



**ESCUELA POLITÉCNICA NACIONAL ESCUELA DE FORMACIÓN DE
TECNÓLOGOS**

TECNOLOGÍA SUPERIOR EN DESARROLLO DE SOFTWARE

BASES DE DATOS

TEMA:

**SISTEMA INTEGRAL DE GESTIÓN PARA ENTORNOS
PROFESIONALES CON BASES DE DATOS RELACIONALES**

INTEGRANTES:

Antony Cisneros

Alan Logroño

Ing. Yadira Franco

30 de Julio de 2025

2025 A

Contenido

Estructura del Proyecto.....	3
Temática Elegida	3
Descripción:	3
Objetivo del Proyecto.....	3
Objetivo General:	3
Objetivos Específicos:	3
Justificación.....	4
Alcance	4
Roles y acciones.....	4
Requerimientos	4
Herramienta Asignada y Tecnología usada	6
Complementos Aplicados:	6
Diseño de Interfaz	7
Modelado de Base de datos	8
Organización del Código	9
Ejecución Completa del Proyecto	10
Evidencias	11
Simulaciones:.....	14
Explicación del Desarrollo	15
Conclusiones y Recomendaciones	16
Anexos:	16
Enlaces de Interés:.....	20

Estructura del Proyecto

Temática Elegida

Nombre del Proyecto: SISTEMA INTEGRAL DE GESTIÓN PARA ENTORNOS
PROFESIONALES CON BASES DE DATOS RELACIONALES

Descripción:

El presente proyecto consiste en el diseño e implementación de un Sistema Integral de Gestión Hotelera utilizando bases de datos relacionales bajo SQL Server. Este sistema está diseñado para administrar eficientemente la información relacionada con habitaciones, reservas, huéspedes, servicios, facturación y auditoría, aplicando buenas prácticas de seguridad, rendimiento y respaldo. La solución simula un entorno profesional real, considerando roles, procedimientos almacenados, triggers y mecanismos de protección de datos sensibles.

Objetivo del Proyecto

Objetivo General:

Diseñar y desarrollar un sistema profesional de gestión hotelera que utilice una base de datos relacional en SQL Server, incorporando modelos normalizados, procedimientos avanzados, roles, auditoría, mecanismos de seguridad, respaldo y recuperación de datos, cumpliendo con estándares reales de gestión de información crítica.

Objetivos Específicos:

- ☐ Modelar lógicamente una base de datos normalizada hasta 3FN para un hotel.
- ☐ Implementar procedimientos almacenados para insertar, actualizar, eliminar y facturar.
- ☐ Diseñar funciones personalizadas que calculen estadías, porcentajes de ocupación y totales facturados.
- ☐ Implementar triggers para auditoría y control automático de acciones sensibles.

- ☐ Aplicar mecanismos de respaldo y restauración (en caliente y en frío).

Justificación

En la actualidad, la gestión hotelera requiere sistemas robustos que aseguren la integridad, disponibilidad y confidencialidad de los datos. Este proyecto permite aplicar de forma práctica los conocimientos adquiridos en modelado relacional, seguridad, programación SQL y administración de bases de datos. Implementar un sistema profesional para hoteles facilita el control de reservas, servicios y facturación, aportando valor tanto en eficiencia operativa como en toma de decisiones basada en datos confiables y seguros. Trabajando de manera colaborativa con las múltiples herramientas que nos ofrecen los entornos de SqlServe, permitiéndonos su correcta gestión e implementación en un entorno crítico.

Alcance

El presente sistema busca la implementación y desarrollo de la gestión básica a contener un entorno de hotelería, centrado en el control y manutención de un entorno confiable y seguro para los clientes del mismo, y de igual manera optimizando procesos para la facturación, consultas y uso de procedimientos almacenados y sus funciones, así de igual manera centrado en la encriptación y manejo seguro de la bases de datos relacionales.

Roles y acciones

Rol	Descripción
Administrador	Tiene acceso completo a todas las tablas, auditorías y configuraciones. Puede crear y modificar usuarios.
Recepcionista	Gestiona huéspedes y reservas, sin acceso a auditoría ni configuraciones.
Camarera	Visualiza solo información relacionada con habitaciones y su estado operativo (ocupada, limpieza, mantenimiento).

Requerimientos

Categoría	Requerimiento	Descripción / Ejemplo
-----------	---------------	-----------------------

Modelado y Normalización	Modelo E-R en 3FN	Diagrama entidad-relación validado con integridad referencial completa.
	Relaciones correctas	Relaciones uno-a-muchos, muchos-a-muchos y atributos multivaluados normalizados.
Estructura de Base	Tablas funcionales	Mínimo 10 tablas principales: habitaciones, huéspedes, reservas, facturas, pagos, etc.
	Tablas de catálogos	Tablas auxiliares como categorías de habitación, formas de pago, servicios, estados.
	Restricciones de campos	Aplicación de NOT NULL, CHECK, UNIQUE, DEFAULT, AUTO_INCREMENT, etc.
	Claves foráneas y reglas referenciales	Uso de ON DELETE CASCADE, ON UPDATE CASCADE, SET NULL según contexto.
Procedimientos Almacenados	Inserciones con validación	Ejemplo: verificar disponibilidad antes de insertar una reserva.
	Actualizaciones masivas	Actualizar tarifas o estado de reservas según condiciones.
	Eliminaciones seguras	Borrado lógico o validación previa para evitar pérdidas de datos clave.
	Reportes por período	Generar ingresos mensuales, reservas por semana, etc.
	Facturación automática	Cálculo por noche + servicios usados.
	Transacciones controladas	Uso de START TRANSACTION, SAVEPOINT, ROLLBACK, COMMIT con manejo de errores.
Funciones SQL	Indicadores y cálculos	Funciones para duración de estadía, % ocupación, totales facturados.
Triggers	Auditoría de acciones	Registro automático en tabla log_acciones (INSERT, UPDATE, DELETE).
	Control automático	Por ejemplo, actualización de stock o validación de fechas.
	Simulación de notificaciones	Inserción en tabla de alertas ante acciones críticas.
Seguridad y Roles	Roles personalizados	Administrador, auditor, recepcionista, cliente, camarera, etc.
	Gestión de privilegios	Uso de GRANT, REVOKE.
	Cifrado de datos	Contraseñas y correos con SHA2, AES ENCRYPT/AES DECRYPT.
	Simulación de ataques	Ejemplo de SQL Injection (' OR '1'='1) con prevención.
	Validaciones de entrada	Uso de expresiones regulares para prevenir datos peligrosos.
Auditoría	Log de acciones	Tabla con campos: usuario, IP, fecha, acción, tabla, ID afectado.
	Reportes de uso	Consultas por usuario, acción, módulo, fechas.

