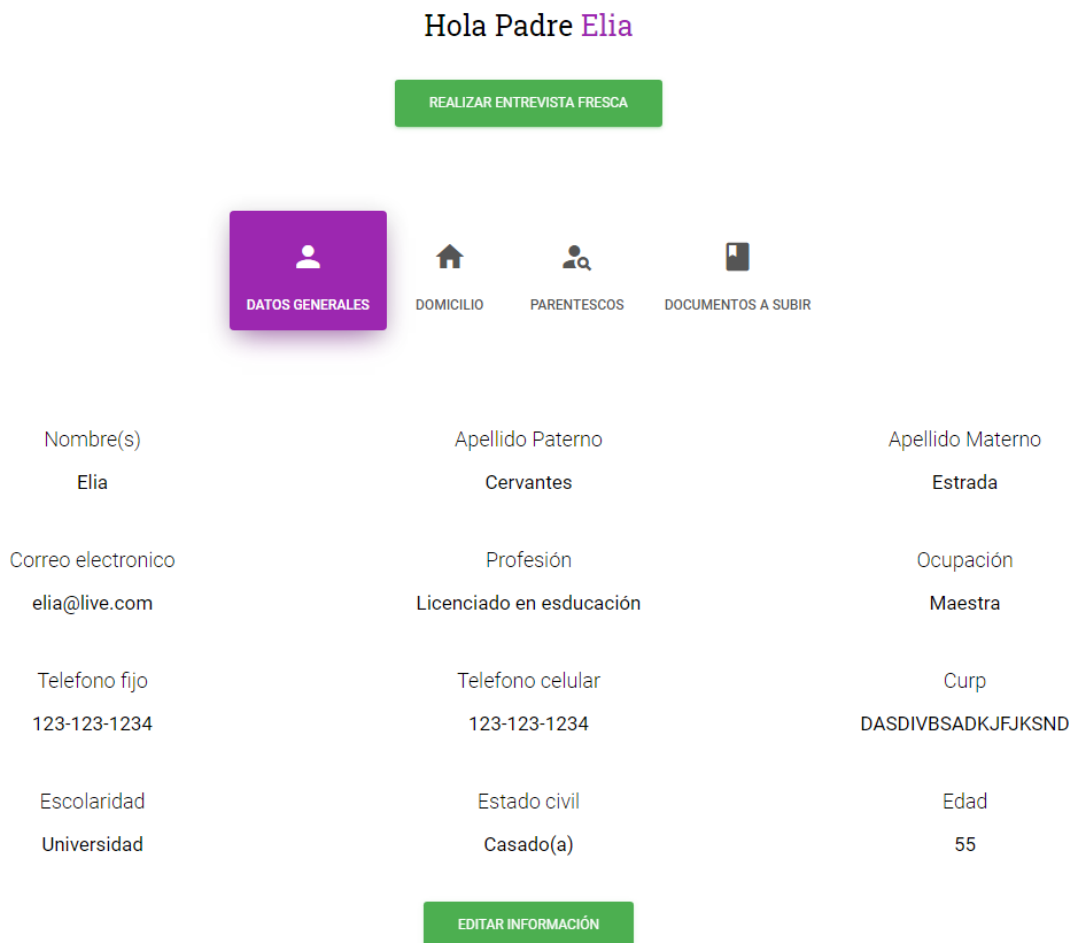


Figura 4.21 Diseño de panel de control. (Pantalla de alta)
Fuente: Elaboración propia

La tabla 14.18 muestra los datos generales del padre de familia y la base de datos.

Codificación	
<pre> 180 <div class="form-group col-md-4"> 181 <h4>Nombre(s)</h4> 182 <p class="h5">{{Auth::user()->name}}</p> 183 </div> 184 <div class="form-group col-md-4"> 185 <h4>Apellido Paterno</h4> 186 <p class="h5">{{Auth::user()->apellidoP}}</p> 187 </div> 188 <div class="form-group col-md-4"> 189 <h4>Apellido Materno</h4> 190 <p class="h5">{{Auth::user()->apellidoM}}</p> 191 </div> 192 <div class="form-group col-md-4"> 193 <h4>Correo electrónico</h4> 194 <p class="h5">{{Auth::user()->email}}</p> 195 </div> 196 <div class="form-group col-md-4"> 197 <h4>Profesión</h4> 198 <p class="h5">{{Auth::user()->profesion}}</p> 199 </div> 200 <div class="form-group col-md-4"> 201 <h4>Ocupación</h4> 202 <p class="h5">{{Auth::user()->ocupacion}}</p> 203 </div> 204 <div class="form-group col-md-4"> 205 <h4>Telefono fijo</h4> 206 <p class="h5">{{Auth::user()->telefono_fijo}}</p> </pre>	
Conexión base de datos	
	<pre> Schema::create('padre_familias', function (Blueprint \$table) { \$table->increments('id'); \$table->string('name')->length(50); //brandom baruch \$table->string('apellidoP')->length(20); \$table->string('apellidoM')->length(20); \$table->string('email')->unique()->length(50); \$table->string('password')->length(60); \$table->tinyInteger('edad')->length(3); \$table->string('curp')->unique()->length(20); \$table->string('telefono_fijo')->length(20); \$table->string('telefono_cel')->length(20); \$table->string('profesion')->length(30); \$table->string('ocupacion')->length(50); \$table->string('escolaridad')->length(20); \$table->string('estado_civil')->length(10); //soltero casado \$table->rememberToken(); \$table->timestamps(); } </pre>

Tabla 4.18 Codificación y conexión de base de datos. Iteración 4
Fuente: Elaboración propia

Historia de usuario	
Numero: 5	Usuario: Docente (puesto de director)
Nombre de la historia: Motor de búsqueda	
Prioridad en negocio: alta	Riegos en desarrollo: alta
Programador responsable: Brandom Baruch González Cervantes	
Descripción: Las páginas de listados de padre de familia y alumno contarán con un buscado que funciona únicamente con el nombre para encontrar al usuario correspondiente.	

Tabla 4.19 Motor de búsqueda
Fuente: Elaboración propia

Iteración 5

Hace referencia a la quinta historia de usuario que es el motor de búsqueda de la información de los usuarios.

Sprint Backlog iteración 5

La tabla 4.20 muestra las tareas específicas a realizar en esta iteración 5.

Tarea asignada a: Brandom Baruch González Cervantes				
ID	Tarea	Inicio	Finalización	Estado
1	Asignación de los componentes en la vista del framework.	14/09/2020	14/09/2020	Completado
2	Creación del método buscar en el controlador.	15/09/2020	15/09/2020	Completado
3	Creación de las vistas del buscador.	15/09/2020	16/09/2020	Completado
4	Pruebas de direccionamiento a otras pantallas.	16/09/2020	16/09/2020	Completado

Tabla 4.20 Tareas específicas iteración 5

Fuente: Elaboración propia

Diseño de la iteración 5

A continuación, se mostrarán los resultados obtenidos en este sprint.

Pantallas de alta y baja

El buscador sirve para localizar los usuarios basándose por las iniciales de su nombre o nombre completo. La figura 4.22 muestra los resultados con dicha inicial y tendrá dos botones, uno es para editar el usuario y el otro es para ver sus datos generales, la figura 4.23 muestra si fue escrito el nombre completo lo redirige a sus datos generales.



Figura 4.22 Pantalla de búsqueda. (Pantalla de alta)
Fuente: Elaboración propia

Información del Padre Elia





 DATOS GENERALES			 DOMICILIO	 PARENTESCOS	 DOCUMENTOS
Nombre(s)	Apellido Paterno	Apellido Materno			
Elia	Cervantes	Estrada			
Correo electronico	Profesión	Curp			
elia@live.com	Licenciado en educación	DASDIVBSADKJFJKSND			
Ocupación	Telefono fijo	Telefono celular			
Maestra	123-123-1234	123-123-1234			
Escolaridad	Estado civil	Edad			
Universidad	Casado(a)	55			
			EDITAR INFORMACIÓN	REGRESAR	

Figura 4.23 Pantalla de datos generales padre. (Pantalla de Fuente: Elaboración propia

La tabla 4.21 muestra el método para buscar el padre de familia, la condición indica que si encuentra un parecido va a mostrar los datos, pero si encuentra el indicado va a mostrar la página donde muestra todos sus datos.

Codificación

```
$request->user()->autorizarPuestos('Director');  
//Guardo el dato en la variable search  
$search = $request->input('search');  
//Realiza una consulta con el dato obtenido  
$padres = padre::select()->where('name', 'like', "%$search%")->get();  
//Si solamente existe un resultado  
if ($padres->count() == 1) {  
    //Obtener su id y dirigirlo a la ruta seleccionada  
    $id = $padres->first()->id;  
    return redirect("director/padre_familia/$id/show");  
}
```

<pre>//Si existen varios resultados, mostrarlo en la vista "padre_search" return view('director.buscador.padre_search')->with(compact('padres', 'search'));</pre>
<p>Conexión base de datos</p> <pre>Schema::create('padre_familias', function (Blueprint \$table) { \$table->increments('id'); \$table->string('name')->length(50); //brandom baruch \$table->string('apellidoP')->length(20); \$table->string('apellidoM')->length(20); \$table->string('email')->unique()->length(50); \$table->string('password')->length(60); \$table->tinyInteger('edad')->length(3); \$table->string('curp')->unique()->length(20); \$table->string('telefono_fijo')->length(20); \$table->string('telefono_cel')->length(20); \$table->string('profesion')->length(30); \$table->string('ocupacion')->length(50); \$table->string('escolaridad')->length(20); \$table->string('estado_civil')->length(10); //soltero casado \$table->rememberToken(); \$table->timestamps(); });</pre>

Tabla 4.21 Codificación y conexión de base de datos. Iteración 5
Fuente: Elaboración propia

Historia de usuario	
Numero: 6	Usuario: Docente (puesto director), alumno y padre de familia
Nombre de la historia: Asignación de parentesco	
Prioridad en negocio: alta	Riegos en desarrollo: alta
Programador responsable: Brandom Baruch González Cervantes	
Descripción: Todos los usuarios (docente, alumno y padre de familia) podrán asignar un parentesco entre el alumno y el padre de familia	

Tabla 4.22 HU - Asignación de parentesco
Fuente: Elaboración propia

Iteración 6

Hace referencia a la sexta historia de usuario. La asignación de parentesco entre padre y alumno. El administrador, el padre de familia y el alumno podrá asignar un parentesco con uno o varios alumnos (caso del padre de familia) o padres de familia (en caso del alumno).

Sprint Backlog iteración 6

La tabla 4.23 muestra las tareas específicas a realizar en la iteración 6.

Tarea asignada a: Brandom Baruch González Cervantes				
ID	Tarea	Inicio	Finalización	Estado
1	Asignación de los componentes en la vista del framework	17/09/2020	17/09/2020	Completado
2	Creación del modelo y controlador para programar	18/09/2020	20/09/2020	Completado

3	Creación de tablas y migración en la base de datos	21/09/2020	21/09/2020	Completado
4	Pruebas de direccionamiento a otras pantallas	21/09/2020	22/09/2020	Completado

Tabla 4.23 Tareas específicas iteración

Fuente: Elaboración propia

Diseño de la iteración 6

A continuación, se mostrarán los resultados de este sprint.

Pantallas de alta

En la figura 4.24 (director), 4.25 (alumno) y 4.26 (padre de familia) se muestra la pantalla para asignar un parentesco entre un alumno y un padre de familia. Cuando se selecciona un parentesco se mostrará el nia del alumno, el nombre y el parentesco que tienen. Este apartado tiene la opción de eliminar el parentesco.

Asignar parentesco(s) para el alumno
Brandom Baruch

Selecciona un familiar

Sanford Hauck Walker

Selecciona un parentesco

Papá

AGREGAR
REGRESAR

#	Nombre completo	Parentesco	Opciones
1	Elia Cervantes Estrada	Mamá	✕

Figura 4.24 Asignación de parentesco del director. (Pantalla de alta)

Fuente: Elaboración propia

Asignar familiar para el alumno **Brenda**

Selecciona un familiar

Elia Cervantes Estrada

Selecciona un parentesco

Papá

AGREGAR
REGRESAR

#	Nombre	Parentesco	Opciones
1	Herman Hoppe Eichmann	Papá	✕

Figura 4.25 Asignación de parentesco del alumno. (Pantalla de alta)

Fuente: Elaboración propia

Asignar familiar para el padre Bernadette

Selecciona un alumno	Selecciona un parentesco
Brandom Baruch González Cervantes	Papá

#	Nombre	Parentesco	Opciones
1	Elias Atonal Carro	Papá	

Figura 4.26 Asignación de parentesco del padre. (Pantalla de alta)

Fuente: Elaboración propia

La tabla 4.24 muestra los siguientes métodos:

- Index: Muestra los parentescos entre un padre y alumno.
- Store: Registra el parentesco entre padre y alumno.
- Delete: Eliminar el parentesco.

Codificación

```
10
11 class AsignarAlumnosPadreController extends Controller
12 {
13
14   public function index(Request $request,$id)
15   {
16     $request->user()->autorizarPuestos('Director');
17     $padre = padre::find($id);
18     $alumnos = Alumno::orderBy('name')->get();
19     return view('director.padre_familia.padre_parentesco')->with(compact('padre','alumnos'));
20   }
21
22
23   public function store(Request $request, $id)
24   {
25
26     $rules = [
27       'parentezco' => 'required'
28     ];
29
30     $message = [
31       'parentezco.required' => 'Debes de escribir el parentesco que tienen'
32     ];
33
34     $this->validate($request, $rules , $message);
35
36     $padre = padre::find($id);
37     $parentezco = new Parentezco;
38     $parentezco->parentezco = $request->input('parentezco');
39     $parentezco->padre_id = $padre->id;
40     $parentezco->save();
41     $mensaje = 'Se ha agregado el alumno exitosamente';
42     return back()->with(compact('mensaje'));
43   }
44
45   public function destroy($padre_id , $alumno_id)
46   {
47     $alumno = Alumno::find($alumno_id);
48     $padre = padre::find($padre_id);
49     $padre->alumnos()->detach($alumno->nia);
50     $mensaje = 'Se ha eliminado el alumno ' . $alumno->name . ' exitosamente ' ;
51     return back()->with(compact('mensaje'));
52   }
53 }
54 }
```

Conexión base de datos

```

public function up()
{
    Schema::create('parentezcos', function (Blueprint $table) {
        $table->increments('id');
        $table->string('parentezco');
        //FK
        $table->integer('alumno_id')->unsigned();
        $table->foreign('alumno_id')->references('nia')->on('alumnos');

        //FK
        $table->integer('padre_id')->unsigned();
        $table->foreign('padre_id')->references('id')->on('padre_familias');

        $table->timestamps();
    });
}

```

Tabla 4.24 Codificación y conexión de base de datos. Iteración
Fuente: Elaboración propia

Historia de usuario	
Numero: 7	Usuario: Docente (puesto de director)
Nombre de la historia: Registro de docentes	
Prioridad en negocio: alta	Riegos en desarrollo: alta
Programador responsable: Brandom Baruch González Cervantes	
Descripción: El docente con puesto de director podrá registrar docentes.	

Tabla 4.25 HU - Registro de docentes
Fuente: Elaboración propia

Iteración 7

Hace referencia a la séptima historia de usuario sobre el registro de los docentes.

Sprint Backlog iteración 7

La tabla 4.26 muestra las tareas correspondientes para la realización de la iteración 7.

