Pontificia Universidad Católica del Ecuador

Facultad de Hábitat, Infraestructura y Creatividad

Carrera de Sistemas de Información

Desarrollo de un prototipo funcional de encuestas digitales con asistencia de inteligencia artificial para apoyar la creación, gestión y organización de formularios de recolección de datos

Trabajo de titulación presentado como requisito previo para la obtención del título de Ingeniero en Sistemas de la Información

Danny Alexander Balseca Olivo

Director y tutor: Mgtr./Ph. D. Rafael Heredia Melgarejo

Quito, 6 de junio de 2026

Contents

RESUMEN iv

INTRODUCCIÓN 1

CAPÍTULO 1: INTRODUCCIÓN Y PLANTEAMIENTO DEL PROYECTO 2

1.1 Contextualización del problema 2

1.2 Planteamiento del problema 2

1.3 Pregunta guía del desarrollo tecnológico 3

1.4 Justificación 3

1.5 Objetivos 4

1.5.1 Objetivo general 4

1.5.2 Objetivos específicos 4

1.6 Alcance y delimitaciones 4

1.7 Estructura final del documento 5

CAPÍTULO 2: FUNDAMENTACIÓN TEÓRICA 6

2.1 Sistemas de información web 6

2.2 Aplicaciones web para gestión de encuestas 6

2.3 Ingeniería de software y prototipado 7

2.4 Arquitectura cliente-servidor 7

2.5 Backend en .NET 8

2.6 Frontend web con Vue 3, Vite y TypeScript 8

2.7 Base de datos relacional y SQL Server 9

2.8 Inteligencia artificial 9

2.9 Inteligencia artificial generativa en aplicaciones web 10

2.10 Uso responsable y controlado de IA 10

2.11 Experiencia de usuario y usabilidad 10

2.12 Modelo TAM como apoyo a la aceptación tecnológica 11

2.13 Escala SUS como instrumento de usabilidad 11
CAPÍTULO 3: MARCO DE DESARROLLO Y VALIDACIÓN DEL PROTOTIPO 12
3.1 Caracterización del proyecto tecnológico 12
3.2 Método de desarrollo del prototipo 13
3.3 Técnicas, evidencias e instrumentos de validación 13
3.4 Participantes de la validación del prototipo 14
3.5 Alcance de la validación y limitaciones del prototipo 14
CAPÍTULO 4: ANÁLISIS, DISEÑO Y DESARROLLO DEL PROTOTIPO 15
4.1 Requerimientos funcionales y no funcionales 15
4.2 Actores y casos de uso 16
4.3 Arquitectura técnica del prototipo 17
4.3.1 Backend implementado 18
4.3.2 Frontend implementado 19
4.3.3 Comunicación entre frontend y backend 19
4.4 Modelamiento e implementación de la base de datos 20
4.4.1 Modelo conceptual 21
4.4.2 Modelo lógico 22
4.4.3 Modelo físico implementado 24
4.5 Módulo de usuarios 26
4.6 Módulo de encuestas, preguntas y respuestas 26
4.7 Integración de asistencia de IA 27
4.8 Estado actual del desarrollo 28
4.9 Verificación técnica del frontend y backend 30
CAPÍTULO 5: RESULTADOS Y VALIDACIÓN 31
5.1 Introducción del capítulo 31
5.2 Resultados funcionales del sistema 31
5.3 Validación técnica del prototipo 32

5.4 Validación experimental 33
5.5 Evaluación de usabilidad mediante SUS 34
5.6 Evaluación de aceptación tecnológica mediante TAM 36
5.7 Evaluación del uso de inteligencia artificial 37
5.8 Comparación de tiempo con y sin apoyo del sistema 38
5.9 Resultados por objetivo 38
5.10 Análisis general de resultados 39
CAPÍTULO 6: CONCLUSIONES Y RECOMENDACIONES 40
6.1 Conclusiones 40
6.2 Recomendaciones 41
BIBLIOGRAFÍA 42
ANEXOS 43
Anexo A. Matriz de avance del proyecto 43
Anexo B. Cronograma referencial de trabajo 44
Anexo D. Plantilla de bitácora de iteraciones 45
Anexo E. Capturas de la aplicación SurveyDanny 46
Anexo F. Manual de usuario de la aplicación SurveyDanny 55
F.1 Objetivo del manual 55
F.2 Requisitos de acceso 55
F.3 Ingreso y registro 55
F.4 Navegación principal 55
F.5 Crear una encuesta manual 56
F.6 Generar encuesta con asistencia de IA 56
F.7 Usar el banco de preguntas 56
F.8 Configurar y publicar una encuesta 57
F.9 Responder una encuesta 57
F.10 Consultar reportes 57

F.11 Mensajes frecuentes 58
Anexo G. Manual administrativo de la aplicación SurveyDanny 58
G.1 Objetivo del manual 58
G.2 Componentes administrados 58
G.3 Gestión de usuarios y roles 58
G.4 Administración de módulos funcionales 59
G.5 Endpoints principales de operación 59
G.6 Administración de base de datos 60
G.7 Puesta en marcha del sistema 60
G.8 Parámetros de configuración 60
G.9 Mantenimiento y respaldo 61
G.10 Atención de incidentes 61
G.11 Checklist administrativo 61

RESUMEN

El presente trabajo de titulación propone el desarrollo de un prototipo funcional de encuestas digitales con asistencia de inteligencia artificial, orientado a apoyar la creación, gestión y organización de formularios de recolección de datos en un entorno académico y tecnológico. La propuesta surge a partir de la dificultad que presentan los procesos manuales o el uso de herramientas generales no integradas, los cuales pueden generar demoras en la elaboración de encuestas, duplicidad de información, baja trazabilidad y limitaciones para consolidar resultados básicos.

El prototipo se plantea bajo una arquitectura cliente-servidor. El frontend se desarrolla con Vue 3, Vite y TypeScript, con apoyo de Pinia para la gestión de sesión, Axios para el consumo de servicios, PrimeVue para componentes de interfaz, Chart.js para visualización básica y lucide-vue-next para iconografía. El backend se implementa con ASP.NET Core,

Entity Framework Core y SQL Server, exponiendo endpoints REST para autenticación, gestión de encuestas, gestión de preguntas, configuración de encuestas públicas, registro de respuestas y consulta básica de resultados.

La inteligencia artificial se incorpora como una funcionalidad asistiva y controlada dentro del sistema. En la implementación actual, el frontend cuenta con una vista para ingresar tema, objetivo y cantidad de preguntas; el backend dispone de un controlador y un servicio que construyen la solicitud, consumen una API externa y guardan la encuesta generada junto con sus preguntas y opciones. Esta funcionalidad debe entenderse como apoyo inicial para la creación de encuestas, no como validación científica automática de la encuesta.

El trabajo se caracteriza como un proyecto de desarrollo tecnológico aplicado, cuyo resultado principal es un prototipo web funcional. La validación propuesta considera pruebas funcionales, comparación entre creación manual o mediante Google Forms y uso del prototipo, medición de tiempo, registro de tareas completadas, encuesta de usabilidad basada en SUS, encuesta de aceptación basada en TAM y valoración de satisfacción de usuarios. La revisión técnica realizada confirma que el frontend supera la verificación de tipos y que el backend compila correctamente, aunque existen advertencias de nulabilidad que deben considerarse como mejora técnica pendiente.

Palabras clave: encuestas digitales, prototipo funcional, sistema web, inteligencia artificial asistiva, .NET, SQL Server, Vue, validación experimental, usabilidad.

INTRODUCCIÓN

La recopilación de datos mediante encuestas constituye una actividad frecuente en instituciones, organizaciones y proyectos académicos que requieren obtener información de manera ordenada para apoyar procesos de evaluación, diagnóstico o toma de decisiones. Sin embargo, cuando la creación, aplicación y almacenamiento de encuestas se realiza mediante herramientas separadas, documentos manuales o formularios generales no integrados, pueden presentarse dificultades relacionadas con la organización de la información, la duplicidad de datos, la pérdida de trazabilidad y la consolidación tardía de resultados.

En este contexto, los sistemas de información web representan una alternativa adecuada para centralizar el proceso de gestión de encuestas digitales. Una aplicación web puede integrar módulos de autenticación, creación de encuestas, administración de preguntas, almacenamiento de respuestas y visualización básica de resultados dentro de una misma solución. Además, una arquitectura cliente-servidor permite separar la interfaz de usuario, la lógica de negocio y la persistencia de datos, lo cual favorece la mantenibilidad del prototipo.

El presente trabajo se orienta al desarrollo de un prototipo funcional de encuestas digitales con asistencia de inteligencia artificial. La IA se concibe como una funcionalidad de apoyo para generar una estructura inicial de encuesta a partir de un tema u objetivo ingresado por el usuario. Esta asistencia no tiene carácter autónomo ni define por sí sola la calidad de la encuesta; por el contrario, se incorpora como un módulo del sistema cuya salida debe ser revisada por el usuario antes de utilizarse como formulario definitivo.

El documento se organiza en seis capítulos. El primer capítulo presenta el planteamiento del proyecto. El segundo desarrolla la fundamentación teórica. El tercer capítulo define el marco de desarrollo y validación del prototipo. El cuarto expone el análisis, diseño y desarrollo del prototipo. El quinto plantea la validación experimental y el análisis de resultados esperados. Finalmente, el sexto capítulo presenta conclusiones y recomendaciones preliminares, sin inventar resultados que dependan de pruebas aún pendientes.

CAPÍTULO 1: INTRODUCCIÓN Y PLANTEAMIENTO DEL PROYECTO

1.1 Contextualización del problema

Las organizaciones requieren información confiable para evaluar procesos, conocer opiniones, identificar necesidades y respaldar decisiones. Las encuestas son uno de los mecanismos más utilizados para obtener datos directamente de usuarios, clientes, estudiantes o grupos de interés. No obstante, su efectividad depende de la forma en que se diseñan, aplican, almacenan y consultan sus resultados.

En la práctica, muchas encuestas se elaboran en documentos, hojas de cálculo, formularios independientes o plataformas generales como Google Forms. Aunque estas herramientas permiten crear formularios de manera rápida, no siempre se ajustan a las necesidades de una solución tecnológica propia, integrada con autenticación, base de datos, módulos de administración, lógica de negocio y trazabilidad del flujo de respuestas.

A esta situación se suma la dificultad que algunos usuarios tienen al redactar preguntas claras y coherentes con el objetivo de una encuesta. La formulación inicial de preguntas puede requerir tiempo, organización del contenido y revisión de opciones de respuesta. Por ello, resulta pertinente desarrollar un prototipo que incorpore asistencia de IA como apoyo funcional, sin eliminar la decisión humana sobre la calidad final de las preguntas.

1.2 Planteamiento del problema

Actualmente, la elaboración y gestión de encuestas puede demandar tiempo cuando se realiza de forma manual o mediante herramientas no integradas. El usuario debe redactar preguntas, organizar opciones, configurar formularios, distribuir enlaces, registrar respuestas y consultar resultados en espacios separados. Esta dispersión puede generar errores de registro, duplicidad de información, dificultad para mantener trazabilidad y retrasos en la consolidación de datos.

Si bien existen plataformas digitales para crear formularios, estas suelen ser herramientas generales y externas al sistema propio de una organización o proyecto. Además, no siempre ofrecen una asistencia inteligente integrada que ayude al usuario a estructurar preguntas, proponer opciones de respuesta o iniciar la construcción de una encuesta desde el propio flujo del sistema.

Desde la perspectiva de Ingeniería en Sistemas de Información, el problema principal no consiste en validar científicamente cuestionarios o modelos académicos, sino en diseñar y desarrollar una solución tecnológica que apoye el proceso de creación, gestión, aplicación y almacenamiento de encuestas digitales. Por tanto, se requiere un prototipo funcional que integre módulos técnicos esenciales y una asistencia de IA limitada, editable y controlada por el usuario.

1.3 Pregunta guía del desarrollo tecnológico

¿Cómo diseñar y desarrollar un prototipo funcional de encuestas digitales con asistencia de inteligencia artificial que apoye la creación, gestión y organización de formularios de recolección de datos dentro de un entorno web?

1.4 Justificación

El desarrollo de un prototipo funcional de encuestas digitales se justifica por la necesidad de contar con una solución tecnológica propia que centralice el proceso de creación, administración, publicación y almacenamiento de encuestas. Frente a procedimientos manuales o herramientas generales no integradas, una plataforma propia permite organizar mejor los datos, controlar el acceso, mantener la persistencia en una base de datos relacional y facilitar la consulta básica de resultados.

La propuesta aporta valor desde el campo de Ingeniería en Sistemas porque integra análisis de requerimientos, diseño de arquitectura, modelamiento de base de datos, desarrollo frontend, desarrollo backend, seguridad básica mediante autenticación, integración controlada de IA y pruebas funcionales. Estos elementos permiten que el trabajo sea evaluado como un proyecto tecnológico de desarrollo de software y no como una propuesta teórica centrada en la construcción de instrumentos.

La asistencia de inteligencia artificial se incorpora como una función complementaria para mejorar la experiencia del usuario durante la creación de encuestas. Su objetivo es sugerir preguntas, opciones o estructuras iniciales que reduzcan el tiempo de elaboración y apoyen la organización del formulario. Sin embargo, las sugerencias generadas no se aplican automáticamente: el usuario debe revisarlas, editarlas, aceptarlas o descartarlas.

Esta delimitación mantiene la factibilidad técnica y académica del proyecto. El prototipo no pretende entrenar modelos propios, reemplazar la revisión humana ni garantizar calidad

científica absoluta de las encuestas; busca demostrar la construcción de una solución web funcional que integre asistencia de IA como apoyo controlado dentro del flujo de trabajo. Por ello, la justificación enfatiza el componente tecnológico y la validación experimental real del prototipo con usuarios, tiempos de ejecución, tareas completadas, SUS, TAM, satisfacción y comparación frente a creación manual o Google Forms.

1.5 Objetivos

1.5.1 Objetivo general

Desarrollar un prototipo funcional de encuestas digitales con asistencia de inteligencia artificial, bajo arquitectura cliente-servidor, que permita la creación, gestión, organización y almacenamiento de formularios de recolección de datos en un entorno web.

1.5.2 Objetivos específicos

1. Levantar los requerimientos funcionales y no funcionales del prototipo, considerando los módulos de usuarios, encuestas, preguntas, respuestas, resultados básicos y asistencia de IA.
2. Diseñar la arquitectura del sistema, la interfaz de usuario y el modelo de base de datos relacional que soporten la operación del prototipo web.
3. Desarrollar el prototipo funcional mediante frontend, backend y base de datos, incorporando módulos de autenticación, gestión de encuestas, gestión de preguntas, almacenamiento de respuestas y visualización básica de resultados.
4. Integrar una funcionalidad de asistencia de IA que permita sugerir preguntas, opciones o estructura inicial de encuesta, manteniendo la revisión y edición humana como control obligatorio.
5. Validar experimentalmente el prototipo con usuarios mediante pruebas funcionales, medición de tiempo, tareas completadas, usabilidad, aceptación tecnológica y satisfacción general.

1.6 Alcance y delimitaciones

El alcance del proyecto comprende el análisis, diseño, desarrollo y validación experimental de un prototipo web de encuestas digitales. El sistema se orienta a usuarios que requieren crear encuestas, administrar preguntas, configurar formularios, publicar enlaces de respuesta, almacenar información y consultar resultados básicos desde una interfaz web.

La asistencia de IA forma parte del alcance como una funcionalidad de apoyo para sugerir preguntas, opciones o estructuras iniciales. La respuesta generada por la API deberá mostrarse como contenido editable y no como resultado definitivo. El usuario conserva la responsabilidad de revisar y ajustar el contenido antes de incorporarlo a una encuesta.

Tabla 1. Delimitación general del alcance del prototipo

Incluye	No incluye
Gestión de usuarios, registro, inicio de sesión y roles básicos.	Aplicaciones móviles nativas.
Creación, edición, consulta y eliminación de encuestas.	Despliegue institucional en producción.
Gestión de preguntas, tipos de respuesta y opciones.	Entrenamiento de modelos propios de inteligencia artificial.
Almacenamiento de respuestas enviadas por perfils.	Validación estadística de la calidad científica de las encuestas.
Asistencia de IA para sugerir preguntas, opciones o estructura inicial.	Sustitución de la revisión humana sobre preguntas sugeridas.
Visualización básica de resultados y reportes iniciales.	Garantía de calidad científica absoluta de las encuestas.
Validación experimental con usuarios en entorno prototipo.	Analítica predictiva, minería de datos o tableros avanzados.

1.7 Estructura final del documento

- Capítulo 1: Introducción y planteamiento del proyecto.
- Capítulo 2: Fundamentación teórica.
- Capítulo 3: Marco de desarrollo y validación del prototipo.
- Capítulo 4: Análisis, diseño y desarrollo del prototipo.
- Capítulo 5: Validación experimental y análisis de resultados.
- Capítulo 6: Conclusiones y recomendaciones.

CAPÍTULO 2: FUNDAMENTACIÓN TEÓRICA

2.1 Sistemas de información web

Los sistemas de información web son soluciones de software accesibles mediante un navegador que permiten registrar, procesar, almacenar y consultar información desde distintos dispositivos conectados a una red. Su arquitectura suele integrar una capa de presentación, una capa de lógica de negocio y una capa de persistencia de datos. Desde la ingeniería de software, este tipo de sistemas debe construirse a partir de requerimientos claros, diseño modular, pruebas y mantenimiento continuo, con el fin de favorecer su evolución y confiabilidad (Pressman & Maxim, 2020; Sommerville, 2016).

