**CARRERA DE INGENIERÍA EN SOFTWARE**

Un conjunto de letras blancas en un fondo blanco

Descripción generada automáticamente con confianza media

**Aplicaciones cliente**

**Autor:**

Muñiz Rivas Leopoldo Miquel

**Tema:**

**Proyecto de asistencia de docentes (Angular)**

**Curso:**

4 - A

**Periodo 2025(1)**

Índice

[1.1. Resumen 15](#_Toc202804141)

[1.2. Finalidad del sistema y alcance 15](#_Toc202804142)

[1.3. Características clave por el papel de usuario 16](#_Toc202804143)

[1.1.3 Características del administrador 16](#_Toc202804144)

[1.2.3 Características del profesor 16](#_Toc202804145)

[1.4. Arquitectura Técnica 17](#_Toc202804146)

[1.1.4 Core Service Layer Architecture 17](#_Toc202804147)

[1.2.4 Aventura y flujo de gestión de usuarios 18](#_Toc202804148)

[1.5. Ejecución del servicio básico 19](#_Toc202804149)

[1.1.5 Operaciones de servicios de asistencia 19](#_Toc202804150)

[1.2.5 Operaciones de Servicio Docente 20](#_Toc202804151)

[1.6. Modelos de datos y persistencia 20](#_Toc202804152)

[1.1.6 Modelos de datos básicos 20](#_Toc202804153)

[1.2.6 Estrategia de persistencia 21](#_Toc202804154)

[1.7. Estructura de Arquitectura y Componente de UI 22](#_Toc202804155)

[1.1.7 Sistema de componentes compartidos 22](#_Toc202804156)

[1.2.7 Apiladora de la tecnología 23](#_Toc202804157)

[2. Empezar 23](#_Toc202804158)

[2.1. Requisitos de prerrequisitos y requisitos del sistema 24](#_Toc202804159)

[2.2. Proceso de instalación 24](#_Toc202804160)

[2.3. Descripción de la arquitectura de aplicaciones 25](#_Toc202804161)

[2.4. Configuración de servidor de desarrollo 26](#_Toc202804162)

[2.5. Configuración del sistema de estilo 26](#_Toc202804163)

[2.6. Server-Side Rendering (SSR) Configuración 27](#_Toc202804164)

[2.7. Comandos de inicio rápido 28](#_Toc202804165)

[3. Configuración del proyecto 28](#_Toc202804166)

[3.1. Prerrequisitos 28](#_Toc202804167)

[3.2. Configuración del entorno para el desarrollo 29](#_Toc202804168)

[3.1.2 Instalación inicial 29](#_Toc202804169)

[3.2.2 Configuración de servidor de desarrollo 29](#_Toc202804170)

[3.3.2 Configuración de construcción 30](#_Toc202804171)

[3.3. Corriendo la aplicación 30](#_Toc202804172)

[3.1.3 Modo de desarrollo 30](#_Toc202804173)

[3.2.3 Construcción de la producción 31](#_Toc202804174)

[3.3.3 Server-Side Rendering (SSR) 31](#_Toc202804175)

[3.4. Estructura de archivos y configuración de activos 31](#_Toc202804176)

[3.1.4 Gasoducto de Activos 31](#_Toc202804177)

[3.2.4 Configuración de estilo 32](#_Toc202804178)

[3.5. Configuración de pruebas 32](#_Toc202804179)

[3.1.5 Pruebas de unidad 32](#_Toc202804180)

[3.2.5 Configuración de prueba 32](#_Toc202804181)

[3.6. Flujo de trabajo para el desarrollo 33](#_Toc202804182)

[3.1.6 Mira Modo 33](#_Toc202804183)

[3.2.6 Estructura del proyecto 34](#_Toc202804184)

[4. Dependencias 34](#_Toc202804185)

[4.1. Estrategia de gestión de paquetes 34](#_Toc202804186)

[4.1.1 Configuración de Administrador de paquetes 35](#_Toc202804187)

[4.2. Dependencias del Marco Básico 35](#_Toc202804188)

[4.1.2 Resumen del ecosistema angular 35](#_Toc202804189)

[4.2.2 Componentes clave del marco 35](#_Toc202804190)

[4.3. Dependencias de IU y Estilo 36](#_Toc202804191)

[4.1.3 Arquitectura de la Biblioteca de IU 37](#_Toc202804192)

[4.2.3 Desintegración de la estrategia de estilo 37](#_Toc202804193)

[4.4. Dependencias de Desarrollo y Construcción 38](#_Toc202804194)

[4.1.4 Plan de herramientas de desarrollo 38](#_Toc202804195)

[4.2.4 Construir Scripts Configuración 38](#_Toc202804196)

[4.5. Configuración de representaciones de servidor-ser 39](#_Toc202804197)

[4.1.5 Cadena de dependencia de la SSR 39](#_Toc202804198)

[4.6. Estrategia de gestión de versiones 40](#_Toc202804199)

[4.1.6 Restricciones de la versión 40](#_Toc202804200)

[4.7. Seguridad y mantenimiento de la dependencia 40](#_Toc202804201)

[4.1.7 Bloquear la estructura de archivos Resumen 40](#_Toc202804202)

[5. Arquitectura 41](#_Toc202804203)

[5.1. Patrón de Arquitectura del Sistema 41](#_Toc202804204)

[5.2. Estructura general del sistema 41](#_Toc202804205)

[5.3. Arquitectura de capa de servicio 42](#_Toc202804206)

[5.1.3 Dependencias e Interacciones de Servicios 42](#_Toc202804207)

[5.4. Arquitectura de componentes 43](#_Toc202804208)

[5.1.4 Organización del módulo de características 43](#_Toc202804209)

[5.5. Arquitectura de flujo de datos 43](#_Toc202804210)

[5.1.5 Gestión del Estado y flujo de datos 44](#_Toc202804211)

[5.6. Patrones de comunicación de componentes 44](#_Toc202804212)

[5.7. Consumo de inyección y señalización de servicio 44](#_Toc202804213)

[5.8. Arquitectura de componentes compartidos 46](#_Toc202804214)

[5.1.8 Estructura de componentes compartidos 46](#_Toc202804215)

[5.9. Arquitectura de enrutamiento 46](#_Toc202804216)

[5.1.9 Estructura de configuración de ruta 47](#_Toc202804217)

[6. Servicios básicos 47](#_Toc202804218)

[6.1. Resumen de la arquitectura de servicio 47](#_Toc202804219)

[6.2. Dependencias e Interacciones de Servicios 48](#_Toc202804220)

[6.3. Servicio de asistencia 49](#_Toc202804221)

[6.1.3 Características clave 49](#_Toc202804222)

[6.2.3 Gestión estatal basada en la señal 49](#_Toc202804223)

[6.1.3 Operaciones críticas 49](#_Toc202804224)

[6.2.3 Justificación Flujo de trabajo 51](#_Toc202804225)

[6.4. MaestroServicio 51](#_Toc202804226)

[6.1.4 Operaciones de datos básicos 51](#_Toc202804227)

[6.2.4 Gestión de los cuadros 51](#_Toc202804228)

[6.3.4 Inicialización de datos de simulacros 52](#_Toc202804229)

[6.5. AuthService 53](#_Toc202804230)

[6.1.5 Estado de autenticación 53](#_Toc202804231)

[6.1.5 Login Flow 53](#_Toc202804232)

[6.2.5 Navegación basada en el papel 54](#_Toc202804233)

[6.6. Patrones de persistencia de datos 54](#_Toc202804234)

[6.7. Cargada de datos y serialización 54](#_Toc202804235)

[6.8. Usage de servicio en componentes 54](#_Toc202804236)

[6.9. Ejemplo de integración de componentes 55](#_Toc202804237)

[6.10. Consumo de datos reactivos 55](#_Toc202804238)

[7. Sistema de enrutamiento 55](#_Toc202804239)

[7.1. Arquitectura general de enrutamiento 56](#_Toc202804240)

[7.1.1 Diagrama de estructura de la ruta 56](#_Toc202804241)

[7.2. Sistema de tanteo encubierto 56](#_Toc202804242)

[7.1.2 Autenticación de la Guardia Civil de Flujo y Rutas 57](#_Toc202804243)

[7.3. Rotas del módulo de características 57](#_Toc202804244)

[7.1.3 Rutas del Portal de Administración 58](#_Toc202804245)

[7.2.3 Rutas del Portal de Maestros 58](#_Toc202804246)

[7.4. Ejecución de carga perezosa 59](#_Toc202804247)

[7.1.4 Cargando la comparación de la estrategia 59](#_Toc202804248)

[7.5. Manejo de la ruta predeterminada 60](#_Toc202804249)