Tarea asignada a: Brandom Baruch González Cervantes				
ID	Tarea	Inicio	Finalización	Estado
1	Asignación de los componentes en la vista del framework	23/09/2020	23/09/2020	Completado
2	Creación del modelo y controlador para programar	24/09/2020	26/09/2020	Completado
3	Creación de tablas y migración en la base de datos	27/09/2020	27/09/2020	Completado
4	Pruebas de direccionamiento a otras pantallas	28/09/2020	28/09/2020	Completado

Tabla 4.26 Tareas específicas iteración
Fuente: Elaboración propia

Pantallas de baja y alta

Las figuras 4.27 y 4.28 muestra la pantalla para registrar un nuevo docente, en la figura 4.29 se muestra el listado de los docentes que están registrados, cada docente tiene la opción de: editar, eliminar y ver sus datos generales a detalle (ver figura 4.30).

A Web Page

https://

Ayuda --

Datos del docente

Nombre del padre o tutor:

Apellido Paterno

Apellido Materno

NIA

cedula profesional

Domicilio

Localidad

Direccion

Código postal

Datos de trabajo

Turno

Genero

Figura 4.27 Pantalla de registro de docente. (Pantalla de baja)
Fuente: Elaboración propia

Registrar nuevo Docente



Nombre(s) Escribe un nombre/s <input type="text"/>	Apellido Paterno Escribe un apellido paterno <input type="text"/>	Apellido Materno Escribe un apellido materno <input type="text"/>
Edad Escribe una edad <input type="text"/>	Correo electronico Escribe un correo electrónico <input type="text"/>	Teléfono fijo Formato: 123-123-1234 <input type="text"/>
Telefono celular Formato: 123-123-1234 <input type="text"/>	Contraseña <input type="password"/>	Confirmar contraseña <input type="password"/>

Formato para teléfono fijo y celular : 123-123-1234

Si solamente tiene un numero telefónico, colócalo en las dos opciones

Figura 4.28 Pantalla de registro de docente. (Pantalla de alta)
Fuente: Elaboración propia

Bienvenidos VASH ESTAMPIDA ▾


































2020

AÑO DE
LEONA VICARIO
BENEMÉRITA MADRE DE LA PATRIA

Listado de Docentes

AGREGAR NUEVO DOCENTE

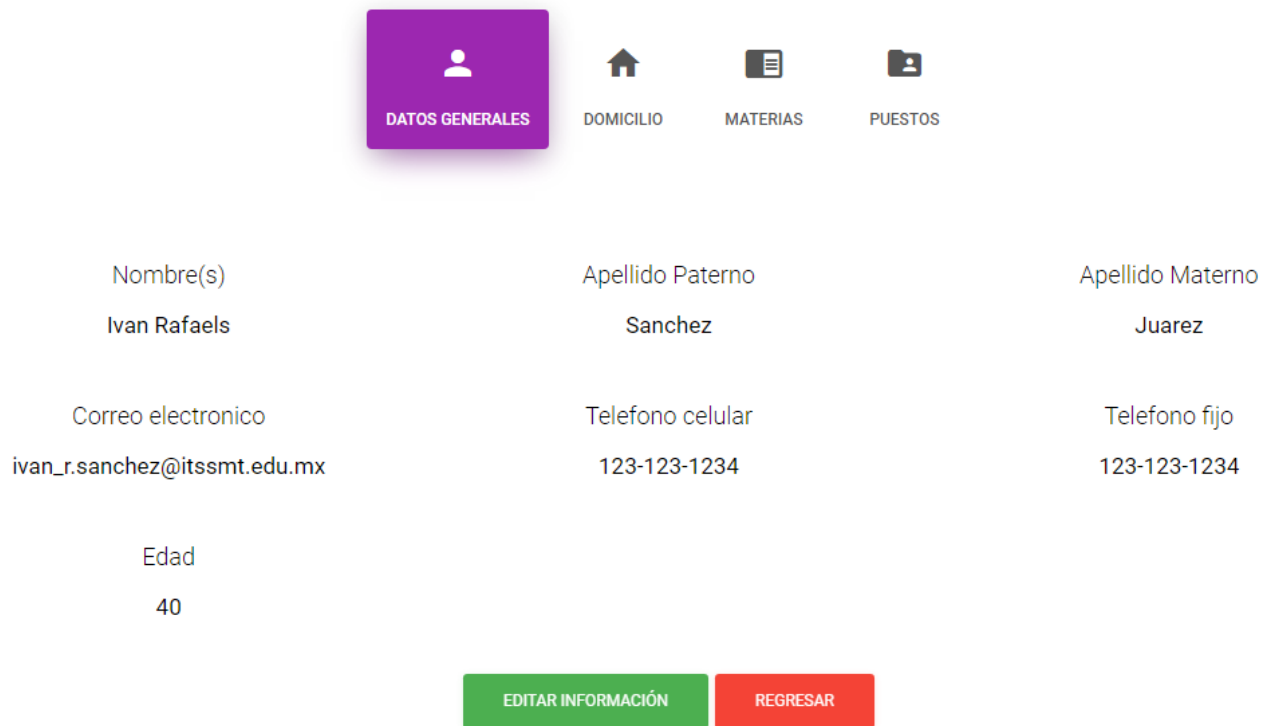
#	Nombre	Apellido Paterno	Apellido Materno	Correo Electronico	Opciones
1	Alanis	Leuschke	Cole	tweimann@example.net	  
2	Alanna	Gutmann	Zulauf	janelle.heidenreich@example.com	  
3	Alexa	Pagac	Simonis	wiza.raymond@example.com	  
4	Bernard	Thiel	Osinski	davis.norberto@example.org	  
5	Brooklyn	Gleichner	Lebsack	agoyette@example.com	  
6	Chesley	Hoeger	Heidenreich	jace.reynolds@example.com	  
7	Ivan Rafaels	Sanchez	Juarez	ivan_r.sanchez@itssmt.edu.mx	  
8	Kiley	VonRueden	Rempel	wisoky.madelynn@example.com	  
9	Mavis	Stiedemann	Moore	dnicolas@example.net	  
10	Michel	Hahn	Fritsch	amir13@example.com	  

◀ 1 2 ▶

© 2020, hecho por SGTEMS

Figura 4.29 Pantalla de listado de docentes. (Pantalla de alta)
Fuente: Elaboración propia

Información del Docente Ivan Rafaels



Nombre(s)	Apellido Paterno	Apellido Materno
Ivan Rafaels	Sanchez	Juarez
Correo electronico	Telefono celular	Telefono fijo
ivan_r.sanchez@itssmt.edu.mx	123-123-1234	123-123-1234
Edad		
40		

[EDITAR INFORMACIÓN](#) [REGRESAR](#)

Figura 4.30 Pantalla de datos generales. (Pantalla de alta)
Fuente: Elaboración propia

Las figuras 4.31 y 4.32 muestra el panel de control del docente y cuentan con los siguientes apartados:

- Datos generales: Muestra su información general como: nombre, apellidos, teléfonos, etc. El mismo docente podrá editar su información.
- Domicilio: Mostrará la información de su domicilio, si el domicilio que escogió no es el correcto. Podrá eliminarlo y escoger uno nuevo o agregar su domicilio.
- Materias: Mostrará las materias que impartirá en este ciclo escolar.
- Puestos: Mostrara los puestos que tenga el docente (director, tutor, profesor, etc.).

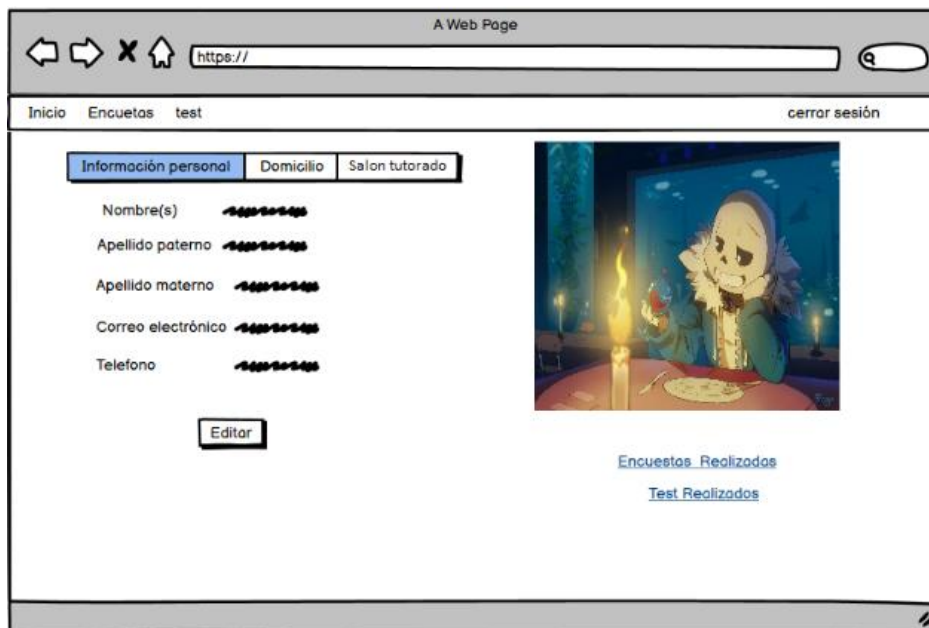
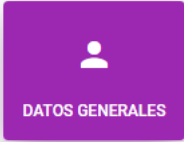


Figura 4.31 Panel de control del Docente. (Pantalla de baja)
Fuente: Elaboración propia

Hola Docente **Ivan Rafael**


DOMICILIO
MATERIAS
PUESTOS

Nombre(s) Ivan Rafael	Apellido Paterno Sanchez	Apellido Materno Juarez
Correo electrónico ivan_r.sanchez@itssmt.edu.mx	Teléfono celular 123-123-1234	Teléfono fijo 123-123-1234
Edad 40		

EDITAR INFORMACIÓN

Figura 4.32 Panel de control del Docente. (Pantalla de alta)
Fuente: Elaboración propia

La tabla 4.27 muestra los siguientes métodos:

- Constructor: El panel de control de docente solo será accesible para el docente autenticado.
- Index: Muestra el panel de control de docente cuando inicia sesión.
- Edit: Muestra el formulario para editar el docente.
- Update: Registra los datos del formulario en la base de datos de docentes.

Codificación	
<pre>15 public function __construct() 16 { 17 \$this->middleware('auth:docente'); //EL pane de control solo puede entrar el docente 18 } 19 20 public function index() 21 { 22 return view('docente.docente_home'); 23 } 24 25 public function edit() 26 { 27 28 return view('docente.docente_edit'); 29 } 30 31 public function update(Request \$request, \$id) 32 { 33 //Reglas de validacion 34 35 \$messages = [36 'name.required' => 'Debes de colocar un nombre', 37 'apellidoP.required' => 'Debes de colocar un apellido paterno', 38 'apellidoM.required' => 'Debes de colocar un apellido materno', 39 'edad.required' => 'Debes de colocar la edad', 40 'edad.min' => 'La edad minima es 18 años', 41 'edad.max' => 'La edad maxima es 80 años', 42 'email.required' => 'Debes de colocar un correo electronico', 43 'email.email' => 'Solo se aceptan correos electronicos', 44 // 'password.required' => 'Debes de colocar una contraseña', 45 // 'password.min' => 'La contraseña debe tener por lo menos 6 digitos', 46 'password.confirmed' => 'La contraseña no coinciden, intentalo de nuevo', 47 'telefono_fijo.required' => 'Debes de colocar un telefono fijo', 48 'telefono_cel.required' => 'Debes de colocar un telefono celular', 49]; 50 51 \$this->validate(\$request,\$rules,\$messages); 52 53 \$docente = Docente::find(\$id); 54 \$docente->name = \$request->input('name'); 55 \$docente->apellidoP = \$request->input('apellidoP'); 56 \$docente->apellidoM = \$request->input('apellidoM'); 57 \$docente->edad = \$request->input('edad'); 58 \$docente->email = \$request->input('email'); 59 \$docente->telefono_fijo = \$request->input('telefono_fijo'); 60 \$docente->telefono_cel = \$request->input('telefono_cel'); 61 if (\$request->password) { 62 \$docente->password = Hash::make(\$request->input('password')); 63 } 64 }</pre>	
Conexión base de datos	
<pre>Schema::create('docentes', function (Blueprint \$table) { \$table->increments('id'); \$table->string('name')->length(50); \$table->string('apellidoP')->length(20); \$table->string('apellidoM')->length(20); \$table->tinyInteger('edad')->length(3); \$table->string('email')->unique()->length(50); //EL \$table->string('password')->length(60); \$table->string('telefono_fijo')->length(20); \$table->string('telefono_cel')->length(20); \$table->rememberToken(); \$table->timestamps(); });</pre>	

Tabla 4.27 Codificación y conexión de base de datos. Iteración 7
Fuente: Elaboración propia

Historia de usuario	
Numero: 8	Usuario: Docente (puesto de director)
Nombre de la historia: Asignar puestos	
Prioridad en negocio: alta	Riesgos en desarrollo: alta
Programador responsable: Brandom Baruch González Cervantes	
Descripción: El docente con puesto de director podrá registrar y asignar puestos a los docentes.	

Tabla 4.28 Asignar puestos
Fuente: Elaboración propia

Iteración 8

Hace referencia a la octava historia de usuario. La asignación de puestos para los docentes, solamente tiene acceso el administrador.

Sprint Backlog iteración 8

En la tabla 4.29 se muestran las tareas que se hicieron durante la iteración 8.

Tarea asignada a: Brandom Baruch González Cervantes				
ID	Tarea	Inicio	Finalización	Estado
1	Asignación de los componentes en la vista del framework.	29/09/2020	29/09/2020	Completado
2	Creación del modelo y controlador para programar	30/09/2020	02/10/2020	Completado
3	Creación de tablas y migración en la base de datos	02/10/2020	03/10/2020	Completado
4	Pruebas de direccionamiento a otras pantallas.	03/10/2020	03/10/2020	Completado

Tabla 4.29 Tareas específicas iteración
Fuente: Elaboración propia

Diseño de la iteración 8

A continuación, se mostrarán los resultados de este sprint.

Pantallas de alta

La figura 4.33 muestra el listado de los puestos que puede tener un docente, el administrador puede registrar (figura 4.34), editar, eliminar y asignar el puesto a uno o varios docentes (figura 4.35).



Listado de Puestos

AGREGAR NUEVO PUESTO

#	Nombre	Descripción	Opciones
1	Director	Descripcion del director	  
2	Tutor Escolar	Persona que es responsable de un grupo en particular	  

Figura 4.33 Pantalla de listado de puestos. (Pantalla de Fuente: Elaboración propia)

Registrar nuevo Puesto

Nombre del puesto

Descripción del puesto

REGISTRAR PUESTO CANCELAR

Figura 4.34 Pantalla para registrar puesto. (Pantalla de Fuente: Elaboración propia)

Asignar puesto de Director

Selecciona un docente

Vash Estampida

AGREGAR DOCENTE REGRESAR


Nombre	Apellido Paterno	Apellido Materno	Opciones
Vash Estampida	Carro	López	 

Figura 4.35 Pantalla de asignar un puesto. (Pantalla de alta) Fuente: Elaboración propia

La tabla 4.30 muestra los métodos para listar, registrar, editar, eliminar un puesto y asignar o quitar dicho puesto a un docente. La función autorizarPuestos('Director') indica que solo el docente con el puesto de director podrá ver las vistas.