RespalDOS y Restauración	Backup en caliente y en frío	Copias sin detener servicio y con servicio detenido.
	Restauración desde scripts	Scripts SQL para recuperar la base completa o parcialmente.
Rendimiento y Monitoreo	Índices	Creación de índices simples y compuestos para consultas más rápidas.
	Evaluación de consultas	Análisis de rendimiento con EXPLAIN.
	Simulación de carga	Inserción de más de 500 registros para pruebas de stress.

Ilustración 1 Requerimientos del proyecto

Herramienta Asignada y Tecnología usada

Para el desarrollo del sistema de gestión hotelera se utilizó SQL Server como sistema de gestión de bases de datos relacional (RDBMS), dada su robustez, soporte para programación avanzada con Transact-SQL y capacidad de gestión de seguridad, auditoría y rendimiento.

Adicionalmente, se emplearon las siguientes herramientas:

- SQL Server Management Studio (SSMS): para la creación, ejecución y gestión de scripts SQL.
- Dbdiagram.io / MySQL Workbench / pgModeler: para el modelado lógico y validación de normalización.
- GitHub: como repositorio para control de versiones y respaldo del código fuente.

Complementos Aplicados:

Con el objetivo de simular un entorno profesional y cumplir con buenas prácticas en bases de datos, se implementaron los siguientes complementos y extensiones funcionales:

- Procedimientos almacenados avanzados: para automatizar operaciones críticas como facturación, inserciones validadas y generación de reportes.

- Triggers (disparadores): que permiten auditar automáticamente acciones sensibles (INSERT, UPDATE, DELETE).
- Cifrado de datos sensibles
- Gestión de roles y privilegios: para limitar el acceso a la información según el perfil del usuario
- Simulación de ataques y seguridad: como prevención ante SQL Injection
- Índices y análisis de rendimiento: para mejorar tiempos de respuesta en consultas complejas con EXPLAIN y pruebas de carga.