En el presente trabajo, el sistema web se materializa en el prototipo SurveyDanny, orientado a la creación, gestión, organización y almacenamiento de encuestas digitales. La interfaz permite que el usuario interactúe con formularios, preguntas, opciones de respuesta y reportes básicos, mientras que el backend procesa las solicitudes y conserva la información en una base de datos relacional.

El uso de sistemas web resulta pertinente en entornos académicos porque reduce la dependencia de instalaciones locales, facilita el acceso centralizado y permite integrar módulos de autenticación, administración de datos y consulta de resultados. En comparación con procesos manuales o herramientas aisladas, una plataforma web propia permite mayor trazabilidad, control de acceso y organización de la información.

2.2 Aplicaciones web para gestión de encuestas

Las aplicaciones web para gestión de encuestas permiten construir formularios digitales, distribuir enlaces de respuesta, almacenar datos y consultar resultados desde un entorno centralizado. Estas herramientas son útiles cuando una organización o proyecto requiere recolectar información de usuarios, estudiantes, clientes o grupos de interés de manera organizada.

Herramientas generales como Google Forms resuelven escenarios rápidos de recolección de datos; sin embargo, no siempre se ajustan a necesidades de integración con módulos propios, autenticación, reglas de negocio, trazabilidad interna o asistencia inteligente dentro del flujo del sistema. Por ello, SurveyDanny se plantea como un prototipo

académico que demuestra una solución propia, extensible y alineada con criterios de ingeniería de software.

2.3 Ingeniería de software y prototipado

La ingeniería de software proporciona métodos, prácticas y criterios para desarrollar sistemas de manera ordenada. Comprende actividades como levantamiento de requerimientos, diseño, implementación, pruebas, validación y mantenimiento. Pressman y Maxim (2020) destacan la importancia de aplicar procesos sistemáticos para construir productos mantenibles y verificables. De forma complementaria, Sommerville (2016) resalta la necesidad de comprender requisitos, arquitectura y evolución del software como elementos centrales de la calidad del producto.

El prototipado es adecuado cuando se busca construir una versión funcional que permita validar decisiones técnicas, observar el comportamiento del sistema y obtener retroalimentación temprana. En esta tesis, el prototipo no se presenta como una solución institucional final, sino como una versión funcional suficiente para demostrar los flujos principales de encuestas digitales con asistencia de inteligencia artificial.

2.4 Arquitectura cliente-servidor

La arquitectura cliente-servidor organiza el sistema en componentes con responsabilidades diferenciadas. El cliente corresponde a la aplicación que utiliza el usuario, normalmente ejecutada en el navegador, mientras que el servidor procesa solicitudes, aplica reglas de negocio, accede a la base de datos y devuelve respuestas estructuradas. Esta separación favorece la mantenibilidad porque cada capa puede evolucionar con menor dependencia directa sobre las demás.

En una aplicación web, el cliente se comunica con el servidor mediante solicitudes y respuestas, generalmente a través de servicios HTTP o API REST. El cliente no debe acceder directamente a la base de datos; en su lugar, consume servicios controlados por el backend. Esta estructura permite centralizar validaciones, mejorar la seguridad, proteger credenciales, controlar reglas de negocio y facilitar la escalabilidad del sistema.

En SurveyDanny, el frontend desarrollado con Vue 3, Vite y TypeScript actúa como cliente web. El backend implementado con ASP.NET Core funciona como servidor de aplicación y expone endpoints para autenticación, encuestas, preguntas, respuestas,

reportes y asistencia de IA. La base de datos SQL Server almacena la información mediante un modelo relacional administrado con Entity Framework Core.

Figura 1. Modelo cliente-servidor aplicado al prototipo SurveyDanny. Nota. Elaboración propia.

Tabla 2. Componentes de la arquitectura del prototipo

Componente	**Función dentro del sistema**
Frontend web Vue 3	Aplicación cliente que permite login, registro, dashboard, encuestas, preguntas, constructor, enlace público, asistencia de IA y reportes básicos.
Backend ASP.NET Core	Servidor API que expone endpoints REST, valida solicitudes, aplica reglas de negocio, autentica usuarios y coordina el acceso a datos.
SQL Server y Entity Framework Core	Capa de persistencia que almacena usuarios, roles, encuestas, preguntas, tipos de respuesta, respuestas y parámetros del sistema.
API externa de IA	Servicio consumido por el backend para generar sugerencias iniciales de preguntas y opciones, bajo revisión humana obligatoria.

2.5 Backend en .NET

.NET es una plataforma de desarrollo utilizada para construir aplicaciones web, servicios y APIs. En el contexto del proyecto, el backend se desarrolla con C# y ASP.NET Core porque permite organizar controladores, aplicar inyección de dependencias, documentar servicios, conectarse con bases de datos relacionales y proteger rutas mediante mecanismos de autenticación.

En SurveyDanny, esta capa concentra validaciones, reglas de negocio, acceso a datos, seguridad básica con JWT y comunicación con servicios externos. Esta separación evita que el frontend acceda directamente a la base de datos y permite mantener un punto central de control sobre las operaciones del sistema.

2.6 Frontend web con Vue 3, Vite y TypeScript

El frontend corresponde a la capa de presentación del sistema. Su función es permitir que el usuario interactúe con las funcionalidades disponibles mediante pantallas, formularios,

botones, mensajes y componentes visuales. En SurveyDanny, esta capa organiza las vistas de autenticación, dashboard, administración de encuestas, gestión de preguntas, constructor, encuesta pública, reportes y generación asistida con IA.

Vue 3 favorece la construcción de interfaces reactivas basadas en componentes reutilizables. Vite proporciona un entorno de desarrollo rápido, mientras que TypeScript aporta tipado estático para reducir errores durante la implementación. El uso de Vue Router, Pinia y Axios permite organizar navegación, sesión y comunicación HTTP con el backend.

2.7 Base de datos relacional y SQL Server

Una base de datos relacional organiza la información mediante tablas relacionadas por claves primarias y foráneas. Este modelo es pertinente para SurveyDanny porque el sistema maneja entidades claramente identificables, como usuarios, roles, encuestas, preguntas, opciones, tipos de respuesta y respuestas registradas.

SQL Server permite conservar la información de forma estructurada y consultar resultados básicos asociados a las encuestas. La integridad referencial es importante para evitar inconsistencias, como respuestas sin encuesta asociada o preguntas sin relación con un formulario.

2.8 Inteligencia artificial

La inteligencia artificial es un campo de la informática orientado al desarrollo de sistemas capaces de realizar tareas que requieren razonamiento, aprendizaje, planificación, percepción, comprensión del lenguaje o toma de decisiones. Russell y Norvig señalan que la IA estudia agentes que perciben su entorno y actúan para alcanzar objetivos, lo que permite comprender su aplicación en sistemas que apoyan actividades humanas mediante automatización o asistencia inteligente.

En sistemas educativos, administrativos o de apoyo a usuarios, la IA puede contribuir al análisis de información, generación de sugerencias, clasificación de datos o reducción de tareas repetitivas. No obstante, su aplicación debe delimitarse de acuerdo con el alcance del sistema y con controles de supervisión humana, especialmente cuando los resultados generados pueden influir en decisiones de usuarios.

2.9 Inteligencia artificial generativa en aplicaciones web

La inteligencia artificial generativa se refiere al uso de modelos capaces de producir contenido nuevo, como texto, preguntas, respuestas, resúmenes o propuestas iniciales a partir de instrucciones ingresadas por el usuario. En aplicaciones web, suele integrarse mediante APIs externas que reciben una solicitud estructurada y devuelven una respuesta generada por el modelo.

En SurveyDanny, la IA generativa se incorpora como funcionalidad asistiva para sugerir una estructura inicial de encuesta. El usuario ingresa un tema, un objetivo y una cantidad aproximada de preguntas; el backend construye la solicitud, consume la API externa y procesa la respuesta para generar preguntas, tipos de respuesta y opciones iniciales.

Este uso no implica que el sistema valide científicamente una encuesta ni que la IA reemplace el criterio humano. La salida generada debe considerarse un borrador editable. Por ello, el sistema debe mantener controles de revisión, edición y aprobación humana antes de que el contenido sea utilizado en un contexto académico o institucional.

2.10 Uso responsable y controlado de IA

El uso responsable de IA implica reconocer que los modelos generativos pueden producir respuestas incompletas, imprecisas, sesgadas o poco adecuadas para un contexto específico. Por esta razón, las sugerencias generadas deben presentarse como apoyo inicial y no como contenido definitivo.

En el prototipo, el control se implementa mediante tres criterios: el usuario define el tema y objetivo de la encuesta, la IA se mantiene como módulo de apoyo dentro del flujo técnico y el usuario conserva la responsabilidad de revisar, editar, aceptar o descartar el contenido generado. Esta delimitación evita atribuir a la IA una función de validación científica que no corresponde al alcance del trabajo.

2.11 Experiencia de usuario y usabilidad

La experiencia de usuario se relaciona con la forma en que una persona percibe e interactúa con un sistema. En una aplicación de encuestas, la interfaz debe facilitar acciones como crear formularios, registrar preguntas, seleccionar tipos de respuesta, publicar enlaces y consultar resultados. La claridad de los mensajes, la consistencia visual y la reducción de pasos innecesarios influyen en la percepción del usuario.

La usabilidad se evalúa considerando si los usuarios logran completar tareas de manera efectiva, eficiente y satisfactoria dentro de un contexto de uso. La norma ISO 9241-11 define la usabilidad a partir de efectividad, eficiencia y satisfacción, mientras que Nielsen destaca atributos como facilidad de aprendizaje, eficiencia de uso, memorabilidad, prevención de errores y satisfacción. Estos criterios son pertinentes para evaluar si SurveyDanny puede ser comprendido y utilizado por usuarios no especializados.

2.12 Modelo TAM como apoyo a la aceptación tecnológica

El Technology Acceptance Model, propuesto por Davis (1989), permite analizar la aceptación de una tecnología a partir de la utilidad percibida y la facilidad de uso percibida. La utilidad percibida se refiere al grado en que una persona considera que el sistema mejora su desempeño o facilita una actividad. La facilidad de uso percibida se relaciona con el esfuerzo requerido para comprender y utilizar la herramienta.

En el contexto de SurveyDanny, TAM permite valorar si los usuarios consideran que el prototipo ayuda a crear y organizar encuestas, si el flujo resulta comprensible y si estarían dispuestos a utilizarlo en escenarios futuros. Esta evaluación es especialmente útil en prototipos académicos porque permite identificar aceptación inicial, barreras de uso y oportunidades de mejora.

2.13 Escala SUS como instrumento de usabilidad

La System Usability Scale, propuesta por Brooke (1996), es un instrumento breve de 10 ítems que permite medir la usabilidad percibida de un sistema. Sus preguntas combinan afirmaciones positivas y negativas, respondidas generalmente en una escala tipo Likert. El resultado se transforma en un puntaje de 0 a 100 que facilita interpretar la percepción general de usabilidad.

En este trabajo, SUS se utiliza como referencia para valorar la facilidad de uso percibida después de que los usuarios ejecuten tareas básicas en el prototipo. Su aplicación permite complementar la validación funcional con una medida de percepción, lo cual resulta adecuado para identificar si el sistema es comprensible, cómodo y suficientemente claro para los usuarios.

CAPÍTULO 3: MARCO DE DESARROLLO Y VALIDACIÓN DEL PROTOTIPO

3.1 Caracterización del proyecto tecnológico

El presente trabajo corresponde a un proyecto de desarrollo tecnológico aplicado, debido a que su objetivo central es construir una solución web funcional para atender una necesidad práctica: apoyar la creación, gestión y organización de encuestas digitales mediante un prototipo con asistencia de IA.

El enfoque es tecnológico con alcance descriptivo. Es tecnológico porque el resultado principal es un sistema web desarrollado bajo una arquitectura cliente-servidor; es descriptivo porque documenta requerimientos, componentes, módulos, flujo de operación, validación y percepción de usuarios sin plantear la comprobación de una hipótesis estadística compleja.

El diseño corresponde a un proceso de prototipado incremental con validación experimental. El prototipo permite comprobar la factibilidad técnica de la solución y observar su comportamiento frente a tareas reales de usuarios en un entorno controlado.

Tabla 3. Caracterización del proyecto tecnológico

Elemento	Definición aplicada al proyecto	Justificación
Tipo de proyecto	Desarrollo tecnológico aplicado	Busca construir una solución tecnológica para un problema práctico de gestión de encuestas.
Enfoque	Tecnológico con alcance descriptivo	Describe el sistema, su arquitectura, módulos y validación desde una perspectiva de Ingeniería en Sistemas.
Diseño	Prototipado incremental con validación experimental	Permite implementar un producto funcional y evaluarlo con usuarios en un entorno controlado.
Producto principal	Prototipo web de encuestas digitales	La IA es un módulo asistivo dentro del sistema, no el centro del trabajo.

3.2 Método de desarrollo del prototipo

El desarrollo se organiza mediante un método incremental basado en prototipado. Este método permite construir el sistema por módulos, verificar avances, ajustar requerimientos y validar el flujo principal antes de ampliar funcionalidades.

La asistencia de IA utilizada durante la documentación o planificación del proyecto se considera apoyo auxiliar del proceso de trabajo. No constituye el objeto de estudio ni sustituye las decisiones técnicas del autor. La IA relevante para la tesis es la funcionalidad integrada al prototipo para generar una encuesta inicial desde el sistema web.

Tabla 4. Fases del desarrollo incremental del prototipo

Fase	Actividades principales	Producto esperado
Análisis	Levantamiento de necesidades, definición de actores, requerimientos funcionales y no funcionales.	Matriz de requerimientos y alcance delimitado.
Diseño	Diseño de arquitectura, modelo de datos, módulos, interfaz y flujo de asistencia de IA.	Diagramas, modelo relacional y estructura funcional.
Desarrollo	Implementación de frontend, backend, base de datos, autenticación, encuestas, preguntas y respuestas.	Prototipo funcional por incrementos.
Integración de IA	Consumo de API externa, presentación de sugerencias editables y control de revisión humana.	Módulo asistivo integrado y verificable.
Pruebas y validación	Ejecución de pruebas funcionales, medición de tiempos, tareas, SUS, TAM y satisfacción.	Matriz de resultados y análisis de validación.

3.3 Técnicas, evidencias e instrumentos de validación

Tabla 5. Técnicas, evidencias e instrumentos de validación

Técnica	Evidencia o instrumento	Aplicación en el proyecto
Revisión documental	Fichas o matriz de fuentes	Sustentar sistemas web, prototipado, IA asistiva, usabilidad, SUS y TAM.
Levantamiento de requerimientos	Matriz de requerimientos e historias de usuario	Definir funcionalidades del prototipo y criterios de aceptación.

Técnica	Evidencia o instrumento	Aplicación en el proyecto
Diseño técnico	Diagramas, modelo de datos y descripciones de módulos	Representar arquitectura, base de datos y flujo de operación.
Pruebas funcionales	Casos de prueba	Verificar autenticación, encuestas, preguntas, respuestas y reportes básicos.
Validación experimental	Registro de tiempos y tareas completadas	Comparar creación manual/Google Forms frente al uso del prototipo.
Evaluación de usabilidad	Encuesta basada en SUS	Medir facilidad de uso percibida después de realizar tareas.
Aceptación tecnológica	Encuesta basada en TAM	Medir utilidad percibida, facilidad de uso percibida e intención de uso.
Satisfacción	Encuesta breve y comentarios cualitativos	Identificar percepción general y oportunidades de mejora.

3.4 Participantes de la validación del prototipo

La validación experimental se realizará con un grupo pequeño de usuarios, entre 5 y 10 participantes, seleccionados por conveniencia dentro de un contexto académico o cercano al uso previsto del prototipo. Este tamaño es adecuado para una evaluación inicial de usabilidad y factibilidad en un prototipo, sin pretender generalizar resultados estadísticos a una población amplia.

Los participantes deberán ejecutar tareas básicas relacionadas con el flujo del sistema: iniciar sesión, crear una encuesta, registrar preguntas, configurar opciones, publicar una encuesta y enviar respuestas. Cuando el módulo de IA esté implementado, también deberán solicitar sugerencias, revisar el contenido generado y aceptar, editar o descartar propuestas.

3.5 Alcance de la validación y limitaciones del prototipo

La validación propuesta permite observar el comportamiento del prototipo y la percepción de usuarios en un escenario controlado. Sin embargo, no valida estadísticamente la calidad científica de las encuestas creadas, no garantiza que todas las preguntas resultantes sean aplicables a cualquier contexto y no sustituye la revisión humana o académica del contenido.

Los resultados deberán presentarse con los datos reales obtenidos durante la validación. Mientras las pruebas no se ejecuten, los apartados de resultados deben mantenerse como campos editables o análisis esperado, evitando inventar cifras finales.

CAPÍTULO 4: ANÁLISIS, DISEÑO Y DESARROLLO DEL PROTOTIPO

4.1 Requerimientos funcionales y no funcionales

Los requerimientos funcionales describen las acciones que el sistema debe permitir al usuario. Los requerimientos no funcionales establecen condiciones de calidad relacionadas con seguridad, usabilidad, mantenibilidad, integridad de datos y control de la asistencia de IA.

