

UNITRABAJO: diseño e implementación de una base de datos para gestionar ofertas laborales estudiantiles en la Facultad de Ingeniería

UNITRABAJO: Design and Implementation of a Database for Managing Student Employment Opportunities in the Faculty of Engineering

  Amambal Coronado, José Martin ¹   Becerra Tafur, Jorge Alberto ¹
  León Valdivia, Emely Bright ¹   Mendoza Condor, José Esnayder ¹
  Morales Cerna, Jimmy Jhair ¹   Suarez Pretell, Gabriel Joe Luis ¹

¹ Escuela de Ingeniería Informática, Universidad Nacional de Trujillo, La Libertad, Perú.

*Autor correspondiente: bleonv@unitru.edu.pe (E-LeonValdivia)

RESUMEN

El objetivo del estudio fue diseñar e implementar UNITRABAJO, una base de datos transaccional para gestionar las ofertas laborales, postulaciones y contrataciones de estudiantes de la Facultad de Ingeniería. El alcance incluyó el modelamiento de los procesos de la Oficina de Trabajo y el registro de estudiantes, carreras, empresas, instituciones, dependencias, llamados laborales, inscripciones, resultados y contrataciones, además de la consulta, actualización y eliminación de estos datos. La metodología comprendió el diseño del diagrama entidad-relación, la normalización hasta la Tercera Forma Normal y la Forma Normal de Boyce-Codd, y la construcción del modelo relacional. El modelo físico se implementó en MySQL mediante claves primarias, claves foráneas y restricciones de integridad. También se formularon operaciones de álgebra y cálculo relacional, consultas SQL e interfaces gráficas en VBA Excel conectadas a MySQL mediante ODBC. Como resultado, se obtuvo una base de datos funcional, estructurada y consistente, capaz de registrar, consultar, actualizar y eliminar información; identificar llamados sin postulantes y estudiantes no contratados; y generar información sobre oportunidades laborales y contrataciones. Se concluye que UNITRABAJO reduce la redundancia, previene anomalías, centraliza la información y mejora el seguimiento de los procesos de inserción laboral estudiantil.

Palabras clave: base de datos transaccional, ofertas laborales, postulaciones estudiantiles, MySQL, gestión de información.

ABSTRACT

The objective of this study was to design and implement UNITRABAJO, a transactional database for managing job offers, applications, and hiring records of students from the Faculty of Engineering. The scope included modeling the processes of the Employment Office and recording students, academic programs, companies, institutions, departments, job postings, applications, results, and hiring records, as well as retrieving, updating, and deleting these data. The methodology comprised the design of the entity-relationship diagram, normalization to Third Normal Form and Boyce-Codd Normal Form, and construction of the relational model. The physical model was implemented in

MySQL using primary keys, foreign keys, and integrity constraints. Relational algebra and relational calculus operations, SQL queries, and graphical interfaces developed in Excel VBA and connected to MySQL through ODBC were also implemented. The resulting database was functional, structured, and consistent; it supported data creation, retrieval, updating, and deletion, identified job postings with no applicants and students who were not hired, and generated information on employment opportunities and hiring. UNITRABAJO therefore reduces redundancy, prevents data anomalies, centralizes information, and improves the monitoring of student employment placement processes.

Keywords: transactional database, job offers, student applications, MySQL, information management.

1. INTRODUCCIÓN

La gestión de oportunidades laborales en las instituciones universitarias exige integrar información de estudiantes, empresas, instituciones y dependencias académicas. El proceso incluye la publicación de convocatorias, el registro de postulantes, la evaluación de candidaturas, la identificación de llamados declarados desiertos y el seguimiento de los estudiantes contratados. Cuando esta información se administra en archivos o estructuras no integradas, se generan duplicidades, inconsistencias y dificultades para consultar el historial de postulaciones y contrataciones.

Las bases de datos relacionales permiten almacenar, vincular y recuperar esta información de forma estructurada [1], [2], [3]. Su diseño parte de la identificación de las entidades, los atributos y las relaciones que representan el dominio analizado [4], [5], [6], conforme a los fundamentos del modelo relacional propuesto por Codd [7]. Posteriormente, la normalización distribuye los datos en tablas relacionadas, reduce la redundancia y evita anomalías de inserción, actualización y eliminación [8], [9].

La implementación requiere un sistema gestor que permita definir tablas, tipos de datos, claves, relaciones, restricciones de integridad y consultas. La creación y ejecución de sentencias SQL se basa en el estándar ISO/IEC 9075 [10], [11]. MySQL ofrece los mecanismos necesarios para implementar modelos relacionales y procesar operaciones transaccionales [12]. Complementariamente, Visual Basic for Applications (VBA) permite desarrollar formularios para la interacción con los datos [13], mientras que Open Database Connectivity (ODBC) establece la comunicación entre aplicaciones y sistemas gestores [14]. Por ello, la integración de MySQL, VBA Excel y ODBC constituye una alternativa accesible para registrar, consultar y actualizar información.

