

Proyecto Interdisciplinario – Clínica Privada Internacional

Proyecto Interdisciplinario Clínica Privada Internacional

1. Presentación

Este proyecto interdisciplinario integra los contenidos desarrollados durante el año en las siguientes asignaturas:

Asignatura	Área de aplicación en el proyecto
Análisis de Sistemas	Relevamiento, modelado y documentación de requisitos.
Administración y Gestión de Bases de Datos	Diseño relacional, stored procedures, triggers y seguridad.
Laboratorio de Programación Orientada a Objetos	Backend en Java con API REST.
Proyecto Informático II	Frontend en React.js.
Redes	Arquitectura de red, seguridad y virtualización.

El trabajo se realizará de forma paralela al avance de los contenidos de cada materia. La entrega final debe ser completa; los proyectos que no incluyan todos los componentes requeridos serán calificados como desaprobados.

Los grupos que no superen la evaluación ordinaria podrán presentar una segunda instancia en diciembre. En esa ocasión, los estudiantes que tengan pendiente alguna de las materias

involucradas previamente deberán presentar y defender el proyecto interdisciplinario completo antes de rendir el contenido de la materia.

- **Trabajo grupal** → la defensa será **individual**. Cada estudiante debe conocer a fondo todos los aspectos del proyecto, sin importar el rol que haya desempeñado dentro del equipo.
- **Evaluaciones internas** → tres entregas a lo largo del año para monitorear el avance, la participación individual y el porcentaje de cumplimiento.

Nota: En la sección de Redes, los datos (direcciones IP, máscaras, etc.) se generarán aleatoriamente y, una vez definidos, no podrán modificarse.

2. Contexto

Una **clínica privada internacional** abre una nueva **sucursal** y requiere un **sistema integral** para la gestión de pacientes, turnos, historias clínicas, control de acceso y seguimiento médico, así como una **infraestructura de red** que garantice disponibilidad, confidencialidad y escalabilidad.

El sistema debe ser **modular**, **seguro** y **escalable**, con una interfaz amigable tanto para los **usuarios finales** (pacientes) como para los **administradores** (médicos, personal de salud y administrativos).

El proyecto integrará conocimientos de:

1. Análisis de Sistemas – Relevamiento de procesos, modelado UML.
2. Programación Orientada a Objetos – Backend en Java (API REST).
3. Proyecto Informático II – Frontend en React.js.
4. Administración de Bases de Datos – Modelo relacional, stored procedures, triggers.
5. Redes – Diseño de topología, servicios de red y virtualización.

3. Requisitos Técnicos

3.1 Análisis de Sistemas

1. Relevamiento

- Entrevistas/encuestas a médicos, pacientes y personal administrativo.
- Documentar procesos actuales (registro de pacientes, emisión de turnos, control de historias clínicas).

2. Diagramas UML

- Casos de uso: "Registrar paciente", "Agendar turno", "Generar historia clínica", "Validar acceso al historial".
- Diagrama de secuencia: "Proceso de consulta y generación de historia clínica".

3. Propuesta funcional

- Módulos: Pacientes, Turnos, Historias Clínicas, Médicos, Reportes, Usuarios / Roles.

3.2 Programación Orientada a Objetos – Backend (Java)

1. API REST (Servlets)

- POST /api/appointments – Agendar turno.
- GET /api/patients/{id} – Obtener datos del paciente.
- POST /api/medical-records – Registrar historia clínica.
- GET /api/access/{token} – Validar acceso al historial clínico.

2. Validaciones

- Evitar solapamientos de horarios.
- Comprobar disponibilidad de médicos.
- Verificar integridad de datos (alergias, especialidad, obra social).

3. Patrones de diseño

- **Singleton** – Conexión a la base de datos.
- **Strategy** – Tipos de consulta (general, especialista, urgencia).

4. Diagramas UML – Diagrama de clases (Entidades: Paciente , Turno , HistoriaClínica , Médico , Usuario).

5. Pruebas unitarias – JUnit para lógica de validación de alergias y conflictos de agenda.

3.3 Proyecto Informático II – Frontend (React.js)

1. Portal de pacientes

- Agenda de turnos.
- Visualización de historial clínico (con QR de acceso).
- Notificaciones (email y push) sobre turnos y resultados.

2. Panel de médicos

- Gestión de pacientes y turnos asignados.
- Registro de diagnósticos y tratamientos.
- Reportes de salud por especialidad.

3. Panel de administradores

- ABM de médicos, pacientes, turnos y usuarios.
- Reportes de ocupación, ingresos, especialidades más demandadas.
- Gestión de roles y permisos.

4. Navegación – react-router-dom para rutas protegidas (ej: /medico, /admin).

5. Estado global – Context API para sesión de usuario, notificaciones y configuración.

6. Consumo de API

```
// Agendar turno
const response = await fetch("http://localhost:8080/api/appointments", {
  method: "POST",
  headers: { "Content-Type": "application/json" },
  body: JSON.stringify(appointmentData)
});
```

3.4 Administración de Bases de Datos

1. Modelo relacional (DER)

- Pacientes (id, nombre, dni, email, teléfono, obraSocial, alergias)
- Turnos (id, idPaciente, idMedico, fecha, hora, estado)
- HistoriasClínicas (id, idPaciente, idMedico, diagnostico, tratamiento, fecha)
- Médicos (id, nombre, especialidad, matrícula, disponibilidad)
- Usuarios (id, nombre, email, rol, contraseñaHash)

2. Normalización – Hasta Tercera Forma Normal (3FN) para evitar redundancia.

3. Scripts SQL

- ≥ 5 stored procedures (sp_AgendarTurno , sp_GenerarHistoria , sp_ReporteMensualIngresos).
- ≥ 4 triggers (trg_RegistrarAccesoHistorial , trg_ActualizarDisponibilidadMedico).