[7.6. Integración de la ruta con Autenticación 60](#_Toc202804250)

[8. Autenticación 60](#_Toc202804251)

[8.1. Resumen del sistema 61](#_Toc202804252)

[8.2. Login de arquitectura de componentes 61](#_Toc202804253)

[8.1.2 Estructura de componentes 61](#_Toc202804254)

[8.2.2 Login Flow 62](#_Toc202804255)

[8.3. Arquitectura de AuthService 63](#_Toc202804256)

[8.1.3 Propiedades de servicio 63](#_Toc202804257)

[8.2.3 Métodos básicos 63](#_Toc202804258)

[8.4. Control de acceso basado en el papel 63](#_Toc202804259)

[8.1.4 Roles de Usuario y enrutamiento 63](#_Toc202804260)

[8.2.4 Lógica de validación de roles 64](#_Toc202804261)

[8.5. Persistencia de los datos 64](#_Toc202804262)

[8.1.5 Aplicación de la instalación 64](#_Toc202804263)

[8.6. Sistema de autenticación simulada 65](#_Toc202804264)

[8.1.6 Datos de Usuarios de Burla 65](#_Toc202804265)

[8.2.6 Mock Integración de datos 66](#_Toc202804266)

[8.7. Consideraciones de seguridad 66](#_Toc202804267)

[8.8. Reglas de validación 67](#_Toc202804268)

[9. Portal de administración 67](#_Toc202804269)

[9.1. Resumen del sistema 67](#_Toc202804270)

[9.1.1 Arquitectura de Portal de Admin 67](#_Toc202804271)

[9.2.1 Estructura de navegación 68](#_Toc202804272)

[9.2. Componentes básicos 68](#_Toc202804273)

[9.1.2 AdminDashboardComponent 68](#_Toc202804274)

[9.2.2 MaestroGestiónComponento 69](#_Toc202804275)

[9.3.2 AsistenciaConcompleto de Gestión 70](#_Toc202804276)

[9.3. Arquitectura de flujo de datos 71](#_Toc202804277)

[9.4. Integración de servicios 72](#_Toc202804278)

[9.1.4 AsistenciaService Integración 72](#_Toc202804279)

[9.2.4 Integración docente 73](#_Toc202804280)

[9.5. Patrones de flujo de trabajo de administración 74](#_Toc202804281)

[9.1.5 Justificación Revisión Flujo de trabajo 74](#_Toc202804282)

[9.2.5 Desplazamiento de trabajo de la Gerencia de 75](#_Toc202804283)

[10. Admin Dashboard 75](#_Toc202804284)

[10.1. Arquitectura de componentes 76](#_Toc202804285)

[10.2. Cálculo de las estadísticas 76](#_Toc202804286)

[10.3. Estructura del menú de navegación 77](#_Toc202804287)

[10.4. Componente de tarjeta de estadísticas 78](#_Toc202804288)

[10.1.4 Componente de acción rápida 78](#_Toc202804289)

[10.5. Dependencias de servicios 79](#_Toc202804290)

[10.1.5 Integración docente 79](#_Toc202804291)

[10.2.5 AsistenciaService Integración 79](#_Toc202804292)

[10.6. Estilo y Animaciones 80](#_Toc202804293)

[11. Gestión del profesorado 80](#_Toc202804294)

[11.1. Finalidad y alcance 80](#_Toc202804295)

[11.2. Arquitectura de sistemas 81](#_Toc202804296)

[11.1.2 Estructura de componentes 81](#_Toc202804297)

[11.2.2 Arquitectura de flujo de datos 81](#_Toc202804298)

[11.3. Componentes básicos 81](#_Toc202804299)

[11.1.3 MaestroGestiónComponento 81](#_Toc202804300)

[11.2.3 MaestroFormDialogComponent 82](#_Toc202804301)

[11.4. Interfaz de usuario 83](#_Toc202804302)

[11.1.4 Vista de lista de maestros 83](#_Toc202804303)

[11.2.4 Diálogo de formación de profesores 84](#_Toc202804304)

[11.5. Puntos de integración 85](#_Toc202804305)

[11.1.5 Dependencias de servicios 85](#_Toc202804306)

[11.2.5 Integración de la navegación 85](#_Toc202804307)

[11.6. Operaciones de datos 86](#_Toc202804308)

[11.1.6 Maestros Operaciones de la CRUD 86](#_Toc202804309)

[11.2.6 Normas de validación de la formación 86](#_Toc202804310)

[11.7. Manejo de errores 86](#_Toc202804311)

[11.1.7 Muestra de error de validación 86](#_Toc202804312)

[11.8. Consideraciones de seguridad 87](#_Toc202804313)

[12. Gestión de los horarios 87](#_Toc202804314)

[12.1. Resumen 87](#_Toc202804315)

[12.2. Componentes básicos 88](#_Toc202804316)

[12.1.2 ScheduleManagementComponent 88](#_Toc202804317)

[12.2.2 ScheduleFormDialogComponent 89](#_Toc202804318)

[12.3. Modelo y estructura de datos 89](#_Toc202804319)

[12.1.3 Programar Estructura de datos 89](#_Toc202804320)

[12.2.3 Agregación de la Lista 89](#_Toc202804321)

[12.4. Interfaz de uso y flujos de trabajo 90](#_Toc202804322)

[12.1.4 Interfaz de gestión de cuadros 90](#_Toc202804323)

[12.2.4 Calendario de formulario Diálogo 90](#_Toc202804324)

[12.5. Integración de servicios 91](#_Toc202804325)

[12.1.5 Integración docente 91](#_Toc202804326)

[12.2.5 Validación de la forma y encuadernación de datos 91](#_Toc202804327)

[12.6. Operaciones de la CRUD 92](#_Toc202804328)

[12.1.6 Crear horario 92](#_Toc202804329)

[12.2.6 Editar el calendario 92](#_Toc202804330)

[12.3.6 Borrar el calendario 93](#_Toc202804331)

[12.7. Navegación y diseño 93](#_Toc202804332)

[13. Gestión de asistencia 94](#_Toc202804333)

[13.1. Arquitectura de componentes 94](#_Toc202804334)

[13.1.1 Reseña del componente 94](#_Toc202804335)

[13.2.1 Integración de los servicios básicos 95](#_Toc202804336)

[13.2. Estructura de interfaz de usuario 95](#_Toc202804337)

[13.1.2 Organización de la Tab 95](#_Toc202804338)

[13.2.2 Asistencia Records Tab 96](#_Toc202804339)

[13.3.2 Justificaciones Tab 96](#_Toc202804340)

[13.3. Flujo de datos y filtrado 97](#_Toc202804341)

[13.1.3 Asistencia Data Flow 97](#_Toc202804342)

[13.2.3 Aplicación de la filtración 97](#_Toc202804343)

[13.4. Gestión de la situación 98](#_Toc202804344)

[13.1.4 Tipos de situación de asistencia 98](#_Toc202804345)

[13.2.4 Tipos de estado de Justificación 98](#_Toc202804346)

[13.5. Justificación Revisión Flujo de trabajo 99](#_Toc202804347)

[13.1.5 Flujo de proceso de revisión 99](#_Toc202804348)

[13.2.5 Estructura de componentes de diálogo 100](#_Toc202804349)

[13.6. Definiciones de columna 100](#_Toc202804350)

[13.1.6 Columnas de la Mesa de asistencia 100](#_Toc202804351)

[13.1.6 Justificación de columnas de mesa 100](#_Toc202804352)

[13.7. Métodos de ayudantes 101](#_Toc202804353)

[14. Maestro Portal 101](#_Toc202804354)

[14.1. Arquitectura Portal 101](#_Toc202804355)

[14.2. Estructura de componentes de Portal de Maestro 102](#_Toc202804356)

[14.3. Sistema de navegación 102](#_Toc202804357)

[14.4. Dependencias de servicios 103](#_Toc202804358)

[14.5. Resolución actual de maestros 103](#_Toc202804359)

[14.6. Registro de asistencia Flujo de Registro 104](#_Toc202804360)

[14.7. Gestión estatal reactiva 105](#_Toc202804361)

[14.8. Integración de componentes de la interfaz de usuario 105](#_Toc202804362)

[14.1.8 Componentes de materiales comunes 105](#_Toc202804363)

[14.9. Manipulación de errores y alimentación de usuario 106](#_Toc202804364)

[15. Maestro Dashboard 107](#_Toc202804365)

[15.1. Arquitectura de componentes 107](#_Toc202804366)

[15.2. Características clave 107](#_Toc202804367)

[15.1.2 Secretarías Dashboard 107](#_Toc202804368)

[15.2.2 La gestión de horarios de hoy 108](#_Toc202804369)