El problema estudiado se ubica en la Oficina de Trabajo de una Facultad de Ingeniería, que recibe ofertas laborales de empresas e instituciones y publica llamados dirigidos a los estudiantes. Durante este proceso se registran datos de estudiantes, carreras, entidades ofertantes, dependencias, ofertas, periodos de inscripción, currículos, resultados y contrataciones. También se deben identificar los llamados declarados

desiertos, ya sea por ausencia de postulantes o porque ninguno cumple el perfil requerido. La cantidad de entidades y operaciones involucradas evidencia la necesidad de centralizar la información en una estructura que permita consultarla y darle seguimiento.

Como respuesta, se desarrolló UNITRABAJO, una base de datos transaccional para gestionar ofertas laborales, postulaciones y contrataciones de estudiantes de la Facultad de Ingeniería. El proyecto incluyó el modelamiento de los procesos de la Oficina de Trabajo, el diseño del diagrama entidad-relación, la normalización hasta la Tercera Forma Normal y la Forma Normal de Boyce-Codd, la construcción del modelo relacional y su implementación física en MySQL. Además, se formularon operaciones de álgebra y cálculo relacional, se ejecutaron consultas SQL y se desarrollaron interfaces gráficas en VBA Excel conectadas mediante ODBC.

El objetivo general fue diseñar e implementar una base de datos transaccional para gestionar las ofertas laborales, postulaciones y contrataciones de los estudiantes de la Facultad de Ingeniería. Los objetivos específicos fueron modelar los procesos de la Oficina de Trabajo; diseñar el modelo entidad-relación; normalizar los datos hasta la Forma Normal de Boyce-Codd; implementar el modelo físico en MySQL; formular consultas mediante álgebra y cálculo relacional; desarrollar interfaces en VBA Excel; y facilitar la consulta de ofertas laborales y postulaciones.

2. MATERIALES Y MÉTODO

2.1 Materiales

Para el diseño e implementación de UNITRABAJO se emplearon recursos de hardware y software destinados al modelamiento, almacenamiento, programación y acceso a los datos. El diagrama entidad-relación se elaboró en DIA [15]. El modelo relacional y la administración de la base de datos se realizaron en MySQL Workbench [16], mientras que el modelo físico se implementó en MySQL mediante SQL. La interfaz gráfica se desarrolló en Microsoft Excel con VBA [17], y la conexión con MySQL se configuró mediante ODBC, la biblioteca ADODB [18], [19] y el controlador MySQL Connector/ODBC [20].

TABLA I
MATERIALES DEL ENTORNO DE DESARROLLO DE UNITRABAJO

Categoría	Componente	Propósito
Hardware y sistema operativo	Equipo de cómputo con Windows	Ejecutar las herramientas de desarrollo
Modelamiento conceptual	DIA	Elaborar el diagrama entidad-relación
Base de datos	MySQL y MySQL Workbench	Elaborar el modelo relacional e implementar la base de datos
Interfaz	VBA Excel	Desarrollar los formularios
Conectividad	MySQL Connector/ODBC y ADODB	Comunicar VBA Excel con MySQL

2.2 Método

El desarrollo de UNITRABAJO se ejecutó en siete etapas: diseño conceptual, normalización, elaboración del modelo relacional, implementación física en MySQL, formulación de consultas, desarrollo de la interfaz gráfica y configuración de la conexión mediante ODBC.

2.2.1 Diseño del modelo conceptual

El modelo conceptual se elaboró en DIA a partir de tres procesos de la Oficina de Trabajo: publicación de llamados laborales, inscripción de estudiantes y registro de contrataciones. A partir de estos procesos se identificaron las entidades, los atributos, las relaciones y las cardinalidades requeridas para representar la información gestionada [4], [5], [6], [7].

La Figura 1 presenta el diagrama entidad-relación de UNITRABAJO. El modelo distingue los llamados ofertados por empresas de los gestionados por dependencias institucionales.

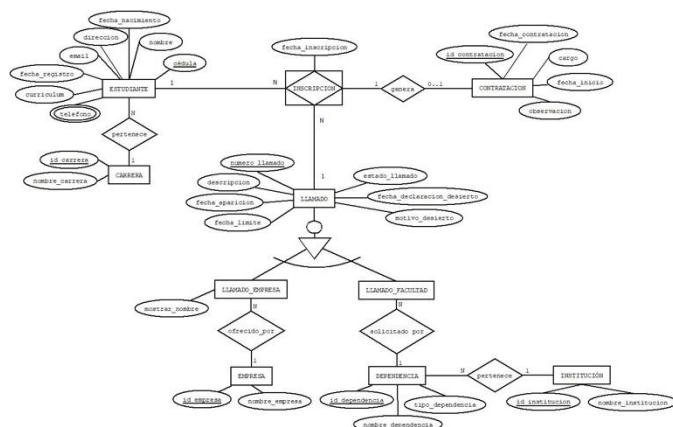


Fig. 1. Diagrama entidad-relación del sistema UNITRABAJO, elaborado en DIA.