Diseño de Interfaz

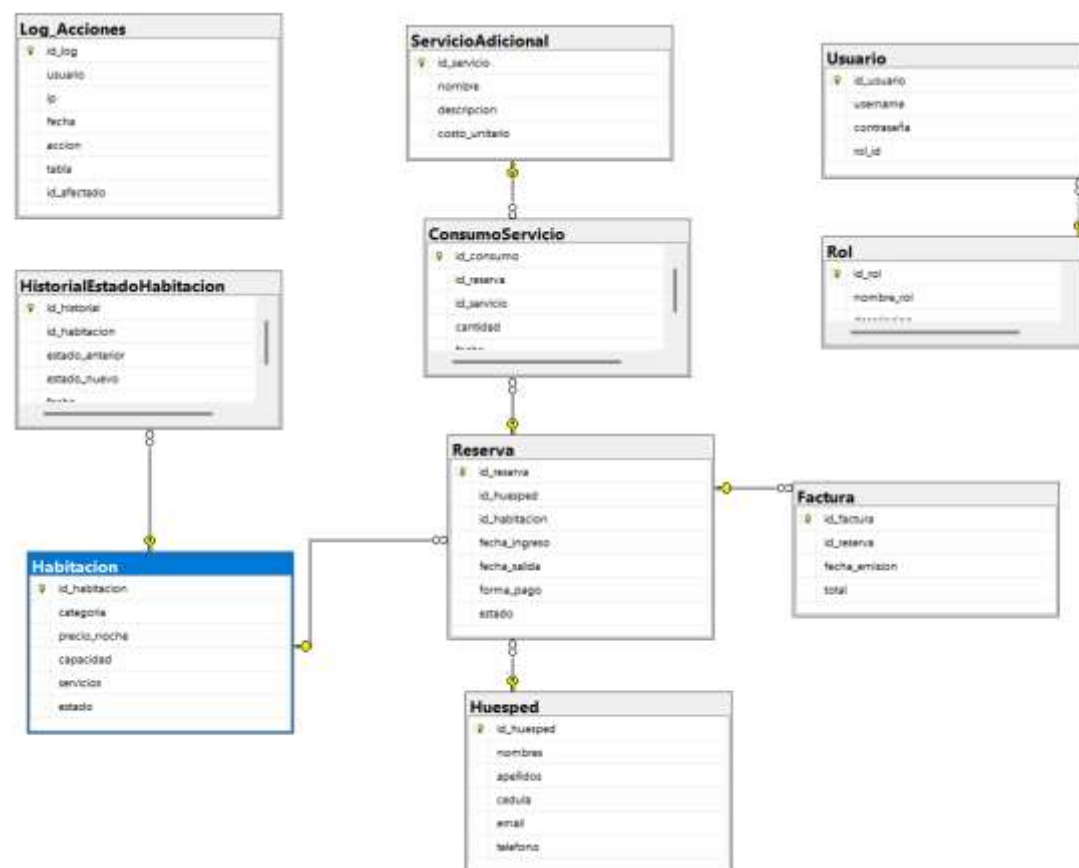


Ilustración 2 Gestion de Sistema en SqlServer

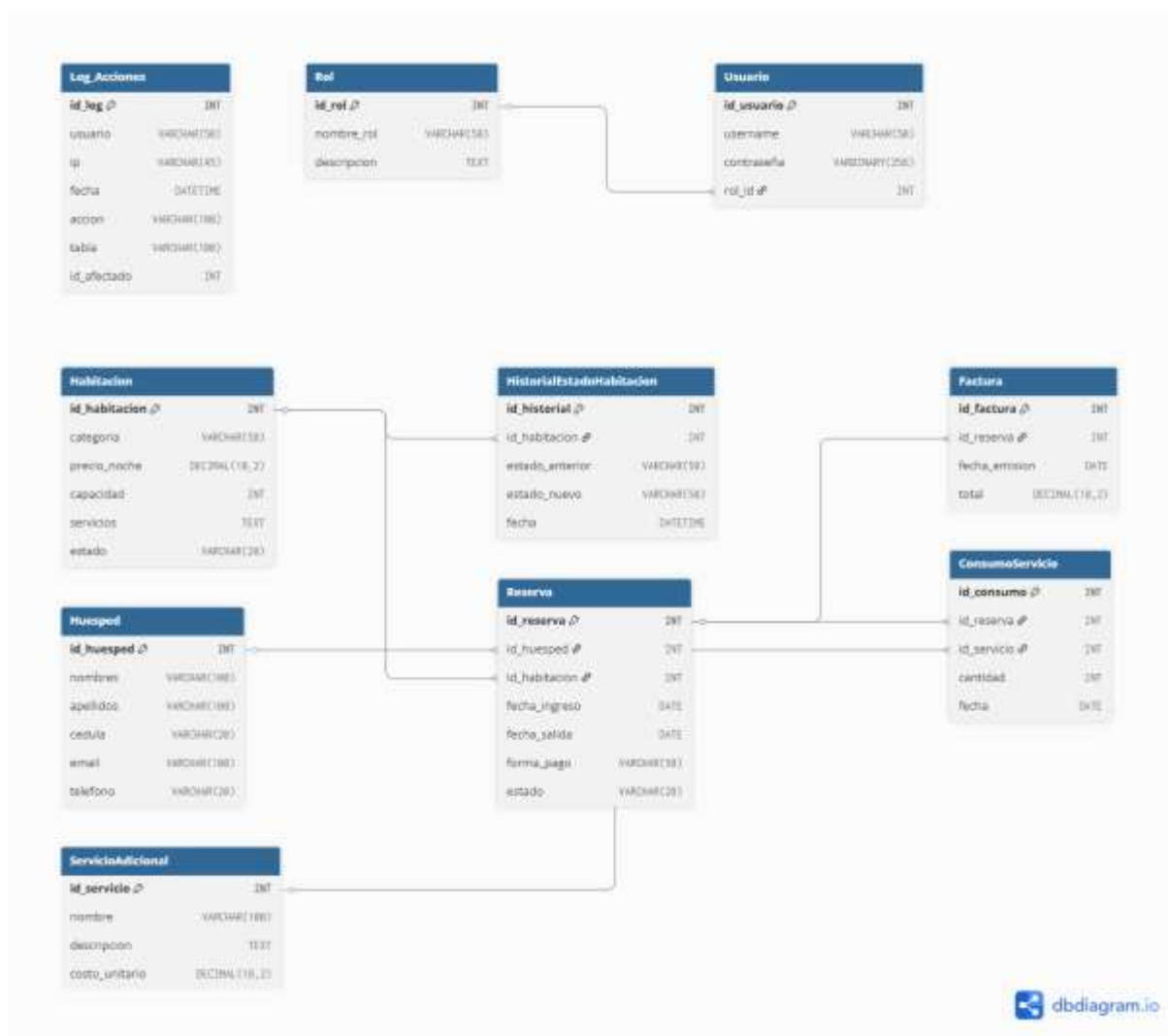


Ilustración 3 Gestion de Sistema en dbdiagram.io

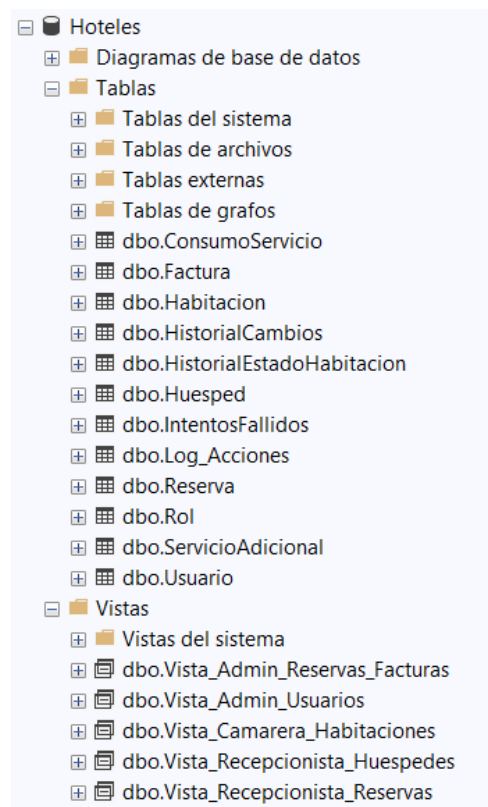
Modelado de Base de datos

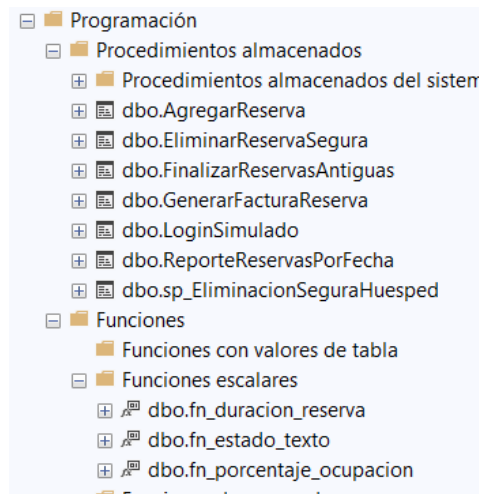
- El modelado de la base de datos fue realizado aplicando una metodología formal desde el diseño conceptual hasta el físico:
- Modelo Entidad-Relación (E-R): elaborado en herramientas como dbdiagram.io, representando entidades clave como Habitaciones, Reservas, Huéspedes, Facturas, Servicios, Pagos.
- Normalización hasta 3FN: evitando redundancia y mejorando la integridad lógica del sistema.