[15.3.2 Panel de medidas rápidas 108](#_Toc202804370)

[15.3. Arquitectura de flujo de datos 109](#_Toc202804371)

[15.4. Estructura de interfaz de usuario 109](#_Toc202804372)

[15.1.4 Integración de diseño 109](#_Toc202804373)

[15.1.4 Arquitectura de la plantilla 110](#_Toc202804374)

[15.5. Integración de la gestión de asistencia 110](#_Toc202804375)

[15.1.5 Actuación de Estado de asistencia 111](#_Toc202804376)

[15.6. Características de depuración y desarrollo 111](#_Toc202804377)

[15.7. Detalles de la ejecución técnica 112](#_Toc202804378)

[15.1.7 Estrategia de detección de cambios 112](#_Toc202804379)

[15.1.7 Inyección de dependencia 112](#_Toc202804380)

[15.1.7 Propiedades computadas 112](#_Toc202804381)

[16. Gestión de los horarios 112](#_Toc202804382)

[16.1. Resumen del sistema 112](#_Toc202804383)

[16.2. Componentes básicos 113](#_Toc202804384)

[16.1.2 MaestroScheduleComponent 113](#_Toc202804385)

[16.2.2 ScheduleChangeRequestDialogComponent 114](#_Toc202804386)

[16.3. Sistema de visualización de horarios 114](#_Toc202804387)

[16.1.3 Vista semanal del calendario 114](#_Toc202804388)

[16.2.3 Vista de mesa detallada 115](#_Toc202804389)

[16.4. Registro de asistencia 115](#_Toc202804390)

[16.1.4 Flujo de registro 115](#_Toc202804391)

[16.2.4 Actuación de Estado de asistencia 116](#_Toc202804392)

[16.5. Sistema de Solicitud de Cambio de Horario 116](#_Toc202804393)

[16.1.5 Solicitar estructura de formulario 116](#_Toc202804394)

[16.2.5 Solicitud de procesamiento 117](#_Toc202804395)

[16.6. Integración de flujo de datos 117](#_Toc202804396)

[16.1.6 Dependencias de servicios 117](#_Toc202804397)

[16.7. Componentes de interfaz de usuario 118](#_Toc202804398)

[16.1.7 Integración de la navegación 118](#_Toc202804399)

[16.2.7 Componentes de diseño de materiales 118](#_Toc202804400)

[17. Seguimiento de la asistencia 119](#_Toc202804401)

[17.1. Arquitectura de componentes 119](#_Toc202804402)

[17.2. Flujo de datos y gestión del Estado 120](#_Toc202804403)

[17.3. Características de Interfaz de Usuario 121](#_Toc202804404)

[17.1.3 Lista de asistencia 121](#_Toc202804405)

[17.2.3 Sistema de visualización de estado 121](#_Toc202804406)

[17.3.3 Fecha de filtración 122](#_Toc202804407)

[17.4. Funcionalidad de check-Out 123](#_Toc202804408)

[17.5. Integración con Servicios Básicos 124](#_Toc202804409)

[17.1.5 AsistenciaService Integración 124](#_Toc202804410)

[17.2.5 Integración docente 124](#_Toc202804411)

[17.3.5 Integración de austro vertebración 124](#_Toc202804412)

[17.6. Navegación y diseño 125](#_Toc202804413)

[18. Justifications 125](#_Toc202804414)

[18.1. Resumen del sistema 125](#_Toc202804415)

[18.2. Interfaz de maestro 127](#_Toc202804416)

[18.1.2 Estructura de componentes 127](#_Toc202804417)

[18.2.2 Mostrar tabla 127](#_Toc202804418)

[18.3.2 Gestión de la situación 127](#_Toc202804419)

[18.3. Presentación de justificación 128](#_Toc202804420)

[18.1.3 Estructura de la forma 128](#_Toc202804421)

[18.2.3 Validación de la forma 128](#_Toc202804422)

[18.3.3 Proceso de presentación 129](#_Toc202804423)

[18.4. Integración de servicios de apoyo 129](#_Toc202804424)

[18.1.4 Métodos de justificación 129](#_Toc202804425)

[18.2.4 Flujo de datos 130](#_Toc202804426)

[18.5. Modelo de datos 130](#_Toc202804427)

[18.1.5 AsistenciaSustificación Propiedades 130](#_Toc202804428)

[18.2.5 Estado ciclo de vida 131](#_Toc202804429)

[18.6. Persistencia de los datos 131](#_Toc202804430)

[18.1.6 Operaciones de almacenamiento 131](#_Toc202804431)

[18.2.6 Serialización de datos 131](#_Toc202804432)

[18.7. Puntos de integración 132](#_Toc202804433)

[18.1.7 Navegación portal docente 132](#_Toc202804434)

[18.1.7 Integración de autenticación 132](#_Toc202804435)

[19. Componentes compartidos 132](#_Toc202804436)

[19.1. Sistema de diseño 133](#_Toc202804437)

[19.1.1 LayoutComponent Architecture 133](#_Toc202804438)

[19.2.1 Interfaz de MenuItem 134](#_Toc202804439)

[19.3.1 Patrón de integración de diseño 134](#_Toc202804440)

[19.2. Componentes de tablero 134](#_Toc202804441)

[19.1.2 StatsCardComponent 134](#_Toc202804442)

[19.2.2 QuickActionsComponent 135](#_Toc202804443)

[19.3.2 Patrón de uso del tablero 136](#_Toc202804444)

[19.3. Componentes de diálogo 136](#_Toc202804445)

[19.1.3 Patrón de estructura de diálogo 136](#_Toc202804446)

[19.2.3 Patrones de estilo de diálogo 137](#_Toc202804447)

[19.4. Sistema de catalogación 138](#_Toc202804448)

[19.1.4 Configuración del tema 138](#_Toc202804449)

[19.2.4 Patternes de estilo 138](#_Toc202804450)

[19.5. Arquitectura de componentes 139](#_Toc202804451)

[19.1.5 Patrones de componentes 139](#_Toc202804452)

[19.6. Directrices de uso 140](#_Toc202804453)

[19.1.6 Uso de componentes de diseño 140](#_Toc202804454)

[19.1.6 Usage de los componentes del tablero 140](#_Toc202804455)

[19.1.6 Integración de diálogo 140](#_Toc202804456)

[20. Sistema de diseño 140](#_Toc202804457)

[20.1. Arquitectura de componentes 141](#_Toc202804458)

[20.2. Componente de diseño de núcleo 141](#_Toc202804459)

[20.1.2 Interfaz de MenuItem 142](#_Toc202804460)

[20.3. Sistema de navegación 142](#_Toc202804461)

[20.1.3 Aplicación de la navegación 142](#_Toc202804462)

[20.4. Arquitectura de diseño sensible 143](#_Toc202804463)

[20.1.4 Responsiva de las eliminatorias 143](#_Toc202804464)

[20.5. Integración de autenticación 144](#_Toc202804465)

[20.1.5 Métodos de autenticación 144](#_Toc202804466)

[20.2.5 Sistema de estilo 145](#_Toc202804467)

[20.3.5 Características de atipleje personalizado 145](#_Toc202804468)

[20.6. Patrones de uso 146](#_Toc202804469)

[20.1.6 Ejemplo de aplicación 146](#_Toc202804470)

[20.7. Integración de diseño de materiales 146](#_Toc202804471)

1. **Introducción general**

## Resumen

Este documento ofrece una visión general del sistema de gestión de la asistencia al profesorado (proyecto-asistencia-docentes-ng), una aplicación web construida con Angular que facilita el seguimiento y la gestión de la asistencia al profesorado en centros educativos. El sistema proporciona interfaces basadas en el papel para administradores y maestros, permitiendo un seguimiento eficiente de la asistencia, la gestión de horarios y los flujos de trabajo de justificación.

## Finalidad del sistema y alcance

El sistema de gestión de la asistencia al profesorado sirve como una plataforma centralizada para gestionar los horarios de los maestros, registrar la asistencia y manejar justificaciones de ausencias. La aplicación soporta dos funciones de usuario principales:

* **Administradores:** Acceso completo al sistema para la gestión de maestros, horarios y revisión de los registros **de** asistencia
* **Maestros:** Embarcación personal para ver horarios, registrar asistencia y presentar justificaciones

El sistema funciona como una aplicación del lado del cliente con la persistencia local de Storage, diseñada para uso institucional donde el seguimiento de la asistencia y la rendición de cuentas son esenciales.