2.2.2 Normalización

La normalización partió de una relación general denominada OFICINA_TRABAJO, que agrupó inicialmente los atributos de estudiantes, carreras,

teléfonos, llamados laborales, empresas, instituciones, dependencias, inscripciones y contrataciones.

A esta relación se le aplicaron la Primera Forma Normal, la Segunda Forma Normal, la Tercera Forma Normal y la Forma Normal de Boyce-Codd [1], [2], [4], [5], [8], [9]. Como resultado, los atributos se distribuyeron en relaciones independientes y se definieron las claves necesarias para mantener la integridad y consistencia de los datos. La Tabla II resume cada etapa.

TABLA II
PROCESO DE NORMALIZACIÓN DE LA BASE DE DATOS UNITRABAJO

Etapas	Criterio aplicado	Aplicación en UNITRABAJO
Relación inicial	Agrupación de todos los atributos.	Se integraron inicialmente datos de estudiantes, llamados, empresas, instituciones, inscripciones y contrataciones en OFICINA TRABAJO.
Primera Forma Normal	Eliminación de grupos repetidos y valores no atómicos.	Los números telefónicos fueron separados para permitir el registro de uno o más teléfonos por estudiante.
Segunda Forma Normal	Eliminación de dependencias parciales.	Los atributos dependientes únicamente del estudiante o del llamado fueron separados de las relaciones con claves compuestas.
Tercera Forma Normal	Eliminación de dependencias transitivas.	Se crearon tablas independientes para carreras, empresas, instituciones y dependencias.
Forma Normal de Boyce-Codd	Verificación de determinantes.	Se comprobó que los determinantes de cada relación correspondieran a claves candidatas.

2.2.3 Elaboración del modelo relacional

Después de la normalización, cada entidad se transformó en una tabla y las relaciones se implementaron mediante claves primarias y foráneas. El modelo, elaborado en MySQL Workbench, incorporó tablas asociativas para resolver las relaciones de muchos a muchos [5], [6], [16]. La Figura 2 presenta la estructura relacional de UNITRABAJO.

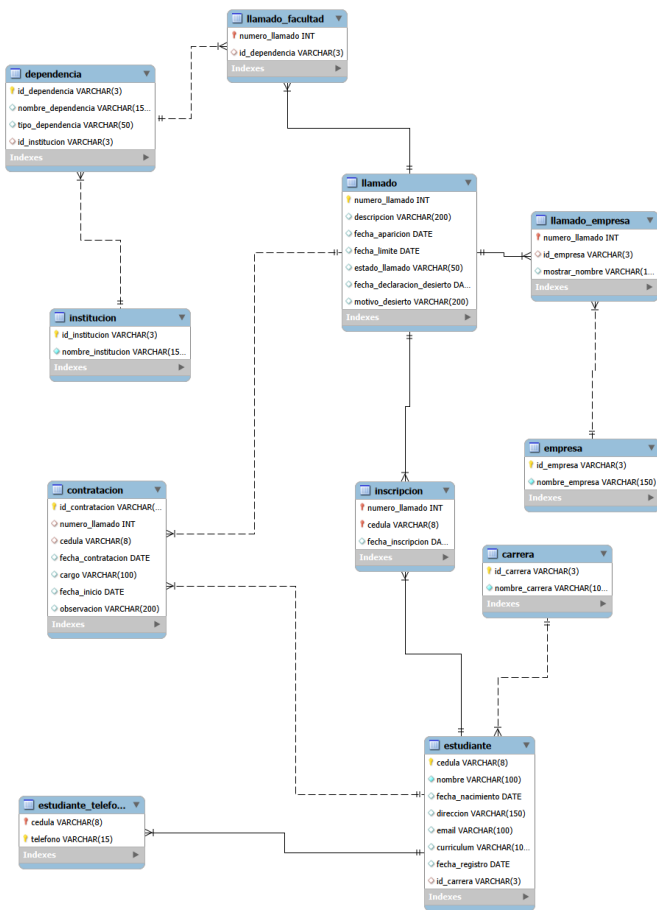


Fig. 2. Modelo relacional de la base de datos UNITRABAJO, elaborado en MySQL Workbench.

2.2.4 Implementación del modelo físico

El modelo relacional se implementó en MySQL mediante sentencias SQL. Para cada tabla se definieron los tipos de datos, las claves primarias, las claves foráneas y las restricciones de integridad requeridas [10], [11], [12]. La Tabla III identifica las tablas creadas y la función que cumple cada una en UNITRABAJO.