4. Seguridad

- Roles: admin (acceso total), medico (lectura/escritura en sus pacientes), paciente (solo lectura de su historial).

3.5 Redes

1. Simulación en Packet Tracer

- Topología: Router perimetral → Switch core → Servidores (API Java, BD, Web).

- VLANs: VLAN-ADMIN (personal administrativo), VLAN-MEDICAL (consultorios y equipos médicos), VLAN-PATIENTS (Wi-Fi para pacientes), VLAN-SECURITY (cámaras y monitoreo).

2. Servicios

- Web Hosting (Nginx) para frontend React.
- DNS interno y externo (clinicaprivada-intl.com).
- DHCP para dispositivos de la clínica.
- SFTP para backups de historiales clínicos y logs de acceso.

3. Presupuesto de red

- Routers (Cisco ISR 4321), switches (Cisco Catalyst 2960), servidores (Dell PowerEdge), cámaras IP, puntos de acceso Wi-Fi, UPS.
- Detalle: costo hardware + licencias + mantenimiento anual.

4. Cálculo de IPs (Subneteo) – Plan de direccionamiento IPv4 (ej. 10.15.0.0/16) con subredes por VLAN.

5. Virtualización – ESXi para servidores de aplicación y base de datos.

4. Entregables Obligatorios

Entregable	Detalle
Código Fuente	Repositorio GitHub con ramas <code>main</code> , <code>dev</code> , <code>feature/*</code> .
Documentación	Relevamiento (actas, encuestas a médicos y pacientes).
Documentación	Tesina – Mínimo 50 páginas.
Documentación	Manual de usuario (pacientes, médicos, administradores).
Diseños	Prototipos UI/UX (Figma o Diagrams.net).
Diseños	Diagramas UML, DER, Gantt, CPM.
Redes	Archivo Packet Tracer (.pkt).
Redes	Plano de red en PDF (cables, rack, etiquetado).
Redes	Subneteo y tabla de IPs.

Redes	Presupuesto detallado.
Redes	Modelo 3D (SketchUp/Blender) de la clínica.
Scrum Artifacts	Product Backlog (Jira/Trello).
Scrum Artifacts	Sprint Backlog y Burndown Chart.
Scrum Artifacts	Actas de Daily Scrum.
Scrum Artifacts	Informe de Sprint Review y Retrospective.

5. Estructura del Proyecto con Scrum

Fase	Actividad
1. Relevamiento	Entrevistas a médicos, pacientes y personal administrativo.
1. Relevamiento	Observación de procesos de registro, turnos y atención médica.
1. Relevamiento	Obtención de requisitos funcionales y no funcionales.
2. Formación del Equipo Scrum	Product Owner: gerente de la nueva sucursal.
2. Formación del Equipo Scrum	Scrum Master: docente o estudiante.
2. Formación del Equipo Scrum	Development Team: todos los miembros del grupo.
3. Creación del Product Backlog	Épicas: "Como paciente, quiero ver mi historial clínico desde mi celular".
4. Sprint Planning	Selección de tareas: "Diseñar modelo de datos", "Crear API de turnos".
5. Daily Scrum	Registro diario: lo hecho, lo próximo, impedimentos.

6. Sprint Backlog	Seguimiento en Trello o Jira.
7. Burndown Chart	Gráfico de progreso del sprint.
8. Sprint Review	Demostración del módulo de turnos al Product Owner.
9. Sprint Retrospective	Análisis de mejora: “Mejorar la seguridad en el acceso a historiales clínicos”.

6. Equipamiento Tecnológico por Departamento

Departamento	Dimensiones	PC	Impresoras de Red	Plotters	Impresoras 3D	Notas
1. Dirección General						Oficina ejecutiva con sala de reuniones
2. Consultorios Médicos						Equipos médicos conectados a red
3. Sala de Espera						Wi-Fi para pacientes, pantallas informativas
4. Farmacia Interna						Impresora de etiquetas para medicamentos
5. Laboratorio Clínico						Lectores de códigos y conectividad

6. Hospitalización						Monitoreo en tiempo real
7. Radiología e Imágenes						Equipos de alta resolución y almacenamiento
8. Administración						Archivo digital y físico seguro
9. Recursos Humanos						Sala de entrevistas y capacitación
10. Seguridad y Monitoreo						Centro de cámaras y control de acceso
11. Finanzas y Facturación						Control de cobros y obras sociales
12. Tecnología y Soporte						Sala de servidores adyacente
13. Urgencias						Red aislada para equipos críticos
14. Cirugía						Acceso restringido, monitoreo constante
15. Nutrición y Bienestar						Consultas personalizadas y seguimiento

16. Psicología y Salud Mental						Confidencialidad reforzada
17. Sostenibilidad						Gestión de residuos médicos
18. Inteligencia de Salud						Pantallas con datos de ocupación y especialidades
19. Alianzas Estratégicas						Convenios con seguros y hospitales
20. Auditoría Médica						Acceso restringido a historiales y tratamientos