

Rúbrica de Evaluación - Proyecto Sistema de Help Desk Web

1. Tabla resumen de criterios y pesos

La siguiente tabla resume los criterios de evaluación y su peso en la nota final del proyecto.

#	Criterio	Peso (%)
1	Modelo de datos y uso de MariaDB	10
2	Funcionalidad de tickets	10
3	Gestión de comentarios	10
4	Autenticación y roles	10
5	Seguridad básica	10
6	Interfaz y UX (HTML + Bootstrap + jQuery)	5
7	Documentación (docs/)	10
8	Mejora adicional	10
9	Uso de GitHub y organización del proyecto	10
10	Presentación (video - ver el final del proyecto para criterios)	15
	Total	100

2. Rúbrica analítica con descriptores por nivel

Se utiliza una escala de cuatro niveles de desempeño: Excelente, Bueno, Aceptable e Insuficiente. Cada criterio se evalúa en función de las descripciones siguientes.

Criterio 1 – Modelo de datos y uso de MariaDB

Nivel	Descriptor
Excelente	Modelo de datos completamente consistente con el enunciado (3 tablas requeridas), claves primarias y foráneas correctamente definidas, tipos de datos adecuados, uso correcto de restricciones. El script SQL se ejecuta sin errores.

Nivel	Descriptor
Bueno	Modelo de datos mayormente correcto; las 3 tablas están presentes y funcionales, con detalles menores en tipos de datos, restricciones o nomenclatura que no impiden el funcionamiento general.
Aceptable	El modelo incluye las 3 tablas, pero presenta errores conceptuales o de implementación (claves foráneas mal definidas, tipos poco adecuados) que requieren correcciones para un uso robusto.
Insuficiente	El modelo no cumple con las 3 tablas requeridas, o presenta errores graves que impiden el uso de la base de datos (fallos de ejecución, ausencia de claves, estructura inconsistente).

Criterio 2 – Funcionalidad de tickets

Nivel	Descriptor
Excelente	El sistema permite crear, listar, visualizar, actualizar estado y asignar tickets de acuerdo con el rol del usuario (ADMIN, AGENT, USER). El flujo es coherente, estable y sin errores críticos.
Bueno	La mayoría de las funcionalidades de tickets está implementada (crear, listar y ver detalle); la actualización de estado y/o asignación puede tener limitaciones menores o pequeños defectos.
Aceptable	Se pueden crear y listar tickets, pero el flujo está incompleto o poco claro (por

Nivel	Descriptor
	ejemplo, problemas frecuentes al actualizar estado o asignar). Requiere intervención adicional del docente para ser utilizable.
Insuficiente	La funcionalidad de tickets es muy limitada o no opera (no se pueden crear, listar o ver tickets de forma consistente). El sistema no cumple el objetivo principal del proyecto.

Criterio 3 – Gestión de comentarios

Nivel	Descriptor
Excelente	Los comentarios pueden añadirse y visualizarse por ticket, con autor y fecha claramente identificados y orden cronológico correcto. La interacción es fluida y estable.
Bueno	La gestión de comentarios funciona, pero presenta detalles menores (por ejemplo, formato de fecha, orden parcial, algunos mensajes de error no claros).
Aceptable	Es posible añadir comentarios, pero la visualización o el vínculo con el ticket/usuario es confuso o incompleto (por ejemplo, falta de nombre de autor, orden irregular).
Insuficiente	La funcionalidad de comentarios no está implementada o no funciona de forma fiable (no se guardan, no se muestran o generan errores críticos).

Criterio 4 – Autenticación y roles

Nivel	Descriptor
Excelente	Autenticación completamente funcional (login/logout), manejo correcto de sesión y control de acceso por rol (ADMIN, AGENT, USER) en las vistas correspondientes. No se puede acceder a vistas restringidas sin credenciales adecuadas.
Bueno	El login y logout funcionan; el control por rol está implementado para la mayoría de las vistas, aunque puedan existir algunas rutas sin protección o casos borde sin manejar.
Aceptable	Autenticación básica funcionando (se puede iniciar sesión), pero con control de acceso por rol incompleto o inconsistente. Es posible acceder a ciertas vistas sin los permisos adecuados.
Insuficiente	La autenticación es inexistente o no funcional; no se diferencia el acceso por rol, o cualquier usuario puede acceder a todas las vistas sin restricciones.