- Integridad Referencial: implementada mediante claves primarias y foráneas, con reglas como ON DELETE CASCADE, SET NULL, según el contexto.
- Tipos de relaciones: uno-a-muchos (una habitación tiene muchas reservas), muchos-a-muchos (reservas y servicios), y multivaluados normalizados.
- Catálogos independientes: para mantener listas como categorías de habitaciones, formas de pago, tipos de servicios, estados de reserva, etc.

Organización del Código

El sistema ha sido estructurado principalmente siguiendo principios de arquitectura modular y buenas prácticas, todo con el fin de facilitar un desarrollo a futuro del sistema. A continuación, se presenta los distintos paquetes utilizados.





Ejecución Completa del Proyecto

El Ejecución Completa del Proyecto:

La ejecución del proyecto se llevó a cabo en varias fases, comenzando con la planificación y el análisis de requerimientos, seguidas por la implementación, pruebas y finalmente el despliegue en el entorno de producción. Todo esto siguiendo una metodología practica basada en etapas progresivas, para la construcción de un sistema robusto y profesional

1. **Modelado lógico:** Se diseñó el modelo entidad-relación utilizando herramientas como dbdiagram.io, garantizando normalización hasta 3FN y relaciones claras entre entidades.
2. **Construcción de la base de datos:** Se implementaron más de 10 tablas funcionales, catálogos auxiliares y relaciones con integridad referencial en SQL Server.
3. **Carga de datos:** Se insertaron más de 500 registros simulados para pruebas de carga, cubriendo habitaciones, reservas, servicios, clientes y facturación.

4. **Programación avanzada:** Se desarrollaron procedimientos almacenados, funciones definidas por el usuario y triggers, cumpliendo con las reglas de negocio del sistema.
5. **Seguridad y roles:** Se crearon roles personalizados y se aplicaron políticas de acceso mediante GRANT, REVOKE y cifrado de datos sensibles.
6. **Auditoría:** Se registraron todas las operaciones críticas en la tabla log_acciones, incluyendo acción, usuario, IP, y fecha.
7. **Respaldos y restauración:** Se ejecutaron respaldos en caliente y en frío, con evidencia y verificación de recuperación.
8. **Optimización:** Se aplicaron índices simples y compuestos, y se midió el rendimiento con EXPLAIN y comparativas antes/después.

Evidencias

Nº	Evidencia	Descripción
1	<pre> CREATE TABLE [Habitacion] ([id_habitacion] INT PRIMARY KEY IDENTITY(1, 1), [categoría] VARCHAR(50), [precio_noche] DECIMAL(10,2), [capacidad] INT, [servicios] TEXT, [estado] VARCHAR(20)) GO CREATE TABLE [Huesped] ([id_huesped] INT PRIMARY KEY IDENTITY(1, 1), [nombre] VARCHAR(100), [apellidos] VARCHAR(100), [cedula] VARCHAR(20) UNIQUE, [email] VARCHAR(100), [telefono] VARCHAR(20)) GO CREATE TABLE [Reserva] ([id_reserva] INT PRIMARY KEY IDENTITY(1, 1), [id_huesped] INT, [id_habitacion] INT, [fecha_ingreso] DATE, [fecha_salida] DATE, [forma_pago] VARCHAR(50), [estado] VARCHAR(20)) GO </pre>	Script SQL ejecutado con estructura de la base de datos.

	<pre> 17 CREATE TABLE [ServicioAdicional] (18 [id_servicio] INT PRIMARY KEY IDENTITY(1, 1), 19 [nombre] VARCHAR(100), 20 [descripcion] TEXT, 21 [costo_unitario] DECIMAL(10,2) 22) 23 GO 24 25 CREATE TABLE [ConsumoServicio] (26 [id_consumo] INT PRIMARY KEY IDENTITY(1, 1), 27 [id_reserva] INT, 28 [id_servicio] INT, 29 [cantidad] INT, 30 [fecha] DATE 31) 32 GO 33 34 CREATE TABLE [Factura] (35 [id_factura] INT PRIMARY KEY IDENTITY(1, 1), 36 [id_reserva] INT, 37 [fecha_emision] DATE, 38 [total] DECIMAL(10,2) 39) 40 GO 41 42 CREATE TABLE [Usuario] (43 [id_usuario] INT PRIMARY KEY IDENTITY(1, 1), 44 [username] VARCHAR(50) UNIQUE, 45 [contraseña] VARCHAR(256), 46 [rol_id] INT 47) 48 GO 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700 701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000 </pre>	
2	<pre> 1 -- HABITACIONES -- 2 INSERT INTO Habitaciones (categoria, precio, noche, capacidad, servicios, estado) VALUES 3 ('Single', 137.79, 3, 'WiFi, TV', 'disponible'), 4 ('Doble', 89.97, 2, 'WiFi, TV, Aire', 'ocupado'), 5 ('Suite', 88.78, 4, 'WiFi, TV, Jacuzzi', 'mantenimiento'), 6 ('Single', 74.88, 1, 'WiFi, TV', 'disponible'), 7 ('Doble', 140.48, 2, 'WiFi, TV, Aire', 'ocupado'), 8 ('Suite', 189.76, 4, 'WiFi, TV, Jacuzzi', 'mantenimiento'), 9 ('Single', 66.14, 1, 'WiFi, TV', 'disponible'), 10 ('Doble', 162.14, 2, 'WiFi, TV, Aire', 'ocupado'), 11 ('Suite', 89.79, 4, 'WiFi, TV, Jacuzzi', 'mantenimiento'), 12 ('Single', 88.88, 1, 'WiFi, TV', 'disponible'), 13 ('Doble', 81.28, 2, 'WiFi, TV, Aire', 'ocupado'), 14 ('Suite', 185.99, 4, 'WiFi, TV, Jacuzzi', 'mantenimiento'), 15 ('Single', 82.61, 1, 'WiFi, TV', 'disponible'), 16 ('Doble', 171.67, 2, 'WiFi, TV, Aire', 'ocupado'), 17 ('Suite', 184.87, 4, 'WiFi, TV, Jacuzzi', 'mantenimiento'), 18 ('Single', 185.71, 1, 'WiFi, TV', 'disponible'), 19 ('Doble', 129.69, 2, 'WiFi, TV, Aire', 'ocupado'), 20 ('Suite', 98.91, 4, 'WiFi, TV, Jacuzzi', 'mantenimiento'), 21 ('Single', 89.81, 1, 'WiFi, TV', 'disponible'), 22 ('Doble', 89.18, 2, 'WiFi, TV, Aire', 'ocupado'), 23 ('Suite', 179.87, 4, 'WiFi, TV, Jacuzzi', 'mantenimiento'), 24 ('Single', 151.89, 1, 'WiFi, TV', 'disponible'), 25 ('Doble', 110.35, 2, 'WiFi, TV, Aire', 'ocupado'), 26 ('Suite', 110.28, 4, 'WiFi, TV, Jacuzzi', 'mantenimiento'), 27 ('Single', 88.63, 1, 'WiFi, TV', 'disponible'), 28 ('Doble', 119.28, 2, 'WiFi, TV, Aire', 'ocupado'), 29 ('Suite', 82.28, 4, 'WiFi, TV, Jacuzzi', 'mantenimiento'), 30 ('Single', 84.14, 1, 'WiFi, TV', 'disponible'), 31 ('Doble', 121.86, 2, 'WiFi, TV, Aire', 'ocupado'), 32 ('Suite', 145.48, 4, 'WiFi, TV, Jacuzzi', 'mantenimiento'), 33 ('Single', 170.88, 1, 'WiFi, TV', 'disponible'), 34 ('Doble', 184.89, 2, 'WiFi, TV, Aire', 'ocupado'), 35 ('Suite', 82.81, 4, 'WiFi, TV, Jacuzzi', 'mantenimiento'), 36 ('Single', 60.59, 1, 'WiFi, TV', 'disponible'), 37 ('Doble', 180.1, 2, 'WiFi, TV, Aire', 'ocupado'), 38 ('Suite', 147.76, 4, 'WiFi, TV, Jacuzzi', 'mantenimiento'), 39 ('Single', 82.61, 1, 'WiFi, TV', 'disponible'), 40 ('Doble', 700.00, 2, 'WiFi, TV, Aire', 'ocupado') </pre>	Comandos de inserción de registros iniciales.