Codificación	
<pre> 12 public function index(Request \$request) 13 { 14 \$request->user()->autorizarPuestos('Director'); 15 \$puestos = Puesto::paginate(5); 16 return view('director.docente.puestos.puesto_index')->with(compact('puestos')); 17 } 18 19 public function create(Request \$request) 20 { 21 \$request->user()->autorizarPuestos('Director'); 22 return view('director.docente.puestos.puesto_create'); 23 } 24 25 public function store(Request \$request) 26 { 27 \$rules = [28]; 29 30 \$message = [31]; 32 33 \$this->validate(\$request,\$rules,\$message); 34 //dd(\$request->all()); 35 \$puesto = new Puesto; 36 \$puesto->puesto = \$request->input('puesto'); 37 \$puesto->descripcion = \$request->input('descripcion'); 38 \$puesto->save(); 39 public function edit(Request \$request,\$id) 40 { 41 \$request->user()->autorizarPuestos('Director'); 42 \$puesto = Puesto::find(\$id); 43 return view('director.docente.puestos.puesto_edit')->with(compact('puesto')); 44 } 45 46 public function update(Request \$request , \$id) 47 { 48 \$rules = [49]; 50 51 \$message = [52]; 53 54 \$this->validate(\$request,\$rules,\$message); 55 //dd(\$request->all()); 56 \$puesto = Puesto::find(\$id); 57 \$puesto->puesto = \$request->input('puesto'); 58 \$puesto->descripcion = \$request->input('descripcion'); 59 60 public function destroy(\$id) 61 { 62 \$puesto = Puesto::find(\$id); 63 \$puesto->delete(); 64 \$eliminador = 'Se ha eliminado el puesto ' . \$puesto->puesto; 65 return back()->with(compact('eliminador')); 66 } 67 68 public function show(Request \$request,\$id) 69 { 70 \$request->user()->autorizarPuestos('Director'); 71 \$puesto = Puesto::find(\$id); 72 \$docentes = Docente::all(); 73 return view('director.docente.puestos.puesto_show')->with(compact('puesto','docentes')); 74 } 75 76 public function agregarPuesto(Request \$request, \$id) 77 { 78 \$puesto = Puesto::find(\$id); 79 \$puesto_asignado = new PuestoAsignado; 80 \$puesto_asignado->docente_id = \$request->input('docente_id'); 81 \$puesto_asignado->puesto_id = \$puesto->id; 82 \$puesto_asignado->save(); 83 \$mensaje = 'Se ha asignado el puesto para el docente exitosamente'; 84 return back()->with(compact('mensaje')); 85 } 86 87 public function eliminarPuesto(\$puesto_id , \$docente_id) 88 { 89 \$docente = Docente::find(\$docente_id); 90 \$docente->puestos()->detach(\$puesto_id); 91 \$eliminador = 'Se ha quitado el puesto del docente: ' . \$docente->name; 92 return back()->with(compact('eliminador')); 93 } 94 } </pre>	
<p>Conexión base de datos</p>	

```

public function up()
{
    Schema::create('puestos', function (Blueprint $table) {
        $table->increments('id');
        $table->string('puesto');
        $table->string('descripcion');
        $table->timestamps();
    });
}

```

Tabla 4.30 Codificación y conexión de base de datos. Iteración 8

Fuente: Elaboración propia

Historia de usuario	
Numero: 9	Usuario: Docente (puesto de director)
Nombre de la historia: Registro de domicilios	
Prioridad en negocio: alta	Riegos en desarrollo: alta
Programador responsable: Brandom Baruch González Cervantes	
Descripción: El docente con puesto de director podrá registrar, editar y eliminar domicilios.	

Tabla 4.31 Registro de domicilios

Fuente: Elaboración propia

Iteración 9

Hace referencia a la novena historia de usuario sobre el registro de domicilio.

Sprint Backlog Iteración 9

La tabla 4.32 muestra las tareas específicas a realizar para la iteración 9.

Tarea asignada a: Brandom Baruch González Cervantes				
ID	Tarea	Inicio	Finalización	Estado
1	Asignación de los componentes en la vista del framework.	04/10/2020	04/10/2020	Completado
2	Creación del modelo y controlador para programar	05/10/2020	07/10/2020	Completado
3	Creación de tablas y migración en la base de datos	08/10/2020	08/10/2020	Completado
4	Pruebas de direccionamiento a otras pantallas.	08/10/2020	09/10/2020	Completado

Tabla 4.32 Tareas específicas iteración

Fuente: Elaboración propia

Diseño de la iteración 9

A continuación, se muestra el resultado de este sprint.

Pantallas de alta

La figura 47 muestra todos los domicilios registrados, el administrador podrá registrar (figura 48) y cada domicilio cuenta con botones para editar y eliminar.

Codigo Postal	Estado	Municipio	Calle	Colonia	No Exterior	No Interior	Opciones
90135	Tlaxcala	Santa Ana Nopalucan	Av. Loma Bonita	Barrio Puebla	11	0	
90135	Tlaxcala	Santa Ana Nopalucan	Av. Loma Bonita	Barrio Puebla	12	0	

Figura 4.36 Pantalla de listado de domicilios. (Pantalla de alta)
Fuente: Elaboración propia

Registrar nuevo Domicilio

Estado	Municipio	Localidad
<input type="text" value="Escribe un estado"/>	<input type="text" value="Coloca un municipio"/>	<input type="text" value="Coloca una localidad"/>
Calle	Colonia	No. Interior
<input type="text" value="Coloca una calle"/>	<input type="text" value="Coloca una colonia"/>	<input type="text" value="Coloca un No. interio"/>
No. Exterior	Código Postal	
<input type="text" value="Coloca un No. exterior"/>	<input type="text" value="Coloca un Código postal"/>	

Figura 4.37 Pantalla para registrar un domicilio. (Pantalla de alta)
Fuente: Elaboración propia

La tabla 4.33 muestra la base de datos de domicilios y los métodos para listar, registrar, editar y eliminar un domicilio.

Codificación

```
12 public function index(Request $request)
13 {
14     $request->user()->autorizarPuestos('Director');
15     $domicilios = Domicilio::paginate(10);
16     return view('director.domicilio.domicilio_index')->with(compact('domicilios'));
17 }
18
19
20 public function create(Request $request)
21 {
22     $request->user()->autorizarPuestos('Director');
23     return view('director.domicilio.domicilio_create');
24 }
25
26
27 public function store(Request $request)
28 {
29     $rules = [
30 ];
31
32     $message = [
33 ];
34
35     $this->validate($request, $rules, $message);
36
37     $domicilio = new Domicilio;
38     $domicilio->localidad = $request->input('localidad');
39     $domicilio->calle = $request->input('calle');
40     $domicilio->cp = $request->input('cp');
41     $domicilio->colonia = $request->input('colonia');
42     $domicilio->no_interior = $request->input('no_interior');
43     $domicilio->no_exterior = $request->input('no_exterior');
44     $domicilio->save();
45     $mensaje = 'Se ha agregado un domicilio exitosamente. ¿Quieres agregar otro domicilio?';
46     return back()->with(compact('mensaje'));
47 }
48
49
50 public function edit(Request $request,$id)
51 {
52     $request->user()->autorizarPuestos('Director');
53     $domicilio = Domicilio::find($id);
54     return view('director.domicilio.domicilio_edit')->with(compact('domicilio'));
55 }
56
57
58 public function update(Request $request, $id)
59 {
60     $domicilio->municipio = $request->input('municipio');
61     $domicilio->localidad = $request->input('localidad');
62     $domicilio->calle = $request->input('calle');
63     $domicilio->cp = $request->input('cp');
64     $domicilio->colonia = $request->input('colonia');
65     $domicilio->no_interior = $request->input('no_interior');
66     $domicilio->no_exterior = $request->input('no_exterior');
67     $domicilio->save();
68     $mensaje = 'Se ha actualizado el domicilio existente de manera exitosa.';
69     return redirect('director/domicilios/index')->with(compact('mensaje'));
70 }
71
72
73 public function destroy($id)
74 {
75     $domicilio = Domicilio::find($id);
76     $domicilio->delete();
77     $eliminado = 'Se ha eliminado el domicilio exitosamente';
78     return back()->with(compact('eliminado'));
79 }
```

Conexión base de datos

```
Schema::create('domicilios', function (Blueprint $table)
{
    $table->increments('id');
    $table->string('estado',50);
    $table->string('municipio',100); //Santa Ana Nopalucan
    $table->string('localidad',100);
    $table->string('calle',100);
    $table->char('cp', 6);
    $table->string('colonia',30);
    $table->char('no_interior',3);
    $table->char('no_exterior',3);
    $table->timestamps();
});
```

Tabla 4.33 Codificación y conexión de base de datos. Iteración 9

Fuente: Elaboración propia

Historia de usuario	
Numero: 10	Usuario: Docente (puesto de director)
Nombre de la historia: Registrar Grupos	
Prioridad en negocio: alta	Riegos en desarrollo: alta
Programador responsable: Brandom Baruch González Cervantes	
Descripción: El docente con puesto de director podrá registrar grupos.	

Tabla 4.34 HU - Registro grupos
Fuente: Elaboración propia

Iteración 10

Hace referencia a la décima historia de usuario. El registro de grupos solamente tiene acceso el administrador.

Sprint Backlog iteración 10

La tabla 4.35 muestra las tareas específicas para la realizar la iteración 10.

Tarea asignada a: Brandom Baruch González Cervantes				
ID	Tarea	Inicio	Finalización	Estado
1	Asignación de los componentes en la vista del framework.	10/10/2020	10/10/2020	Completado
2	Creación del modelo y controlador para programar	11/10/2020	13/10/2020	Completado
3	Creación de tablas y migración en la base de datos	13/10/2020	14/10/2020	Completado
4	Pruebas de direccionamiento a otras pantallas.	14/10/2020	15/10/2020	Completado

Tabla 4.35 Tareas específicas iteración 10
Fuente: Elaboración propia

Diseño iteración 10

En el siguiente apartado se mostrarán los resultados tras la realización de las actividades de este sprint.

Pantallas de alta

La figura 4.38 muestra el listado de grupos registrados (figura 4.39), cada grupo puede editar y eliminar un grupo. La página tiene un botón para observar que alumnos y materias están en el grupo (figuras 4.40 y 4.41).

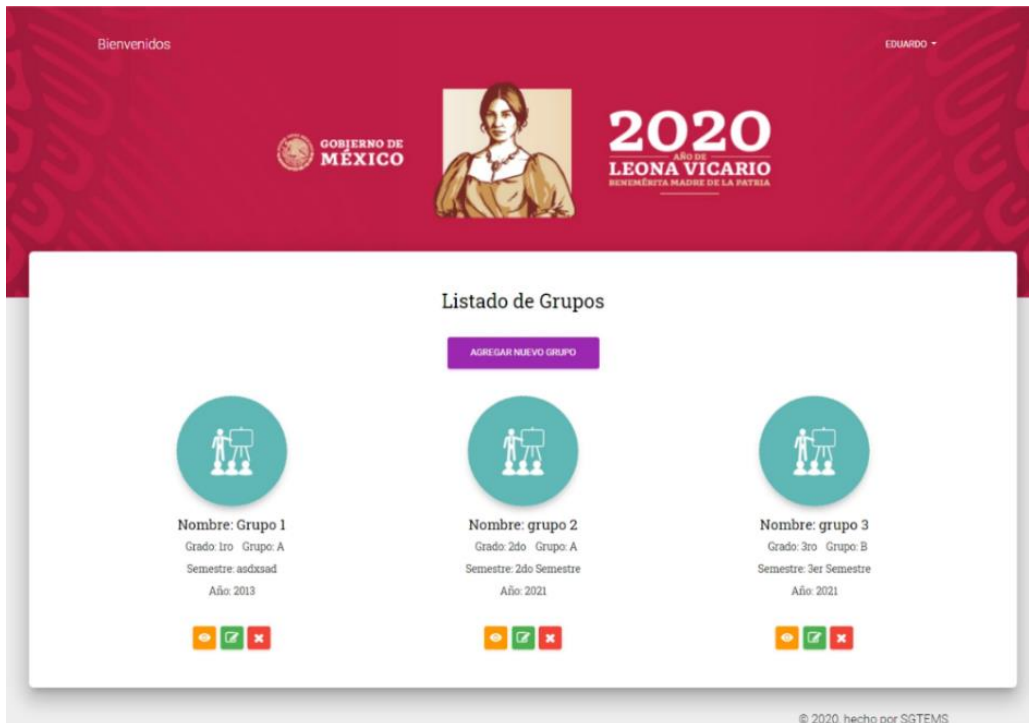


Figura 4.38 Pantalla de listado de grupos. (Pantalla de alta)

Fuente: Elaboración propia

Registrar nuevo Grupo

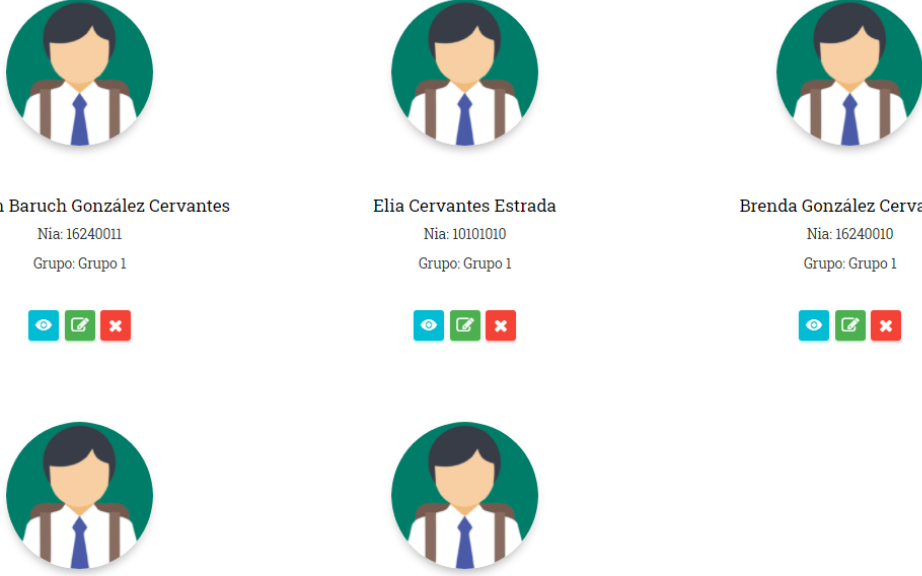
Nombre	Grado	Grupo
<input type="text" value="Escribe un nombre para el grupo"/>	<input type="text" value="Escribe el grado del grupo"/>	<input type="text" value="Ejemplo: A"/>
Semestre	Año	
<input type="text" value="Escribe el semestre del grupo"/>	<input type="text" value="Escribe el año del grupo"/>	
<input type="button" value="AGREGAR GRUPO"/>		<input type="button" value="CANCELAR"/>

Figura 4.39 Pantalla de registrar un grupo. (Pantalla de alta)

Fuente: Elaboración propia

Alumnos que pertenecen al grupo Grupo 1

MATERIAS DEL GRUPO REGRESAR



Brandom Baruch González Cervantes
Nia: 16240011
Grupo: Grupo 1

Elia Cervantes Estrada
Nia: 10101010
Grupo: Grupo 1

Brenda González Cervantes
Nia: 16240010
Grupo: Grupo 1

Fernando Barragan Santana
Nia: 250525
Grupo: Grupo 1

Elias Atonal Carro
Nia: 159753
Grupo: Grupo 1

Each student profile includes three icons: a blue eye icon, a green notepad icon, and a red 'X' icon.