Los requerimientos se definen de acuerdo con el alcance tecnológico del prototipo y deben ser validados mediante casos de prueba. La asistencia de IA se considera un requerimiento funcional específico, pero su salida debe pasar por revisión humana antes de incorporarse al formulario.

Tabla 6. Requerimientos funcionales del prototipo

Código	Requerimiento funcional	Descripción
RF-01	Registro de usuarios	El sistema debe permitir registrar usuarios con datos básicos de acceso.
RF-02	Inicio de sesión	El sistema debe autenticar usuarios registrados y generar una sesión protegida.
RF-03	Gestión de roles	El sistema debe permitir asignar roles básicos para controlar permisos.
RF-04	Gestión de encuestas	El usuario debe poder crear, editar, consultar, eliminar o cambiar el estado de una encuesta.
RF-05	Gestión de preguntas	El usuario debe poder registrar, editar, consultar y eliminar preguntas reutilizables.
RF-06	Tipos y opciones de respuesta	El sistema debe permitir definir tipos de respuesta y opciones asociadas.
RF-07	Constructor de encuesta	El sistema debe permitir asociar preguntas a una encuesta y configurar su flujo.

Código	Requerimiento funcional	Descripción
RF-08	Encuesta pública	El sistema debe permitir responder una encuesta mediante enlace público.
RF-09	Almacenamiento de respuestas	El sistema debe registrar respuestas seleccionadas por los perfils.
RF-10	Resultados básicos	El sistema debe mostrar información resumida de las respuestas registradas.
RF-11	Asistencia de IA	El sistema debe sugerir preguntas, opciones o estructura inicial a partir de un tema u objetivo.
RF-12	Revisión de sugerencias	El usuario debe poder aceptar, editar o descartar las sugerencias generadas por IA.

Tabla 7. Requerimientos no funcionales del prototipo

Código	Requerimiento no funcional	Descripción
RNF-01	Usabilidad	La interfaz debe permitir ejecutar las tareas principales sin complejidad innecesaria.
RNF-02	Seguridad básica	Los módulos internos deben protegerse mediante autenticación y uso de token.
RNF-03	Mantenibilidad	El código debe organizarse por capas, servicios, componentes y responsabilidades.
RNF-04	Integridad de datos	La base de datos debe mantener relaciones y restricciones coherentes.
RNF-05	Rendimiento básico	Las operaciones principales deben responder en tiempos aceptables para un entorno prototipo.
RNF-06	Control de IA	Las sugerencias generadas deben tratarse como borradores editables y no aplicarse automáticamente.
RNF-07	Trazabilidad	Cuando sea posible, el sistema deberá registrar información mínima de solicitudes de IA y acciones del usuario.

4.2 Actores y casos de uso

Tabla 8. Actores principales del sistema

Código	Actor	Descripción
ACT-01	Usuario registrado	Persona que accede al sistema para crear encuestas, gestionar preguntas, configurar formularios y consultar resultados básicos.
ACT-02	Perfil	Persona que responde una encuesta generada desde el sistema mediante un enlace público.
ACT-03	Administrador o rol autorizado	Usuario con permisos para administrar información interna, según la configuración de roles del prototipo.

Tabla 9. Casos de uso principales del prototipo

Caso de uso	Actor principal	Descripción
Registrar usuario	Usuario registrado	Crear una cuenta para acceder al sistema.
Iniciar sesión	Usuario registrado	Validar credenciales y acceder a módulos protegidos.
Crear encuesta	Usuario registrado	Registrar una encuesta con título, descripción y estado.
Crear pregunta	Usuario registrado	Registrar una pregunta para el banco reutilizable.
Asociar pregunta a encuesta	Usuario registrado	Vincular preguntas con una encuesta mediante el constructor.
Definir tipo de respuesta	Usuario registrado	Seleccionar o configurar el tipo de respuesta disponible.
Solicitar sugerencias con IA	Usuario registrado	Enviar tema u objetivo para recibir sugerencias editables.
Responder encuesta	Perfil	Registrar respuestas mediante enlace público.
Consultar resultados	Usuario registrado	Revisar información básica de respuestas registradas.

4.3 Arquitectura técnica del prototipo

La arquitectura técnica del prototipo se organiza en frontend web, backend API, base de datos y servicio externo de IA. El frontend envía solicitudes al backend mediante endpoints

REST; el backend valida la información, ejecuta reglas de negocio, accede a la base de datos y devuelve respuestas estructuradas al cliente.

El backend funciona como punto central de control. Esta decisión evita que el frontend se conecte directamente con la base de datos y permite concentrar autenticación, autorización, reglas de negocio, documentación de endpoints, configuración CORS y consumo controlado de servicios externos.

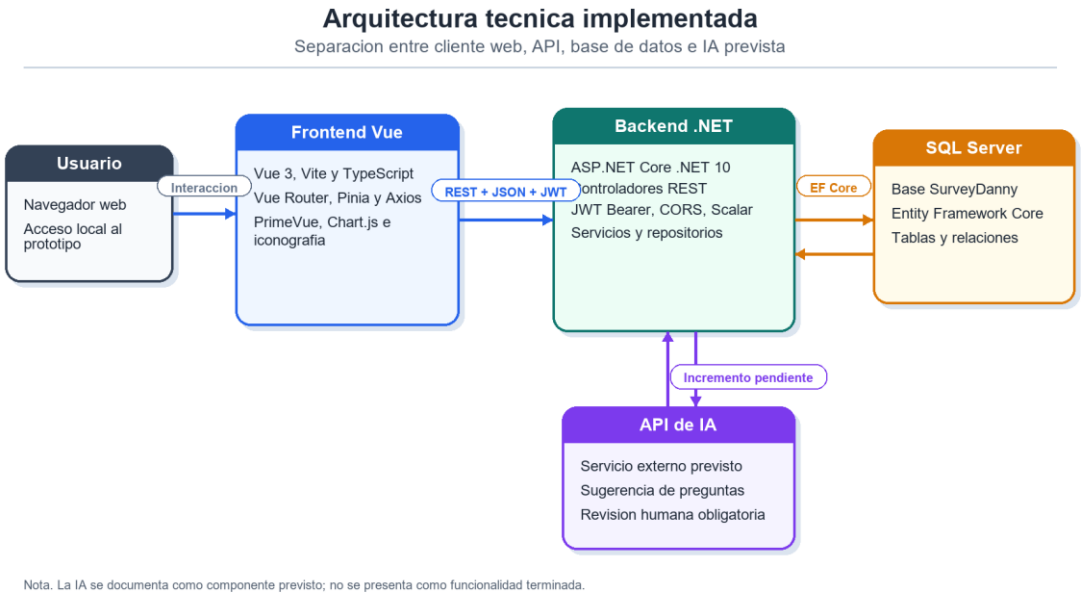


Figura 2. Arquitectura técnica del prototipo SurveyDanny. Nota. Elaboración propia a partir de la estructura del sistema.

4.3.1 Backend implementado

El backend se encuentra organizado en proyectos separados para API, servicios, repositorios y acceso a datos. El proyecto principal expone controladores de autenticación, encuestas, preguntas y encuestas para participantes. La capa de servicios concentra reglas de negocio; la capa de repositorio gestiona consultas y persistencia; y el proyecto de acceso a datos contiene entidades y contexto de Entity Framework Core.

La configuración principal se realiza en Program.cs, donde se registra el contexto de base de datos, la política CORS, la autenticación JWT Bearer, la autorización, OpenAPI y Scalar. Esta estructura permite probar endpoints desde la documentación interactiva y revisar contratos durante el desarrollo.

4.3.2 Frontend implementado

El frontend se ubica en C:\SurveyDannyWeb y está organizado por vistas, rutas, componentes, servicios API, tipos de datos y almacenamiento de sesión. Vue Router define rutas públicas y protegidas; Pinia conserva información de autenticación; Axios centraliza solicitudes HTTP; y los componentes reutilizables mantienen consistencia en formularios, listados, indicadores y reportes.

Las rutas principales incluyen login, registro, dashboard, encuestas, formulario de encuesta, preguntas, formulario de pregunta, constructor de encuesta, encuesta pública y reportes. Esta organización permite separar pantallas administrativas, vista pública y servicios de comunicación con el backend.

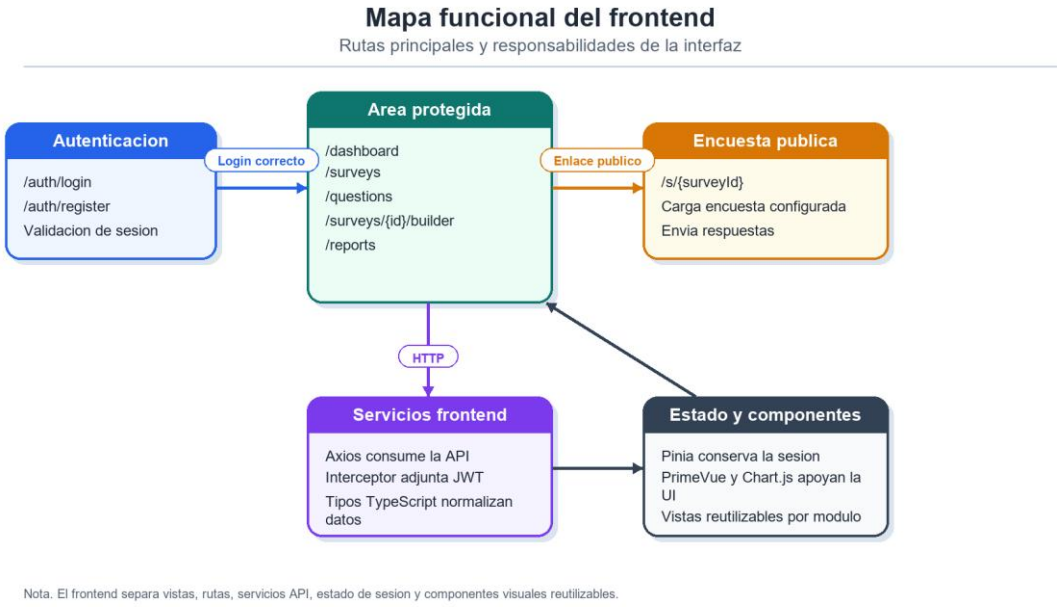


Figura 3. Mapa funcional del frontend del prototipo. Nota. Elaboración propia a partir de la estructura ubicada en C:\SurveyDannyWeb.

4.3.3 Comunicación entre frontend y backend

La comunicación entre frontend y backend se realiza mediante solicitudes HTTP en formato JSON. Cuando el usuario inicia sesión correctamente, el backend devuelve un token JWT; posteriormente, el frontend lo conserva y lo adjunta en las solicitudes protegidas.

Este esquema evita exponer directamente la base de datos a la interfaz y facilita que las reglas de negocio se mantengan dentro de la API. Además, la configuración CORS permite que el frontend local consuma el backend durante el desarrollo.

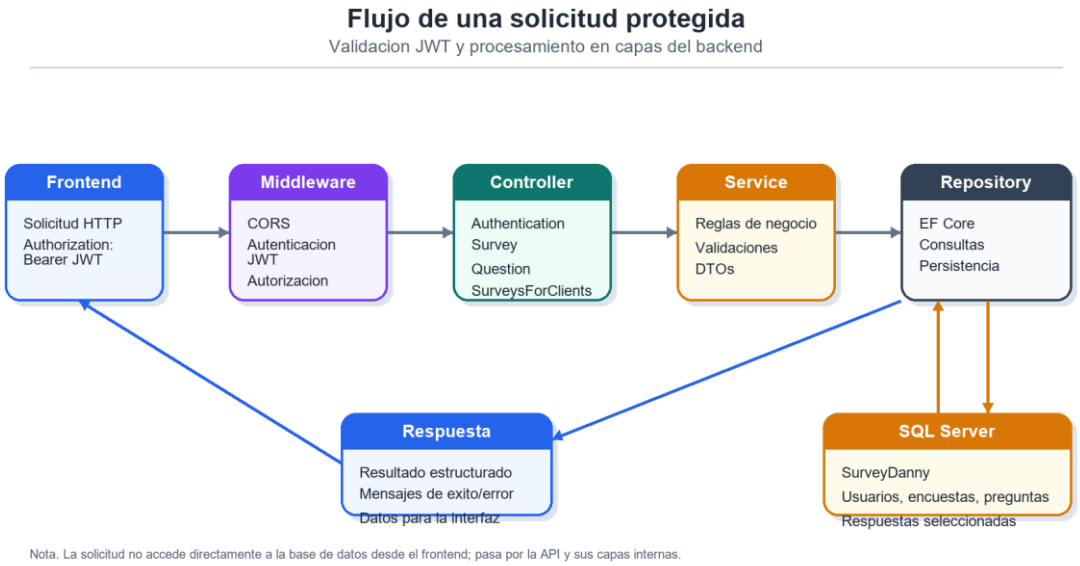


Figura 4. Flujo de una solicitud protegida entre frontend, backend y base de datos. Nota. Elaboración propia.

4.4 Modelamiento e implementación de la base de datos

La base de datos del prototipo se denomina SurveyDanny y fue implementada en SQL Server de acuerdo con las necesidades funcionales del sistema. Su estructura permite gestionar usuarios, roles, encuestas, preguntas, tipos de respuesta, textos de respuesta, respuestas disponibles, respuestas seleccionadas, reportes básicos y parámetros iniciales.

El modelo de datos responde al flujo principal del sistema: un usuario registrado puede crear encuestas; las preguntas se gestionan como catálogo reutilizable; las preguntas pueden asociarse a encuestas; los tipos de respuesta permiten clasificar respuestas disponibles; y las respuestas seleccionadas registran información entregada por participantes.

4.4.1 Modelo conceptual

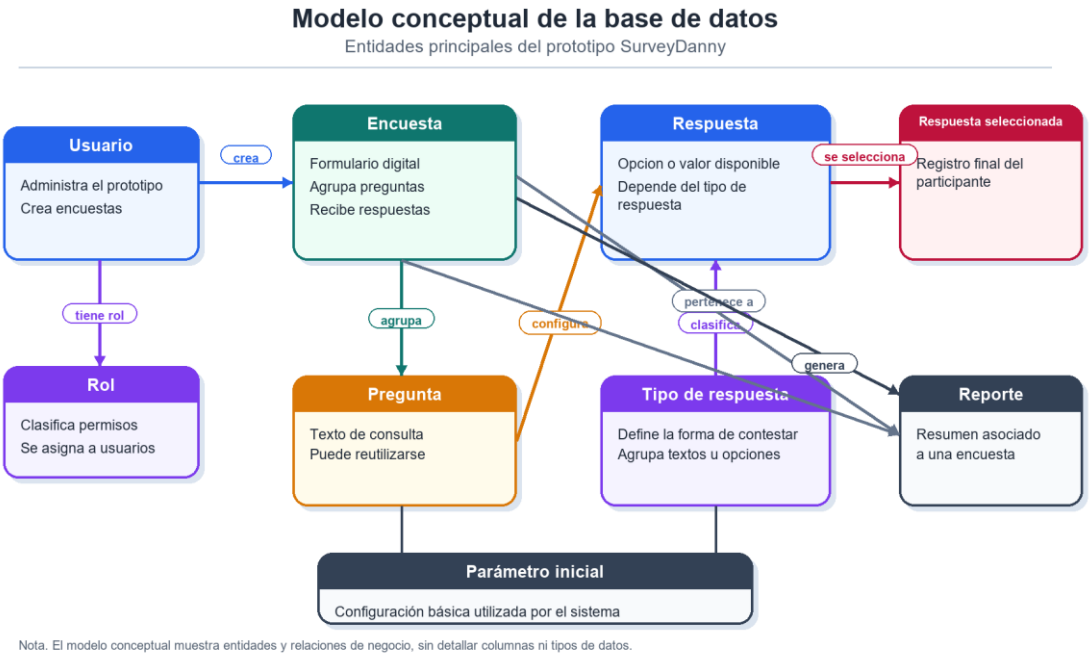


Figura 5. Modelo conceptual de la base de datos SurveyDanny. Nota. Elaboración propia con base en las entidades del prototipo.

Tabla 10. Modelo conceptual de entidades

Entidad	Descripción	Relación general
Usuario	Persona que accede al sistema para crear y administrar encuestas.	Crea encuestas y administra contenido.
Rol	Clasificación de permisos o tipo de usuario dentro del sistema.	Se asigna a usuarios.
Encuesta	Formulario digital creado por un usuario.	Agrupa preguntas y respuestas.
Pregunta	Elemento textual que solicita información dentro de una encuesta.	Puede asociarse a una o varias encuestas.
Tipo de respuesta	Catálogo que define la forma de respuesta.	Se relaciona con respuestas.
Respuesta	Opción o valor disponible para responder una pregunta.	Se relaciona con tipo de respuesta y pregunta dentro de encuesta.
Respuesta seleccionada	Registro de la respuesta entregada por un perfil.	Relaciona encuesta, pregunta y respuesta.

Entidad	Descripción	Relación general
Reporte	Información resumida asociada a una encuesta.	Se vincula con una encuesta.
Parámetro inicial	Configuración básica del sistema.	Apoya la configuración del prototipo.