	<pre> -- RESERVAS -- INSERT INTO Reserva (id_huesped, id_habitacion, fecha_ingreso, fecha_salida, forma_pago, estado) VALUES (100, 10, '2020-09-13', '2020-09-18', 'Transferencia', 'cancelado'), (40, 20, '2020-10-20', '2020-10-25', 'Transferencia', 'finalizado'), (70, 3, '2020-09-09', '2020-09-12', 'Transferencia', 'cancelado'), (20, 00, '2020-11-20', '2020-12-01', 'Efectivo', 'finalizado'), (10, 14, '2020-10-22', '2020-10-26', 'Transferencia', 'cancelado'), (10, 00, '2020-09-23', '2020-09-26', 'Transferencia', 'finalizado'), (9, 1, '2020-08-02', '2020-08-08', 'Efectivo', 'cancelado'), (30, 10, '2020-09-16', '2020-09-18', 'Tarjeta', 'cancelado'), (80, 00, '2020-09-26', '2020-09-28', 'Efectivo', 'cancelado'), (100, 3, '2020-07-10', '2020-07-14', 'Transferencia', 'cancelado'), (90, 10, '2020-10-12', '2020-10-13', 'Tarjeta', 'cancelado'), (10, 30, '2020-09-14', '2020-09-18', 'Tarjeta', 'finalizado'), (80, 3, '2020-09-22', '2020-09-26', 'Tarjeta', 'finalizado'), (10, 4, '2020-10-28', '2020-11-01', 'Efectivo', 'finalizado'), (80, 30, '2020-11-09', '2020-11-21', 'Tarjeta', 'cancelado'), (20, 00, '2020-07-01', '2020-08-01', 'Transferencia', 'cancelado'), (10, 10, '2020-11-09', '2020-11-11', 'Efectivo', 'finalizado'), (80, 30, '2020-10-09', '2020-10-09', 'Efectivo', 'finalizado'), (110, 40, '2020-09-03', '2020-09-03', 'Tarjeta', 'cancelado'), (10, 40, '2020-10-20', '2020-10-20', 'Transferencia', 'cancelado'), (40, 00, '2020-09-25', '2020-10-01', 'Efectivo', 'cancelado'), (30, 40, '2020-10-19', '2020-10-19', 'Tarjeta', 'cancelado'), (70, 30, '2020-10-27', '2020-11-01', 'Tarjeta', 'cancelado'), (100, 7, '2020-11-20', '2020-11-21', 'Tarjeta', 'finalizado'), (80, 10, '2020-09-03', '2020-09-08', 'Efectivo', 'cancelado'), (90, 00, '2020-10-27', '2020-10-01', 'Tarjeta', 'cancelado'), (8, 30, '2020-09-28', '2020-09-28', 'Efectivo', 'cancelado'), (110, 8, '2020-11-23', '2020-11-26', 'Tarjeta', 'finalizado'), (110, 3, '2020-10-20', '2020-10-20', 'Tarjeta', 'cancelado'), (110, 3, '2020-10-27', '2020-10-28', 'Efectivo', 'cancelado'), (80, 00, '2020-09-09', '2020-09-09', 'Tarjeta', 'cancelado'), (80, 10, '2020-09-27', '2020-10-01', 'Transferencia', 'cancelado'), (10, 30, '2020-11-09', '2020-11-10', 'Efectivo', 'cancelado'), </pre>	
3	<div> <div>Programación</div> <div> <div>Procedimientos almacenados</div> <div> <div>Procedimientos almacenados del sisten</div> <div> <div>dbo.AgregarReserva</div> <div>dbo.EliminarReservaSegura</div> <div>dbo.FinalizarReservasAntiguas</div> <div>dbo.GenerarFacturaReserva</div> <div>dbo.LoginSimulado</div> <div>dbo.ReporteReservasPorFecha</div> <div>dbo.sp_EliminacionSeguraHuesped</div> </div> </div> </div> </div>	Captura de Procedimientos Creados
4	<div> <div>Funciones escalares</div> <div> <div>dbo.fn_duracion_reserva</div> <div>dbo.fn_estado_texto</div> <div>dbo.fn_porcentaje_ocupacion</div> </div> </div>	Ejecución de funciones que calculan duración o porcentajes.
5	<pre> -- TRIGGERS -- -- Trigger 1: Auditoria - registrar inserciones en Reserva -- CREATE TRIGGER trg_log_insert_reserva ON Reserva AFTER INSERT AS BEGIN INSERT INTO Log_Acciones (usuario, ip, accion, tabla, id_afectado) SELECT SYSTEM_USER, '127.0.0.1', 'INSERT', 'Reserva', id_reserva FROM inserted; END; GO -- Trigger 2: Cambiar estado de habitación automáticamente al reservar -- CREATE TRIGGER trg_actualizar_estado_habitacion ON Reserva AFTER INSERT AS BEGIN UPDATE Habitacion SET estado = 'ocupada' WHERE id_habitacion IN (SELECT id_habitacion FROM inserted); END; GO </pre>	Captura de registros insertados automáticamente en tabla log_acciones.