TABLA III
TABLAS IMPLEMENTADAS EN LA BASE DE DATOS UNITRABAJO

Tabla	Función dentro de la base de datos
CARRERA	Almacena las carreras profesionales de los estudiantes
ESTUDIANTE	Registra los datos personales y académicos de los estudiantes
ESTUDIANTE_TELEFONO	Permite registrar uno o más teléfonos por estudiante
EMPRESA	Almacena la información de las empresas que ofrecen oportunidades laborales
INSTITUCION	Registra las instituciones vinculadas con la Facultad
DEPENDENCIA	Almacena las oficinas o dependencias pertenecientes a una institución

Tabla	Función dentro de la base de datos
LLAMADO	Registra la descripción, fechas y estado de las convocatorias laborales
LLAMADO_EMPRESA	Relaciona un llamado con una empresa
LLAMADO_FACULTAD	Relaciona un llamado con una dependencia institucional
INSCRIPCION	Relaciona a los estudiantes con los llamados en los que participan
CONTRATACION	Registra a los estudiantes seleccionados o contratados
RESULTADO_POSTULACION	Almacena el resultado obtenido por cada postulante
HABILIDAD	Registra las habilidades consideradas en los llamados
LLAMADO_HABILIDAD	Relaciona las habilidades requeridas con un llamado
LLAMADO_CARRERA	Relaciona las carreras solicitadas con un llamado

2.2.5 Formulación de operaciones relacionales y consultas SQL

Las necesidades de información se expresaron primero mediante operaciones de álgebra y cálculo relacional. Luego, estas expresiones se tradujeron a consultas SQL y se ejecutaron en MySQL mediante operaciones de selección, proyección, combinación y exclusión [3], [4], [10], [11], [12].

Las consultas se diseñaron para obtener cuatro conjuntos de información: estudiantes inscritos en llamados ofertados por empresas; postulantes vinculados con llamados declarados desiertos; estudiantes inscritos sin contratación registrada; y estudiantes seleccionados en oportunidades provenientes de empresas o dependencias institucionales.

2.2.6 Desarrollo de la interfaz gráfica

La interfaz gráfica de UNITRABAJO se desarrolló en Microsoft Excel mediante VBA [13], [17]. El menú principal dio acceso a los módulos de ofertas laborales, llamados, estudiantes, empresas, dependencias, postulaciones, resultados y reportes. Cada formulario incorporó controles para ingresar, seleccionar, consultar, actualizar y eliminar los datos almacenados.

La Tabla IV describe los módulos implementados, la información administrada y las operaciones disponibles.

TABLA IV
MÓDULOS DESARROLLADOS PARA LA INTERFAZ GRÁFICA DE UNITRABAJO

Módulo	Información o función administrada	Operaciones principales
Menú principal	Acceso a los módulos y presentación del resumen general	Navegar y visualizar indicadores

Módulo	Información o función administrada	Operaciones principales
Ofertas de empleo	Entidad ofertante, descripción, fechas, carreras y habilidades requeridas	Guardar y limpiar
Llamados	Procedencia, descripción, fechas y estado del llamado	Guardar, actualizar, eliminar y listar
Estudiantes	Datos personales, carrera, teléfonos y currículum	Guardar, actualizar, eliminar y listar
Empresa/ Dependencia	Selección del tipo de entidad que se desea registrar	Abrir el formulario de empresa, abrir el formulario de dependencia o regresar al menú
Empresas	Identificación y nombre de la empresa	Guardar, actualizar, eliminar y listar
Dependencias	Institución, tipo y nombre de la dependencia	Guardar, actualizar, eliminar y listar
Postulaciones	Llamado, estudiante, fecha y observaciones	Buscar y registrar
Resultados	Postulante, resultado final, fecha y observaciones	Buscar y guardar
Reportes	Información seleccionada por el usuario	Generar y visualizar reportes

2.2.7 Conexión mediante ODBC

La conexión entre la interfaz en VBA Excel y la base de datos MySQL se configuró mediante ODBC. Se creó un origen de datos denominado BD_MySQL y se declaró un objeto global ADODB.Connection. El procedimiento verificó el estado de la conexión antes de abrir una nueva sesión e incorporó un controlador de errores para gestionar fallos de comunicación [14], [18], [19], [20].

La Figura 3 muestra el procedimiento programado en VBA para establecer la conexión con MySQL mediante el origen de datos BD_MySQL.

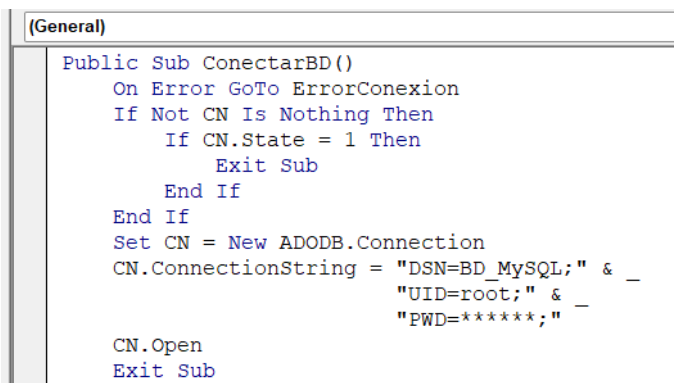


Fig. 3. Procedimiento de conexión entre VBA Excel y MySQL mediante ODBC.