Criterio 5 – Seguridad básica

Nivel	Descriptor
Excelente	Se usan hashes para contraseñas, parámetros en consultas SQL, se protegen las rutas críticas y se manejan errores sin exponer información sensible. El código demuestra un cuidado explícito por buenas prácticas básicas de seguridad.

Nivel	Descriptor
Bueno	Las prácticas de seguridad clave están presentes (hash de contraseñas y parámetros SQL), aunque algunos aspectos (mensajes de error, manejo de sesiones, validaciones adicionales) podrían mejorar.
Aceptable	Se observa intención de aplicar seguridad (por ejemplo, uso parcial de hash o parámetros), pero se combina con prácticas inseguras (concatenación de SQL, mensajes de error muy detallados, etc.).
Insuficiente	Las contraseñas se almacenan en texto plano, se construye SQL con concatenación directa, no se protege el acceso a recursos, o se exponen datos sensibles. Riesgo de seguridad evidente.

Criterio 6 – Interfaz y UX (HTML + Bootstrap + jQuery)

Nivel	Descriptor
Excelente	La interfaz es clara, ordenada y responsiva; se aprovechan adecuadamente componentes de Bootstrap. jQuery se utiliza para mejorar la experiencia de usuario (validaciones, interacciones suaves). Navegación intuitiva.
Bueno	La interfaz es funcional y mayormente responsiva; usa Bootstrap de forma correcta aunque simple. jQuery se utiliza en algunos puntos clave, pero podría

Nivel	Descriptor
	integrarse mejor para la experiencia global.
Aceptable	La interfaz es básica pero utilizable; el uso de Bootstrap es mínimo o inconsistente. jQuery se utiliza poco o solo en detalles simples. La navegación puede resultar confusa en algunos puntos.
Insuficiente	La interfaz es desorganizada, difícil de usar o no responsiva. No se observa uso significativo de Bootstrap ni de jQuery, o la app resulta confusa incluso para tareas básicas.

Criterio 7 – Documentación (docs/)

Nivel	Descriptor
Excelente	La carpeta docs/ contiene manual de usuario, manual técnico, diagrama ER y capturas de pantalla bien organizadas. La documentación es clara, detallada y suficiente para que un tercero instale y use el sistema sin ayuda adicional.
Bueno	La documentación está mayormente completa (manuales, diagrama, capturas), con algunos detalles menores faltantes o poco claros, pero en general permite instalar y entender el sistema.
Aceptable	Existe documentación básica (por ejemplo, solo un documento mezclado o instrucciones parciales) que requiere la intervención del equipo o del docente para completar información importante.

Nivel	Descriptor
Insuficiente	La documentación es muy limitada, confusa o inexistente. No se proveen instrucciones claras para instalación ni uso del sistema.

Criterio 8 – Mejora adicional

Nivel	Descriptor
Excelente	Se implementa al menos una mejora adicional claramente identificada (filtros, estadísticas, validaciones avanzadas, etc.), bien integrada al sistema y documentada en el manual técnico.
Bueno	La mejora adicional está implementada y funciona, pero su integración o documentación es parcial (no se explica del todo o su uso no es evidente).
Aceptable	La mejora se describe en la documentación, pero su implementación es incompleta o poco estable (errores frecuentes o solo parcialmente funcional).
Insuficiente	No se observa ninguna mejora adicional, o la propuesta no es relevante frente al alcance del proyecto.

Criterio 9 – Uso de GitHub y organización del proyecto

Nivel	Descriptor
Excelente	El repositorio GitHub está bien organizado, con estructura clara de carpetas, README.md completo, requirements.txt, carpeta docs/, y exclusión adecuada de archivos sensibles

Nivel	Descriptor
	como .env. Commits con mensajes razonables.
Bueno	El repositorio contiene la mayor parte de los elementos requeridos (código, docs, README), aunque con algunos detalles de organización mejorables o pequeños descuidos (por ejemplo, falta de información breve en el README).
Aceptable	El código está en GitHub pero con organización limitada, sin documentación mínima en el README o con estructuras de carpetas desordenadas. Aun así, el proyecto puede revisarse.
Insuficiente	No se entrega repositorio en GitHub, o el repositorio está vacío, incompleto o no permite revisar el proyecto correctamente.