	<pre> -- Trigger 3: Historial de estado de habitación (auditoria de cambio manual) -- CREATE TRIGGER trg_historial_estado_habitacion ON Habitación AFTER UPDATE AS BEGIN INSERT INTO HistorialEstadoHabitación (id_habitacion, estado_anterior, estado_nuevo, fecha) SELECT i.id_habitacion, d.estado, i.estado, GETDATE() FROM inserted i INNER JOIN deleted d ON i.id_habitacion = d.id_habitacion WHERE i.estado <> d.estado; END; GO </pre>	
6	Creación y asignación de roles	Ejecución de comandos GRANT y REVOKE.
7	Resultados de respaldos	Captura de consola mostrando archivos de respaldo generados.

Simulaciones:

Simulación	Descripción
Inserción masiva (500 registros)	Evaluación de rendimiento ante carga alta.
SQL Injection	Prueba con ' OR '1'='1 y validación de prevención activa.
Fallos de integridad referencial	Intento de eliminación de registros con dependencias activas.

Inserción Masiva:

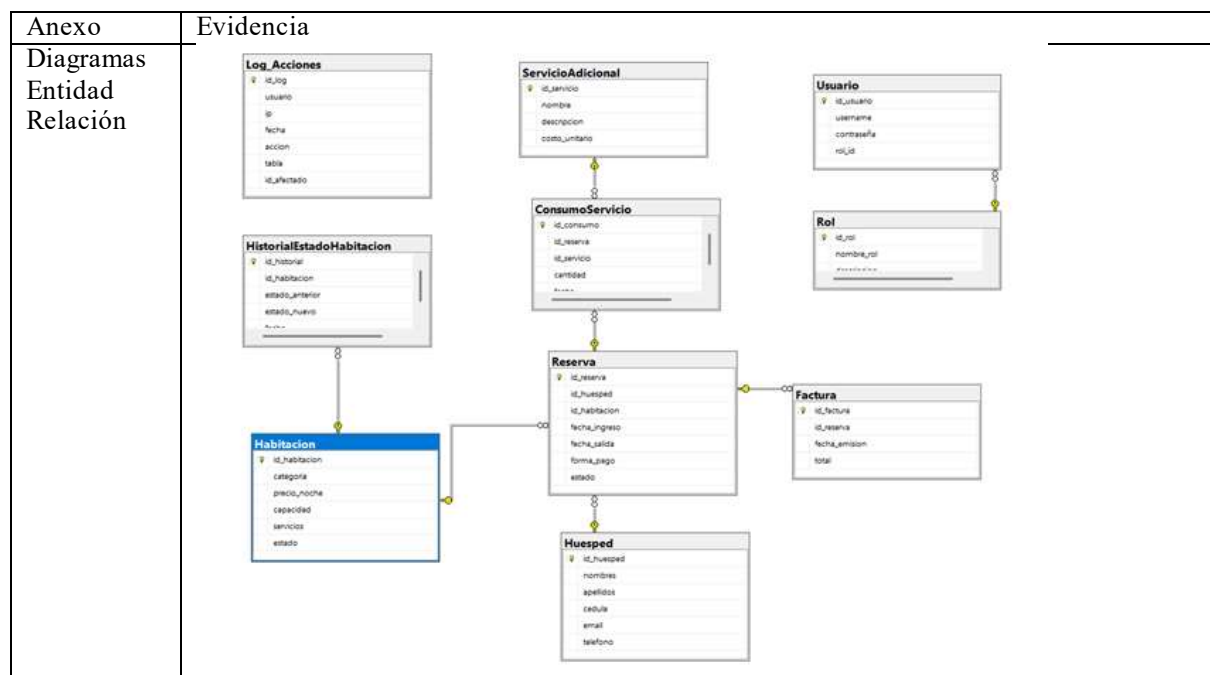
Conclusiones y Recomendaciones

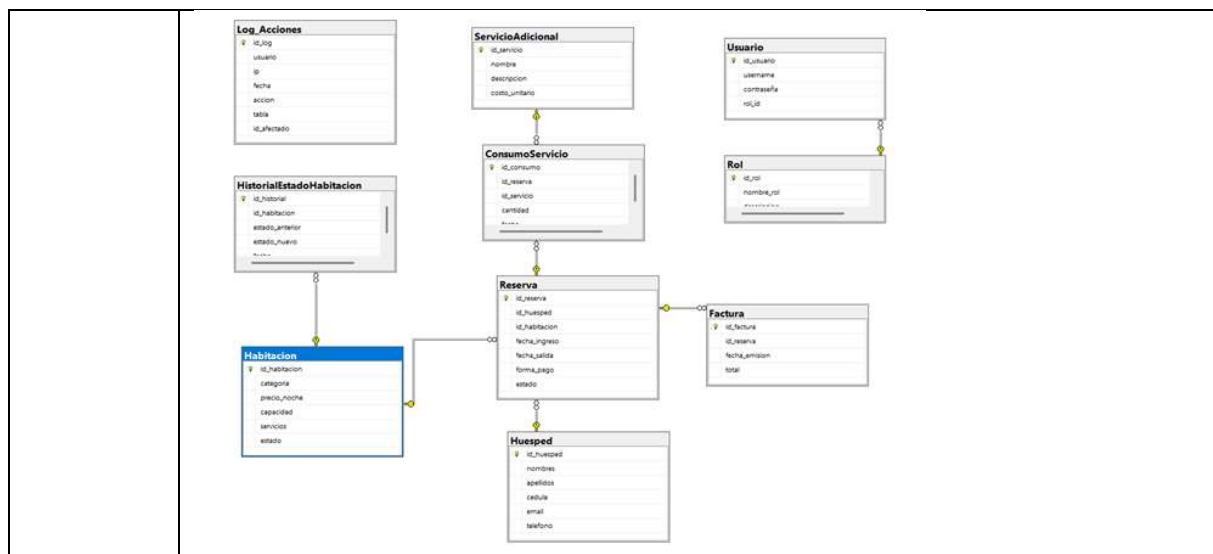
En conclusión, el desarrollo del sistema cumple con los estándares de una base de datos relaciona segura y eficiente, demostrando seguridad e integridad en el manejo de datos con roles asignados, seguido de una programación avanzada con procedimientos, funciones y triggers para el manejo de tareas, demostrando un sistema escalable y eficiente

Recomendaciones

- Programar respaldos automáticos periódicos y centralizar su almacenamiento.
- Implementar notificaciones en tiempo real usando triggers + servicios externos.
- Aplicar autenticación fuerte a nivel de conexión y monitoreo activo de uso en producción.
- Continuar realizando pruebas de seguridad más avanzadas

Anexos:

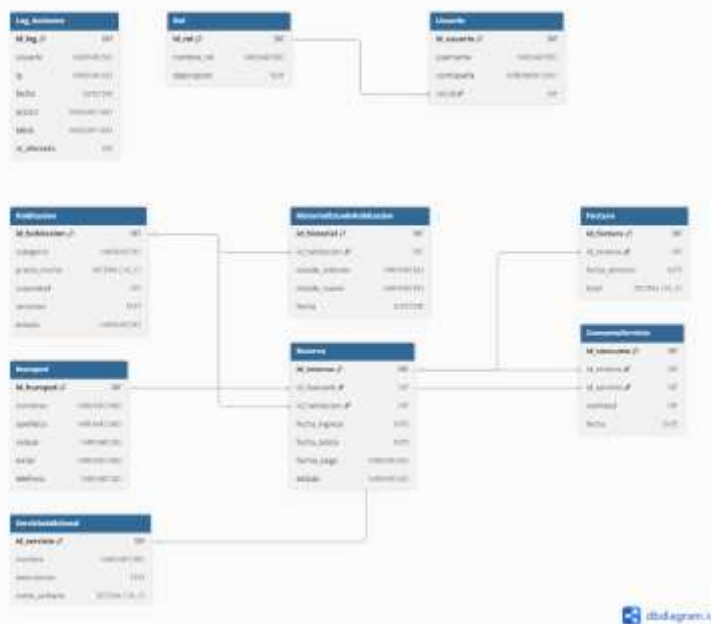




Enlace
GitHub

<https://github.com/AntonyCis/ProyectoBasesHosteleria.git>

Estructura
de Tablas



Ejecucion de Consultas, funciones y triggers

```

7  -- Consulta 1: Reporte de reservas con datos del huésped y habitación --
8  SELECT
9      R.id_reserva,
10     H.nombres + ' ' + H.apellidos AS nombre_huesped,
11     Hab.categoria,
12     R.fecha_ingreso,