3. RESULTADOS

3.1 Implementación de la base de datos

La implementación física generó una base de datos relacional en MySQL integrada por 15 tablas: CARRERA,

ESTUDIANTE, ESTUDIANTE_TELEFONO, EMPRESA, INSTITUCION, DEPENDENCIA, LLAMADO, LLAMADO_EMPRESA, LLAMADO_FACULTAD, INSCRIPCION, CONTRATACION, RESULTADO_POSTULACION, HABILIDAD, LLAMADO_HABILIDAD y LLAMADO_CARRERA. Las relaciones entre estas tablas se establecieron mediante claves primarias y foráneas, conforme al modelo relacional de UNITRABAJO.

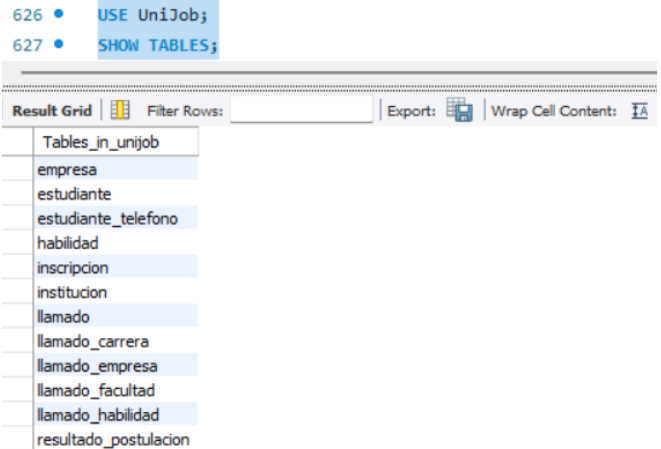


Fig. 4. Tablas implementadas en la base de datos UNITRABAJO.

3.2 Resultados de las consultas

Las consultas ejecutadas en MySQL recuperaron información sobre la participación de los estudiantes en los llamados laborales. Se obtuvieron registros de estudiantes inscritos en oportunidades ofertadas por empresas, postulantes asociados con llamados desiertos, estudiantes inscritos sin contratación y estudiantes seleccionados en llamados de empresas o dependencias institucionales. Para ello, se combinaron principalmente las tablas ESTUDIANTE, LLAMADO, INSCRIPCION, LLAMADO_EMPRESA, LLAMADO_FACULTAD, RESULTADO_POSTULACION y CONTRATACION.

La Tabla V resume la cantidad de registros obtenida en las consultas ejecutadas.

TABLA V
RESULTADOS DE LAS CONSULTAS SQL

Consulta ejecutada	Número de registros
Estudiantes registrados con su carrera	10
Llamados laborales registrados	10
Estudiantes inscritos en llamados laborales	11
Llamados sin estudiantes inscritos	3
Estudiantes contratados	10

3.2.1 Consulta de estudiantes inscritos en llamados laborales

Para obtener los estudiantes inscritos en cada convocatoria, se relacionaron las tablas INSCRIPCION,

ESTUDIANTE, CARRERA y LLAMADO. El resultado mostró, por cada registro, el número y la descripción del llamado, la cédula y el nombre del estudiante, la carrera profesional y la fecha de inscripción.

```

626 SELECT
627     l.numero_llamado,
628     l.descripcion AS llamado,
629     e.cedula,
630     e.nombre AS estudiante,
631     c.nombre_carrera AS carrera,
632     i.fecha_inscripcion
633 FROM INSCRIPCION i
634 INNER JOIN ESTUDIANTE e

```

numero_llamado	llamado	cedula	estudiante	carrera	fecha_inscripcion
1001	Practicante de desarrollo de software	70000001	Ana Torres Ruiz	Ingeniería de Sistemas	2026-08-02
1001	Practicante de desarrollo de software	70000008	Pedro Castillo	Ingeniería de Telecomunicaciones	2026-08-03
1002	Asistente de investigación en estructuras	70000006	Diego López	Ingeniería Mecánica	2026-08-05
1002	Asistente de investigación en estructuras	70000003	María Rojas Ledón	Ingeniería Civil	2026-08-04
1003	Practicante de supervisión de obras civiles	70000006	Diego López	Ingeniería Mecánica	2026-08-06
1003	Practicante de supervisión de obras civiles	70000003	María Rojas Ledón	Ingeniería Civil	2026-08-05
1004	Apoyo en proyectos de automatización	70000005	Carmen Díaz	Ingeniería Mecatrónica	2026-08-07
1005	Practicante de mantenimiento industrial	70000006	Diego López	Ingeniería Mecánica	2026-08-08
1005	Practicante de mantenimiento industrial	70000007	Rosa García Silva	Ingeniería Electrónica	2026-08-09
1007	Practicante de gestión ambiental	70000004	José Ruiz García	Ingeniería Ambiental	2026-08-10
1007	Practicante de gestión ambiental	70000009	Lucía Ramos	Ingeniería de Energía	2026-08-11

Fig. 5. Estudiantes inscritos en los llamados laborales.