Figura 4.40 Pantalla de alumnos que pertenecen al grupo. (Pantalla de alta)


Fuente: Elaboración propia

Materias del grupo Grupo 1

Grado:2do y Grupo:A

Este grupo ya tiene asignada la materia de **Tutorias 1s**

AGREGAR MATERIAS REGRESAR



Tutorias 1s
Clave: TUTORIAS11
asfwrqefwrasdasdas

✖

Figura 4.41 Pantalla de materias que pertenecen al grupo. (Pantalla de alta)

Fuente: Elaboración propia

La tabla 4.36 muestra la tabla de grupos y los métodos para listar, registrar, editar, eliminar y mostrar los alumnos y materias que pertenecen al grupo.

Codificación	
14	<code>public function index(Request \$request)</code>
15	<code>{</code>
16	<code> \$request->user()->autorizarPuestos('Director');</code>
17	<code> \$grupos = Grupo::paginate(10);</code>
18	<code> return view('director.grupo.grupo_index')->with(compact('grupos'));</code>
19	<code>}</code>
20	
21	<code>public function create(Request \$request)</code>
22	<code>{</code>
23	<code> \$request->user()->autorizarPuestos('Director');</code>
24	<code> return view('director.grupo.grupo_create');</code>
25	<code>}</code>
26	
27	<code>public function store(Request \$request)</code>
28	<code>{</code>
29	<code> \$rules = [</code>
30	<code> 'name' => 'required',</code>
31	<code> 'grado' => 'required',</code>
32	<code> 'semestre' => 'required',</code>
33	<code> 'year' => 'required',</code>
34	<code> 'grupo' => 'required',</code>
35	<code>];</code>
36	<code> \$message = [</code>
37	<code> 'name' => 'El nombre es obligatorio.',</code>
38	<code> 'grado' => 'El grado es obligatorio.',</code>
39	<code> 'semestre' => 'El semestre es obligatorio.',</code>
40	<code> 'year' => 'El año es obligatorio.',</code>
41	<code> 'grupo' => 'El grupo es obligatorio.',</code>
42	<code>];</code>
43	<code> \$this->validate(\$request, \$rules, \$message);</code>
44	<code> \$grupo = new Grupo;</code>
45	<code> \$grupo->name = \$request->input('name');</code>
46	<code> \$grupo->grado = \$request->input('grado');</code>
47	<code> \$grupo->semestre = \$request->input('semestre');</code>
48	<code> \$grupo->year = \$request->input('year');</code>
49	<code> \$grupo->grupo = \$request->input('grupo');</code>
50	<code> \$grupo->save();</code>
51	<code> \$mensaje = 'Se ha agregado un grupo nuevo. ¿Quieres agregar otro grupo?';</code>
52	<code> return back()->with(compact('mensaje'));</code>
53	<code>}</code>
54	
55	<code>public function edit(Request \$request,\$id)</code>
56	<code>{</code>
57	<code> \$grupo = Grupo::find(\$id);</code>
58	<code> return view('director.grupo.grupo_edit')->with(compact('grupo'));</code>
59	<code>}</code>
60	
61	<code>public function update(Request \$request, \$id)</code>
62	<code>{</code>
63	<code> \$rules = [</code>
64	<code> 'name' => 'required',</code>
65	<code> 'grado' => 'required',</code>
66	<code> 'semestre' => 'required',</code>
67	<code> 'year' => 'required',</code>
68	<code> 'grupo' => 'required',</code>
69	<code>];</code>
70	<code> \$message = [</code>
71	<code> 'name' => 'El nombre es obligatorio.',</code>
72	<code> 'grado' => 'El grado es obligatorio.',</code>
73	<code> 'semestre' => 'El semestre es obligatorio.',</code>
74	<code> 'year' => 'El año es obligatorio.',</code>
75	<code> 'grupo' => 'El grupo es obligatorio.',</code>
76	<code>];</code>
77	<code> \$this->validate(\$request, \$rules, \$message);</code>
78	<code> \$grupo = Grupo::find(\$id);</code>
79	<code> \$grupo->name = \$request->input('name');</code>
80	<code> \$grupo->grado = \$request->input('grado');</code>
81	<code> \$grupo->semestre = \$request->input('semestre');</code>
82	<code> \$grupo->year = \$request->input('year');</code>
83	<code> \$grupo->grupo = \$request->input('grupo');</code>
84	<code> \$grupo->save();</code>
85	<code> return back()->with(compact('mensaje'));</code>
86	<code>}</code>
87	
88	<code>public function show(Request \$request, Grupo \$grupo)</code>
89	<code>{</code>
90	<code> \$request->user()->autorizarPuestos('Director');</code>
91	<code> \$materias = \$grupo->materias()->paginate(10);</code>
92	<code> \$tutorias = \$grupo->materias()->where('name','like','%tutorias%')->first();</code>
93	<code> //dd(\$tutorias->name);</code>
94	<code> return view('director.grupo.materia_grupo.grupo_materia_show')->with(compact('grupo', 'materias', 'tutorias'));</code>
95	<code>}</code>
96	
97	<code>public function create(Request \$request,\$id)</code>
98	<code>{</code>
99	<code> \$request->user()->autorizarPuestos('Director');</code>
100	<code> \$grupo = Grupo::find(\$id);</code>
101	<code> \$tutorias = \$grupo->materias()->where('name','like','%tutorias%')->first();</code>
102	<code> //dd(\$t->name);</code>
103	<code> \$materias = Materia::orderBy('id', 'desc')->get();</code>
104	<code> return view('director.grupo.materia_grupo.grupo_materias')->with(compact('grupo', 'materias', 'tutorias'));</code>
105	<code>}</code>
106	
107	<code>public function store(Request \$request)</code>
108	<code>{</code>
109	<code> \$asignada = new Asignada;</code>
110	<code> \$asignada->grupo_id = \$request->input('grupo_id');</code>
111	<code> \$asignada->materia_id = \$request->input('materia_id');</code>
112	<code> \$asignada->save();</code>
113	<code> \$mensaje = 'Se ha agregado la materia de ' . \$request->input('materia_name') . ' a ' . \$request->input('grupo_id') . ' con éxito.';</code>
114	<code> return back()->with(compact('mensaje'));</code>
115	<code>public function destroy(\$grupo_id, \$materia_id)</code>
116	<code>{</code>
117	<code> \$materia = Materia::find(\$materia_id);</code>
118	<code> \$materia->grupos()->detach(\$grupo_id);</code>
119	<code> \$eliminada = 'Se ha quitado la materia ' . \$materia->name . ' del grupo exitosamente.';</code>
120	<code> return back()->with(compact('eliminada'));</code>
121	<code>}</code>

Conexión base de datos

```

public function up()
{
    Schema::create('grupos', function (Blueprint $table) {
        $table->increments('id');
        $table->string('name');
        $table->string('grado',100);
        $table->char('grupo',2);
        $table->string('semestre',100);
        $table->year('year');
        $table->timestamps();
    });
}

```

Tabla 4.36 Codificación y conexión de base de datos. Iteración 10
Fuente: Elaboración propia

Historia de usuario	
Numero: 11	Usuario: Docente (puesto de director)
Nombre de la historia: Registro de materias	
Prioridad en negocio: alta	Riegos en desarrollo: alta
Programador responsable: Brandom Baruch González Cervantes	
Descripción: El docente con puesto de director podrá registrar, editar y eliminar materias.	

Tabla 4.37 Registro de materias
Fuente: Elaboración propia

Iteración 11

Hace referencia a la décima primera historia de usuario. El registro de materias únicamente el acceso es del administrador.

Sprint Backlog iteración 11

La tabla 4.38 muestra las tareas a realizar en la iteración 11.

Tarea asignada a: Brandom Baruch González Cervantes				
ID	Tarea	Inicio	Finalización	Estado
1	Asignación de los componentes en la vista del framework.	16/10/2020	16/10/2020	Completado
2	Creación del modelo y controlador para programar	17/10/2020	20/10/2020	Completado
3	Creación de tablas y migración en la base de datos	21/10/2020	21/10/2020	Completado
4	Pruebas de direccionamiento a otras pantallas.	21/10/2020	22/10/2020	Completado

Tabla 4.38 Tareas específicas iteración 11
Fuente: Elaboración propia

Diseño de la iteración 11

A continuación, se muestran los resultados de la realización del sprint.

Pantallas de alta

La figura 4.42 muestra todas las materias registradas (figura 4.43), cada materia puede editar, eliminar y asignar dicha materia a un docente (figura 4.44).

#	Nombre	Clave	Descripción	Opciones
1	Tutorias 1	TUTORIAS11	Clase de tutorías en la mañana	  
2	Contabilidad	CON-6701	Materia de especialidad	  
3	Tutorias 1	TUTORIAS1_1	Clase de tutorías de la tarde	  

Figura 4.42 Pantalla de listado de materias. (Pantalla de alta)
Fuente: Elaboración propia

Registrar nueva Materia

Nombre de la materia	Clave de la materia
<input type="text" value="Escribe el nombre de la materia"/>	<input type="text" value="ESCRIBE LA CLAVE DE LA MATERIA"/>
Descripción de la materia	
<input type="text" value="Coloca una descripción corta"/>	
<input type="button" value="REGISTRAR MATERIA"/>	<input type="button" value="CANCELAR"/>

Figura 4.43 Pantalla de registro de materia. (Pantalla de alta)
Fuente: Elaboración propia

La tabla 4.39 muestra los métodos para listar, registrar, editar, eliminar y asignar materias.

Codificación
<pre> 9 class MateriaController extends Controller 10 { 11 public function index (Request \$request) 12 { 13 \$request->user()->autorizarPuestos('Director'); 14 \$materias = Materia::paginate(10); 15 return view('director.materia.materia_index')->with(compact('materias')); 16 } 17 18 19 public function create(Request \$request) 20 { 21 \$request->user()->autorizarPuestos('Director'); 22 return view('director.materia.materia_create'); 23 } 24 25 public function store(Request \$request) 26 { 27 \$rules = [31]; 32 33 \$message = [39]; 40 41 \$this->validate(\$request , \$rules , \$message); 42 //dd(\$request->all()); 43 \$materia = new Materia; 44 \$materia->name = \$request->input('name'); 45 \$materia->descripcion = \$request->input('descripcion'); 46 \$materia->clave = \$request->input('clave'); 47 \$materia->save(); 48 \$mensaje = 'Se ha agregado la materia ' . \$materia->name . ' exitosamente.'; 49 return redirect('director/materias/index')->with(compact('mensaje')); 50 } 51 52 public function edit(Request \$request,\$id) 53 { 54 \$request->user()->autorizarPuestos('Director'); 55 \$materia = Materia::find(\$id); 56 return view ('director.materia.materia_edit')->with(compact('materia')); 57 } 58 59 public function update(Request \$request, \$id) 60 { 61 \$rules = [65]; 74 \$this->validate(\$request , \$rules , \$message); 75 \$materia = Materia::find(\$id); 76 \$materia->name = \$request->input('name'); 77 \$materia->descripcion = \$request->input('descripcion'); 78 \$materia->clave = \$request->input('clave'); 79 \$materia->save(); 80 \$mensaje = 'Se ha editado la materia ' . \$materia->name . ' exitosamente.'; 81 return redirect('director/materias/index')->with(compact('mensaje')); 82 } 84 public function destroy (\$id) 85 { 86 \$materia = Materia::find(\$id); 87 \$materia->delete(); 88 \$eliminado = 'Se ha eliminado la materia ' . \$materia->name; 89 return back()->with(compact('eliminado')); 90 } 91 92 public function store(Request \$request , \$id) 93 { 94 \$materia = Materia::find(\$id); 95 \$imparte = new Imparte; 96 \$imparte->docente_id = \$request->input('docente_id'); 97 \$imparte->materia_id = \$materia->id; 98 \$imparte->save(); 99 \$mensaje = 'Se ha asignado un docente para la materia ' . \$materia->name; 100 return back()->with(compact('mensaje')); 101 } </pre>
Conexión base de datos
<pre> public function up() { Schema::create('materias', function (Blueprint \$table) { \$table->increments('id'); \$table->string('name',255); \$table->string('descripcion',255); \$table->string('clave',50); \$table->timestamps(); }); } </pre>

Tabla 4.39 Codificación y conexión de base de datos. Iteración 11
Fuente: Elaboración propia

Historia de usuario	
Numero: 12	Usuario: Padre de familia y alumno
Nombre de la historia: Apartado de entrevistas	
Prioridad en negocio: alta	Riesgos en desarrollo: medio
Programador responsable: Brandom Baruch González Cervantes	
Descripción: El padre de familia y el alumno contarán con un apartado donde deben de realizar las entrevistas correspondientes.	

Tabla 4.40 HU - Apartado de entrevistas

Fuente: Elaboración propia

Iteración 12

Hace referencia a la décima segunda historia de usuario. Para el caso del Padre de familia cuando pulsa el botón de “Realizar entrevista fresca” lo redirigirá a otra página que debe de realizar las entrevistas con el parentesco que tenga con el alumno. El alumno en la parte superior derecho encontrará su nombre, al dar “clic” se desplegará las opciones que cuenta el panel. La opción de las entrevistas es “panel de entrevistas”.

Sprint Backlog iteración 12

En la tabla 41 se muestran las tareas a realizar para la iteración 12.

Tarea asignada a: Brandom Baruch González Cervantes				
ID	Tarea	Inicio	Finalización	Estado
1	Asignación de los componentes en la vista del framework.	23/10/2020	23/10/2020	Completado
2	Creación del modelo y controlador para programar	24/10/2020	29/10/2020	Completado
3	Creación de tablas y migración en la base de datos	30/10/2020	30/10/2020	Completado
4	Pruebas de direccionamiento a otras pantallas.	31/10/2020	31/10/2020	Completado

Tabla 4.41 Tareas específicas iteración 12

Fuente: Elaboración propia

Diseño de la iteración 12

En este apartado se mostrará el resultado de la iteración 12.

Pantallas de alta y baja

La figura 4.44 muestra la estructura para el apartado de las encuestas. El padre de familia solo contestara la entrevista fresca por cada parentesco que tenga con el alumno (si tiene dos parentescos, tendrá que contestar dos veces la entrevista) (figura 4.45).

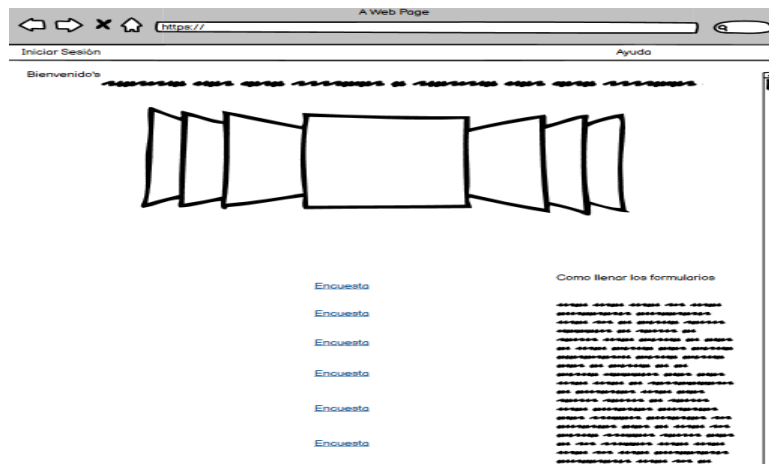


Figura 4.44 Panel de encuestas. (Pantalla de baja)
Fuente: Elaboración propia

Entrevista fresca para el padre de familia

Realiza las encuestas para cada alumnos

#	Nombre completo	Parentesco	Opciones
16240011	Brandom Baruch González Cervantes	Mamá	Finalizo la entrevista
16240010	Brenda González Cervantes	Mamá	EMPEZAR ENTREVISTA

[REGRESAR](#)

Figura 4.45 Panel de encuestas. (Pantalla de alta)
Fuente: Elaboración propia

La figura 4.46 muestra la ubicación para mostrar las encuestas que debe de realizar (figura 4.47). En los casos de las encuestas: Entrevista fresca, Cuestionario anexos y Test sobre estilos de aprendizaje cuentan con varias secciones. Las figuras 4.48, 4.49 y 4.50 muestran las secciones para los casos mencionados.