**Procesos de Negocio Core:**

* Gestión y asignación de horarios de maestros
* Seguimiento de check-in/check-out de asistencia en tiempo real
* Ausencia de la presentación de justificaciones y flujo de **trabajo de aprobación**
* **Informes de asistencia y análisis exhaustivos**

**Fuentes:**

[**paquete.json1 a 51**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L1-L51)[**src/app/core/services/auth.service.ts1-97**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/auth.service.ts#L1-L97)[**src/app/core/services/teacher.serv.ts1-179**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/teacher.service.ts#L1-L179)[**src/app/core/services/attendance.service.ts**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/attendance.service.ts#L1-L234)

## Características clave por el papel de usuario

### Características del administrador

Los administradores acceden al sistema a través del portal administrativo, proporcionando capacidades de gestión integrales:

|  |  |  |
| --- | --- | --- |
| Característica | Componente | Objeto |
| Gestión del profesorado | TeacherManagementComponent | Operaciones de CRUD para expedientes de maestros |
| Gestión de los cuadros | Componentes de horario administrativo | Asignar y modificar horarios de los maestros |
| Supervisión de la asistencia | Componentes de asistencia administrativa | Ver todos los registros de asistencia |
| Revisión de justificación | Componentes de justificación administrativa | Aprobar/rechazar justificaciones de ausencia |

### Características del profesor

Los profesores interactúan con un salpicadero personal enfocado en sus horarios individuales y asistencia:

|  |  |  |
| --- | --- | --- |
| Característica | Componente | Objeto |
| Vista del horario | Componentes de horario de maestros | Ver clases y horarios asignados |
| Registro de asistencia | Componentes de asistencia a maestros | Check-in/check-out para las clases |
| Presentación de justificación | JustificationFormDialogComponent | Presentar justificaciones de ausencias |
| Historia de asistencia | Componentes de asistencia a maestros | Ver registros de asistencia personal |

**Fuentes:** [**src/app/features/admin/teachers/teacher-management.component.ts1-67**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-management.component.ts#L1-L67)[**src/app/features/maestro/justificaciones/justificación-form-dialog.s1-65**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/justifications/justification-form-dialog.component.ts#L1-L65)

## Arquitectura Técnica

### Core Service Layer Architecture

**A screenshot of a computer

AI-generated content may be incorrect.**

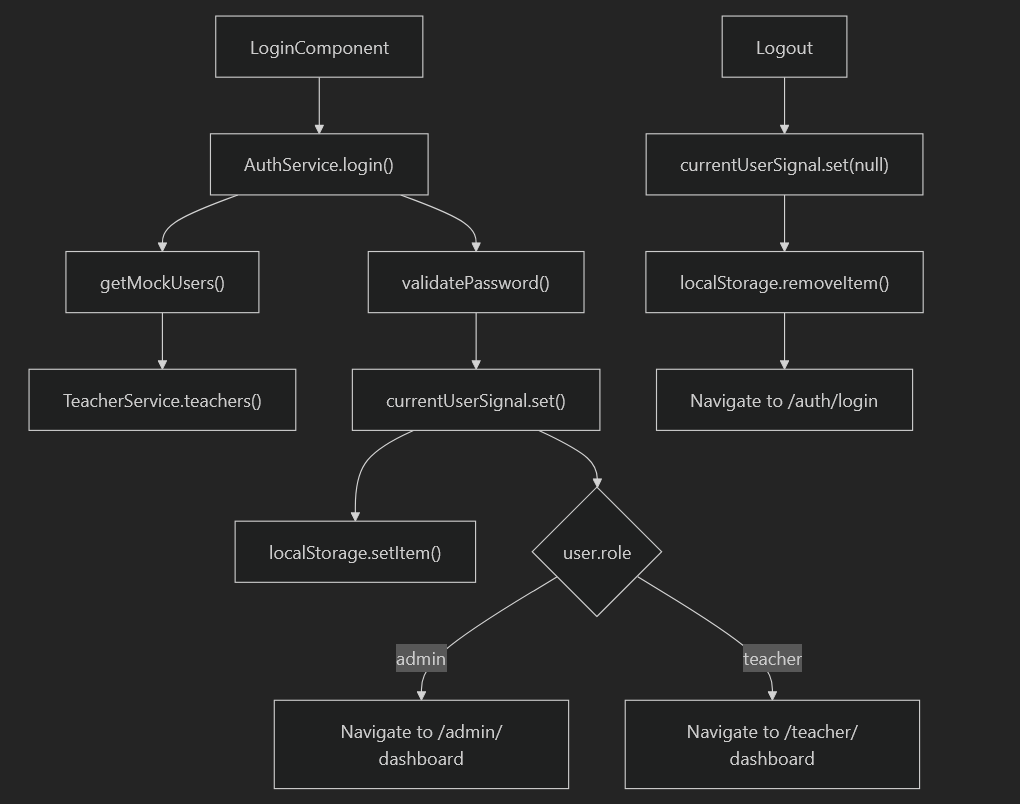
La aplicación sigue una arquitectura en capas con una clara separación de preocupaciones:

* Capa de componentes : Componentes angulares que implementan interfaces de usuario específicas
* Capa de servicio : Lógica de negocio encapsulado en servicios inyectables
* Capa de datos : interfaces TypeScript que definen estructuras de datos
* Capa de persistencia: persistencia de datos local basado en Storage

**Fuentes:**

[**src/app/core/services/auth.service.ts8-11**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/auth.service.ts#L8-L11)[**src/app/core/services/teacher.serv.ts6-11**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/teacher.service.ts#L6-L11)[**src/app/core/services/attendance.service.ts4 a 7**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/attendance.service.ts#L4-L7)

### Aventura y flujo de gestión de usuarios

****

El sistema de autenticación utiliza señales angulares para la gestión estatal reactiva e implementa el enrutamiento basado en el papel. El AuthServicegestiona sesiones de usuarios a través de la persistencia local de Storage y proporciona señales calculadas para el control de acceso basado en el rol.

**Fuentes:**

[**src/app/core/services/auth.service.ts42-70**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/auth.service.ts#L42-L70)[**src/app/core/services/auth.service.ts72 a 78**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/auth.service.ts#L72-L78)[**src/app/core/services/auth.service.ts13 a 18**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/auth.service.ts#L13-L18)

## Ejecución del servicio básico

### Operaciones de servicios de asistencia

El AttendanceServiceofrece una funcionalidad de seguimiento de asistencia completa:

**Métodos primarios:**

* checkIn(teacherId: string, scheduleId: string): Llega maestra de registros
* checkOut(teacherId: string, scheduleId: string): Salida del profesor de registros
* submitJustification(): Manipulas de las presentaciones de justificación de ausencia
* reviewJustification(): Examen de la justificación administrativa de los procesos

**Gestión de datos:**

* Utiliza señales angulares (s)attendancesSignal, justificationsSignal) para el estado reactivo
* Implementa persistencia local de almacenamiento con serialización JSON
* Proporciona filtrado basado en fechas y consultas específicas para maestros

Fuentes:

[**src/app/core/services/attendance.service.ts68-110**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/attendance.service.ts#L68-L110)[**src/app/core/services/attendance.service.ts112-137**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/attendance.service.ts#L112-L137)[**src/app/core/services/attendance.service.ts197 - 207**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/attendance.service.ts#L197-L207)

### Operaciones de Servicio Docente

El TeacherServicegestiona los registros del profesorado y las tareas de horario:

**Funcionalidad clave:**

* Operaciones de CRUD para expedientes de maestros
* Asignación y gestión del calendario
* Seriación de datos de casos de severo para el desarrollo
* Gestión estatal reactiva con señales

**Estructura de datos:**

* Los registros de maestros incluyen información personal, departamento y horarios asignados
* Los objetos de horario contienen cronometraje, asignatura e información de las aulas
* Relaciones mantenidas a través de teacherIdllaves extranjeras

Fuentes:

[**src/app/core/services/teacher.serv.ts106-116**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/teacher.service.ts#L106-L116)[**src/app/core/services/teacher.serv.ts140-159**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/teacher.service.ts#L140-L159)[**src/app/core/services/teacher.serv.ts39 a 96**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/teacher.service.ts#L39-L96)

## Modelos de datos y persistencia

### Modelos de datos básicos

**La aplicación utiliza interfaces TypeScript para definir estructuras de datos:**

* Usuario : Interfaz base para autenticación (role, email, name)
* Profesor: Extende el Usuario con los detalles del empleo (employeeId, department, schedules)
* Horario: Define el tiempo y los detalles de la clase (dayOfWeek, startTime, endTime, subject)
* Asistencia : Eventos de asistencia a registros (teacherId, scheduleId, checkIn, checkOut, status)
* AsistenciaSustificación : Adivimática justificaciones de ausencia (reason, status, reviewedBy)

### Estrategia de persistencia

El sistema utiliza localStorage para la persistencia del cliente con el siguiente enfoque:

**Claves de almacenamiento:**

* teachers: Registros y horarios de maestros
* attendances: Registros de asistencia
* justifications: Presentaciones de justificación
* currentUser: Sesión activa de usuario

**Serialización de datos:**

* JSON serialización para objetos complejos
* Objetos de fecha manipulados con el parsing/stringificación adecuado
* Manejo de errores para datos dañados con retroceso para burlarse de los datos

**Fuentes:**

[**src/app/core/services/attendance.service.ts18 - 54**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/attendance.service.ts#L18-L54)[**src/app/core/services/teacher.serv.ts22 a 37**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/teacher.service.ts#L22-L37)[**src/app/core/services/auth.service.ts27 - 40**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/auth.service.ts#L27-L40)

## Estructura de Arquitectura y Componente de UI

### Sistema de componentes compartidos

La aplicación utiliza una arquitectura de componentes compartidos para patrones de interfaz de usuario consistentes:

**Componentes compartidos básicos:**

* LayoutComponent: Concha de aplicación con navegación
* StatsCardComponent: Visualización de estadísticas del tablero
* QuickActionsComponent: Botones de acción para tareas comunes

**Integración de diseño de materiales:**

* Componentes de material angular para un estilo consistente
* Tailwind CSS para estilismo personalizado y diseños
* Patrones de diseño sensibles en todo

**Comunicación de componentes:**

* Señales angulares para la gestión estatal reactiva
* Inyección de servicio para el acceso a la lógica empresarial
* Componentes de diálogo para formularios y confirmaciones

**Fuentes:**

[**src/app/features/admin/dashboard/stats-card.component.html1-33**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/stats-card.component.html#L1-L33)[**src/app/features/admin/dashboard/quick-actions.component.html1-52**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/quick-actions.component.html#L1-L52)[**src/app/features/admin/teachers/teacher-management.component.ts9-11**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-management.component.ts#L9-L11)

### Apiladora de la tecnología

|  |  |  |
| --- | --- | --- |
| Tecnología | Versión | Objeto |
| Angular | 20.0.0 | Marco básico |
| Material angular | 20,0.2 | Componentes de la interfaz de |
| CSS de Tailwind | 4.1.10 | Sistema de estilo |
| RxJS | 7.8.0 | Programación reactivo |
| TipoScript | 5.8.2 | Seguridad de tipo |
| Express.js | 5.1.0 | Servidor de SSR |

**La aplicación aprovecha las características modernas de Angular, incluyendo componentes independientes, gestión estatal basada en señales y capacidades de renderizado del lado del servidor.**

**Fuentes:**

[**paquete.json13 a 35**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L13-L35)[**paquete.json36 a 50**](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L36-L50)

# Empezar

Este documento proporciona instrucciones iniciales de configuración, visión general de la dependencia y configuración básica para el sistema de gestión de la asistencia al profesorado. Abarca los pasos esenciales para poner en marcha el entorno de desarrollo y comprender la estructura básica del proyecto.

## Requisitos de prerrequisitos y requisitos del sistema

El sistema de gestión de asistencia al profesorado se basa en Angular 20 con modernas tecnologías web. Para el desarrollo se necesitan los siguientes componentes:

|  |  |  |
| --- | --- | --- |
| Componente | Versión | Objeto |
| Node.js | 20.17.19 | Entorno por períodos de ejecución |
| Bun | La última vez | Gestor de paquetes |
| CLI angular | 20.0.1 | Herramientas de desarrollo |
| TipoScript | 5.8.2 | JavaScript de tipos seguros |

**Panorama general de configuración del entorno para el desarrollo**

A screenshot of a computer

AI-generated content may be incorrect.

Fuentes: [paquete.json1 a 51](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L1-L51) [angular.json1-95](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/angular.json#L1-L95)

## Proceso de instalación

La aplicación utiliza Bun como gestor de paquetes para una instalación de dependencia más rápida y mejor rendimiento a lo largo de npm. El proceso de instalación consiste en clonar el repositorio y la instalación de dependencias.

**Pasos de instalación de núcleo:**

1. **Instalar Bun** (si no está ya instalado)
2. **Clonar el repositorio**
3. **Instalaciones de instalación** usando bun install
4. **Verificar la instalación** con bun run ng version

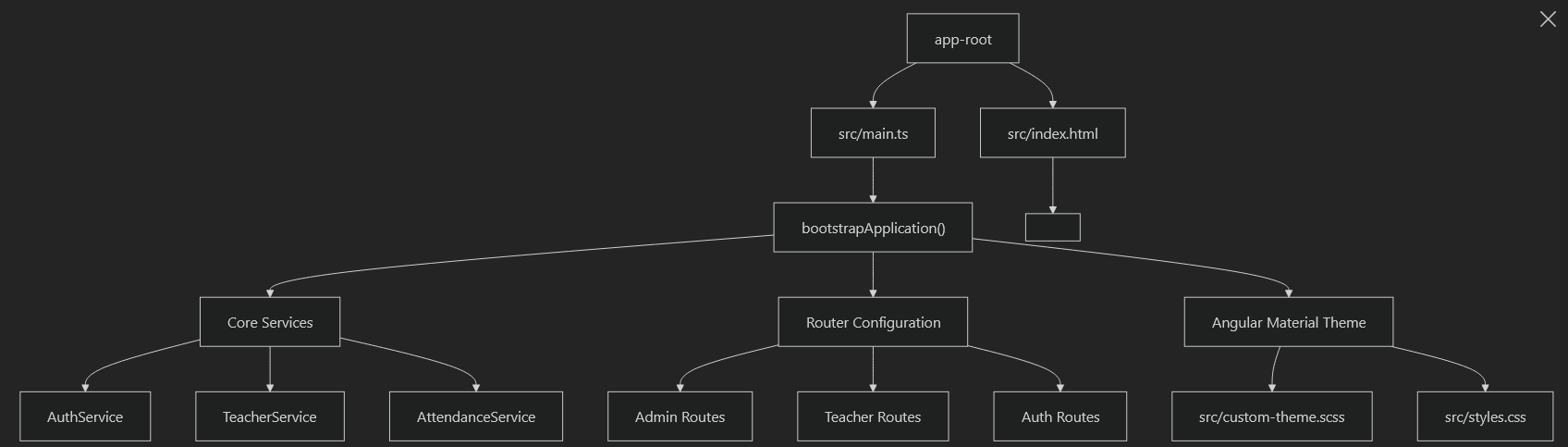
El package.jsondefine la estructura del proyecto y los scripts disponibles:

|  |  |  |
| --- | --- | --- |
| Script | Command | Objeto |
| start | ng serve | Development server |
| build | ng build | Production build |
| watch | ng build --watch --configuration development | Watch mode build |
| test | ng test | Unit tests |
| serve:ssr:proyecto-asistencia-docentes-ng | node dist/proyecto-asistencia-docentes-ng/server/server.mjs | Servidor de SSR |

Fuentes: [paquete.json4-10](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L4-L10) [Bunlock.1-46](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/bun.lock#L1-L46)

## Descripción de la arquitectura de aplicaciones

La aplicación sigue la arquitectura modular de Angular con una organización basada en características. La estructura básica incluye autenticación, portal de administración, portal de profesores y componentes compartidos.

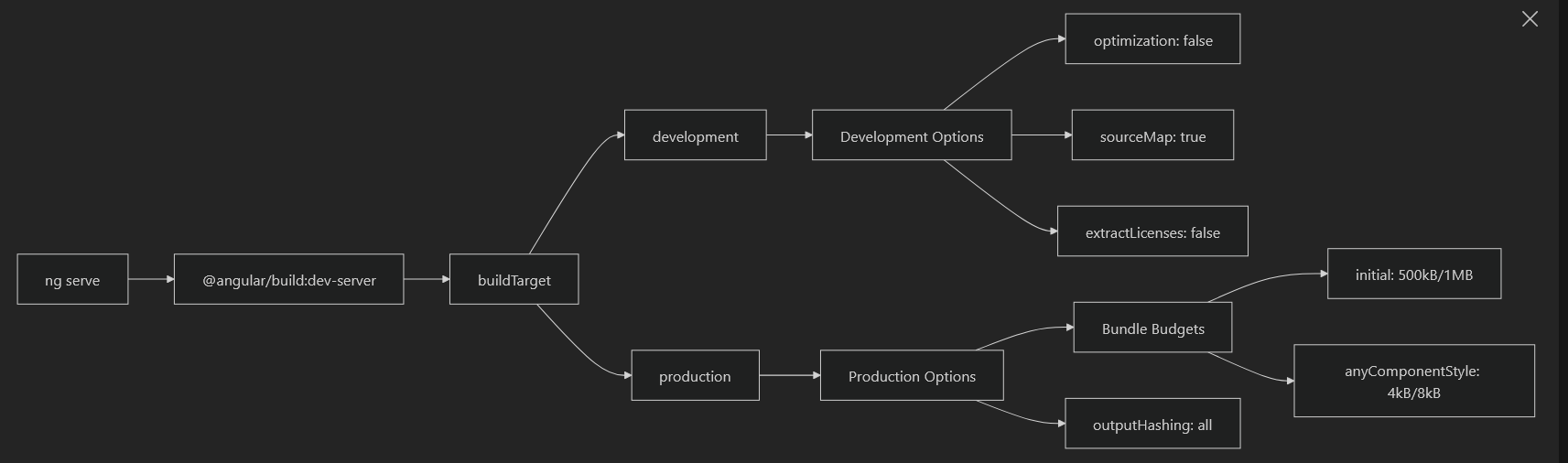


Sources: [src/index.html12-14](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/index.html#L12-L14) [angular.json16-32](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/angular.json#L16-L32) [src/styles.css1-6](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/styles.css#L1-L6)

## Configuración de servidor de desarrollo

El servidor de desarrollo Angular se configura a través de angular.json con objetivos específicos de construcción para los entornos de desarrollo y producción.