3.3 Interfaces gráficas desarrolladas

El desarrollo en VBA Excel produjo una interfaz compuesta por un menú principal y formularios para gestionar estudiantes, empresas, dependencias, ofertas laborales, llamados, postulaciones, resultados y reportes. El menú principal permitió abrir cada módulo, y los formularios mostraron los campos y controles correspondientes a la tabla o proceso administrado.

Los formularios permitieron registrar, buscar, actualizar y eliminar datos. Además, se incorporó un formulario de selección que dirige al usuario al registro de empresas o al registro de dependencias, lo que concentró ambas opciones sin recargar el menú principal.

Sistema Oficina de Trabajo - Menú Principal

UNI TRABAJO
Sistema de Gestión de Ofertas y Empleo Universitario

Bienvenido(a), Administrador

- Ofertas Empleo
- Llamados
- Estudiantes
- Empresas / Instituciones
- Postulación
- Resultados
- Reportes
- Salir

RESUMEN GENERAL

Ofertas activas	8	Llamados abiertos	1
Postulaciones	15	Estudiantes	10
Contrataciones	10	Llamados Desiertos	3

Fig. 6. Menú principal y resumen general de UNITRABAJO.

La Figura 6 presenta el menú principal de UNITRABAJO, desde el cual se accede a los módulos y

se visualizan indicadores calculados con los registros de la base de datos.

La Figura 7 muestra el formulario intermedio que permite elegir entre el módulo de empresas y el módulo de dependencias institucionales.

Empresa / Dependencia

UNI TRABAJO
Sistema de Gestión de Ofertas y Empleo Universitario

Registrar Empresa

Registrar Dependencia

Regresar al Menú

Fig. 7. Formulario de selección entre empresas y dependencias.

Estudiantes

DATOS DEL ESTUDIANTE

Cedula Estudiante: Nombres:

Fecha de nacimiento: Email:

Carrera:

TÉLEFONO

Tipo:

Número:

AGREGAR CURRÍCULUM

Archivo:

LISTADO DE ESTUDIANTES

Cedula	Nombre	Carrera	Email
70000001	Ana Torres Ruiz	Ingeniería de Sistemas	ana.torres@mail.com
70000002	Luis Pérez Díaz	Ingeniería Industrial	luis.perez@mail.com
70000003	María Rojas Ledón	Ingeniería Civil	maria.rojas@mail.com
70000004	José Ruiz García	Ingeniería Ambiental	jose.ruiz@mail.com
70000005	Carmen Díaz	Ingeniería Mecatrónica	carmen.diaz@mail.com
70000006	Diego López	Ingeniería Mecánica	diego.lopez@mail.com
70000007	Rosa García Silva	Ingeniería Electrónica	rosa.garcia@mail.com
70000008	Pedro Castillo	Ingeniería de Telecomunicaciones	pedro.castillo@mail.com
70000009	Lucía Ramos	Ingeniería de Energía	lucia.ramos@mail.com
70000010	Carlos Vega Peña	Ingeniería de Minas	carlos.vega@mail.com

Fig. 8. Formulario de gestión de estudiantes.

Postulaciones

REGISTRO DE POSTULACIONES

Nro llamado: Fecha Postulación:

Cédula Estudiante:

Estudiante:

Carrera: Currículum:

Observación:

Fig. 9. Formulario de registro de postulaciones.

Cada formulario incluyó campos de entrada, listas desplegables, tablas de registros y botones para ejecutar las operaciones del módulo. Por tanto, el usuario pudo administrar la información desde Excel sin ingresar directamente a MySQL Workbench.

3.4 Integración entre VBA Excel y MySQL

La conexión entre VBA Excel y MySQL se configuró mediante ODBC, utilizando la biblioteca ADODB y el origen de datos BD_MySQL. Desde los formularios, la aplicación ejecutó instrucciones SQL para insertar, consultar, actualizar o eliminar registros en las tablas de UNITRABAJO. A su vez, los datos recuperados desde MySQL se mostraron en los controles de la interfaz [18], [19], [20].

La integración se comprobó mediante dos operaciones sobre la tabla EMPRESA: inserción y eliminación. En cada caso, la acción se ejecutó desde el formulario de empresas y se verificó mediante una consulta SQL en MySQL.

Para verificar la inserción, en el campo id_empresa se ingresó el código "E11" y en el campo nombre_empresa la razón social "Deveru S.A.C.". Al presionar el botón Guardar, el sistema mostró el mensaje "Empresa registrada correctamente", como se observa en la Figura 10.