4.4.2 Modelo lógico

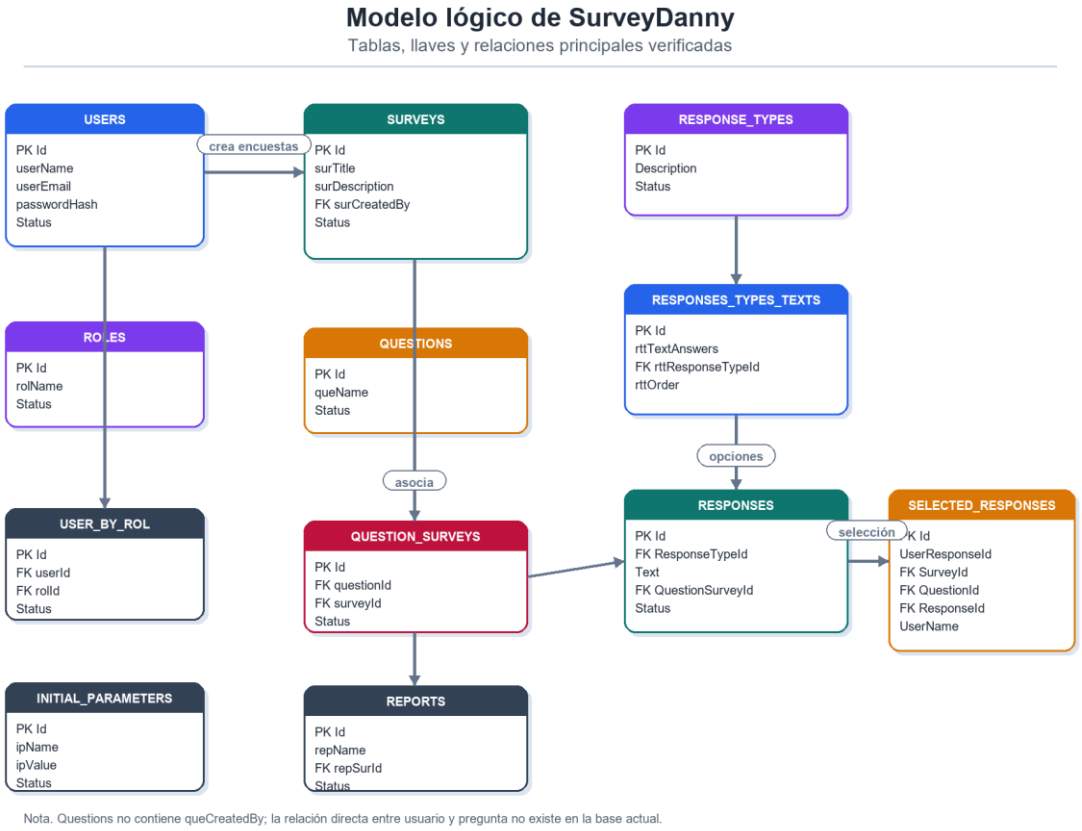


Figura 6. Modelo lógico resumido de la base de datos SurveyDanny. Nota. Elaboración propia con base en las entidades implementadas.

Tabla 11. Modelo lógico de relaciones

Entidad origen	Entidad destino	Relación lógica
Users	UserByRol	Un usuario puede tener un rol asociado.
Roles	UserByRol	Un rol puede asignarse a varios usuarios.
Users	Surveys	Un usuario puede crear varias encuestas.
Surveys	QuestionSurveys	Una encuesta puede contener varias preguntas.

Entidad origen	Entidad destino	Relación lógica
Questions	QuestionSurveys	Una pregunta puede formar parte de una o varias encuestas.
ResponseTypes	Responses	Un tipo de respuesta puede tener varias respuestas asociadas.
QuestionSurveys	Responses	Una respuesta puede vincularse con una pregunta dentro de una encuesta.
Surveys	SelectedResponses	Una encuesta puede tener respuestas seleccionadas.
Questions	SelectedResponses	Una pregunta puede recibir respuestas seleccionadas.
Responses	SelectedResponses	Una respuesta puede ser seleccionada por un perfil.
Surveys	Reports	Una encuesta puede tener reportes básicos asociados.

4.4.3 Modelo físico implementado

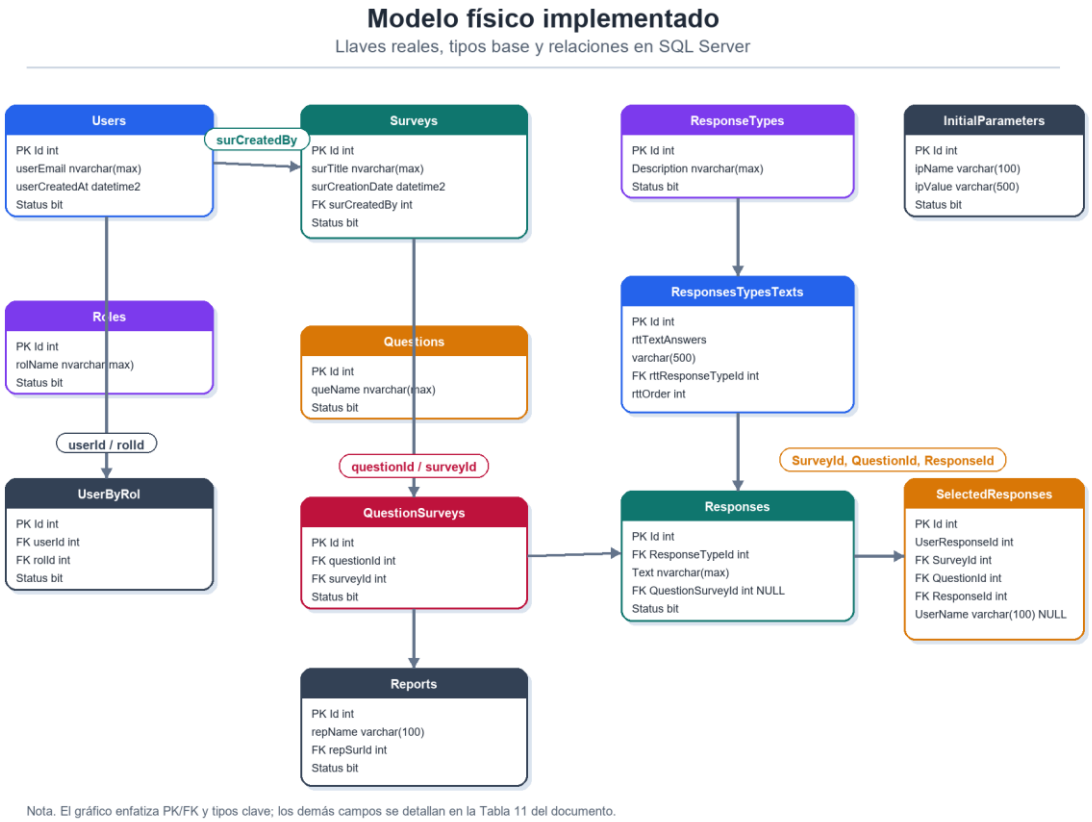


Figura 7. Modelo físico implementado en SQL Server. Nota. Elaboración propia con base en las tablas y claves de SurveyDanny.

Tabla 12. Tablas implementadas en la base de datos SurveyDanny

Tabla	Campos principales	Propósito
Users	Id, userName, userEmail, userPasswordHash, userPasswordSalt, userImageUrl, userCreatedAt, Status	Almacenar datos de usuarios registrados.
Roles	Id, rolName, Status	Definir roles disponibles.
UserByRol	Id, userId, rolId, Status	Relacionar usuarios con roles.
Surveys	Id, surTitle, surDescription, surCreationDate, surCreatedBy, Status	Registrar encuestas creadas.
Questions	Id, queName, Status	Almacenar preguntas disponibles.
QuestionSurveys	Id, questionId, surveyId, Status	Relacionar preguntas con encuestas.
ResponseTypes	Id, Description, Status	Definir tipos de respuesta.

Tabla	Campos principales	Propósito
ResponsesTypes Texts	Id, rttTextAnswers, rttResponseTypeId, rttOrder	Almacenar textos asociados a determinados tipos de respuesta.
Responses	Id, ResponseTypeId, Text, Status, QuestionSurveyId	Registrar respuestas disponibles o asociadas a preguntas.
SelectedResponses	Id, UserResponseId, SurveyId, QuestionId, ResponseId, Status, UserName	Almacenar respuestas seleccionadas por perfils.
Reports	Id, repName, repSurId, Status	Asociar reportes básicos con encuestas.
InitialParameters	Id, ipName, ipValue, Status	Almacenar parámetros iniciales de configuración.

Tabla 13. Relaciones implementadas en la base de datos SurveyDanny

Entidad origen	Entidad destino	Descripción
Users	Surveys	Un usuario puede crear varias encuestas mediante el campo surCreatedBy.
Surveys	QuestionSurveys	Una encuesta puede asociarse con varias preguntas.
Questions	QuestionSurveys	Una pregunta puede formar parte de una o varias encuestas.
ResponseTypes	Responses	Un tipo de respuesta puede asociarse con varias respuestas.
QuestionSurveys	Responses	Las respuestas pueden vincularse con una pregunta dentro de una encuesta.
Surveys	SelectedResponses	Una encuesta puede tener varias respuestas seleccionadas por perfils.
Questions	SelectedResponses	Cada respuesta seleccionada se relaciona con una pregunta.
Responses	SelectedResponses	Cada selección apunta a una respuesta registrada.
Surveys	Reports	Una encuesta puede tener reportes básicos asociados.

Entidad origen	Entidad destino	Descripción
Roles	UserByRol	Un rol puede asignarse a varios usuarios.
Users	UserByRol	Un usuario puede tener un rol asociado.

4.5 Módulo de usuarios

El módulo de usuarios permite registrar y autenticar a las personas que utilizan el sistema. El registro almacena nombre de usuario, correo electrónico y credenciales protegidas mediante hash y salt. El inicio de sesión valida credenciales y, en caso de éxito, genera un token JWT que el frontend conserva para consumir endpoints protegidos.

El backend incorpora asignación de roles mediante endpoints para perfiles autorizados. En el frontend, la sesión se gestiona con Pinia y el token se adjunta automáticamente en las solicitudes realizadas con Axios. Esta implementación restringe el acceso a módulos internos como encuestas y preguntas, y establece una base de seguridad suficiente para el alcance académico del prototipo.

4.6 Módulo de encuestas, preguntas y respuestas

El módulo de encuestas permite crear, listar, editar y eliminar formularios digitales desde el frontend. Cada encuesta cuenta con título, descripción, fecha de creación, estado y usuario creador. El backend expone estas operaciones mediante controladores y las persiste en SQL Server.

El módulo de preguntas permite registrar, consultar, modificar y eliminar preguntas dentro de un banco reutilizable. El constructor de encuestas permite asociar preguntas existentes o crear una pregunta manual desde la misma vista, seleccionar un tipo de respuesta y guardar la configuración del formulario.

El módulo de respuestas permite cargar una encuesta pública mediante la ruta correspondiente, presentar una secuencia de preguntas al participante y enviar respuestas seleccionadas al backend. Las respuestas se almacenan en SelectedResponses y se relacionan con encuesta, pregunta, respuesta y usuario o identificador disponible.



Figura 8. Ciclo funcional de creación, publicación y respuesta de una encuesta. Nota. Elaboración propia.

4.7 Integración de asistencia de IA

La integración con inteligencia artificial se define como un módulo funcional del prototipo. En el frontend existe la ruta /surveys/generate y la vista SurveyGptView, donde el usuario ingresa el tema de la encuesta, el objetivo y la cantidad de preguntas. Esta información se envía al backend mediante el servicio surveyGptApi.

En el backend, el controlador SurveyGPT expone el endpoint GenerateSurveyWithGpt y utiliza el servicio SurveyGPT para construir el prompt, solicitar una respuesta estructurada a una API externa y guardar la encuesta generada. La respuesta esperada incluye título, descripción, preguntas, tipo de respuesta y opciones cuando corresponda.

El incremento actual genera y guarda una encuesta inicial que luego puede revisarse desde el constructor, la vista pública y los reportes. Por coherencia con el enfoque de IA asistida, esta funcionalidad debe presentarse como apoyo funcional y no como validación automática de instrumentos. Como mejora recomendada, el flujo puede incorporar una pantalla de revisión previa para editar, aceptar o descartar sugerencias antes del guardado definitivo.

Si se requiere mayor trazabilidad, podrá añadirse una tabla o bitácora que registre solicitud, respuesta, usuario, encuesta y fecha de ejecución. Esta trazabilidad ayudaría a documentar el uso controlado del módulo sin convertir la IA en el eje central del proyecto.



Figura 9. Flujo previsto para generación asistida de preguntas mediante API de IA. Nota. Elaboración propia.

Tabla 14. Flujo previsto para la asistencia de IA

Paso	Actor	Descripción
1	Usuario	Ingresa tema, objetivo y cantidad aproximada de preguntas.
2	Frontend	Envía la solicitud al backend desde la vista /surveys/generate.
3	Backend	Valida la información, construye el prompt y consume la API externa.
4	API de IA	Genera una estructura de encuesta con título, descripción, preguntas, tipos y opciones.
5	Backend	Procesa la respuesta, guarda encuesta, preguntas y relaciones en la base de datos.
6	Usuario	Revisa la encuesta generada desde el constructor, la vista pública y los reportes.

4.8 Estado actual del desarrollo

Al momento de la actualización de este documento, el proyecto cuenta con avances verificables en base de datos, backend, frontend y servidor local. La base de datos SurveyDanny se encuentra implementada en SQL Server con tablas y relaciones para usuarios, roles, encuestas, preguntas, tipos de respuesta, respuestas, respuestas seleccionadas, reportes y parámetros iniciales.

En el backend se dispone de una API ASP.NET Core conectada mediante Entity Framework Core, con documentación de endpoints en Scalar, configuración CORS, autenticación JWT y módulos para registro, inicio de sesión, roles, encuestas, preguntas, configuración de encuestas para participantes, almacenamiento de respuestas, consulta de respuestas por encuesta y generación asistida mediante API externa.

El frontend ubicado en C:\SurveyDannyWeb incluye pantallas de login, registro, dashboard, listado y formularios de encuestas, listado y formularios de preguntas, generación de encuesta con IA, constructor manual, enlace público de respuesta y vista de reportes con indicadores, gráficos y tablas. Los pendientes principales son ejecutar pruebas integrales con datos reales, validar el consumo externo de IA con credenciales activas, reforzar la revisión previa de sugerencias generadas y aplicar la validación experimental con usuarios.

Tabla 15. Estado actual y pendientes del prototipo

Componente	Estado actual	Observación
Base de datos SurveyDanny	Implementado	Modelo relacional en SQL Server con entidades principales.
Backend ASP.NET Core	Implementado	API con Entity Framework Core, JWT, CORS y Scalar.
Autenticación y roles	Implementado	Registro, login, hash/salt, JWT y asignación de roles.
Frontend Vue/Vite	Implementado	Login, registro, dashboard, encuestas, preguntas, generación con IA, constructor, encuesta pública y reportes.
Encuestas y preguntas	Implementado	CRUD funcional y relación encuesta-preguntas mediante constructor.
Respuestas públicas	Implementado	Carga de encuesta pública y almacenamiento de respuestas seleccionadas de usuarios autenticados.
Reportes	Implementado parcialmente	Vista con indicadores, gráficos y tablas; depende de datos reales y procedimiento de resumen.
Asistencia de IA	Implementado parcialmente	Existe vista frontend, endpoint backend y guardado de encuesta generada; falta prueba integral con API real y revisión previa editable.

Componente	Estado actual	Observación
Catálogo de tipos de respuesta	Parcial	El backend usa tablas de tipos; el frontend mantiene catálogo de respaldo si no existe endpoint dedicado.

4.9 Verificación técnica del frontend y backend

Para ajustar el documento al estado real del desarrollo, se revisaron las carpetas C:\SurveyDannyWeb y C:\Users\USER\source\repos\BackendEncuestas. Esta revisión permitió confirmar que el sistema cuenta con una aplicación frontend Vue/Vite/TypeScript y una solución backend organizada en proyectos de API, servicios, repositorio y acceso a datos.

La revisión técnica confirmó que el frontend supera la verificación de tipos y que el backend compila correctamente. En el backend se identificaron advertencias de nulabilidad y referencias posiblemente nulas, por lo que se recomienda tratarlas como mejora técnica antes de una entrega final o despliegue real.

Tabla 16. Evidencia técnica verificada

Elemento revisado	Evidencia encontrada	Estado
Frontend	Proyecto C:\SurveyDannyWeb con Vue 3, Vite, TypeScript, Pinia, Axios, PrimeVue, Chart.js y rutas protegidas.	Verificado
Rutas frontend	Login, registro, dashboard, encuestas, generación con IA, constructor, responder encuesta, preguntas y reportes.	Verificado
Backend	Solución BackendEncuestas con proyectos AccessData, Repository, Services y TesisProyectoD1.	Verificado
Controladores	Authentication, Survey, Question, SurveysForClients y SurveyGPT.	Verificado
Base de datos	Contexto SurveyDannyContext con usuarios, roles, encuestas, preguntas, tipos, respuestas, seleccionadas y reportes.	Verificado
Compilación frontend	La verificación de tipos de TypeScript finalizó sin errores.	Aprobado

Elemento revisado	Evidencia encontrada	Estado
Compilación backend	La compilación .NET finalizó sin errores y con advertencias de nulabilidad.	Aprobado con advertencias
IA asistida	Vista, endpoint y guardado asistido.	Parcial

CAPÍTULO 5: RESULTADOS Y VALIDACIÓN

5.1 Introducción del capítulo

Este capítulo presenta la estructura de validación del prototipo SurveyDanny y un escenario de validación modelado con valores referenciales. Debido a que el sistema corresponde a un prototipo funcional académico y algunas funcionalidades todavía requieren estabilización técnica, los datos incluidos en las tablas de resultados se declaran como valores referenciales modelados. Su finalidad es mostrar cómo deben organizarse los indicadores de validación, sin reemplazar una evaluación experimental definitiva con usuarios reales.