Hola Alumno **Brandom Baruch**

Figura 4.46 Panel de entrevistas. (Pantalla de alta)
Fuente: Elaboración propia

Encuestas que debes de realizar

#	Nombre	Enlace
1	Entrevista fresca	INICIAR
2	Cuestionario Anexos	INICIAR
3	Cuestionario "No Abandono"	INICIAR
4	Test sobre estilos de aprendizaje	INICIAR
5	Test para detectar tutorados que requieran atención individual	INICIAR

Figura 4.47 Pantalla de listado de encuestas alumno. (Pantalla de alta)
Fuente: Elaboración propia

1. Encuesta sobre Datos Familiares
2. Encuesta sobre Datos Academicos
3. Habitos de Estudio
4. Otras Actividades
5. Datos Adicionales

Figura 4.48 Pantalla de sección "Entrevista fresca". (Pantalla de alta)
Fuente: Elaboración propia

1. Atribuciones
2. Niveles de empatía
3. Tipo de mentalidad

Figura 4.49 Pantalla de sección "Cuestionarios anexos". (Pantalla de alta)
Fuente: Elaboración propia

1. Conociendo los estilos de aprendizaje
2. Encontrar estilo de aprendizaje
3. **HÁBITOS DE ESTUDIO**

Figura 4.50 Pantalla de sección "Test sobre estilos de aprendizaje". (Pantalla de alta)
Fuente: Elaboración propia

La tabla 4.42 muestra los métodos para registrar un dato en la tabla entrevista fresca alumno cuando el alumno inicia la entrevista y el método update se ejecuta cuando termina todas las secciones de la entrevista fresca. Esto se repite para las demás encuestas.

Codificación
<pre> 1 class EntrevistaPadreController extends Controller 2 { 3 public function index() 4 { 5 return view('padre_familia.entrevista_fresca.entrevista_index'); 6 } 7 8 public function store(Request \$request) 9 { 10 \$entrevista = new Entrevista_Fresca_Padre; 11 \$entrevista->fecha_aplicacion = null; 12 \$entrevista->descripcion = 'Inicio la entrevista'; 13 \$entrevista->alumno_id = \$request->input('alumno_id'); 14 \$entrevista->padre_id = auth()->user()->id; 15 \$entrevista->save(); 16 return redirect('/padre_familia/entrevista/'.\$request->input('alumno_id').'/secciones'); 17 } 18 19 public function secciones(\$alumno_id) 20 { 21 \$alumno = Alumno::find(\$alumno_id); 22 //dd(Auth::user()->entrevista->marca_si_no); 23 return view('padre_familia.entrevista_fresca.secciones')->with(compact('alumno')); 24 } 25 26 public function update(\$alumno_id) 27 { 28 \$alumno = Alumno::find(\$alumno_id); 29 \$entrevista = Entrevista_Fresca_Padre::where('alumno_id',\$alumno->nia)->first(); 30 \$entrevista->fecha_aplicacion = Carbon::now(); 31 \$entrevista->descripcion = 'Finalizo la entrevista'; 32 \$entrevista->save(); 33 } 34 } </pre>
Conexión base de datos

```

Schema::create('entrevista_fresca_alumnos', function (Blueprint $table) {
    $table->increments('id');
    $table->date('fecha_aplicacion')->nullable();
    $table->string('descripcion',30); //Inicio, Finalizo
    //FK
    $table->integer('alumno_id')->unsigned();
    $table->foreign('alumno_id')->references('nia')->on('alumnos');
    $table->timestamps();
Schema::create('test', function (Blueprint $table) {
    $table->increments('id');
    $table->date('fecha_aplicacion')->nullable();
    $table->string('descripcion'); //Inicio, Finalizo
    //FK
    $table->integer('alumno_id')->unsigned();
    $table->foreign('alumno_id')->references('nia')->on('alumnos');
    $table->timestamps();
Schema::create('cuestionario_anexos', function (Blueprint $table) {
    $table->increments('id');
    $table->date('fecha_aplicacion')->nullable();
    $table->string('descripcion'); //Inicio, Finalizo
    //FK
    $table->integer('alumno_id')->unsigned();
    $table->foreign('alumno_id')->references('nia')->on('alumnos');
});

Schema::create('perfil_academico_alumno', function (Blueprint $table) {
    $table->increments('id');
    $table->bigInteger('alumno_id')->unsigned();
    $table->foreign('alumno_id')->references('nia')->on('alumnos');
    $table->string('escuela_procedencia');
    $table->string('ubicacion_escuela');
    $table->string('respuesta1');
    $table->string('respuesta2');
    $table->string('respuesta3');
    $table->string('respuesta4');
    $table->string('respuesta5');
    $table->string('respuesta6');
    $table->string('respuesta7');
    $table->string('respuesta8');
    $table->string('respuesta9');
    $table->string('respuesta10');
    $table->string('respuesta11');
    $table->string('respuesta12');
    $table->string('respuesta13');
    $table->string('respuesta14');
    $table->string('respuesta15');
    $table->timestamps();
});

Schema::create('atencion_individualizada', function (Blueprint $table) {
    $table->increments('id');
    $table->string('nombre_docente');
    $table->string('grupo');
    $table->string('descripcion');
    $table->date('fecha_aplicacion');
    $table->bigInteger('alumno_id')->unsigned();
    $table->foreign('alumno_id')->references('nia')->on('alumnos');
    $table->string('respuesta1')->nullable();
    $table->string('respuesta2')->nullable();
    $table->string('respuesta3')->nullable();
    $table->string('respuesta4')->nullable();
    $table->string('respuesta5')->nullable();
    $table->string('respuesta6')->nullable();
    $table->string('respuesta7')->nullable();
    $table->string('respuesta8')->nullable();
    $table->string('respuesta9')->nullable();
    $table->string('respuesta10')->nullable();
    $table->string('respuesta11')->nullable();
    $table->string('respuesta12')->nullable();
    $table->string('respuesta13')->nullable();
    $table->string('respuesta14')->nullable();
    $table->string('respuesta15')->nullable();
    $table->string('respuesta16')->nullable();
    $table->string('respuesta17')->nullable();
    $table->string('respuesta18')->nullable();
    $table->string('respuesta19')->nullable();
    $table->string('respuesta20')->nullable();
    $table->string('respuesta21')->nullable();
});

```

Tabla 4.42 Codificación y conexión de base de datos. Iteración 12
Fuente: Elaboración propia

Historia de usuario	
Numero: 13	Usuario: Docente (puesto de tutor)
Nombre de la historia: Generar PDF	
Prioridad en negocio: alta	Riegos en desarrollo: alta
Programador responsable: Bandom Baruch González Cervantes	
Descripción: El docente (puesto de tutor) podrá descargar los resultados de cada encuesta de los alumnos en PDF.	

Tabla 4.43 HU - Generar PDF
Fuente: Elaboración propia

Iteración 13

Hace referencia a la décima tercera historia de usuario, se encarga de generar PDF con información de las encuestas contestadas del alumno y el padre de familia.

Sprint Backlog iteración 13

La tabla 44 muestra las tareas específicas para la iteración 13.

Tarea asignada a: Brandom Baruch González Cervantes				
ID	Tarea	Inicio	Finalización	Estado
1	Asignación de los componentes en la vista del framework.	01/11/2020	01/11/2020	Completado
2	Creación del panel de encuestas del tutor	02/11/2020	10/11/2020	Completado
3	Creación de vistas para generar pdf	11/11/2020	13/11/2020	Completado
4	Pruebas de direccionamiento a otras pantallas.	13/11/2020	14/11/2020	Completado

Tabla 4.44 Tareas específicas iteración 13
Fuente: Elaboración propia



Diseño iteración 13

La figura 62 muestra los alumnos que han iniciado, finalizado y no han realizado las encuestas, la figura 63 muestra los resultados del alumno de la encuesta “Perfil Académico”, para descargar el pdf solo hay que presionar en “descargar PDF” y muestra el PDF con los datos del alumno (figura 64).

Pantallas de alta y baja

En la figura 4.51 muestra los cada una de las encuestas los alumnos que han contestado dicha encuesta. En el caso de la “entrevista fresca” muestra los alumnos y padres de familia. Cada alumno y padre de familia tiene un botón donde indica que si han realizado la encuesta el tutor podrá ver la información (figura 4.52) que proporcione el usuario y podrá descargar la información en un PDF (figura 4.53).


Bienvenidos VASH ESTAMPIDA ▾


2020

ARJO DE
LEONA VICARIO
RENERMERTA MADRE DE LA PATRIA


Alumnos del grupo **Grupo 1 2do A**




ENTREVISTA FRESCA




CUESTIONARIO ANEXOS



TEST SOBRE ESTILOS
DE APRENDIZAJE



TEST PARA DETECTAR TUTORADOS
QUE REQUIERAN ATENCIÓN INDIVIDUAL



PERFIL ACADEMICO

Entrevista Fresca Alumno

Nia	Alumno	Ver datos
16240011	Brandom Baruch González Cervantes	✓
10101010	Elia Cervantes Estrada	✓
16240010	Brenda González Cervantes	Inicio la entrevista
250525	Fernando Barragan Santana	No ha iniciado entrevista
159753	Elias Atonal Carro	No ha iniciado entrevista



Entrevista Fresca Padres

Nia	Alumno	Padre	Ver datos
16240011	Brandom Baruch	Elia Cervantes Estrada	✓
16240010	Brenda	Herman Hoppe Eichmann	No ha iniciado la entrevista
16240010	Brenda	Elia Cervantes Estrada	✓
159753	Elias	Bernadette O'Connell Schimmel	No ha iniciado la entrevista

© 2020, hecho por SGTEMS

Figura 4.51 Pantalla de panel de encuestas. (Pantalla de alta)
Fuente: Elaboración propia

Bienvenidos DIRECTOR ▾

2020

AÑO DE
LEONA VICARIO
BENEMÉRITA MADRE DE LA PATRIA

Perfil Académico

Mariana Cante Ramírez

REGRESAR
DESCARGAR PDF

Test para detectar tutorados que requieran atención individualizada

Escuela de procedencia: Técnica 61 Ubicación de la escuela secundaria de procedencia: En el arenal

N°	Descripción	Respuesta (V/F)
1	Mi promedio de la secundaria es de igual o mayor a 7	V
2	Reprobé 2 o más materias en la secundaria	F
3	Yo elegí el plantel de bachillerato al que asisto	F
4	Asisto a la Media Superior porque creo que me sería útil	V
5	Me gusta la escuela	V
6	Soy bueno para estudiar	V
7	En la secundaria sentía confianza con algún maestro(a) como para platicar con él(ella)	F
8	En la secundaria preguntaba mis dudas al maestro(a)	F
9	En la secundaria me molestaban con frecuencia otro u otros compañeros (burlas, intimidación, golpes, extorsiones, etc.)	V
10	Si yo reprobara alguna materia, se lo platicaría a mi mamá o mi papá	F
11	Si tuviera un problema personal, lo platicaría con mi mamá o mi papá	V
12	Me gustaría emigrar a los Estados Unidos en los próximos 5 años	F
13	Es posible que me case o viva con mi pareja antes de terminar el bachillerato	F
14	En mi casa hemos tenido problemas económicos serios (deudas, no alcanza para los gastos del diario, etc).	V
15	Es más importante trabajar que estudiar	F

REGRESAR
DESCARGAR PDF

© 2020, hecho por SGTEMS

Figura 4.52 Pantalla para ver resultados. (Pantalla de alta)

Fuente: Elaboración propia



**Bachillerato General Profesional Ignacia Islas en San Martín
Texmelucan , San Cristobal Tepatlaxco**



Alumno: Brandom Baruch González Cervantes

Test para detectar tutorados que requieran atención individualizada

Escuela de procedencia:
Colegio de Bachilleres Cobat 11 Panotla

Ubicación de la escuela secundaria de procedencia:
Panotla, Tlaxcala

N°	Descripción	Respuesta (V/F)
1	Mi promedio de la secundaria es de igual o mayor a 7	V
2	Reprobé 2 o más materias en la secundaria	F
3	Yo elegí el plantel de bachillerato al que asisto	V
4	Asisto a la Media Superior porque creo que me sería útil	F
5	Me gusta la escuela	V
6	Soy bueno para estudiar	V
7	En la secundaria sentía confianza con algún maestro(a) como para platicar con él(ella)	F
8	En la secundaria preguntaba mis dudas al maestro(a)	F
9	En la secundaria me molestaban con frecuencia otro u otros compañeros (burlas, intimidación, golpes, extorsiones, etc.)	V
10	Si yo reprobara alguna materia, se lo platicaría a mi mamá o mi papá	F
11	Si tuviera un problema personal, lo platicaría con mi mamá o mi papá	V
12	Me gustaría emigrar a los Estados Unidos en los próximos 5 años	F
13	Es posible que me case o viva con mi pareja antes de terminar el bachillerato	V
14	En mi casa hemos tenido problemas económicos serios (deudas, no alcanza para los gastos del día, etc).	F
15	Es más importante trabajar que estudiar	V

Gracias por compartir esta información usted es lo más importante para nosotros.

Francisco Villa No. 1 Colonia el Barrio en San Cristobal Tepatlaxco, C.P. 74120

Teléfono: 1124085

Figura 4.53 Pantalla del PDF. (Pantalla de alta)

Fuente: Elaboración propia

La tabla 4.45 muestra los métodos para la creación de cada PDF de las encuestas realizadas por cada alumno. El constructor indica el tiempo máximo para que cargue el PDF, si se pasa del límite se ejecutara un error.