ID	Nombre
E01	Soluciones Tecnológicas
E02	Constructora Ingeniería
E03	Automatización y Control
E04	Consultoría Ambiental
E05	Energía y Mantenimiento
E06	Redes y Telecomunicaciones
E07	Ingeniería Minera Regional
E08	Proyectos Electromecánicos
E09	Gestión Industrial Peruana S.A.C.
E10	Infraestructura y Servicios Técnicos S.A.C.

Fig. 10. Registro de la empresa con código E11 mediante la interfaz desarrollada en VBA Excel.

Luego, se ejecutó la consulta `SELECT id_empresa, nombre_empresa FROM EMPRESA WHERE id_empresa = 'E11'`. El resultado devolvió el código "E11" en el campo id_empresa y la razón social "Deveru S.A.C." en el campo nombre_empresa, lo que confirmó que ambos valores se almacenaron en la tabla EMPRESA, como se muestra en la Figura 11.

```

626 • USE UniJob;
627 • SELECT id_empresa, nombre_empresa
628 FROM EMPRESA
629 WHERE id_empresa = 'E11';
    
```

id_empresa	nombre_empresa
E11	Deveru S.A.C.

Fig. 11. Verificación del registro E11 en la tabla EMPRESA de MySQL.

Para verificar la eliminación, se seleccionó en el listado el registro con id_empresa "E11" y nombre_empresa "Deveru S.A.C.", y se presionó el botón Eliminar. El formulario mostró el mensaje "Empresa eliminada correctamente", como se observa en la Figura 12.

ID	Nombre
E01	Soluciones Tecnológicas
E02	Constructora Ingeniería
E03	Automatización y Control
E04	Consultoría Ambiental
E05	Energía y Mantenimiento
E06	Redes y Telecomunicaciones
E07	Ingeniería Minera Regional
E08	Proyectos Electromecánicos
E09	Gestión Industrial Peruana S.A.C.
E10	Infraestructura y Servicios Técnicos S.A.C.

Fig. 12. Eliminación del registro E11 mediante la interfaz desarrollada en VBA Excel.

A continuación, se ejecutó la consulta `SELECT id_empresa, nombre_empresa FROM EMPRESA` para listar los registros almacenados. El resultado ya no incluyó el código "E11" ni la razón social "Deveru S.A.C.", lo que confirmó que el registro fue eliminado de la tabla EMPRESA, como se presenta en la Figura 13.

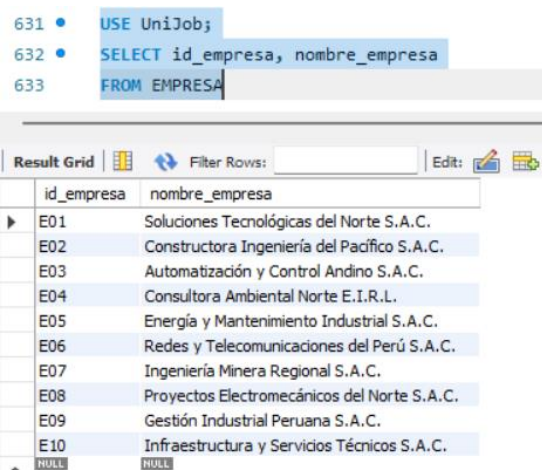


Fig. 13. Verificación de la eliminación del registro E11 en la tabla EMPRESA de MySQL.

La coincidencia entre los mensajes emitidos por el formulario y los resultados de las consultas SQL confirmó que la conexión ODBC ejecutó correctamente las operaciones de inserción y eliminación desde VBA Excel hacia MySQL. En consecuencia, los cambios realizados en la interfaz se reflejaron de forma inmediata en la base de datos UNITRABAJO.

4. DISCUSIÓN

Los resultados demuestran que UNITRABAJO cumple el objetivo de gestionar, en una estructura centralizada, las ofertas laborales, postulaciones y contrataciones estudiantiles.

1. La normalización de la relación inicial OFICINA_TRABAJO hasta la Forma Normal de Boyce-Codd distribuyó los datos en tablas relacionadas, redujo la repetición de atributos y evitó dependencias que podían originar anomalías de inserción, actualización y eliminación, de acuerdo con los fundamentos del diseño relacional [1], [2], [4], [5], [8], [9].
2. El modelo físico implementado en MySQL mantuvo la integridad referencial mediante claves primarias, claves foráneas y restricciones. Esta estructura vinculó de manera consistente los registros de estudiantes, carreras, llamados, inscripciones, resultados y contrataciones [10], [11], [12].
3. Las consultas SQL recuperaron los conjuntos de datos requeridos por la Oficina de Trabajo, entre ellos estudiantes inscritos, llamados sin postulantes, postulantes no contratados y estudiantes seleccionados. Los resultados obtenidos muestran correspondencia entre las necesidades de información y las relaciones definidas en el modelo [3], [10], [11], [12].
4. La interfaz en VBA Excel permitió ejecutar operaciones sobre los registros sin acceder directamente al gestor de base de datos. La conexión ODBC y la biblioteca ADODB trasladaron las operaciones realizadas en los formularios hacia

MySQL y devolvieron los datos consultados a la interfaz [13], [14], [17], [18], [19], [20].