Nota metodológica. Debido a que el sistema corresponde a un prototipo funcional en etapa académica, este capítulo presenta un escenario de validación modelado con cinco perfiles de usuario referenciales. Los valores reportados tienen finalidad preliminar y permiten estructurar el análisis de funcionalidad, usabilidad, aceptación tecnológica, uso de IA y reducción de tiempo. Estos resultados no sustituyen una validación experimental definitiva con usuarios reales, la cual se plantea como actividad pendiente para una etapa posterior.

5.2 Resultados funcionales del sistema

La validación funcional permite comprobar si los módulos principales del prototipo cumplen las acciones esperadas. Para este escenario referencial se consideran funcionalidades esenciales del flujo de uso: autenticación, creación de encuestas, gestión de preguntas, asistencia con IA, respuesta pública y visualización de reportes básicos.

Tabla 17. Resultados funcionales modelados del prototipo

Nº	Funcionalidad evaluada	Resultado esperado	Resultado obtenido	Estado
1	Registro e inicio de sesión	Acceso correcto al panel principal.	El flujo permitió acceder con usuario	Aprobado

			registrado.	
2	Creación de encuesta	La encuesta se registra correctamente.	El prototipo permitió crear una encuesta básica.	Aprobado
3	Gestión de preguntas	Las preguntas se guardan y se muestran correctamente.	Se pudieron registrar preguntas y asociarlas a una encuesta.	Aprobado
4	Generación con IA	El sistema genera preguntas y opciones editables.	Se obtuvo una estructura inicial, pero requiere revisión humana y estabilización técnica.	Parcial
5	Respuesta de encuesta pública	Las respuestas se almacenan correctamente.	El flujo permitió responder una encuesta publicada.	Aprobado
6	Reportes básicos	Los resultados se presentan de forma clara.	Se visualizaron resultados generales, pero falta fortalecer la agregación completa.	Parcial

5.3 Validación técnica del prototipo

La validación técnica se orienta a comprobar que los componentes principales del prototipo se encuentren implementados y permitan ejecutar el flujo básico del sistema. Esta revisión no sustituye una auditoría de calidad de software, pero permite documentar el estado funcional del prototipo en su etapa académica.

Tabla 18. Estado técnico referencial del prototipo

Componente	**Resultado referencial**	**Observación**
Base de datos SurveyDanny	Implementado	Modelo relacional con entidades principales para usuarios, encuestas, preguntas y respuestas.
Backend ASP.NET Core	Implementado	API con controladores, Entity Framework Core, JWT, CORS y documentación técnica.
Frontend Vue/Vite	Implementado	Incluye vistas de autenticación,

		dashboard, encuestas, preguntas, constructor, encuesta pública y reportes.
Asistencia con IA	Implementado parcialmente	Existe flujo de generación, pero requiere estabilizar pruebas con API real y revisión editable previa.
Reportes	Parcial	Se requiere mejorar agregación, visualización y exportación de resultados.
Validación con usuarios reales	Pendiente	Debe ejecutarse para reemplazar los datos modelados de este capítulo.

5.4 Validación experimental

La validación experimental propuesta compara un proceso manual o una herramienta general de creación de encuestas frente al uso del prototipo SurveyDanny. Para el escenario modelado se consideran cinco perfiles de usuario referenciales, cada uno con tareas básicas asociadas al flujo principal del sistema.

Tabla 19. Escenario de validación experimental modelado

Elemento	**Descripción**
Objetivo de la prueba	Evaluar el funcionamiento del prototipo web en un escenario de creación, gestión y respuesta de encuestas digitales.
Perfiles de usuario	5 perfiles de usuario referenciales.
Perfil de perfils	Usuarios generales con experiencia básica en navegación web.
Número de encuestas evaluadas	1 encuesta de prueba por perfil.
Criterios aplicados	Cumplimiento de tareas, tiempo, errores, usabilidad, aceptación tecnológica y percepción de IA.
Instrumentos utilizados	Lista de tareas, registro de tiempos, matriz SUS, matriz TAM y evaluación referencial de IA.
Fecha de aplicación	[Completar cuando se realice la prueba real].

Resultados generales	Datos modelados para estructurar el análisis preliminar; deben reemplazarse por evidencia real.

Tabla 20. Resultados modelados por perfil

Perfil	**Tiempo manual**	**Tiempo con SurveyDanny**	**Tareas completadas**	**Observación referencial**
P1	18 min	12 min	6 de 7	Completó el flujo principal; requirió aclaración menor en reportes.
P2	20 min	13 min	5 de 7	Presentó dudas en asistencia de IA y consulta de reportes.
P3	17 min	12 min	6 de 7	Completó creación y respuesta; sugirió mejorar mensajes de confirmación.
P4	21 min	15 min	5 de 7	Identificó lentitud o inconsistencia en el flujo de IA.
P5	19 min	13 min	6 de 7	Usó el sistema con pocas dificultades; recomendó mejorar reportes.
Promedio	19 min	13 min	5,6 de 7	Reducción modelada aproximada de 31,6 %.

5.5 Evaluación de usabilidad mediante SUS

La escala SUS permite evaluar la usabilidad percibida del sistema mediante 10 ítems. Para esta versión del documento, los valores corresponden a valores referenciales modelados y deben reemplazarse por respuestas reales de usuarios cuando se aplique la prueba.

Tabla 21. Evaluación SUS modelada

Nº	Ítem SUS adaptado al sistema	Valor referencial modelado	Observación
1	Considero que me gustaría utilizar este sistema con frecuencia.	4,0	Percepción favorable de uso futuro.
2	Considero que el sistema es innecesariamente complejo.	2,4	Complejidad moderada, con margen de mejora.
3	Considero que el sistema es fácil de usar.	4,2	Buena percepción de facilidad de uso.
4	Considero que necesitaría apoyo técnico para usar el sistema.	2,0	Baja dependencia de apoyo técnico.
5	Considero que las funciones del sistema están bien integradas.	3,8	Integración aceptable de módulos.
6	Considero que existe demasiada inconsistencia en el sistema.	2,2	Se perciben algunas inconsistencias propias del prototipo.
7	Considero que la mayoría de usuarios aprendería a usar el sistema rápidamente.	4,2	Buena facilidad de aprendizaje.
8	Considero que el sistema es incómodo de utilizar.	1,8	Baja incomodidad percibida.
9	Me sentí seguro al utilizar el sistema.	4,2	Percepción positiva de control durante el uso.
10	Necesité aprender muchas cosas antes de poder utilizar el sistema.	2,4	Requiere orientación inicial moderada.

Puntaje SUS referencial modelado: 74/100.

Interpretación: el puntaje ubica al prototipo en un nivel de buena usabilidad, aunque con aspectos técnicos por mejorar en consistencia, estabilidad y claridad del flujo.

Tabla 22. Interpretación de puntaje SUS

Puntaje SUS	**Interpretación**
0 - 50	Usabilidad baja.
51 - 68	Usabilidad aceptable con aspectos por mejorar.
69 - 80	Buena usabilidad.
81 - 100	Excelente usabilidad.

5.6 Evaluación de aceptación tecnológica mediante TAM

El modelo TAM permite analizar la aceptación tecnológica del sistema mediante utilidad percibida, facilidad de uso percibida e intención de uso. Los valores presentados a continuación son referenciales y modelados, por lo que tienen finalidad preliminar.

Tabla 23. Evaluación TAM modelada

Dimensión TAM	**Indicador**	**Pregunta sugerida**	**Escala**	Valor referencial modelado
Utilidad percibida	Mejora del proceso	El sistema mejora el proceso de creación y gestión de encuestas digitales.	Likert 1-5	4,2
Utilidad percibida	Apoyo al usuario	El sistema facilita la organización de preguntas y respuestas.	Likert 1-5	4,0
Facilidad de uso percibida	Claridad de uso	El sistema es fácil de comprender y utilizar.	Likert 1-5	3,8
Facilidad de uso percibida	Aprendizaje	Aprender a usar el sistema fue sencillo.	Likert 1-5	3,8
Intención de uso	Uso futuro	Estaría dispuesto a utilizar el sistema si estuviera más completo.	Likert 1-5	4,0
Satisfacción general	Valoración global	Considero que el sistema es útil para	Likert 1-5	4,0

		apoyar la creación de encuestas.		

El promedio TAM referencial modelado es de 4,0/5, lo que representa una aceptación tecnológica favorable para un prototipo académico. Sin embargo, este resultado debe confirmarse con usuarios reales antes de redactar conclusiones definitivas.

5.7 Evaluación del uso de inteligencia artificial

El documento declara que SurveyDanny incorpora inteligencia artificial generativa como funcionalidad asistiva para sugerir preguntas, opciones o una estructura inicial de encuesta. Por ello, se incluye una evaluación específica del componente de IA. La matriz siguiente presenta resultados modelados y debe sustituirse por datos reales una vez probada la integración con usuarios.

Tabla 24. Evaluación modelada del componente de IA

Nº	**Criterio evaluado**	**Descripción**	Valor referencial modelado	**Observación**
1	Pertinencia de preguntas generadas	Evalúa si las preguntas generadas son adecuadas al tema y objetivo ingresado.	4,0/5	Las sugerencias se consideran pertinentes en un escenario general.
2	Coherencia temática	Evalúa si el contenido mantiene relación con el tema de la encuesta.	4,0/5	La coherencia es favorable, aunque requiere revisión.
3	Necesidad de edición humana	Evalúa el grado de ajuste requerido antes de usar el contenido.	3,0/5	Se requiere edición moderada del usuario.
4	Tiempo ahorrado	Evalúa si la IA reduce el tiempo inicial de elaboración.	4,0/5	Se observa ahorro referencial en generación inicial.
5	Utilidad percibida	Evalúa si el usuario considera útil la asistencia de IA.	4,2/5	La utilidad percibida es favorable.

6	Claridad del contenido generado	Evalúa si las preguntas y opciones son comprensibles.	3,8/5	La claridad es aceptable, con ajustes menores.
7	Alineación con el objetivo	Evalúa si la IA respeta el objetivo definido por el usuario.	4,0/5	La alineación es favorable en el escenario modelado.

La evaluación modelada muestra que la IA puede aportar utilidad como apoyo inicial, pero no debe reemplazar la revisión humana. La necesidad de edición moderada confirma que el contenido generado debe mantenerse editable y sujeto a aprobación del usuario.

5.8 Comparación de tiempo con y sin apoyo del sistema

La comparación de tiempo permite estimar si el prototipo reduce el esfuerzo requerido para crear y organizar una encuesta. Los valores presentados son referenciales y modelados, por lo que deben ser reemplazados por mediciones reales.

Tabla 25. Comparación modelada de tiempo

Actividad	**Tiempo manual**	**Tiempo con sistema**	**Diferencia**	**Porcentaje de reducción**
Revisión inicial de la tarea	4 min	3 min	1 min	25 %
Creación de encuesta	6 min	4 min	2 min	33,3 %
Registro de preguntas	6 min	4 min	2 min	33,3 %
Generación o ajuste inicial con IA	3 min	1 min	2 min	66,7 %
Consulta de resultado básico	2 min	1 min	1 min	50 %
Promedio total	19 min	13 min	6 min	31,6 %

5.9 Resultados por objetivo

La siguiente tabla relaciona los objetivos específicos con las actividades realizadas y los resultados referenciales. Esta trazabilidad permite verificar que las conclusiones respondan a los objetivos planteados.

Tabla 26. Relación entre objetivos específicos y resultados

Objetivo específico	Actividad realizada	Resultado obtenido	Evidencia en el documento
Levantar requerimientos funcionales y no funcionales.	Identificación de módulos, actores y requerimientos del prototipo.	Requerimientos definidos para usuarios, encuestas, preguntas, respuestas, resultados e IA.	Capítulos 1, 3 y 4.
Diseñar arquitectura, interfaz y base de datos.	Diseño de arquitectura cliente-servidor, modelo relacional y módulos frontend/backend.	Arquitectura estructurada en frontend, backend y base de datos SQL Server.	Capítulos 2 y 4.
Desarrollar el prototipo funcional.	Implementación de autenticación, encuestas, preguntas, respuestas públicas y reportes básicos.	Prototipo funcional con módulos principales implementados y algunos componentes parciales.	Capítulo 4 y tabla 18.
Integrar asistencia de IA.	Incorporación de flujo para generar estructura inicial de encuesta.	Funcionalidad de IA implementada parcialmente y evaluada de forma modelada.	Capítulo 4 y sección 5.7.
Validar experimentalmente el prototipo.	Estructuración de pruebas funcionales, SUS, TAM, tiempos y satisfacción.	Validación modelada referencial pendiente de reemplazo por datos reales.	Capítulo 5.

5.10 Análisis general de resultados

Los resultados modelados permiten observar que SurveyDanny presenta un comportamiento funcional favorable en los módulos principales de autenticación, creación de encuestas, gestión de preguntas y respuesta pública. No obstante, las funcionalidades de

asistencia con IA y reportes se mantienen como áreas que requieren mayor estabilización técnica antes de una validación real con usuarios.

El puntaje SUS referencial modelado de 74/100 ubica al prototipo en un nivel de buena usabilidad, con oportunidades de mejora relacionadas con consistencia, claridad del flujo y estabilidad. Por su parte, el promedio TAM referencial modelado de 4,0/5 muestra una percepción favorable de utilidad y facilidad de uso, aunque estos datos deben interpretarse únicamente como referencia metodológica.

La comparación modelada de tiempos muestra una reducción aproximada de 31,6 % al utilizar el prototipo frente a un proceso manual o herramienta general. Esta reducción debe comprobarse posteriormente mediante mediciones reales con usuarios. En conjunto, el capítulo evidencia que el prototipo tiene una estructura validable, pero todavía requiere completar la aplicación experimental para sustentar conclusiones definitivas.

CAPÍTULO 6: CONCLUSIONES Y RECOMENDACIONES

6.1 Conclusiones

6. El levantamiento de requerimientos permitió delimitar el alcance funcional del prototipo SurveyDanny, identificando módulos de autenticación, gestión de encuestas, gestión de preguntas, almacenamiento de respuestas, reportes básicos y asistencia de inteligencia artificial. Esta definición permitió orientar el desarrollo hacia una solución web centrada en la creación y organización de encuestas digitales.
7. El diseño de la arquitectura cliente-servidor permitió separar la interfaz de usuario, la lógica de negocio y la persistencia de datos. Esta separación favorece la mantenibilidad del prototipo, reduce el acoplamiento entre componentes y permite que el frontend consuma servicios del backend sin acceder directamente a la base de datos.
8. El desarrollo del prototipo evidenció que los módulos principales permiten ejecutar el flujo base de una aplicación de encuestas digitales: registro e inicio de sesión, creación de encuestas, gestión de preguntas, respuesta pública y consulta de resultados básicos. Sin embargo, los módulos de reportes y asistencia con IA requieren mayor estabilización antes de una validación definitiva.
9. La integración de inteligencia artificial se incorporó como funcionalidad asistiva para generar una estructura inicial de encuesta. Su uso debe mantenerse bajo revisión

humana obligatoria, debido a que el contenido generado puede requerir ajustes de pertinencia, claridad y coherencia antes de ser utilizado.

10. El escenario modelado de validación permitió estructurar indicadores de usabilidad, aceptación tecnológica, tiempo y evaluación de IA. El puntaje SUS referencial modelado de 74/100 y el promedio TAM referencial modelado de 4,0/5 sugieren una percepción favorable del prototipo, pero estos valores no sustituyen una validación experimental con usuarios reales.
11. La validación real del prototipo constituye una actividad pendiente. Para cerrar de forma definitiva los resultados, se deben aplicar pruebas con usuarios, registrar tiempos, calcular SUS y TAM, documentar errores observados y reemplazar los datos modelados/referenciales por evidencia empírica.

Tabla 27. Trazabilidad entre objetivos, resultados y conclusiones

Objetivo	**Resultado relacionado**	**Conclusión asociada**
Levantar requerimientos.	Matriz de requerimientos y alcance funcional del prototipo.	El levantamiento permitió delimitar módulos y orientar el desarrollo.
Diseñar arquitectura e interfaz.	Arquitectura cliente-servidor, frontend, backend y base de datos.	La separación de responsabilidades favorece mantenibilidad y organización.
Desarrollar el prototipo.	Módulos principales implementados y componentes parciales identificados.	El prototipo ejecuta el flujo base, pero requiere estabilizar reportes e IA.
Integrar asistencia de IA.	Flujo de generación inicial de preguntas y opciones.	La IA funciona como apoyo editable y no reemplaza revisión humana.
Validar experimentalmente.	Escenario modelado de SUS, TAM, tiempos e IA.	La validación real queda pendiente para sustentar resultados definitivos.