**Estructura de configuración construida**



Fuentes: [angular.json58-68](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/angular.json#L58-L68) [angular.json34 - 56](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/angular.json#L34-L56)

## Configuración del sistema de estilo

La aplicación utiliza un enfoque de estilo híbrido que combina componentes de material angular con Tailwind CSS para el estilo de utilidad-primera. La configuración de estilo se define en varios archivos.

**Arquitectura de Estilo**

|  |  |  |
| --- | --- | --- |
| Archivo | Objeto | Contenido |
| src/styles.css | Estilos globales e importación de Tailwind | Estilos de base, importación Tailwind CSS |
| src/custom-theme.scss | Tema de material angular | Configuración del tema de diseño de materiales |
| angular.json | Configuración de construcción | Orden de inclusión de archivos de estilo |

Los estilos globales son mínimos y se centran en la configuración esencial:

@import "tailwindcss";

html, body { height: 100%; }

body { margin: 0; font-family: Roboto, "Helvetica Neue", sans-serif; }

Fuentes: [src/styles.css1 a 6](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/styles.css#L1-L6) [angular.json24-27](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/angular.json#L24-L27)

## Server-Side Rendering (SSR) Configuración

La aplicación incluye capacidades de SSR para mejorar el rendimiento y SEO. La configuración de la SSR está integrada en el proceso de construcción angular.

**Proceso de construcción de SSR**

A diagram of a server

AI-generated content may be incorrect.

Fuentes: [angular.json28 a 32](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/angular.json#L28-L32) [paquete.json10](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L10-L10)

## Comandos de inicio rápido

Una vez que se configure el entorno, utilice estos comandos para iniciar el desarrollo:

|  |  |  |
| --- | --- | --- |
| Comando | Objeto | Salida esperada |
| bun install | Instalaciones de instalación | Terminación de instalación de paquetes |
| bun run start | Arregón de desarrollo de inicio | Servidor en marcha http://localhost:4200 |
| bun run build | Construcción para la producción | Paquete listo para la producción en dist/ |
| bun run test | Pruebas de unidad de ejecución | Resultados de la ejecución de pruebas |

El servidor de desarrollo proporciona capacidades de recarga en caliente y sirve a la aplicación con la caracterización completa de material angular y el estilo CSS de Tailwind aplicado.

Fuentes:

[paquete.json4 a 10](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L4-L10) [angular.json58-68](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/angular.json#L58-L68)

# Configuración del proyecto

Este documento ofrece instrucciones exhaustivas para la creación del entorno de desarrollo del sistema de gestión de la asistencia al profesorado. Abarca los procedimientos de instalación, configuración y configuración inicial necesarios para ejecutar la aplicación localmente.

## Prerrequisitos

Antes de configurar el proyecto, asegúrese de tener instalados los siguientes:

|  |  |  |
| --- | --- | --- |
| Herramienta | Versión | Objeto |
| Node.js | 18.x o posterior | Entorno por períodos de ejecución |
| npm/yarn/bun | La última vez | Gestor de paquetes |
| CLI angular | 20.0.1 | Herramientas de desarrollo |

## Configuración del entorno para el desarrollo

### Instalación inicial

El proyecto utiliza Angular 20.0.0 con Material Angular y Tailwind CSS. Instalaciones de instalación usando el administrador de paquetes preferido:

npm install o yarn install o bun install

### Configuración de servidor de desarrollo

La aplicación incluye múltiples configuraciones de desarrollo definidas en la configuración Angular CLI:

**Servicio de Desarrollo Flujo de Trabajo**

A black screen with white text

AI-generated content may be incorrect.

Fuentes: [paquete.json6](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L6-L6) [angular.json58-69](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/angular.json#L58-L69)

### Configuración de construcción

El proyecto apoya objetivos de compilación múltiple con diferentes niveles de optimización:

**Configuración de destino**

A diagram of a computer

AI-generated content may be incorrect.

Fuentes: [angular.json34 - 56](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/angular.json#L34-L56) [paquete.json7 a 8](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L7-L8)

## Corriendo la aplicación

### Modo de desarrollo

Inicie el servidor de desarrollo con recarga en caliente:

npm start

Este comando ejecuta ng serve con la configuración de desarrollo, sirviendo la aplicación en http://localhost:4200.

### Construcción de la producción

Generar una construcción de producción:

npm run build

### Server-Side Rendering (SSR)

La aplicación incluye soporte SSR usando Express.js:

npm run serve:ssr:proyecto-asistencia-docentes-ng

Esto ejecuta el servidor SSR pre-construido de dist/proyecto-asistencia-docentes-ng/server/server.mjs.

Fuentes: [paquete.json4 a 10](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L4-L10) [angular.json28 a 32](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/angular.json#L28-L32)

## Estructura de archivos y configuración de activos

### Gasoducto de Activos

El sistema de construcción Angular procesa activos y estilos estáticos a través de un oleoducto definido:

**Pipeitido de procesamiento de activos**

A screenshot of a computer

AI-generated content may be incorrect.

Fuentes: [angular.json18 a 32](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/angular.json#L18-L32) [src/styles.css1 a 6](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/styles.css#L1-L6)

### Configuración de estilo

El proyecto utiliza un enfoque de estilo híbrido que combina Tailwind CSS y Material Angular:

**CSS de Tailwind**: Importado en

* [src/styles.css2](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/styles.css#L2-L2) para el estilo de la utilidad-primera
* **Material angular** : Tema personalizado definido en src/custom-theme.scss
* **Estilos globales**: Roboto fuente familia y estilos básicos de reinicio en [src/styles.css4 a 5](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/styles.css#L4-L5)

Fuentes: [src/styles.css1 a 6](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/styles.css#L1-L6) [angular.json24-27](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/angular.json#L24-L27)

## Configuración de pruebas

### Pruebas de unidad

El proyecto utiliza Karma con Jasmine para pruebas de unidad:

npm test

### Configuración de prueba

La configuración de pruebas incluye:

* **Test Runner:** Karma 6.4.0
* **Marco de pruebas:** Jasmine 5.7.0
* **Navegadores** : Chrome (via karma-chrome-launcher)
* **Cobertura** : karma-coverage para informes de cobertura de código

**Flujo de ejecución de pruebas**



Fuentes: [paquete.json36 a 49](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L36-L49) [angular.json73 - 87](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/angular.json#L73-L87)

## Flujo de trabajo para el desarrollo

### Mira Modo

Para un desarrollo continuo con reconstrucciones automáticas:

npm run watch

Esto ejecuta ng build --watch --configuration development, proporcionando retroalimentación inmediata sobre los cambios de código.