Codificación	
28	<code>public function pdf_padre(\$alumno_id)</code>
29	<code>{</code>
30	<code> \$entrevista = Entrevista_Fresca_Padre::where('alumno_id',\$alumno_id)->first();</code>
31	<code> \$pdf = \PDF::loadView('docente.tutorias.pdf.pdf_padre',compact('entrevista'))->setPaper('letter');//->set_time_limit(300);</code>
32	<code> return \$pdf->stream("Entrevista_padre_".\$entrevista->padres[0]->name.'_'. \$entrevista->padres[0]->apellidoP.'_'. \$entrevista->padres[0]->apellidoM.'.pdf');</code>
33	<code>}</code>
34	<code>public function entrevista_fresca_pdf(\$alumno_id)</code>
35	<code>{</code>
36	<code> \$alumno = Alumno::find(\$alumno_id);</code>
37	<code> \$entrevista = Entrevista_Fresca_Alumno::where('alumno_id',\$alumno->nia)->first();</code>
38	<code> \$pdf = \PDF::loadView('docente.tutorias.pdf.Entrevista_fresca_pdf',compact('entrevista','alumno'))->setPaper('letter');</code>
39	<code> return \$pdf->stream("Entrevista_Alumno_". \$alumno->nia.'.pdf');</code>
40	<code>}</code>
41	<code>public function cuestionario_anexos_pdf(\$alumno_id)</code>
42	<code>{</code>
43	<code> \$alumno = Alumno::find(\$alumno_id);</code>
44	<code> \$cuestionario = Cuestionario_Anexos::where('alumno_id',\$alumno->nia)->first();</code>
45	<code> \$pdf = \PDF::loadView('docente.tutorias.pdf.cuestionario_anexos_pdf',compact('cuestionario','alumno'))->setPaper('letter');</code>
46	<code> return \$pdf->stream("Cuestionario_Anexos_Alumno_". \$alumno->nia.'.pdf');</code>
47	<code>}</code>
48	<code>public function estilo_aprendizaje_pdf(\$alumno_id)</code>
49	<code>{</code>
50	<code> \$alumno = Alumno::find(\$alumno_id);</code>
51	<code> \$test = Test::where('alumno_id',\$alumno->nia)->first();</code>
52	<code> \$pdf = \PDF::loadView('docente.tutorias.pdf.estilo_aprendizaje_pdf',compact('test','alumno'))->setPaper('letter');</code>
53	<code> return \$pdf->stream("Estilo_aprendizaje_Alumno_". \$alumno->nia.'.pdf');</code>
54	<code>}</code>
55	<code>public function atencion_individual_pdf(\$alumno_id)</code>
56	<code>{</code>
57	<code> \$alumno = Alumno::find(\$alumno_id);</code>
58	<code> \$test = Atencion_Individualizada::where('alumno_id',\$alumno->nia)->first();</code>
59	<code> \$pdf = \PDF::loadView('docente.tutorias.pdf.atencion_individual_pdf',compact('test','alumno'))->setPaper('letter');</code>
60	<code> return \$pdf->stream("Atención_Individual_Alumno_". \$alumno->nia.'.pdf');</code>
61	<code>}</code>
62	<code>public function perfil_academico_pdf(\$alumno_id)</code>
63	<code>{</code>
64	<code> \$alumno = Alumno::find(\$alumno_id);</code>
65	<code> \$cuestionario = Perfil_Academico_Alumno::where('alumno_id',\$alumno->nia)->first();</code>
66	<code> \$pdf = \PDF::loadView('docente.tutorias.pdf.perfil_academico_pdf',compact('cuestionario','alumno'))->setPaper('letter');</code>
67	<code> return \$pdf->stream("Perfil_Academico_Alumno_". \$alumno->nia.'.pdf');</code>
68	<code>}</code>
69	<code>}</code>
70	<code>}</code>
71	<code>}</code>
72	<code>}</code>
73	<code>}</code>

Tabla 4.45 Codificación y conexión de base de datos. Iteración 13

Fuente: Elaboración propia

4.7. Fase de pruebas

A continuación, se mencionará algunos errores encontrados durante la creación del proyecto.

Primera migración: Este problema surge cuando se realiza la primera migración y muestra el siguiente error (ver figura 4.54).

```
SQLSTATE[42000]: Syntax error or access violation: 1071 Specified key was too long; max key length is 767 bytes (SQL: alter table `users` add unique `users_email_unique`(`email`))
```

Figura 4.54 Error al ejecutar la primera migración

Fuente: <https://ourcodeworld.co/articulos/leer/705/arreglando-la-excepcion-la-clave-especificada-era-un-error-demasiado-largo-en-laravel-54>

Este error aparece en la versión de Laravel 5.4 y en versiones de MySQL inferiores a la 5.7. Para solucionar este error debemos de buscar un archivo llamado **AppServiceProvider.php**. Dentro del archivo en el método **boot** se tiene que agregar lo siguiente: **Schema::defaultStringLength(191)** y agregar la clase **Schema** al inicio: **use Illuminate\Support\Facades\Schema** (ver figura 4.55).

```
AppServiceProvider.php
1 <?php
2
3 namespace App\Providers;
4
5 use Illuminate\Support\ServiceProvider;
6 use Illuminate\Support\Facades\Schema;
7
8 class AppServiceProvider extends ServiceProvider
9 {
10     /**
11      * Bootstrap any application services.
12      *
13      * @return void
14      */
15     public function boot()
16     {
17         Schema::defaultStringLength(191);
18     }
19 }
```

Figura 4.55 Archivo AppServicesProvider.php
Fuente: Elaboración propia.

Relación de modelos: Es un tipo de error cuando no se relaciona de manera adecuada los modelos. Los modelos son las tablas de la base de datos. La figura 4.56 muestra que “no encontró la columna **alumno_nia** en la tabla **parentezco**”.

```
← → ↻ 🏠 ⓘ 127.0.0.1:8000/alumno
ErrorException (E_ERROR)
SQLSTATE[42S22]: Column not
found: 1054 Unknown column
'parentezcos.alumno_nia' in 'field list'
(SQL: select `padre_familias`.*,
`parentezcos`.`alumno_nia` as
`parentezcos_alumno_nia`
from `parentezcos`
join `padre_familias` on `parentezcos`.`alumno_nia` = `padre_familias`.`alumno_nia`)

C:\xampp64\www\tutorias-correcto\vendor\laravel\framework\src\Illuminate\Database\Connection.php
654. // run the SQL against the PDO connection. Then we can calculate the time it
655. // took to execute and log the query SQL, bindings and time in our memory.
656. try {
657.     $result = $callback($query, $bindings);
658. }
659. // If an exception occurs when attempting to run a query, we'll format the error
660. // message to include the bindings with SQL, which will make this exception a
661. // lot more helpful to the developer instead of just the database error.
662.
```

Figura 4.56 Error de relación de modelos
Fuente: Elaboración propia.

La figura 4.57 muestra el método **padres** en el cual tiene la relación entre el **padre de familia** y la tabla intermedia **parentezcos**.

```
public function padres()
{
    return $this->belongsToMany(Padre_familia::class, 'parentezcos');
}
```

Figura 4.57 Método padres
Fuente: Elaboración propia.

Sin embargo, laravel al relacionar tablas tiene una nomenclatura que es primero el nombre de la tabla y después un guion bajo id, por ejemplo, users_id, pero se puede omitir la nomenclatura y colocar el nombre que sea. La figura 4.58 muestra la solución

In addition to customizing the name of the joining table, you may also customize the column names of the keys on the table by passing additional arguments to the method. The third argument is the foreign key name of the model on which you are defining the relationship, while the fourth argument is the foreign key name of the model that you are joining to: `belongsToMany`

```
return $this->belongsToMany('App\Role', 'role_user', 'user_id', 'role_id');
```

Figura 4.58 Relación de muchos a muchos

Fuente: <https://laravel.com/docs/5.5/eloquent-relationships#many-to-many>

Observando la solución de la figura 4.58, el método padres queda de la siguiente manera: pasando como parámetros el modelo padre, la tabla intermedia (parentezcos) y las llaves foráneas de la tabla intermedia (alumno_id y padre_id) (ver figura 4.59).

```
public function padres()
{
    return $this->belongsToMany(Padre_familia::class, 'parentezcos', 'alumno_id', 'padre_id');
}
```

Figura 4.59 Agregando columnas para el método padres

Fuente: Elaboración

La figura 4.60 muestra el resultado de la relación padres y alumnos.

Hola Alumno **Brandom Baruch**



#	Nombre	Parentesco	Opciones
1	Elia Cervantes Estrada	Mamá	

Figura 4.60 Visualización del panel del alumno

Fuente: Elaboración propia.


```
C:\wamp64\www\tutorias-correcto>php artisan migrate:rollback
Rolling back: 2021_04_21_101028_create_b_domicilio_table
Rolled back: 2021_04_21_101028_create_b_domicilio_table
Rolling back: 2021_04_21_100244_create_parentezcos_table
Rolled back: 2021_04_21_100244_create_parentezcos_table
Rolling back: 2021_04_21_094958_create_asignadas_table
Rolled back: 2021_04_21_094958_create_asignadas_table
Rolling back: 2021_04_21_091034_create_puesto_asignados_table
Rolled back: 2021_04_21_091034_create_puesto_asignados_table
```

Figura 4.63 Ejecución del comando php artisan migrate:rollback

Fuente: Elaboración propia.

```
C:\wamp64\www\tutorias-correcto>php artisan migrate
Migrating: 2014_10_12_100000_create_password_resets_table
Migrated: 2014_10_12_100000_create_password_resets_table
Migrating: 2020_10_03_010407_create_docentes_table
Migrated: 2020_10_03_010407_create_docentes_table
Migrating: 2020_10_07_001317_create_padre_familias_table
Migrated: 2020_10_07_001317_create_padre_familias_table
```

Figura 4.64 Ejecución del comando php artisan migrate

Fuente: Elaboración propia.

Se puede observar que la tabla materias fue modificada la columna descripción.

```
mysql> desc materias;
+-----+-----+-----+-----+-----+-----+
| Field      | Type                | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id         | int(10) unsigned   | NO   | PRI | NULL    | auto_increment |
| name      | varchar(191)       | NO   |     | NULL    |                |
| descripcion| varchar(255)       | NO   |     | NULL    |                |
| clave     | varchar(191)       | NO   |     | NULL    |                |
| created_at| timestamp          | YES  |     | NULL    |                |
| updated_at| timestamp          | YES  |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
```

Figura 4.65 Columna "descripción" modificada

Fuente: Elaboración propia.

La segunda opción es crear una regla de validación al crear y editar una materia, utilizando la validación **max** permite especificar una longitud. La figura 4.66 se muestra de la siguiente manera.

```
$rules = [
    'name' => 'required',
    'descripcion' => 'required|max:10',
    'clave' => 'required|unique:materias',
];

$message = [
    'name.required' => 'Debes de colocar un nombre para la materia',
    'descripcion.required' => 'Debes de colocar una descripción',
    'descripcion.max' => 'Solo puedes colocar un max de 10 caracteres',
    'clave.required' => 'Debes de colocar una clave para la materia',
    'clave.unique' => 'Esta clave ya esta en uso en una materia',
];
```

Figura 4.66 Reglas de validación

Fuente: Elaboración propia.

Se crea dos arreglos, el primero arreglo se coloca los nombres de los inputs y después

La figura 4.66 muestra dos arreglos, el primero arreglo coloca los nombres de los inputs y después las reglas a validar, **required** quiere decir que el campo es requerido, **unique** especifica los y no se podrán repetir para la tabla materias. El segundo arreglo describe las validaciones mencionadas. La figura 4.67 muestra la regla **max** funcionando correctamente.

Registrar nueva Materia

Nombre de la materia	Clave de la materia
Tutorías 1	TUTORIAS1
Descripción de la materia	
Ejemplo de la descripción que dará un error por la cantidad de caracteres	
Solo puedes colocar un max de 10 caracteres	

REGISTRAR MATERIA CANCELAR

Figura 4.67 Ejecutando la regla de validación max
Fuente: Elaboración propia.

CAPÍTULO V

RESULTADOS

5.1. Análisis de Tiempo de Captura

El análisis e interpretación de los resultados es la última etapa del proceso de investigación, todas las anteriores, comprendidas en el diseño, concurren hacia la realización de esta importante operación.

Dentro del Bachillerato “Profa. Ignacia Islas” ubicado en San Cristóbal Tepatlaxco, San Martín Texmelucan, Puebla. Cada tutor de grupo invito a los alumnos y padres de familia que contestaran: encuestas y test del programa “Yo no abandono”. Al mismo tiempo el director de la escuela se le invito a registrar alumnos, docentes, padres de familiar. A continuación, se muestra las actividades y el tiempo que tardaron en realizarlo sin utilizar el sistema web (ver tabla 5.1).

No.	Actividades	Tiempo de registro minutos
1	Responder encuesta de cada alumno	50 a 60
2	Contestar entrevistas fresca alumnos	30 a 60
3	Contestar entrevista fresca para el padre de familia	10 a 20
4	Llenar datos generales de alumnos	20
5	Llenar datos generales de docentes	20
6	Llenar datos generales de padres de familia	20
7	Registrar materias	15
8	Llenar domicilio de padres y alumnos	15
9	Registrar grupos	5
10	Registrar puestos	5

Tabla 5.1 Tiempo de registro

Fuente: Elaboración propia.

Actualmente el Bachillerato “Profa. Ignacia Islas” concluyo el ciclo escolar y se pretendía que todos los padres y alumnos realizaran las encuestas y test, sin embargo, solo se cuenta con 20 alumnos de diferentes semestres y 10 padres de familia para realizar las encuestas y test. Para saber los tiempos que tardaron en contestar se implementó un

cronometro, cuando realiza una encuesta o test automáticamente se activa el cronometro. Al finalizar el tutor puede ver cuánto tiempo tardo el alumno o padre de familia en contestar (ver figura 5.1).



Figura 5.1 Tiempo en finalizar la entrevista fresca
Fuente: Elaboración propia.

5.2. Análisis de las actividades

A continuación, se va a describir a detalle de las actividades vistas en la tabla 5.1.

5.2.1. Actividad 1: Responder encuestas de cada alumno

Cada alumno debe de contestar diferentes encuestas y test con el objetivo de identificar si el alumno pueda tener problemas e identificarlo a tiempo. De acuerdo a los datos obtenidos por parte del Bachillerato “Profa. Ignacia Islas” (ver tabla 5.1) cada alumno tarda en contestar las encuestas y test del programa “Yo no abandono” en un rango entre 50 y 60 min cada uno. A continuación, se muestra la comparación en responder encuestas de manera tradicional y en responder encuestas en el sistema web (ver tabla 5.2 y figura 5.2).

No. Alumnos	Tiempo en responder las encuestas y test/minutos	Tiempo en responder las encuestas y test en el sistema web/minutos	Diferencia de tiempo	Porcentaje de tiempo optimizado
1	56	9.41	46.59	83.20%
2	59	17.55	41.45	70.25%
3	60	57.15	2.85	4.75%
4	57	24.19	32.81	57.56%
5	58	18.34	39.66	68.38%
6	55	18	37	67.27%
7	60	55.37	4.63	7.72%
8	57	30.5	26.5	46.49%
9	60	10.41	49.59	82.65%
10	50	45.19	4.81	9.62%
11	52	50	2	3.85%
12	55	29.5	25.5	46.36%
13	55	52.24	2.76	5.02%
14	60	18.4	41.6	69.33%
15	59	40.1	18.9	32.03%
16	58	10.5	47.5	81.90%
17	58	19	39	67.24%
18	57	39.4	17.6	30.88%
19	53	46	7	13.21%
20	50	48.2	1.8	3.60%
Promedio general			24.48	42.57%

Tabla 5.2 Tiempo en responder capturas y test
Fuente: Elaboración propia.

Se puede identificar que el uso del sistema reduce en un promedio de 42.57% el tiempo de contestar las encuestas y test como se muestra en la tabla 5.2 de manera gráfica se puede ver los resultados en la figura 5.2.