6.2 Recomendaciones

12. Ejecutar una validación real con usuarios para reemplazar los datos modelados/referenciales del capítulo 5 por evidencia empírica verificable.
13. Ampliar la muestra de validación con estudiantes, docentes o usuarios que requieran crear y gestionar encuestas digitales en un contexto académico.

14. Fortalecer el módulo de reportes mediante agregación completa de resultados, filtros, exportación de datos y visualizaciones más claras.
15. Estabilizar la integración con la API de inteligencia artificial, protegiendo credenciales, controlando errores y permitiendo revisión previa del contenido generado antes de guardarlo definitivamente.
16. Mejorar la interfaz de usuario de acuerdo con los resultados de SUS, priorizando consistencia visual, mensajes de confirmación, prevención de errores y claridad de navegación.
17. Aplicar TAM con usuarios reales para identificar barreras de aceptación, utilidad percibida, facilidad de uso e intención de uso futuro.
18. Documentar errores, limitaciones y decisiones técnicas durante cada iteración, con el fin de diferenciar claramente el alcance del prototipo frente a una solución institucional en producción.
19. Considerar como trabajo futuro la integración con plataformas educativas, reportes avanzados, analítica de resultados, trazabilidad de solicitudes de IA y despliegue controlado en un entorno académico real.

BIBLIOGRAFÍA

Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., y Thomas, D. (2001). Manifesto for Agile Software Development. https://agilemanifesto.org/

Brooke, J. (1996). SUS: A quick and dirty usability scale. En P. W. Jordan, B. Thomas, B. A. Weerdmeester y I. L. McClelland (Eds.), Usability Evaluation in Industry. Taylor & Francis.

Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. MIS Quarterly, 13(3), 319-340.

Hevner, A. R., March, S. T., Park, J., y Ram, S. (2004). Design science in information systems research. MIS Quarterly, 28(1), 75-105.

International Organization for Standardization. (2018). ISO 9241-11:2018 Ergonomics of human-system interaction — Part 11: Usability: Definitions and concepts. [Verificar datos bibliográficos exactos en APA 7].

Microsoft. (2026). ASP.NET documentation. Microsoft Learn. Recuperado el 6 de junio de 2026, de https://learn.microsoft.com/en-us/aspnet/core/

Microsoft. (2026). Microsoft SQL documentation. Microsoft Learn. Recuperado el 6 de junio de 2026, de https://learn.microsoft.com/en-us/sql/

Nielsen, J. (1993). Usability Engineering. Academic Press. [Verificar datos bibliográficos exactos en APA 7].

OpenAI. (2026). OpenAI API Platform Documentation. OpenAI Developers. Recuperado el 6 de junio de 2026, de https://developers.openai.com/api/docs/

Pressman, R. S., y Maxim, B. R. (2020). Software Engineering: A Practitioner's Approach (9th ed.). McGraw-Hill.

Russell, S., y Norvig, P. (2021). Artificial Intelligence: A Modern Approach (4th ed.). Pearson. [Verificar datos bibliográficos exactos en APA 7].

Sommerville, I. (2016). Software Engineering (10th ed.). Pearson.

TypeScript. (2026). The TypeScript Handbook. Recuperado el 6 de junio de 2026, de https://www.typescriptlang.org/docs/handbook/intro.html

Vite. (2026). Why Vite. Recuperado el 6 de junio de 2026, de https://vite.dev/guide/why.html

Vue.js. (2026). Introduction. Recuperado el 6 de junio de 2026, de https://vuejs.org/guide/introduction.html

Fuente reciente sobre inteligencia artificial generativa aplicada a educación. [Verificar datos bibliográficos exactos en APA 7].

ANEXOS

Anexo A. Matriz de avance del proyecto

Tabla 23. Avances actuales y pendientes

Componente	Estado	Observación
Base de datos SurveyDanny	Implementado	Modelo relacional en SQL Server con entidades principales del sistema.

Componente	Estado	Observación
Backend ASP.NET Core	Implementado	API con Entity Framework Core, JWT, CORS y Scalar.
Autenticación y roles	Implementado	Registro, login, hash/salt, JWT y asignación de roles.
Frontend Vue/Vite	Implementado	Login, registro, dashboard, encuestas, preguntas, constructor, encuesta pública y reportes.
Encuestas y preguntas	Implementado	CRUD funcional y relación encuesta-preguntas mediante constructor.
Respuestas públicas	Implementado	Carga de encuesta pública y almacenamiento de respuestas seleccionadas.
Reportes	Parcial	Falta agregación completa de resultados desde backend.
Asistencia con IA	Implementado parcialmente	Existe endpoint, servicio y vista frontend; falta evidencia de prueba integral con API real y revisión previa editable.
Validación experimental	Pendiente	Debe ejecutarse con 5 a 10 usuarios y métricas definidas.

Anexo B. Cronograma referencial de trabajo

Tabla 24. Cronograma referencial abril-julio

Actividad	Abril	Mayo	Junio	Julio
Revisión bibliográfica y delimitación del problema	X	X		
Levantamiento de requerimientos y alcance	X	X		
Diseño de arquitectura, interfaz y base de datos		X	X	
Implementación del prototipo web			X	X
Integración del servicio externo de IA			X	X
Pruebas funcionales y evaluación técnica			X	X
Validación con usuarios y análisis de resultados				X

Actividad	Abril	Mayo	Junio	Julio
Redacción y consolidación del documento final		X	X	X

Anexo D. Plantilla de bitácora de iteraciones

La siguiente plantilla puede utilizarse para registrar avances técnicos, decisiones tomadas, evidencias de prueba y ajustes realizados durante el desarrollo del prototipo.

Tabla 25. Plantilla de bitácora de iteraciones

Iteración	Objetivo técnico	Evidencia y validación
Iteración 1	Definir requerimientos, alcance e historias de usuario.	Backlog inicial, matriz de requerimientos y revisión del autor.
Iteración 2	Diseñar arquitectura, modelo de datos e interfaz.	Modelo de base de datos, diagramas y decisiones técnicas.
Iteración 3	Implementar módulos principales del prototipo.	Módulos funcionales, pruebas por módulo y capturas de evidencia.
Iteración 4	Integrar asistencia de IA y reportes básicos.	Endpoint, evidencias de consumo, sugerencias editables y pruebas.
Iteración 5	Ejecutar validación experimental.	Matriz de tiempos, tareas, SUS, TAM, satisfacción y comentarios.

Anexo E. Capturas de la aplicación SurveyDanny

Las siguientes capturas evidencian las pantallas principales del prototipo funcional SurveyDanny. Se incluyen módulos de autenticación, gestión de encuestas, banco de preguntas, constructor, generación asistida con IA, vista pública de respuesta y reportes.

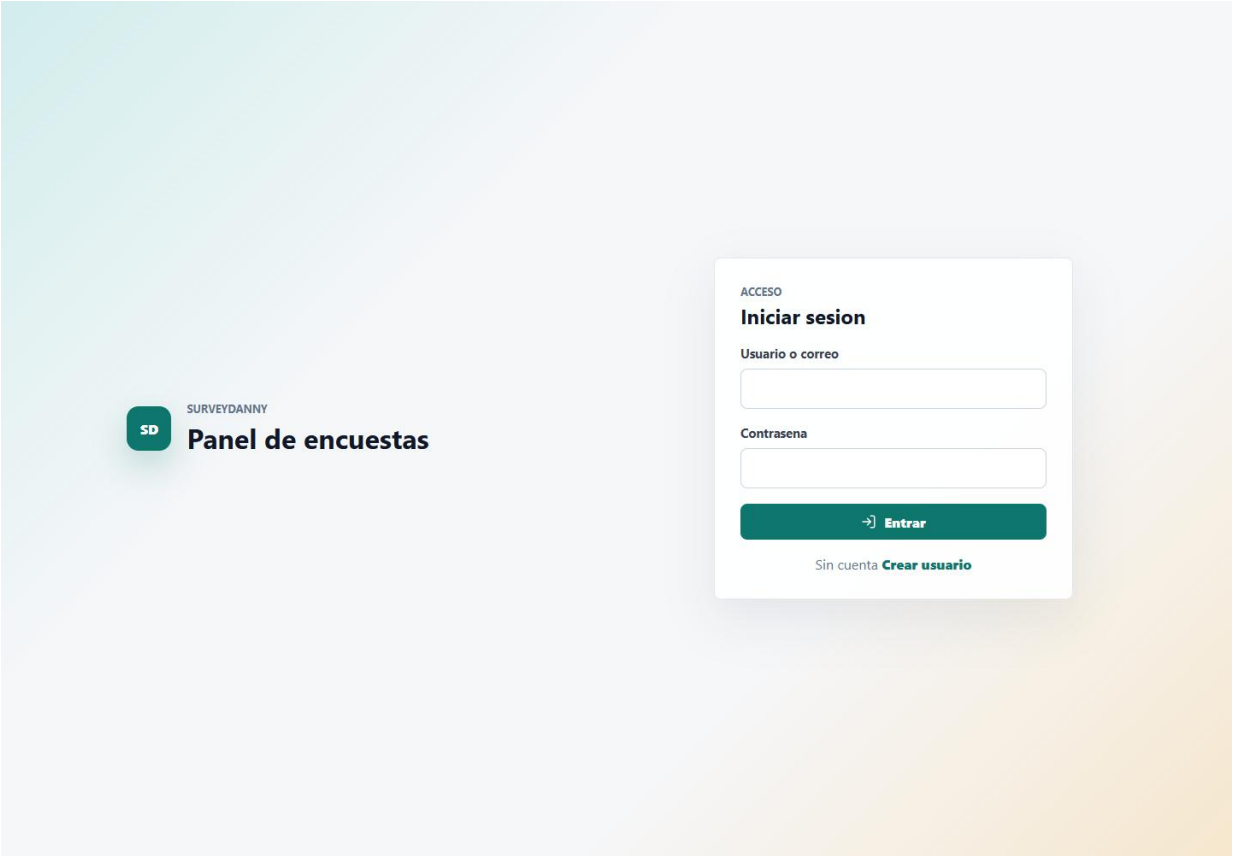


Figura E1. Pantalla de inicio de sesión del prototipo SurveyDanny.

Nota. Captura tomada del prototipo SurveyDanny en ambiente local con datos de demostración.

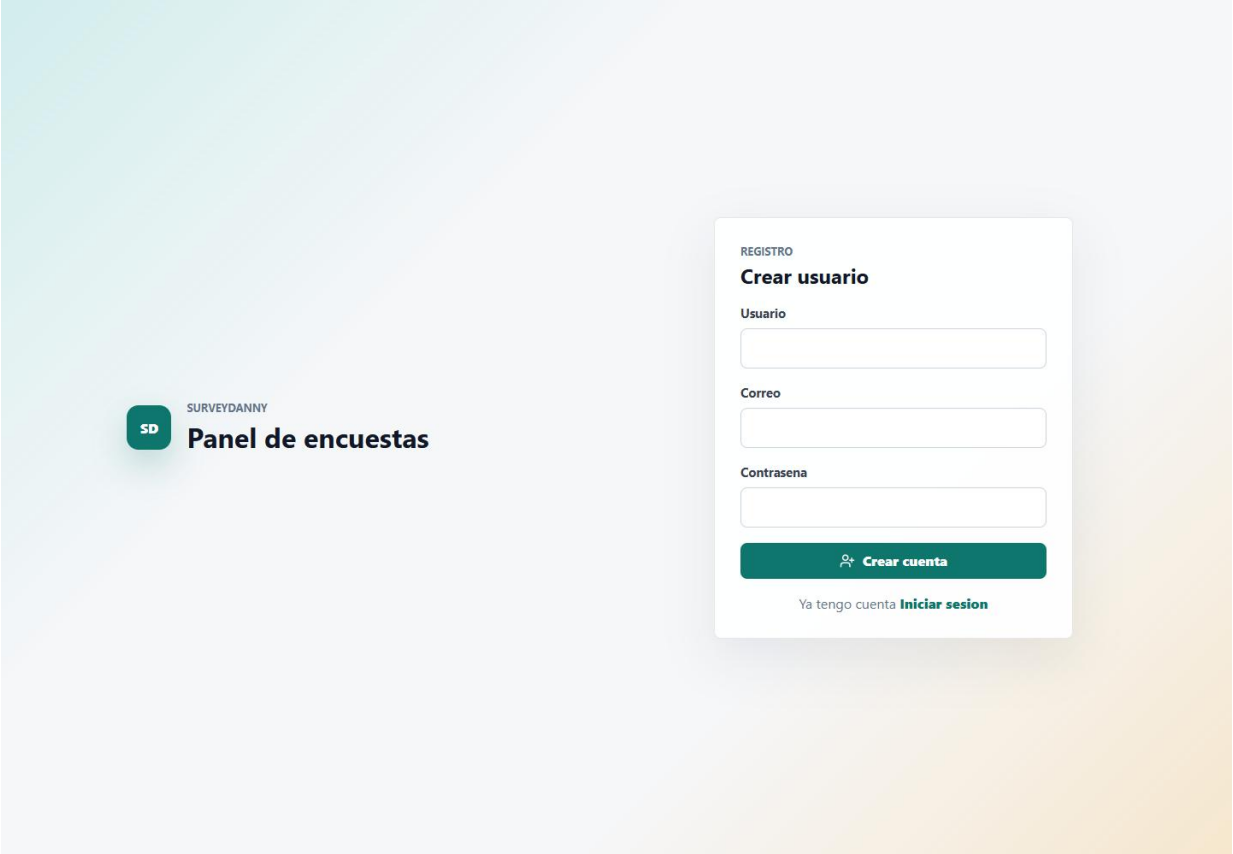


Figura E2. Pantalla de registro de usuarios.

Nota. Captura tomada del prototipo SurveyDanny en ambiente local con datos de demostración.

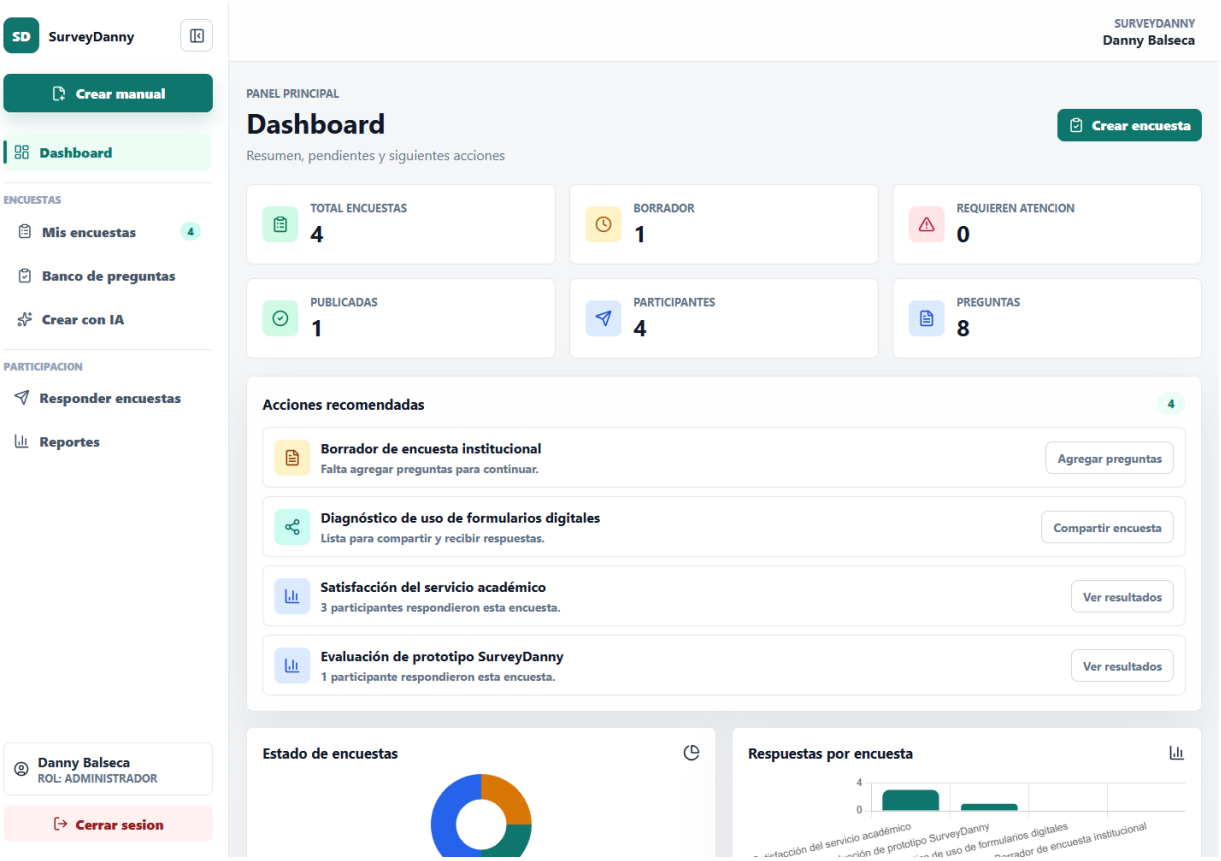


Figura E3. Dashboard principal con indicadores y acciones recomendadas.

Nota. Captura tomada del prototipo SurveyDanny en ambiente local con datos de demostración.