### Estructura del proyecto

La aplicación sigue la estructura estándar del proyecto de Angular:

|  |  |
| --- | --- |
| Directory/File | Purpose |
| src/main.ts | Browser application entry point |
| src/main.server.ts | SSR application entry point |
| src/server.ts | Express server configuration |
| src/styles.css | Global styles and Tailwind imports |
| src/custom-theme.scss | Tema de material angular |
| public/ | Activos estáticos |
| dist/ | Construcción de directorio de salida |

Fuentes: [angular.json10 a 32](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/angular.json#L10-L32) [src/index.html1 a 15](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/index.html#L1-L15)

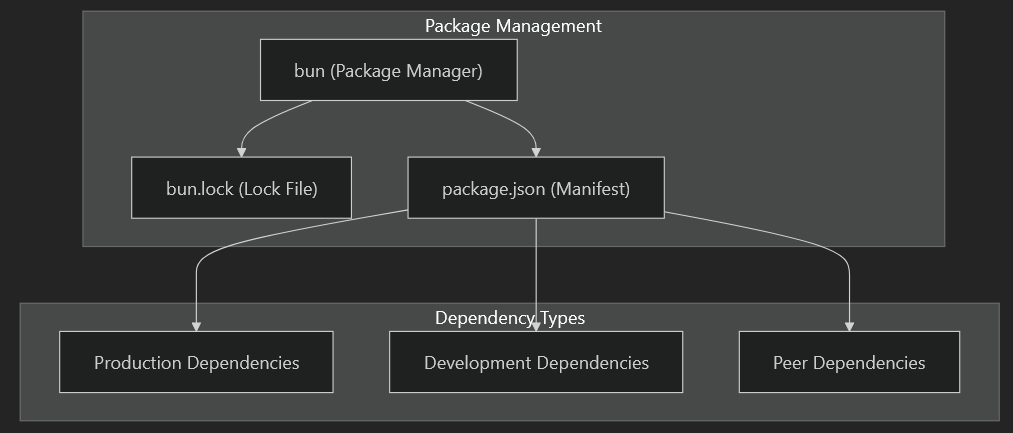
# Dependencias

Este documento ofrece una visión global de la estrategia de gestión de la dependencia y la configuración de paquetes para el sistema de gestión de la asistencia al profesorado. Abarca el enfoque de gestión de paquetes, las dependencias del marco básico, las bibliotecas de IU, el desarrollo y la configuración de la construcción.

## Estrategia de gestión de paquetes

El proyecto utiliza **a Bun** como gestor de paquetes primarios, como lo demuestra la presencia de bun.locken vez de lo tradicional package-lock.jsono o yarn.lockArchivos. Esto proporciona una instalación y resolución más rápidas en comparación con npm.

### Configuración de Administrador de paquetes



Fuentes: [Bunlock.1-50](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/bun.lock#L1-L50) [paquete.json1 a 51](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L1-L51)

## Dependencias del Marco Básico

La aplicación se basa en **Angular 20**, representando la última versión del marco con características modernas y un mejor rendimiento.

### Resumen del ecosistema angular

A screenshot of a computer

AI-generated content may be incorrect.

Fuentes: [paquete.json14 a 23](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L14-L23) [Bunlock.59 a 81](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/bun.lock#L59-L81)

### Componentes clave del marco

|  |  |  |
| --- | --- | --- |
| Package | Versión | Objeto |
| @angular/core | 20.0.0 | Core framework functionality, dependency injection, component system |
| @angular/common | 20.0.0 | Common directives, pipes, and services |
| @angular/forms | 20.0.0 | Reactive and template-driven form handling |
| @angular/router | 20.0.0 | Enrutamiento y navegación del lado del cliente |
| @angular/cdk | 20,0.2 | Kit de desarrollo de componentes para patrones avanzados de IU |
| @angular/material | 20,0.2 | Biblioteca de componentes de diseño de materiales |

Fuentes: [paquete.json14 a 19](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L14-L19)

## Dependencias de IU y Estilo

La aplicación emplea un enfoque multibibliotecario para el diseño de la UI, combinando Materiales angulares, Material-UI (MUI) y Tailwind CSS para capacidades de estilo integrales.

### Arquitectura de la Biblioteca de IU

A screenshot of a computer

AI-generated content may be incorrect.

Fuentes: [paquete.json19](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L19-L19) [paquete.json24 a 28](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L24-L28) [paquete.json30](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L30-L30) [paquete.json33](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L33-L33)

### Desintegración de la estrategia de estilo

|  |  |  |
| --- | --- | --- |
| Categoría de Biblioteca | Paquetes | Caso de uso |
| **Utilidad-Primer CSS** | tailwindcss, @tailwindcss/postcss | Prototipado rápido, diseño sensible, clases de utilidad |
| **Biblioteca de componentes** | @angular/material, @angular/cdk | Componentes angulares preconstruidos siguiendo el diseño de materiales |
| **Reaccionando Integración** | @mui/material, @emotion/react, @emotion/styled | Componentes avanzados de diseño de materiales con CSS-in-JS |
| **Sistemas de iconos** | lucide-angular, @mui/icons-material | Cobertura integral de iconos para elementos de IU |

Fuentes: [paquete.json19](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L19-L19) [paquete.json24 a 28](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L24-L28) [paquete.json30](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L30-L30) [paquete.json33](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L33-L33)

## Dependencias de Desarrollo y Construcción

El entorno de desarrollo aprovecha el CLIngular con herramientas de construcción modernas y marcos de pruebas integrales.

### Plan de herramientas de desarrollo

A screenshot of a computer

AI-generated content may be incorrect.

Fuentes: [paquete.json37 a 39](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L37-L39) [paquete.json40-42](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L40-L42) [paquete.json43-49](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L43-L49)

### Construir Scripts Configuración

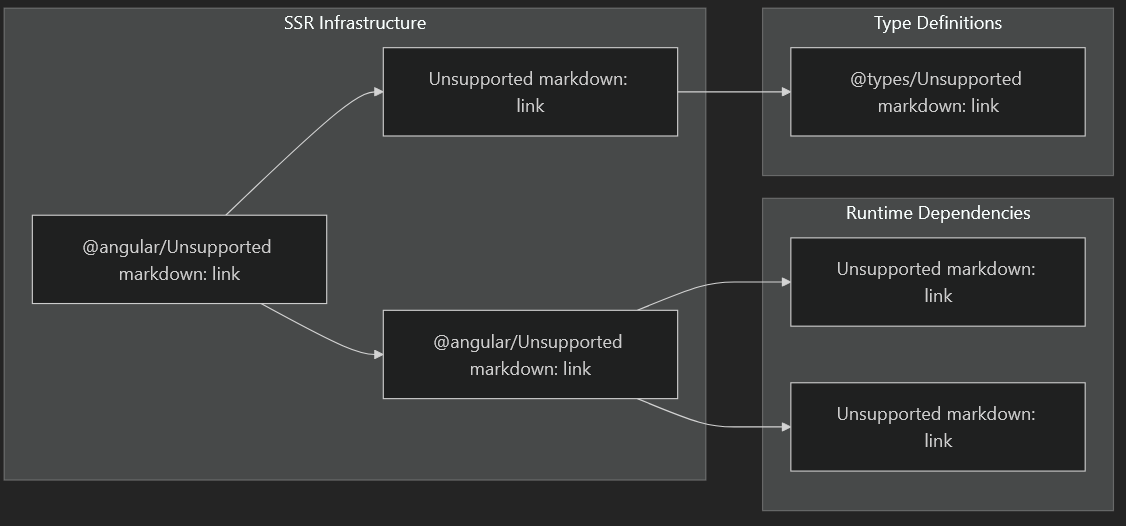
|  |  |  |
| --- | --- | --- |
| Guión | Comando | Objeto |
| ng | ng | Interfaz de comando CLI Angular |
| start | ng serve | Servidor de desarrollo con recarga en caliente |
| build | ng build | Recopilación de construcción de la producción |
| watch | ng build --watch --configuration development | Se construyen el desarrollo continuo |
| test | ng test | Ejecución de pruebas de unidad con Karma/Jasmine |
| serve:ssr:proyecto-asistencia-docentes-ng | node dist/proyecto-asistencia-docentes-ng/server/server.mjs | Ejecución de la ejecución de la entrega del lado del servidor |

Fuentes: [paquete.json4 a 10](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L4-L10)

## Configuración de representaciones de servidor-ser

La aplicación incluye un amplio soporte de SSR para mejorar el rendimiento y las capacidades de SEO.

### Cadena de dependencia de la SSR



Fuentes: [paquete.json23](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L23-L23) [paquete.json21](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L21-L21) [paquete.json29](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L29-L29) [paquete.json32](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L32-L32) [paquete.json34](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L34-L34) [paquete.json40](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L40-L40)

## Estrategia de gestión de versiones

El proyecto mantiene una alineación coherente de la versión en todo el ecosistema Angular, al tiempo que permite flexibilidad para bibliotecas de terceros.

### Restricciones de la versión

A screenshot of a computer

AI-generated content may be incorrect.

|  |  |  |  |
| --- | --- | --- | --- |
| Tipo de restricción | Símbolo | Ejemplo | Pattern de uso |
| Compatible | ^ | ^20.0.0 | Paquetes de ecosistemas angulares |
| Aproximadamente | ~ | ~5.8.2 | Equipos de script y herramientas de prueba |
| Exacto | ninguno | 7.8.0 | RxJS para la estabilidad |

Fuentes: [paquete.json13-35](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L13-L35) [paquete.json36-50](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L36-L50)

## Seguridad y mantenimiento de la dependencia

El archivo de bloqueo garantiza construcciones deterministas y seguridad a través de la versión exacta sujetando a todas las dependencias transitivas.