13     R.fecha_salida,
14     R.estado
15 FROM Reserva R
16 JOIN Huesped H ON R.id_huesped = H.id_huesped
17 JOIN Habitacion Hab ON R.id_habitacion = Hab.id_habitacion
18 ORDER BY R.fecha_ingreso DESC
19
20 -- Consulta 2: Total consumido en servicios por cada reserva --

```

id_reserva	nombre_huesped	categoria	fecha_ingreso	fecha_salida	estado
135	Gabriel Vargas	Suite	2025-12-10	2025-12-13	reservado
145	Gasf Garcia	Simple	2025-12-07	2025-12-09	cancelado
125	Emiliano Zapata	Simple	2025-12-01	2025-12-05	reservado
84	Verónica Silva	Simple	2025-12-01	2025-12-04	cancelado
4	Daniela Rojas	Doble	2025-11-29	2025-12-03	finalizado
144	Thalia Sod	Doble	2025-11-29	2025-12-02	reservado
38	Valeria Guerrero	Doble	2025-11-24	2025-11-28	reservado
45	Fernando Moreno	Simple	2025-11-24	2025-11-26	cancelado
85	Martina Guerrero	Doble	2025-11-23	2025-11-27	finalizado
14	Isabel Morales	Suite	2025-11-23	2025-11-26	finalizado
29	Santiago Reyes	Doble	2025-11-23	2025-11-24	finalizado
99	Francisco Benitez	Doble	2025-11-23	2025-11-25	cancelado
140	Agustín Lara	Simple	2025-11-22	2025-11-26	reservado
87	Sara Peña	Doble	2025-11-21	2025-11-23	reservado
64	Oscar Padilla	Suite	2025-11-21	2025-11-23	reservado
24	Beatriz Santos	Simple	2025-11-20	2025-11-21	finalizado
15	Montserrat Luna	Doble	2025-11-20	2025-11-21	cancelado
73	Santiago Reyes	Suite	2025-11-20	2025-11-25	finalizado

```

20 -- Consulta 2: Total consumido en servicios por cada reserva --
21 SELECT
22     R.id_reserva,
23     H.nombres + ' ' + H.apellidos AS huesped,
24     SUM(CS.cantidad * SA.costo_unitario) AS total_servicios
25 FROM Reserva R
26 JOIN Huesped H ON R.id_huesped = H.id_huesped
27 JOIN ConsumoServicio CS ON R.id_reserva = CS.id_reserva
28 JOIN ServicioAdicional SA ON CS.id_servicio = SA.id_servicio
29 GROUP BY R.id_reserva, H.nombres, H.apellidos
30 ORDER BY total_servicios DESC
31
32 -- Consulta 3: habitaciones ocupadas hoy (sin usar estado) --
33 SELECT DISTINCT
34     Hab.id_habitacion,
35     Hab.categoria,
36     Hab.precio_noche,
37     R.fecha_ingreso,
38     R.fecha_salida,
39     H.nombres + ' ' + H.apellidos AS huesped