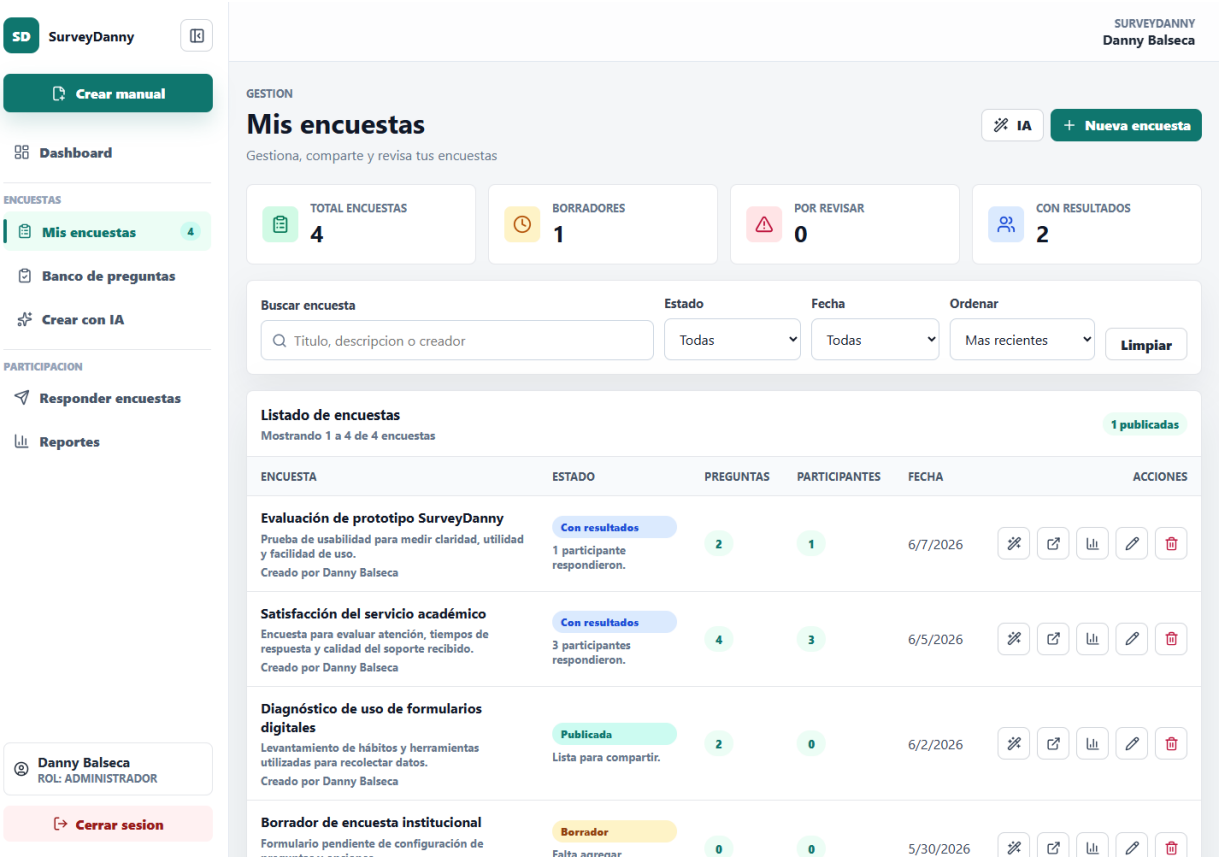


Figura E4. Módulo de gestión y listado de encuestas.

Nota. Captura tomada del prototipo SurveyDanny en ambiente local con datos de demostración.

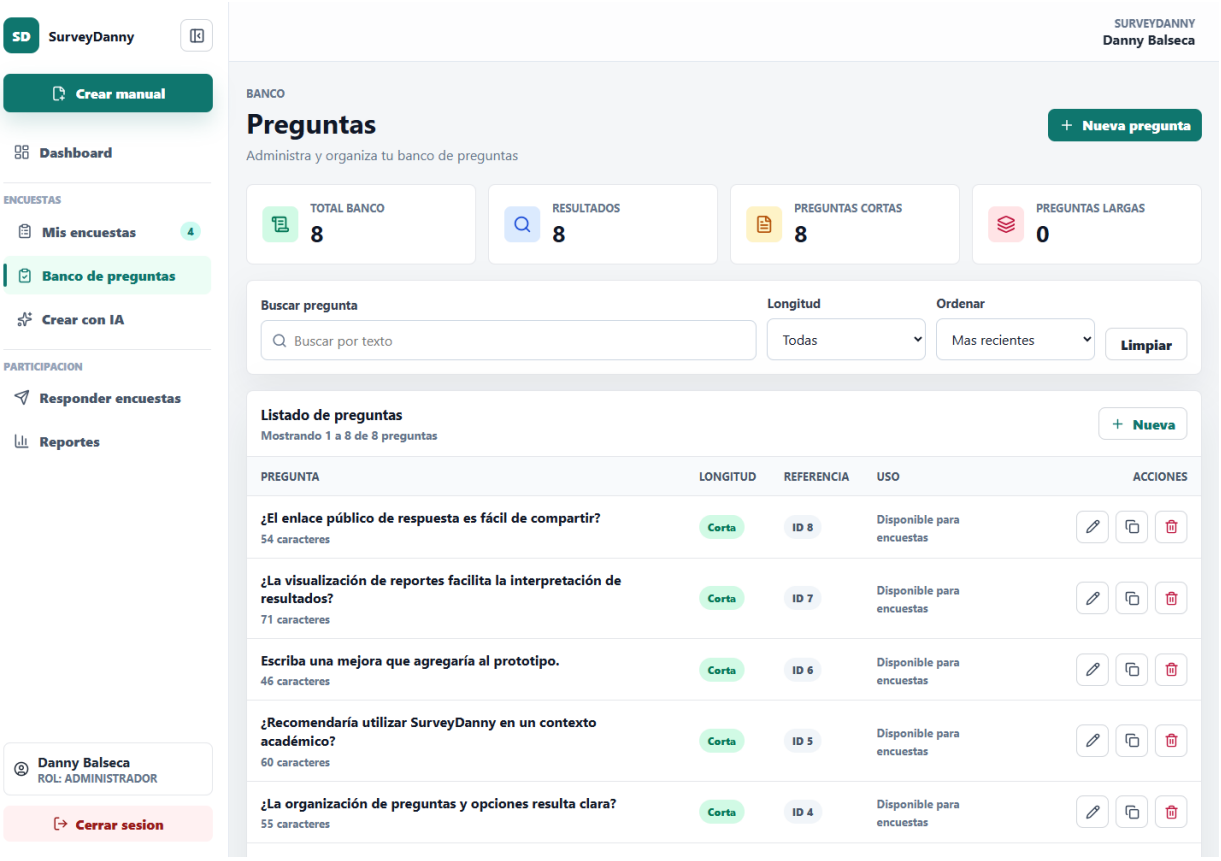


Figura E5. Banco de preguntas del sistema.

Nota. Captura tomada del prototipo SurveyDanny en ambiente local con datos de demostración.

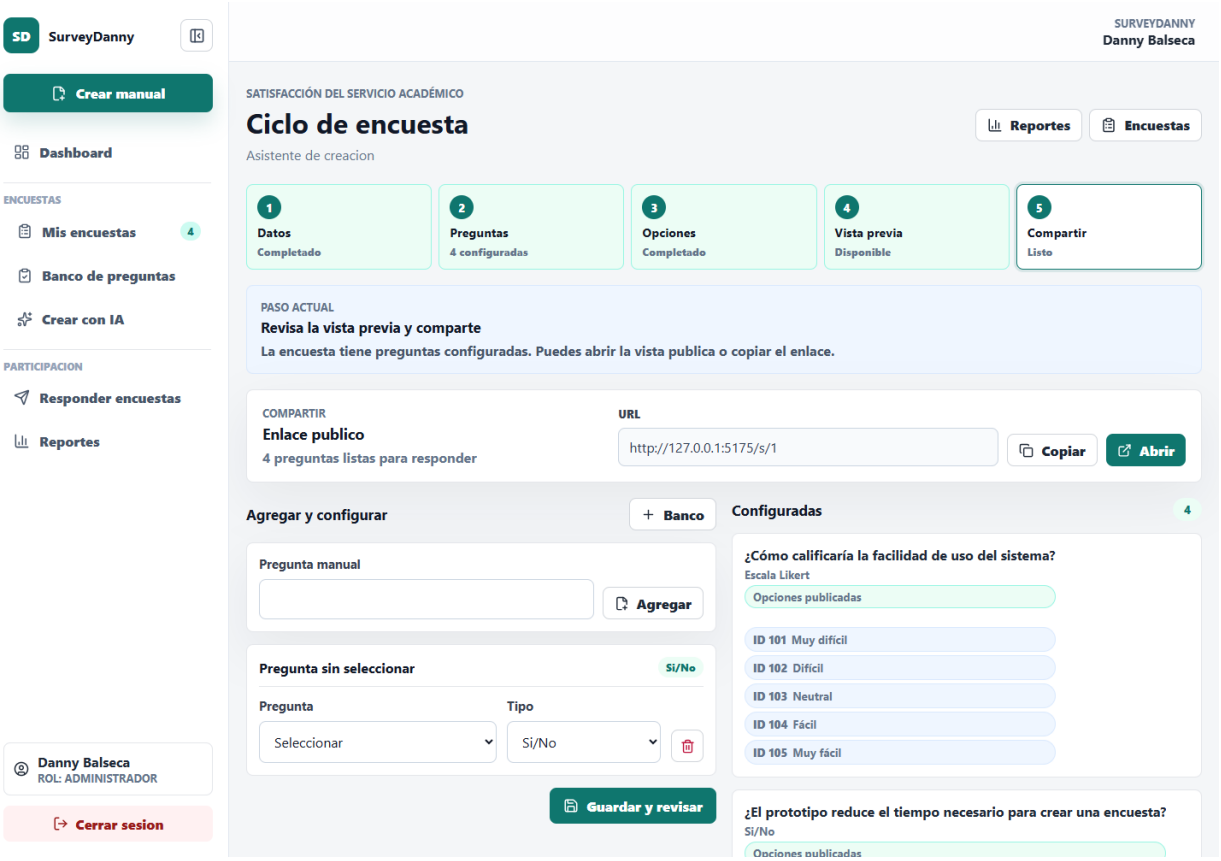


Figura E6. Constructor de encuesta y enlace público.

Nota. Captura tomada del prototipo SurveyDanny en ambiente local con datos de demostración.

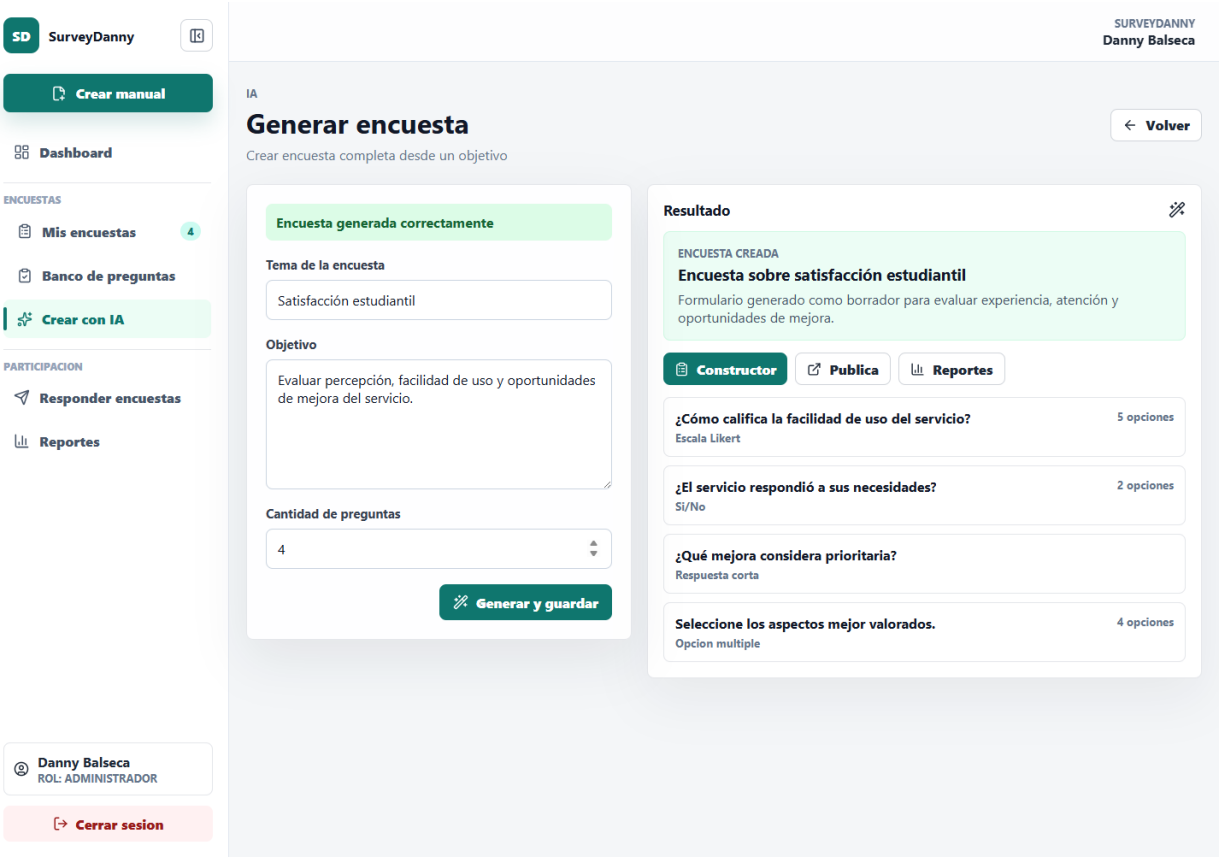


Figura E7. Generación asistida de encuesta mediante IA.

Nota. Captura tomada del prototipo SurveyDanny en ambiente local con datos de demostración.

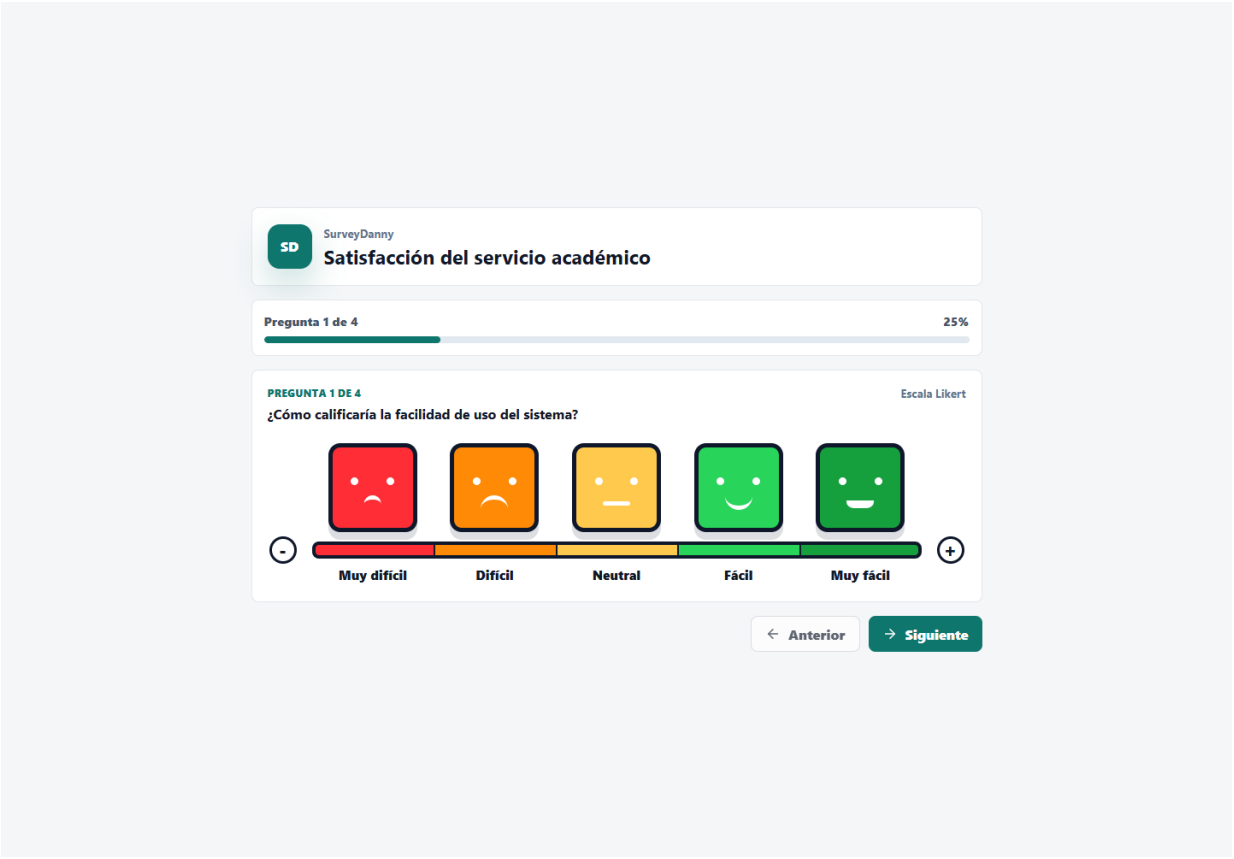


Figura E8. Vista pública para responder una encuesta.

Nota. Captura tomada del prototipo SurveyDanny en ambiente local con datos de demostración.