5. CONCLUSIÓN

Se concluye que el diseño e implementación de UNITRABAJO permitió centralizar la información de estudiantes, ofertas laborales, postulaciones y contrataciones de la Facultad de Ingeniería. La normalización hasta la Forma Normal de Boyce-Codd y la definición de claves y restricciones en MySQL redujeron la redundancia y mantuvieron la consistencia de los datos. Las consultas SQL recuperaron información relevante para el seguimiento de los llamados laborales, mientras que la integración de VBA Excel con MySQL mediante ODBC permitió registrar, consultar, actualizar y eliminar datos desde una interfaz gráfica. Como trabajo futuro, se propone incorporar autenticación, roles de usuario, controles de seguridad, copias de respaldo y pruebas de uso con el personal de la Oficina de Trabajo.

6. RECONOCIMIENTOS

Los autores agradecen al Dr. Ing. José Arturo Díaz Pulido, docente de la Universidad Nacional de Trujillo, por la orientación y asesoría brindadas durante el desarrollo del proyecto. Asimismo, agradecen a la Escuela de Ingeniería Informática por proporcionar el entorno académico necesario para su ejecución.

7. REFERENCIAS

- [1] C. Coronel and S. Morris, *Database Systems: Design, Implementation, & Management*, 13th ed. Boston, MA, USA: Cengage, 2019.
- [2] R. Elmasri and S. B. Navathe, *Fundamentals of Database Systems*, 7th ed. Hoboken, NJ, USA: Pearson, 2016.
- [3] A. Silberschatz, H. F. Korth, and S. Sudarshan, *Database System Concepts*, 7th ed. New York, NY, USA: McGraw-Hill, 2020.
- [4] J. Date, *An Introduction to Database Systems*, 8th ed. Boston, MA, USA: Pearson/Addison-Wesley, 2004.
- [5] T. M. Connolly and C. E. Begg, *Database Systems: A Practical Approach to Design, Implementation, and Management*, 6th ed. Harlow, U.K.: Pearson, 2015.
- [6] H. Garcia-Molina, J. D. Ullman, and J. Widom, *Database Systems: The Complete Book*, 2nd ed. Upper Saddle River, NJ, USA: Pearson Prentice Hall, 2009.
- [7] E. F. Codd, "A relational model of data for large shared data banks," *Commun. ACM*, vol. 13, no. 6, pp. 377–387, Jun. 1970, doi: 10.1145/362384.362685.
- [8] E. F. Codd, "Further normalization of the data base relational model," IBM Research Laboratory, San Jose, CA, USA, Res. Rep. RJ 909, Aug. 1971.
- [9] R. Fagin, "Multivalued dependencies and a new normal form for relational databases," *ACM Trans. Database Syst.*, vol. 2, no. 3, pp. 262–278, Sep. 1977, doi: 10.1145/320557.320571.
- [10] *Information Technology—Database Languages SQL—Part 1: Framework (SQL/Framework)*, ISO/IEC 9075-1:2023, 2023.

- [11] *Information Technology—Database Languages SQL—Part 2: Foundation (SQL/Foundation)*, ISO/IEC 9075-2:2023, 2023.
- [12] Oracle Corporation, *MySQL 8.0 Reference Manual*, 2026. [Online]. Accessed: Jul. 20, 2026.
- [13] Microsoft, "Office Visual Basic for Applications (VBA) reference," *Microsoft Learn*, Sep. 13, 2021. [Online]. Accessed: Jul. 20, 2026.
- [14] Microsoft, ODBC Programmer's Reference, Microsoft Learn, Mar. 30, 2026. [Online]. Accessed: Jul. 20, 2026.
- [15] GNOME Project, "Dia Diagram Editor documentation," *GNOME Wiki Archive*, Oct. 23, 2024. [Online]. Accessed: Jul. 20, 2026.
- [16] Oracle Corporation, *MySQL Workbench Manual*, 2026. [Online]. Accessed: Jul. 20, 2026.
- [17] Microsoft, "Excel Visual Basic for Applications (VBA) reference," Microsoft Learn, Jul. 12, 2022. [Online]. Accessed: Jul. 20, 2026.
- [18] Microsoft, "ActiveX Data Objects (ADO) API reference," *Microsoft Learn*, Jun. 3, 2022. [Online]. Accessed: Jul. 20, 2026.
- [19] Microsoft, "ADO programmer's guide," *Microsoft Learn*, Sep. 14, 2021. [Online]. Accessed: Jul. 20, 2026.
- [20] Oracle Corporation, *MySQL Connector/ODBC Developer Guide*, 2026. [Online]. Accessed: Jul. 20, 2026.