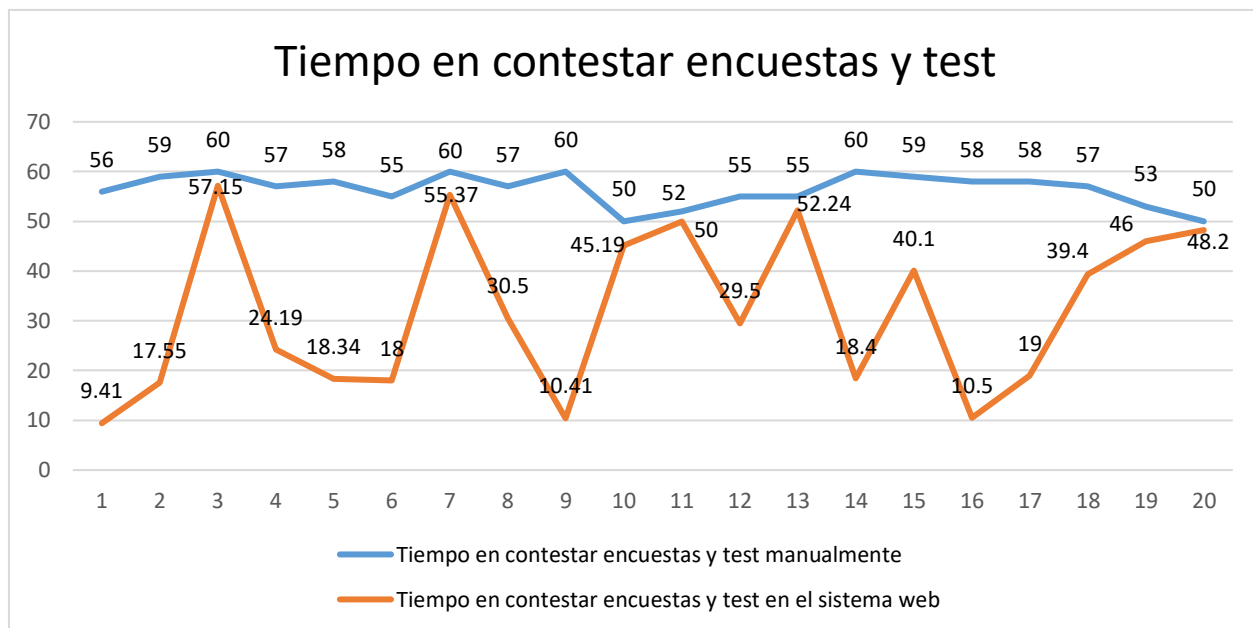


Figura 5.2 Grafica de tiempo en contestar encuestas y test

Fuente: Elaboración propia.

5.2.2. Actividad 2: Llenar entrevistas fresca alumnos

El Bachillerato “Profa. Ignacia Islas” creo una entrevista llamada “entrevista fresca” con el objetivo de conocer al alumno, la entrevista fresca cuenta con varias secciones como:

- Datos familiares: En esta sección habla acerca de su entorno familiar, por ejemplo: con quien vive, como es la relación entre él y sus padres, cuantos hermanos tiene, etc.
- Datos académicos: En sección identifica donde curso la secundaria, que materias fue de su agrado y cómo fue su desempeño durante esa etapa.
- Hábitos de estudio: En esta sección identifica cuanto tiempo brinda para estudiar, realizar tareas, leer, quien le ayuda a estudiar en casa, en qué lugar se siente más cómodo a la hora de estudiar, etc.
- Otras actividades: En esta sección identifica que hace en su tiempo libre, que momentos son felices o tristes para el alumno, etc.
- Datos adicionales: En esta sección habla acerca si padece de alguna enfermedad, si se encuentra en tratamiento médico, si ha probado alguna droga, etc.

Con los datos obtenidos del Bachillerato “Profa. Ignacia Islas” se realizó un análisis para comprobar la eficacia del sistema web en contestar la entrevista fresca para los alumnos (ver tabla 5.3).

No. Alumnos	Tiempo en contestar entrevista fresca/minutos	Tiempo en contestar entrevista fresca en el sistema web/minutos	Diferencia de tiempo	Porcentaje de tiempo optimizado
1	46	3	43	93.48%
2	31	5	26	83.87%
3	35	11	24	68.57%
4	46	31	15	32.61%
5	60	40	20	33.33%
6	59	8	51	86.44%
7	30	6	24	80.00%
8	40	5	35	87.50%
9	35	7	28	80.00%
10	34	12	22	64.71%
11	60	50	10	16.67%
12	53	39	14	26.42%
13	42	10	32	76.19%
14	32	20	12	37.50%
15	37	26	11	29.73%
16	46	13	33	71.74%
17	43	3	40	93.02%
18	60	55	5	8.33%
19	39	26	13	33.33%
20	51	40	11	21.57%
Promedio general			23.45	56.25%

Tabla 5.3 Tiempo en contestar la entrevista fresca alumnos

Fuente: Elaboración propia.

Como se muestra en la tabla 5.3 permite identificar que el uso del sistema reduce en un promedio de 56.25% el tiempo en contestar la entrevista fresca, de manera gráfica se puede ver los resultados en la figura 5.3.

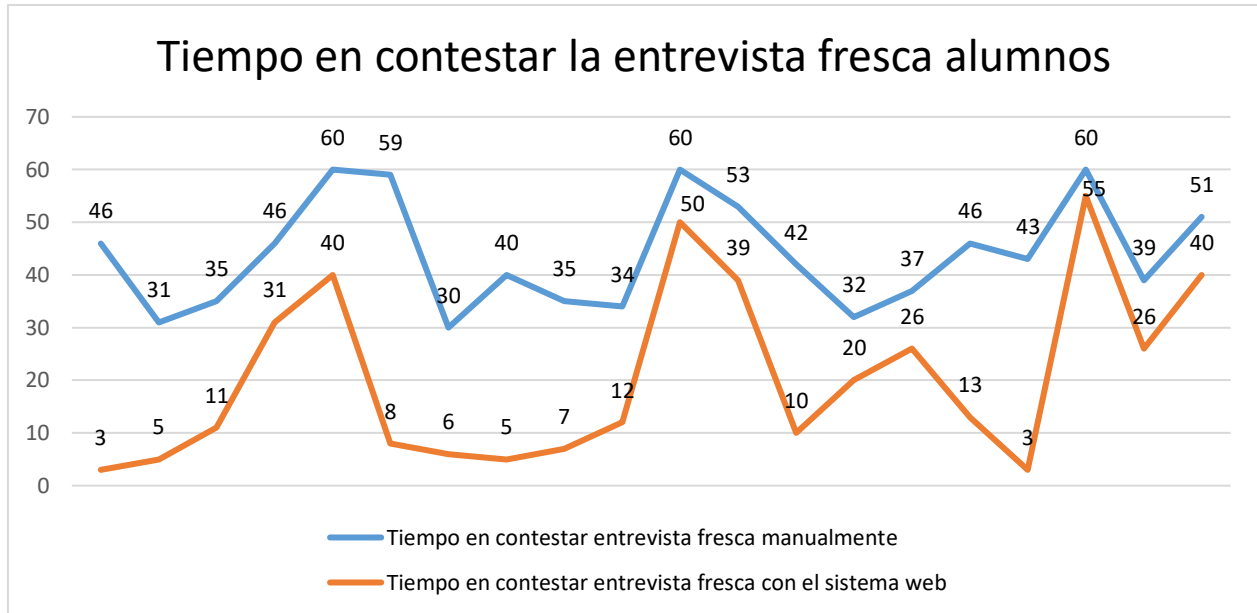


Figura 5.3 Grafica de tiempo en contestar la entrevista fresca alumnos

Fuente: Elaboración propia.

5.2.3. Actividad 3: Contestar entrevista fresca padres

El Bachillerato “Profa. Ignacia Islas” realizó una entrevista fresca para el padre de familia con el objetivo de verificar si existe un vínculo entre padre e hijo/a. El bachillerato brindó información en cuanto tiempo se tarda en realizar la entrevista en un rango de 10 a 20 min aproximadamente cada padre de familia, también, informo que el número máximo de padres que realizaron la entrevista fue de 10 personas. El padre de familia debe de contestar la entrevista fresca dependiendo el número de parentesco que tenga con uno o varios alumnos (si tiene dos alumnos que tiene parentesco, entonces debe de contestar dos veces la entrevista fresca). En la tabla 5.4 muestra el análisis para identificar la eficacia en realizar la entrevista fresca en el sitio web.

No. padres	Tiempo en contestar entrevista fresca/minutos	Tiempo en contestar entrevista fresca en el sistema web/minutos	Diferencia de tiempo	Porcentaje de tiempo optimizado
1	10.40	5.20	5.20	50.00%
2	15.10	11.32	3.78	25.03%
3	12.55	6.42	6.13	48.84%
4	20.48	13.48	7.00	34.18%
5	16.35	8.10	8.25	50.46%
6	18.22	12.30	5.92	32.49%
7	14.50	10.31	4.19	28.90%
8	17.25	11.10	6.15	35.65%
9	11.50	7.54	3.96	34.43%
10	18.39	9.45	8.94	48.61%
Promedio general			5.95	38.86%

Tabla 5.4 Tiempo en contestar entrevista fresca padres

Fuente: Elaboración propia.

Obteniendo los datos necesarios se identifica que el uso del sistema web reduce en un promedio de 38.86% el tiempo de realizar la entrevista fresca para los padres de familia, de manera gráfica se pueden ver los resultados en la figura 5.3.

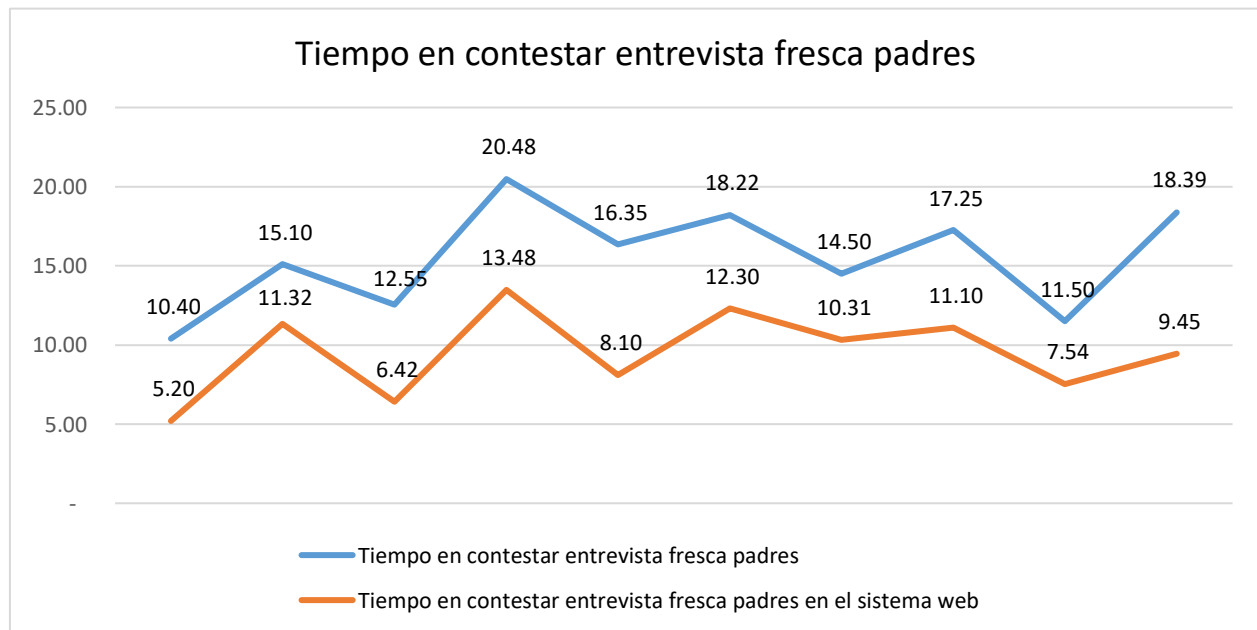


Figura 5.4 Grafica de tiempo en contestar la entrevista fresca padres

Fuente: Elaboración propia.

5.2.4. Actividades del director

Las actividades 4 hasta la 10 son realizadas por el director del bachiller, el director tiene la tarea de administrar y registrar la información de alumnos, padres, docentes, materias, grupos, domicilios y puestos. La información obtenida se registraba en libretas, el problema aparece cuando la información es bastante y es difícil de identificar la información de un alumno, docente, etc. El director proporciono datos acerca del número de registros por cada actividad y cuánto tiempo tarda en registrar los datos mencionados, teniendo en cuenta la información obtenida se realizó un análisis para identificar si utilizando el sistema web puede reducir el tiempo en registrar información (ver tabla 5.5).

Datos a registrar	Número max de registro	Tiempo en registrar/minutos	Tiempo en registrar en el sistema web/minutos	Diferencia de tiempo	Porcentaje de tiempo optimizado
Alumnos	20	20	16.11	3.89	19.45%
Padres de familia	10	20	8.08	11.92	59.6%
Docentes	10	20	6.07	13.93	69.65%
Llenar domicilio de padres y alumnos	10	15	5.37	9.63	64.2%
Materias	10	15	8.28	6.72	44.8%
Grupos	5	5	2.14	2.86	57.2%
puestos	5	5	1.16	3.84	76.8%
Promedio general				7.54	55.96%

Tabla 5.5 Tiempo de captura de datos

Fuente: Elaboración propia.

En la tabla 5.5 se muestra que utilizando el sitio web puede reducir en un 55.96% en registrar alumnos, docentes, materias, llenar domicilio de padres y alumnos, materias, grupos y puestos. De manera gráfica se puede ver en la figura 5.5.

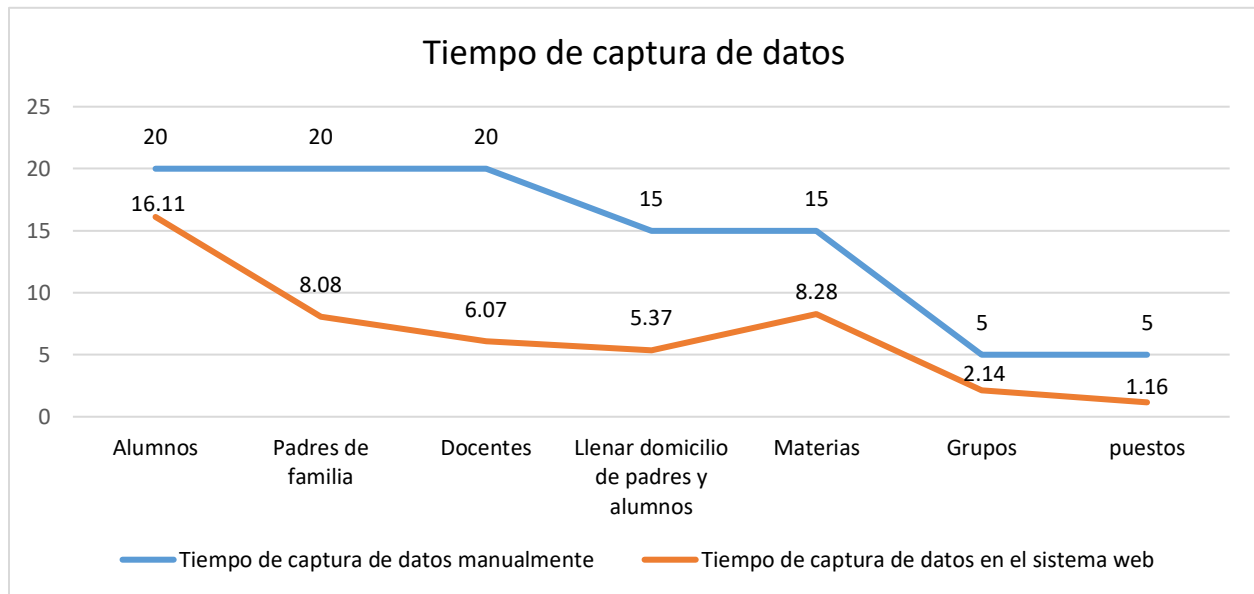


Figura 5.5 Grafica de tiempo de captura de datos

Fuente: Elaboración propia.

5.3. Evaluación del sistema web

El uso del sistema web determino la eficacia en optimizar el tiempo en registrar y administrar la información de alumnos, docentes, materias, etc. Por otra parte, en contestar la “entrevista fresca” y las encuestas y test del programa “Yo no abandono” se reflejó una mejoría en cuestión del tiempo. El tutor de grupo puede saber que alumnos y padres de familia han realizado o no las encuestas, test y “entrevista fresca” para brindar apoyo quien lo requiera (para el caso del alumno).