```

id_reserva	huesped	total_servicios
31	Santiago Reyes	131.00
57	Andrés Rodríguez	119.00
58	Diego Romero	90.00
16	Alejandra Salazar	85.00
104	Oscar Padilla	80.00
112	Arturo Mendoza	75.00
90	Gabriel López	75.00
30	Sebastián Jiménez	75.00
82	Valeria Guerrero	74.00
84	Verónica Silva	72.00
26	Paola Delgado	71.00
73	Santiago Reyes	65.00
27	Mariana Vargas	64.00
17	Claudia Guzmán	60.00
14	Isabel Morales	60.00
24	Beatriz Santos	60.00

FUNCIONES

```

230 RETURNS VARCHAR(100)
231 AS
232 BEGIN
233     DECLARE @estado VARCHAR(20);
234
235     SELECT @estado = estado FROM Reserva WHERE id_reserva = @id.
236
237     RETURN
238     CASE
239         WHEN @estado = 'reservado' THEN 'Reserva activa'
240         WHEN @estado = 'finalizado' THEN 'Reserva finalizado'
241         WHEN @estado = 'cancelado' THEN 'Reserva cancelada'
242         ELSE 'Desconocido'
243     END;
244 END;
245 GO
246
247 SELECT dbo.fn_estado_texto(5) AS estado_legible; --USO--
248 GO
249

```

100 %	11	0	↑	↓
Resultados	Mensajes			
estado_legible				
1	Reserva cancelada			

TRIGGERS

```

SQLQuery16.s...O\dgcis (80))  3.Consultas.sq...GO\dgcis (63))  10.Simulacion...GO\dgcis (51))  4.Indices y Op...GO\dgcis
1 SELECT TOP (1000) [id_log]
2     ,[usuario]
3     ,[rol_actual]
4     ,[terminal]
5     ,[ip]
6     ,[fecha]
7     ,[accion]
8     ,[tabla]
9     ,[transaccion_sql]
10    ,[id_afectado]
11 FROM [Hoteles].[dbo].[Log_Acciones]
12

```

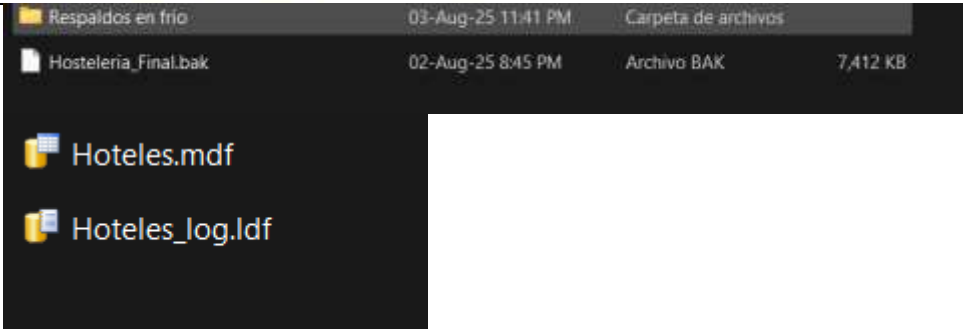
100 % No se encontraron problemas.

Resultados Mensajes

	id_log	usuario	rol_actual	terminal	ip	fecha	accion	tabla	transaccion_sql	id_afectado
1	1	Diego\dgcis	dbo	DIEGO	<local machine>	2025-08-02 19:40:34.563	INSERT	Usuario	INSERT en Usuario	3
2	2	Diego\dgcis	dbo	DIEGO	<local machine>	2025-08-02 19:41:47.073	UPDATE	Usuario	UPDATE en Usuario	3
3	3	Diego\dgcis	dbo	DIEGO	<local machine>	2025-08-02 19:41:56.770	DELETE	Usuario	DELETE en Usuario	3
4	4	Diego\dgcis	dbo	DIEGO	<local machine>	2025-08-02 20:19:32.173	INSERT	Habitacion	INSERT en Habitacion	51
5	5	Diego\dgcis	dbo	DIEGO	<local machine>	2025-08-02 20:23:23.343	INSERT	Reserva	INSERT en Reserva	151
6	6	Diego\dgcis	NULL	NULL	127.0.0.1	2025-08-02 20:23:23.347	INSERT	Reserva	NULL	151
7	7	Diego\dgcis	dbo	DIEGO	<local machine>	2025-08-02 20:23:23.367	UPDATE	Habitacion	UPDATE en Habitacion	1
8	8	Diego\dgcis	dbo	DIEGO	<local machine>	2025-08-02 20:29:45.863	INSERT	Factura	INSERT en Factura	51
9	9	Diego\dgcis	dbo	DIEGO	<local machine>	2025-08-02 20:30:43.870	DELETE	Factura	DELETE en Factura	51

Roles y Privilegios

-  RolAdministrador
-  RolAuditor
-  RolCamarera
-  RolCliente
-  RolRecepcionista

	<pre>-- Asignacion de permisos con GRANT -- -- otorgar permisos SELECT e INSERT a recepcionistas GRANT SELECT, INSERT ON Reserva TO RolRecepcionista; GRANT SELECT ON Huesped TO RolRecepcionista; -- Auditor solo con permisos de lectura GRANT SELECT ON Log_Acciones TO RolAuditor; -- Administrador con control total GRANT SELECT, INSERT, UPDATE, DELETE ON Habitacion TO RolAdministrador; GRANT CONTROL ON DATABASE::[Hoteles] TO RolAdministrador;</pre>
Respaldo caliente y frio	

Enlaces de Interés:

Github: <https://github.com/AntonyCis/ProyectoBasesHosteleria.git>

Youtube: https://youtu.be/o1yei_iA-ik