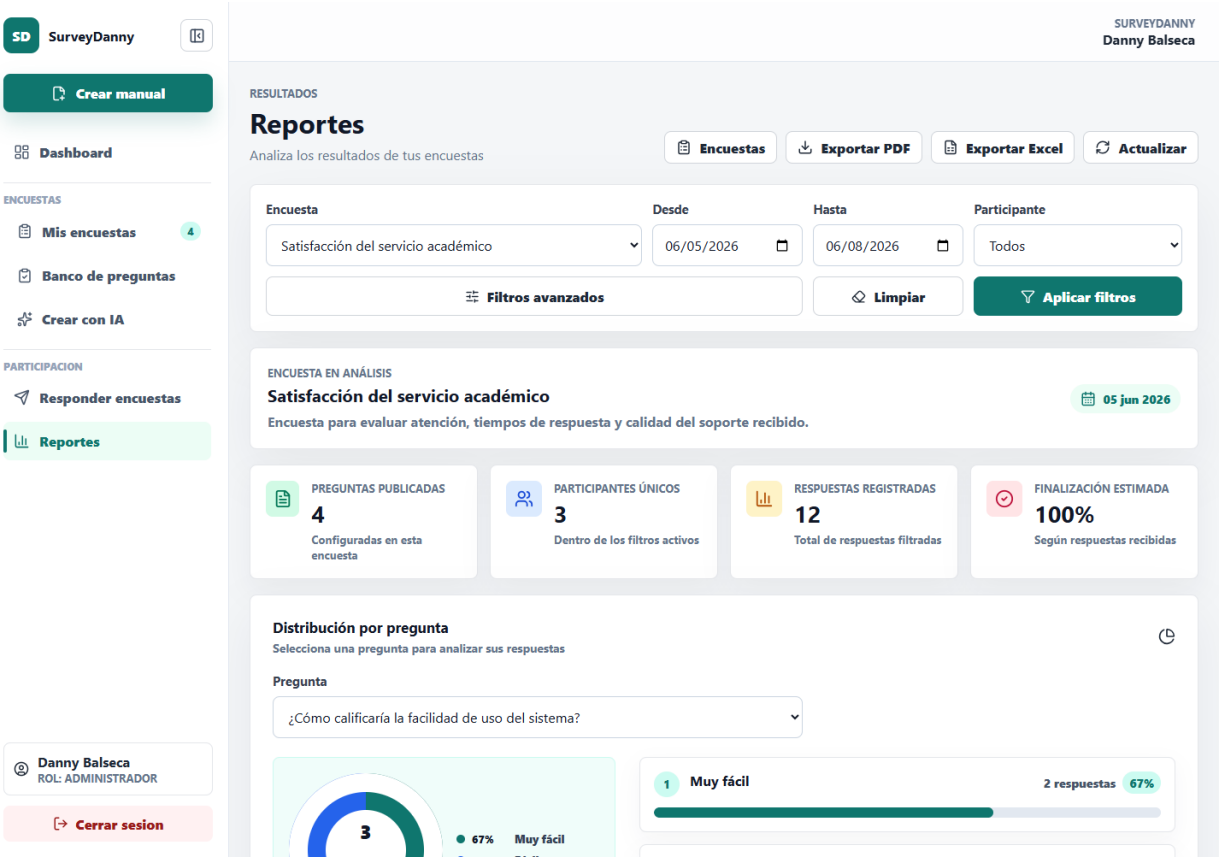


Figura E9. Módulo de reportes y análisis de resultados.

Nota. Captura tomada del prototipo SurveyDanny en ambiente local con datos de demostración.

Anexo F. Manual de usuario de la aplicación SurveyDanny

F.1 Objetivo del manual

El presente manual orienta al usuario final en el uso de SurveyDanny, aplicación web destinada a crear, organizar, publicar y responder encuestas digitales. El documento describe las acciones principales disponibles desde la interfaz: autenticación, administración básica de encuestas, banco de preguntas, generación asistida con IA, publicación de enlaces, respuesta de encuestas y revisión de reportes.

Nota. Las instrucciones se elaboran con base en la versión funcional del prototipo SurveyDanny revisada en ambiente local. Las pantallas referidas corresponden a las capturas presentadas en el Anexo E.

F.2 Requisitos de acceso

Tabla F1. Requisitos para el usuario final

Elemento	Descripción
Navegador	Usar un navegador web moderno con acceso al servidor donde se ejecuta SurveyDanny.
Cuenta de usuario	Contar con usuario y contraseña registrados en el sistema.
Sesión activa	Iniciar sesión para guardar respuestas y acceder a los módulos protegidos.
Datos mínimos	Para crear una encuesta se requiere título y descripción; para usar IA se requiere tema, objetivo y cantidad de preguntas.

F.3 Ingreso y registro

1. Abrir la dirección web de SurveyDanny desde el navegador.
2. En la pantalla de inicio de sesión, escribir el nombre de usuario y la contraseña.
3. Seleccionar la opción de ingreso para acceder al panel principal.
4. Si el usuario no posee cuenta, seleccionar Crear usuario y completar nombre de usuario, correo electrónico y contraseña.
5. Después del registro, volver al inicio de sesión e ingresar con las credenciales creadas.

F.4 Navegación principal

Tabla F2. Opciones del menú principal

Opción	Uso
Dashboard	Muestra indicadores generales, encuestas recientes, pendientes y acciones

	recomendadas.
Crear manual	Permite crear una encuesta desde cero mediante título y descripción.
Mis encuestas	Lista las encuestas registradas y permite editarlas, eliminarlas, abrir el constructor, consultar reportes o revisar la vista pública.
Banco de preguntas	Permite registrar, buscar, duplicar, editar y eliminar preguntas reutilizables.
Crear con IA	Genera una encuesta inicial a partir de un tema, objetivo y cantidad de preguntas.
Responder encuestas	Muestra encuestas disponibles para responder desde la aplicación.
Reportes	Permite consultar resultados, preguntas publicadas y exportar información consolidada.

F.5 Crear una encuesta manual

1. Seleccionar Crear manual o ingresar a Mis encuestas y elegir Nueva encuesta.
2. Escribir el título de la encuesta.
3. Escribir una descripción que explique el propósito del formulario.
4. Seleccionar Guardar y continuar.
5. El sistema guardará la encuesta y abrirá el constructor para asociar preguntas y tipos de respuesta.

F.6 Generar encuesta con asistencia de IA

1. Ingresar al módulo Crear con IA.
2. Completar el tema de la encuesta, por ejemplo satisfacción del cliente.
3. Describir el objetivo de la encuesta.
4. Definir la cantidad de preguntas, con un valor permitido entre 1 y 20.
5. Seleccionar Generar y guardar.
6. Revisar el resultado generado y abrir el constructor para validar o ajustar las preguntas antes de publicar.

Recomendación. La encuesta generada por IA debe ser revisada por el usuario antes de su publicación, especialmente cuando se utilice para actividades académicas o institucionales.

F.7 Usar el banco de preguntas

1. Abrir Banco de preguntas desde el menú lateral.
2. Usar el buscador para localizar preguntas existentes.
3. Seleccionar Nueva pregunta para registrar un texto de pregunta.

4. Guardar la pregunta para incorporarla al banco reutilizable.
5. Cuando sea necesario, editar, duplicar o eliminar una pregunta desde las acciones del listado.

F.8 Configurar y publicar una encuesta

1. Abrir el constructor desde la encuesta seleccionada.
2. Agregar preguntas existentes desde el banco o crear una pregunta manual desde el constructor.
3. Seleccionar el tipo de respuesta correspondiente: Sí/No, escala, abierta, opción múltiple, verdadero/falso, clasificación, acuerdo/desacuerdo, Likert, fecha, respuesta corta, casillas o desplegable.
4. Registrar las opciones de respuesta cuando el tipo seleccionado lo requiera.
5. Guardar la configuración de preguntas y respuestas.
6. Copiar el enlace público generado para compartir la encuesta con los participantes.

F.9 Responder una encuesta

1. Abrir el enlace público de la encuesta o ingresar al módulo Responder encuestas.
2. Revisar el nombre de la encuesta y el indicador de progreso.
3. Responder cada pregunta según el tipo presentado: opción, escala, Likert, fecha o respuesta escrita.
4. Usar Siguiente y Anterior para desplazarse entre preguntas.
5. Al finalizar, seleccionar Enviar.
6. Si el sistema solicita autenticación, iniciar sesión y continuar con el envío de respuestas.

F.10 Consultar reportes

1. Ingresar al módulo Reportes.
2. Seleccionar la encuesta que se desea analizar.
3. Revisar el resumen de participantes, preguntas publicadas y respuestas registradas.
4. Ordenar o filtrar la información cuando la vista lo permita.
5. Descargar el archivo CSV si se requiere conservar evidencia de resultados.

F.11 Mensajes frecuentes

Tabla F3. Mensajes y acciones recomendadas para el usuario

Situación	Acción sugerida
No se puede iniciar sesión	Verificar usuario y contraseña; si el error continúa, solicitar revisión al administrador.
La encuesta no tiene preguntas	Abrir el constructor y asociar al menos una pregunta antes de compartir el enlace.
No se puede enviar una respuesta	Comprobar que todas las preguntas estén respondidas y que la sesión esté activa.
No aparecen resultados	Confirmar que la encuesta ya fue respondida por perfils y recargar el reporte.
La generación con IA falla	Verificar tema, objetivo, cantidad de preguntas y disponibilidad de la configuración administrativa de IA.

Anexo G. Manual administrativo de la aplicación SurveyDanny

G.1 Objetivo del manual

El manual administrativo describe las actividades de configuración, operación y mantenimiento del prototipo SurveyDanny. Está dirigido al responsable técnico o administrador encargado de mantener la aplicación, revisar usuarios, controlar roles, supervisar la API, validar la base de datos y atender incidentes de funcionamiento.

G.2 Componentes administrados

Tabla G1. Componentes técnicos de SurveyDanny

Componente	Descripción administrativa
Frontend	Aplicación Vue 3, Vite y TypeScript ubicada en C:\SurveyDannyWeb. Contiene rutas, vistas, componentes, servicios API y almacenamiento de sesión.
Backend	API ASP.NET Core organizada en controladores, servicios, repositorios y acceso a datos. Expone endpoints para autenticación, encuestas, preguntas, respuestas, reportes e IA.
Base de datos	Base SurveyDanny en SQL Server, administrada mediante Entity Framework Core y tablas relacionales.
Seguridad	Autenticación mediante JWT; el frontend conserva el token y lo envía en solicitudes protegidas.
Documentación API	Referencia técnica disponible en la ruta /scalar cuando el backend está en ejecución.
IA	Servicio de generación de encuestas que consume una clave de API configurada en el backend.

G.3 Gestión de usuarios y roles

El sistema permite registrar usuarios desde la interfaz y asignar roles desde endpoints administrativos del backend. Los roles identificados en la implementación son

Admin, User y Report. La asignación de roles se refleja en el token JWT y permite mostrar la sesión como usuario administrador cuando corresponde.

Tabla G2. Operaciones administrativas de usuarios

Operación	Endpoint o módulo	Uso
Registrar usuario	POST /api/Authentication/register	Crea una cuenta con nombre de usuario, correo y contraseña.
Iniciar sesión	POST /api/Authentication/Login	Genera el token JWT utilizado por el frontend.
Asignar Admin	POST /api/Authentication/AssignRolAdmin	Asigna privilegio administrativo a un usuario existente.
Asignar User	POST /api/Authentication/AssignRolUser	Asigna rol de usuario estándar.
Asignar Report	POST /api/Authentication/AssignRolReport	Asigna rol orientado a consulta de reportes.

Control recomendado. La asignación de roles debe realizarse únicamente por personal autorizado y registrarse en una bitácora de administración.

G.4 Administración de módulos funcionales

Tabla G3. Módulos y responsabilidades administrativas

Módulo	Responsabilidad del administrador
Dashboard	Verificar que los indicadores carguen correctamente y que las encuestas tengan estado coherente.
Encuestas	Supervisar creación, actualización y eliminación de encuestas; validar que no se elimine información necesaria para evidencia.
Banco de preguntas	Revisar calidad de redacción, duplicados y pertinencia de las preguntas reutilizables.
Constructor	Confirmar que las preguntas tengan tipo de respuesta y opciones cuando aplique.
Vista pública	Comprobar que el enlace /s/{id} cargue la encuesta publicada y permita responder con sesión válida.
Reportes	Validar que las respuestas registradas se reflejen en tablas, indicadores y exportación CSV.
Generación con IA	Revisar que las encuestas generadas sean guardadas y que el usuario pueda validarlas antes de publicar.

G.5 Endpoints principales de operación

Tabla G4. Resumen de endpoints del backend

Área	Endpoint base	Acciones principales
Autenticación	/api/Authentication	Registro, login y asignación de roles.
Encuestas	/api/Survey	Crear, listar, consultar por id, actualizar y eliminar encuestas protegidas por JWT.
Preguntas	/api/Question	Crear, listar, consultar por id, actualizar y eliminar preguntas protegidas por JWT.
Publicación y respuestas	/api/SurveysForClients	Asociar preguntas a encuestas, obtener

		encuesta pública, guardar respuestas y consultar respuestas por encuesta.
IA	/api/SurveyGPT	Generar encuesta con IA y guardar encuestas generadas.

G.6 Administración de base de datos

La base SurveyDanny almacena usuarios, roles, encuestas, preguntas, tipos de respuesta, opciones, respuestas seleccionadas, reportes y parámetros iniciales. El administrador debe mantener respaldos periódicos, revisar integridad referencial y evitar cambios directos sin copia previa de seguridad.

Tabla G5. Tablas principales y finalidad

Tabla o entidad	Finalidad
Users	Registra credenciales y datos básicos de los usuarios.
Roles y UserByRol	Controlan los perfiles asignados a cada usuario.
Surveys	Almacena título, descripción, fecha y usuario creador de encuestas.
Questions	Contiene el banco de preguntas reutilizables.
ResponseTypes	Define los tipos de respuesta disponibles en el constructor.
Responses y ResponsesTypesText	Guardan opciones y textos asociados a preguntas publicadas.
QuestionSurvey	Relaciona preguntas con encuestas configuradas.
SelectedResponses	Registra respuestas seleccionadas o escritas por los perfils.
Reports e InitialParameters	Apoyan la consulta y parametrización del sistema.

G.7 Puesta en marcha del sistema

1. Verificar que SQL Server esté activo y que la cadena de conexión del backend apunte a la base SurveyDanny.
2. Iniciar el backend ASP.NET Core y confirmar que los controladores se carguen sin errores.
3. Abrir la ruta /scalar para revisar la documentación de endpoints disponibles.
4. Iniciar el frontend Vue/Vite y validar que pueda comunicarse con la API mediante la URL configurada.
5. Crear o usar un usuario de prueba y ejecutar un flujo completo: login, creación de encuesta, configuración de preguntas, respuesta y reporte.

G.8 Parámetros de configuración

Tabla G6. Parámetros sensibles o críticos

Parámetro	Control administrativo
Cadena de conexión	Mantenerla fuera de documentos públicos y actualizarla solo cuando cambie servidor, base o credenciales.
Clave JWT	Protegerla y renovarla si se sospecha exposición.

CORS	Permitir únicamente los orígenes requeridos por el frontend.
Clave de API de IA	Conservarla en configuración segura; no incluirla en capturas, anexos ni repositorios públicos.
URL del frontend	Validar que coincida con el ambiente de uso local, pruebas o despliegue.

G.9 Mantenimiento y respaldo

- Realizar respaldo de la base de datos antes de cambios estructurales, limpieza de datos o pruebas masivas.
- Revisar periódicamente usuarios y roles asignados.
- Conservar evidencias de pruebas funcionales y resultados exportados cuando sean parte de la validación académica.
- Ejecutar pruebas del flujo completo después de cambios en frontend, backend o base de datos.
- Actualizar dependencias únicamente después de comprobar compatibilidad con Vue, Vite, ASP.NET Core, Entity Framework Core y SQL Server.
- Evitar publicar claves, tokens o cadenas de conexión en documentos, capturas o repositorios.

G.10 Atención de incidentes

Tabla G7. Incidentes comunes y acciones administrativas

Incidente	Posible causa	Acción administrativa
El usuario no puede ingresar	Credenciales incorrectas, usuario inexistente o token inválido.	Verificar registro, estado del backend y respuesta del endpoint de login.
No cargan encuestas	Backend detenido, error de CORS o token vencido.	Reiniciar API, revisar consola del navegador y validar autenticación.
La encuesta pública no aparece	Encuesta sin preguntas configuradas o id incorrecto.	Abrir constructor, asociar preguntas y verificar el enlace /s/{id}.
No se guardan respuestas	Sesión inválida, preguntas incompletas o error de base de datos.	Validar sesión, revisar payload enviado y comprobar tablas de respuestas.
No funciona la IA	Clave de API ausente, servicio externo no disponible o error de formato.	Revisar configuración de OpenAI, logs del backend y parámetros enviados por el usuario.
Los reportes aparecen vacíos	No existen respuestas registradas o la consulta no devuelve datos.	Confirmar respuestas en base de datos y revisar endpoint de consulta de respuestas.

G.11 Checklist administrativo

- Base de datos SurveyDanny activa y accesible.
- Backend iniciado y documentación /scalar disponible.

- Frontend conectado a la API correcta.
- Usuario administrador y roles validados de acuerdo con la responsabilidad de cada usuario.
- Flujo de encuesta probado: crear, configurar, publicar, responder y reportar.