En la tabla 5.6 se muestra un rubrica para evaluar el sistema si cumplió la expectativa del Bachillerato “Profa. Ignacia Islas”, la evaluación se realizó únicamente el director Ismael Hernández Salinas, debido a que solamente contamos contacto con él, sin embargo, el director dio observaciones positivas y algunas mejoras que se pueden realizar al sistema web, por ejemplo: crear una gráfica general y ver por cada pregunta cuantos alumnos contestaron cierta pregunta y tener un panorama mejor para brindar un mejor apoyo al alumno y otra observación es acerca de las interfaces que en algunos botones abrían nuevas ventanas y los docentes que llegaron a utilizarlo se confundían debido el número de pestañas abiertas.

Preguntas	SI	NO	TAL VEZ
¿Con el uso del sistema web considera que podría mejorar la eficiencia de saber que estudiante ha realizado o no las encuestas?	x		
¿Es fácil el registro de información al sitio web?	x		
¿El método de recuperación de contraseña es fácil?	x		
¿Son agradables las interfaces para usted?	x		
¿La interacción de las interfaces la cree correcta?			x
¿Cree correctamente la tipografía de las interfaces?	x		
¿El formato del PDF lo cree correcto?	x		
¿El uso del sistema web lo considera que es intuitivo y fácil de utilizar?			x
¿Considera que la forma de buscar un alumno o padre de familia por el nombre es apropiada?	x		
¿Al asignar un puesto a un docente, lo cree apropiado?	x		
¿La autenticación para el docente es correcto?	x		
¿Es adecuado los roles puestos para realizar ciertas actividades?	x		
¿Es fácil contestar las encuestas?			x
¿Considera que reduciría el tiempo en registrar la información?	x		
¿Con el uso del sitio web considera que podría mejorar la eficiencia al elaborar los reportes de tutorías?	x		
¿Mediante el sitio web cree que se reduciría el tiempo para obtener los resultados de las encuestas?	x		
¿Al utilizar el sitio web, cree que los tiempos de búsqueda de información se reduzcan?	x		

Tabla 5.6 Rubrica para evaluar el sistema web

Fuente: Elaboración propia.

Conclusiones

El objetivo general del proyecto consistió en administrar la información obtenida de alumnos, docentes, padres de familia, materias, grupos, domicilios, puestos y apoyar la labor tutorial dando seguimiento a la “entrevista fresca” por parte del Bachillerato “Profa. Ignacia Islas” y el programa “Yo no abandono” para reducir las deserciones escolares y el uso del papel.

Durante el desarrollo del sistema se aprendió en utilizar frameworks como Laravel para el back-end y Bootstrap para el front-end, GitHub se utilizó para tener un repositorio remoto y compartir el proyecto con el asesor para que viera los avances obtenidos. Gracias al asesor se pudo subir el proyecto a un servidor que fue DigitalOcean para mostrar los avances al director para que diera sus observaciones.

En el desarrollo del proyecto se comprobó la hipótesis a través de pruebas realizadas por alumnos, padres y el director teniendo resultados la comparación de tiempos en contestar encuestas, entrevista, test y tener una mejor administración de datos del bachillerato.

Recomendaciones para el uso del sistema

A continuación, se recomienda algunos navegadores para el óptimo rendimiento del sistema:

- Google Chrome versión 91.0.4472.124 versión resiente.
- Microsoft Edge versión 91.0.864.67 versión resiente.

Trabajos futuros

Durante el desarrollo del proyecto se han identificado algunas mejoras para el sistema tomando en cuenta las observaciones del bachillerato y se esperan en abordar en el futuro. A continuación, los trabajos futuros para mejorar el uso del sistema web:

- Generar graficas para la entrevista fresca y encuestas del programa “Yo no abandono”: El tutor de grupo menciona que sería de utilidad crear graficas para tener un análisis general de los alumnos e identificar alumnos que necesiten apoyo.
- Mostrar los resultados de la entrevista, cuestionarios y test a los alumnos: El director comento acerca que un alumno le gustaría un apartado para ver sus respuestas y poder editarlas fuera el caso. Para que el tutor tenga un mejor análisis y dar seguimiento.

Referencias bibliográficas

1. Antonio Javier Gallego. (2020). Estructura de un proyecto. 6/05/2021, de Gitbook Sitio web: https://ajgallego.gitbook.io/laravel-5/capitulo_1/capitulo_1_funcionamiento_basico#estructura-de-un-proyecto
2. Antonio Javier Gallego. (2020). MVC: Modelo - Vista – Controlador 6/05/2021, de Gitbook Sitio web: <https://ajgallego.gitbook.io/laravel-5/introduccion#mvc-modelo-vista-controlador>
3. Antonio Javier Gallego. (2020). ¿Qué es Laravel? 6/05/2021, de Gitbook Sitio web: <https://ajgallego.gitbook.io/laravel-5/introduccion>
4. Álvaro Fontela. (14/10/2020). Qué es Bootstrap y cómo usarlo. 07/05/2021, de Raiola networks Sitio web: https://raiolanetworks.es/blog/bootstrap/#que_es_bootstrap
5. Alberth Bompert. (20/01/2018). Sublime Text 3 completo. 08/05/2021, de AficionWeb Sitio web: <https://aficionweb.wordpress.com/2018/01/20/sublime-text-full/>
6. Anna Pérez. (27/11/2014). La implementación del método Scrum. 08/05/2021, de OBS Business School Sitio web: <https://www.obsbusiness.school/blog/la-implementacion-del-metodo-scrum>
7. Encarna Abellán M. (2020). Metodología Scrum: qué es y cómo funciona. 2020, de we are marketing Sitio web: <https://www.wearemarketing.com/es/blog/metodologia-scrum-que-es-y-como-funciona.html>
8. EquiposyTalento. (2018). Los 7 beneficios que tiene implantar una metodología ágil. 2020, de equiposytalento.com Sitio web: <https://www.equipsytalento.com/noticias/2018/12/13/los-7-beneficios-que-tiene-implantar-una-metodologia-agile#:~:text=Permite%20dimensionar%20mejor%20los%20proyectos,priorizaci%20n%20y%20toma%20de%20decisiones.>

9. Felipe Rodríguez Ramírez, Irene García Ortega, Saira Antonieta Vásquez Gamboa. (2016). Sistema web: herramienta para la gestión del programa de tutorías. 12/06/2021, de Anfei Digital Sitio web: <https://anfei.mx/revista/index.php/revista/article/view/314>
10. Fernando E. Sánchez Martínez, Ricardo A. Barrera Cámara, Ana Canepa Sáenz. (enero de 2020). Estudio evaluativo del Sistema Institucional de Tutorías para el seguimiento de trayectorias escolares. 21/02/2021, de Scielo Sitio web: http://www.scielo.org.mx/scielo.php?pid=S2007-74672020000100113&script=sci_arttext
11. Flor del Carmen Montes Rodríguez, Guadalupe Minerva Castillo, Sandra Nelly Martínez Cantú. (2017). La tutoría a través de una plataforma web. 13/06/2021, de Escuela Normal "Miguel F. Martínez" Centenaria y Benemérita Sitio web: <http://www.conisen.mx/memorias/memorias/1/C180117-H063.docx.pdf>
12. Francisco Miranda López. (16 agosto 2018). Abandono escolar en educación media superior: conocimiento y aportaciones de política pública. 21/02/2021, de Sinética. Revista electrónica de educación Sitio web: <https://www.redalyc.org/jatsRepo/998/99859321010/html/index.html>.
13. Google Cloud. (16/12/2020). Enviar correo electrónico con Mailgun. 07/05/2021, de Google Cloud Sitio web: <https://cloud.google.com/compute/docs/tutorials/sending-mail/using-mailgun?hl=es-419#:~:text=Mailgun%20es%20un%20servicio%20de,correo%20electr%C3%B3nico%20gratuitos%20por%20mes>.
14. Hamid Jabbari Ilkhechi. (01/02/2017). DigitalOcean. 07/05/2021, de Prezi Sitio web: <https://prezi.com/b5tb5tnw0g79/digitalocean/>
15. Irene Cañete. (2018). Qué es un product backlog y cuál es su función. 08/05/2021, de BBVA Sitio web: <https://www.bbva.com/es/que-es-un-product-backlog-y-cual-es-su-funcion/>

16. Ivonne Rodríguez Pérez. (Julio de 2017). La calidad de la educación superior y la reestructuración del programa de tutoría. 21/02/2021, de Scielo Sitio web: http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S2007-74672017000200135
17. José Miguel Loaiza Granda. (2019). Implementación de la metodología scrum, en el desarrollo del software de la dirección nacional de comunicaciones, en la sección de desarrollo de software de la policía nacional del ecuador. 08/05/2021, de Universidad Tecnológica Israel Sitio web: <http://repositorio.uisrael.edu.ec/bitstream/47000/2164/1/UISRAEL-EC-SIS-378.242-2019-047.pdf>
18. Juan Diego Pérez Jiménez. (20/01/2019). Qué es HTML5. 07/05/2021, de OpenWebinars Sitio web: <https://openwebinars.net/blog/que-es-html5/>
19. Josseph Muñoz Pte. (23/10/2017). WampServer. 07/05/2021, de Prezi Sitio web: <https://prezi.com/zvlhjdyxtzb/wampserver/>
20. Luigy. (22/07/2016). SQL Power Architect herramienta de modelado de datos. 07/05/2021, de Tu informática fácil Sitio web: <https://www.tuinformaticafacil.com/desarrollo-programacion/herramientas/sql-power-architect-herramienta-de-modelado-de-datos>
21. María Jesenia Pachay López, María Rodríguez Gámez. (2020). La deserción escolar: Una perspectiva compleja en tiempos de pandemia. 21/02/2021, de Polo del Conocimiento Sitio web: <https://polodelconocimiento.com/ojs/index.php/es/article/view/2129>
22. MartechForum. (s.f). Balsamiq. 07/05/2021, de MartechForum Sitio web: <https://www.martechforum.com/herramienta/balsamiq-2/>
23. Miguel Ángel Álvarez. (02/11/2020). Generar PDF en Laravel con DomPDF. 07/05/2021, de Desarrolloweb Sitio web: <https://desarrolloweb.com/articulos/generar-pdf-laravel-dompdf.html#:~:text=La%20librer%C3%ADa%20DomPDF%20es%20una,una%20cadena%20de%20texto%2C%20etc.>

24. Miriam Zulma Sánchez, Yaneth Vega Flores, Christian Nivardy Marín, J. Omar Hernández Esquivel. (01/05/2017). Diseño e Implementación de una Plataforma Digital para la Gestión de Tutorías y su Impacto en la Deserción de Estudiantes de Nivel Superior. 13/06/2021, de Redalyc Sitio web: <https://www.redalyc.org/jatsRepo/5122/512253717008/512253717008.pdf>.
25. Norma Mendoza Molina, Patrick Hernández Cuamatzi, Carolina Rocío Sánchez Pérez, Marva Angélica Mora Lumbreras, María del Rocío Ochoa Montiel y Aydee Rojas Escobar. (14 de julio de 2016). Sistema de Automatización para el Programa “Yo No Abandono” del CBTIS No. 61. 21/02/2021, de Revista Iztatl computación Sitio web: <https://ingenieria.uatx.mx/docs/RevistaIztatlComputacionNo10.pdf#page=48>
26. PcHardwarePro. (s.f). Cmder es un emulador de consola para sistemas Windows. 08/05/2021, de PcHardwarePro Sitio web: <https://www.pchardwarepro.com/cmdr-es-un-emulador-de-consola-para-sistemas-windows/#:~:text=CMDER%20es%20un%20emulador%20de%20consola%20por%20t%C3%A1til%20y%20gratuito%20para,m%C3%A1s%20caracter%C3%ADsticas%20y%20experiencia%20intuitiva>.
27. Rocío González. (09/04/2021). Programador PHP: un maestro inigualable en análisis de código y desarrollo web. 07/05/2021, de Crehana Sitio web: <https://www.crehana.com/mx/blog/tech/programador-php/>
28. Secretaria de Educación Media Superior. (15/03/2017). Yo No Abandono. 08/05/2021, de Secretaria de Educación Media Superior Sitio web: http://www.sems.gob.mx/swb/sems/yo_no_abandono
29. Yair. (23/12/2019). Qué es Composer y cómo usarlo. 07/05/2021, de Styde Sitio web: <https://styde.net/que-es-composer-y-como-usarlo/>
30. Yúbal Fernández. (30/10/2019). Qué es Github y qué es lo que les ofrece a los desarrolladores. 08/05/2021, de Xataka Sitio web: <https://www.xataka.com/basics/que-github-que-que-le-ofrece-a-desarrolladores>

31. Xavier Albaladejo. (s.f). Reunión diaria de sincronización del equipo (Scrum daily meeting). 08/05/2021, de Proyectos Agiles Sitio web: <https://proyectosagiles.org/reunion-diaria-de-sincronizacion-scrum-daily-meeting/>

ANEXO A1 MANUAL DE USUARIO

ANEXO A2 MANUAL TÉCNICO

GLOSARIO

B

Base de datos: Es una colección de información organizada de forma que un programa de ordenador pueda seleccionar rápidamente los fragmentos de datos que necesite.

D

Dirección General del Bachillerato (DGB): Es una unidad administrativa de la Subsecretaría de Educación Media Superior (SEMS), se encarga de coordinar la educación que se imparte en el Bachillerato General en los aspectos técnicos y pedagógicos.

E

Educación Media Superior (EMS): Es un espacio para formar personas con conocimientos y habilidades que les permitan desarrollarse en sus estudios superiores o en el trabajo y en la vida.

Editor de código: Es un editor de texto diseñado específicamente para editar el código fuente de programas informáticos.

F

Framework: Es un esquema o estructura que se establece y que se aprovecha para desarrollar y organizar un software determinado.

H

Hardware: Conjunto de elementos físicos o materiales que constituyen una computadora o un sistema informático.

Historia de usuario: Son utilizadas en las metodologías de desarrollo ágiles para la especificación de requerimientos, para que puedan ser entendidos por todos (clientes, desarrolladores y usuarios), representando los requerimientos que debe de cumplir el sistema.

HTML (Hypertext Markup Language): Se utiliza para el desarrollo y creación de páginas web. Se compone de una serie de etiquetas que el navegador interpreta.

I

Implementar: Poner en funcionamiento o llevar a cabo una cosa determinada.

Interfaz: Conjunto de elementos de la pantalla que permiten al usuario realizar ciertas acciones.

Infraestructura como servicio (IaaS): Es un tipo de servicio de informática en la nube que ofrece recursos esenciales de proceso, almacenamiento y redes a petición que son de pago por uso.

L

Lenguaje de programación: Es una herramienta que permite desarrollar software o programas para computadora.

M

Manual técnico: Documento que contiene información válida y clasificada sobre la estructura y modo de operar un aparato.

Modelado: El uso de modelos ayuda al ingeniero de software a “visualizar” el sistema a construir.

Manual de usuario: Es una publicación que incluye los aspectos fundamentales, se trata de una guía que ayuda a entender el funcionamiento del sistema.

Migraciones: En Laravel son una herramienta que permite crear una especie de sistema de control de versiones de bases de datos.

S

Secretaría de Educación Pública (SEP): Se encarga de la vigilancia, operación e implementación de todos los programas relacionados con la educación e instrucción pública aplicados en el ámbito federal.

Sistema gestor de base de datos: Es un conjunto de programas no visibles que administran y gestionan la información que contiene una base de datos.

SQL: Es un lenguaje de dominio específico utilizado en programación, diseñado para administrar y recuperar información de sistemas de gestión de bases de datos.

U

Usuario: Es una persona que utiliza una computadora o un servicio de red.