### Bloquear la estructura de archivos Resumen



Fuentes: [Bunlock.1-1000](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/bun.lock#L1-L1000)

El bun.lockEl archivo contiene más de 1000 entradas de paquete con hachís de integridad SHA-512, asegurando que todas las dependencias mantengan versiones consistentes en diferentes entornos de desarrollo y despliegues.

Fuentes: [Bunlock.1-50](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/bun.lock#L1-L50) [paquete.json1 a 51](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L1-L51)

# Arquitectura

Este documento describe la arquitectura de alto nivel del sistema de gestión de asistencia al profesorado, centrándose en los patrones estructurales, el diseño de capas de servicio y la organización de componentes. La arquitectura sigue los patrones recomendados de Angular con un enfoque orientado al servicio y una gestión estatal reactiva usando señales angulares.

## Patrón de Arquitectura del Sistema

La aplicación sigue una arquitectura en capas con una clara separación de preocupaciones entre la presentación, la lógica empresarial y las capas de persistencia de datos.

## Estructura general del sistema

A screenshot of a computer

AI-generated content may be incorrect.

La arquitectura implementa un flujo de datos unidireccional donde:

* Componentes inyectan servicios para acceder a la lógica empresarial
* Servicios gestionan el estado de aplicación usando Señales Angular
* La persistencia de los datos se maneja a través de la Storage local
* Los componentes compartidos proporcionan patrones de interfaz de usuario consistentes

**Fuentes:** [src/app/core/services/attendance.service.ts1-235](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/attendance.service.ts#L1-L235) [src/app/core/services/teacher.serv.ts1-179](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/teacher.service.ts#L1-L179) [src/app/core/services/auth.service.ts1-96](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/auth.service.ts#L1-L96)

## Arquitectura de capa de servicio

La lógica empresarial central está centralizada en tres servicios primarios que implementan el patrón de la segregación de responsabilidad de comandos (CQRS) a través de señales angulares.

### Dependencias e Interacciones de Servicios

A screenshot of a computer

AI-generated content may be incorrect.

Cada servicio sigue el mismo patrón arquitectónico:

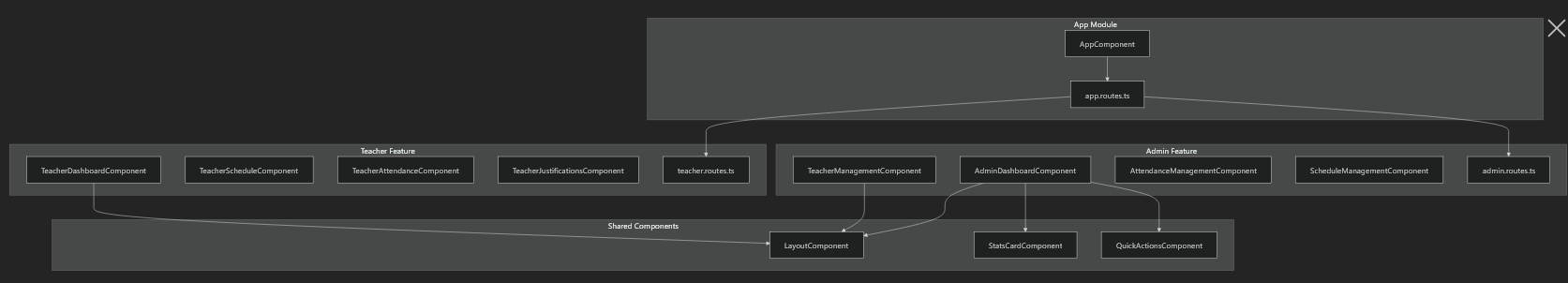
* **Estado basado en la señal:** Señales privadas con accesores públicos de lectura
* **Data Persistencia** : Sincronización automática local de almacenamiento
* **Lógica de Negocios:** Métodos que modifican los cambios de estado y persisten
* **Métodos de consulta** : Valores computados y acceso a datos filtrados

**Fuentes:** [src/app/core/services/auth.service.ts11 a 18](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/auth.service.ts#L11-L18) [src/app/core/services/teacher.serv.ts10-11](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/teacher.service.ts#L10-L11) [src/app/core/services/attendance.service.ts8 a 12](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/attendance.service.ts#L8-L12)

## Arquitectura de componentes

La aplicación se organiza en módulos de características con un sistema de componentes compartidos que proporciona patrones de interfaz de usuario consistentes en los portales de administración y profesores.

### Organización del módulo de características



**Fuentes:** [src/app/features/admin/admin.routes.ts1 a 29](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/admin.routes.ts#L1-L29) [src/app/features/teacher/teacher.routes.ts1 a 26](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/teacher.routes.ts#L1-L26) [src/app/features/admin/teachers/teacher-management.component.ts8-11](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-management.component.ts#L8-L11)

## Arquitectura de flujo de datos

La aplicación implementa un flujo de datos reactivo usando señales angulares para la gestión del estado con persistencia automática a localStorage.

### Gestión del Estado y flujo de datos

A screenshot of a computer

AI-generated content may be incorrect.

El flujo de datos sigue estos patrones:

* **Acciones de usuario** activan métodos de servicio
* **Métodos de servicio** actualizan señales internas
* **Actualizaciones de señales** disparan automáticamente los re-re-re-renders de IU
* **Métodos de persistencia** salvan estado a la casa de pisoteo local
* **Componentes de IU** responden de forma reactiva a los cambios de señal

**Fuentes:** [src/app/core/services/auth.service.ts42-70](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/auth.service.ts#L42-L70) [src/app/core/services/attendance.service.ts68-110](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/attendance.service.ts#L68-L110) [src/app/core/services/teacher.serv.ts106-116](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/teacher.service.ts#L106-L116)

## Patrones de comunicación de componentes

La aplicación utiliza varios patrones de comunicación para mantener el acoplamiento suelto entre componentes, garantizando al mismo tiempo la consistencia de los datos.

## Consumo de inyección y señalización de servicio

A screenshot of a computer

AI-generated content may be incorrect.

Los patrones arquitectónicos clave incluyen:

* **Inyección de constructor:** Los servicios se inyectan en los componentes utilizando Angular DI
* **Reactividad basada en la señal** : Los componentes consumen solo señales para actualizaciones reactivas
* **Comunicación de diálogos:** Los diálogos con los mecanismos reciben datos a través de MAT.DIALOG-DATA
* **Integración de forma:** Los formularios reactivos se integran con los métodos de servicio para actualizaciones estatales

**Fuentes:** [src/app/features/admin/teachers/teacher-management.component.ts49-52](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-management.component.ts#L49-L52) [src/app/features/maestro/justificaciones/justificación-form-dialog.s38-40](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/justifications/justification-form-dialog.component.ts#L38-L40)

## Arquitectura de componentes compartidos

El sistema de componentes compartidos proporciona patrones de interfaz de usuario consistentes y gestión de diseño en toda la aplicación.

### Estructura de componentes compartidos

A screenshot of a computer

AI-generated content may be incorrect.

Los componentes compartidos siguen estos patrones:

* **Componente de diseño** : Proporciona navegación y estructura consistentes
* **Componentes Data-Driven** : Aceptar objetos de configuración para una reutilización flexible
* **Componentes de diálogo** : Implementar patrones de forma estándar con validación
* **Integración de diseño de materiales:** Uso consistente de componentes de materiales angulares

**Fuentes:** [src/app/features/admin/dashboard/stats-card.component.html1-33](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/stats-card.component.html#L1-L33) [src/app/features/admin/dashboard/quick-actions.component.html1-52](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/quick-actions.component.html#L1-L52) [src/app/features/admin/teachers/teacher-management.component.ts33 a 38](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-management.component.ts#L33-L38)

## Arquitectura de enrutamiento

El sistema de enrutamiento implementa la navegación basada en el rol con carga perezosa para un rendimiento óptimo.

### Estructura de configuración de ruta

A screenshot of a computer

AI-generated content may be incorrect.

La arquitectura de enrutamiento implementa:

* **Carga perezosa:** Los módulos de características se cargan bajo demanda
* **Guardias de ruta** : Autenticación y control de acceso basado en roles
* **Redirects por defecto:** Redirección automática a los paneles apropiados
* **Organización modular** : Configuración de ruta separada para cada característica

**Fuentes:** [src/app/features/admin/admin.routes.ts7 a 28](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/admin.routes.ts#L7-L28) [src/app/features/teacher/teacher.routes.ts3 a 25](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/teacher.routes.ts#L3-L25)

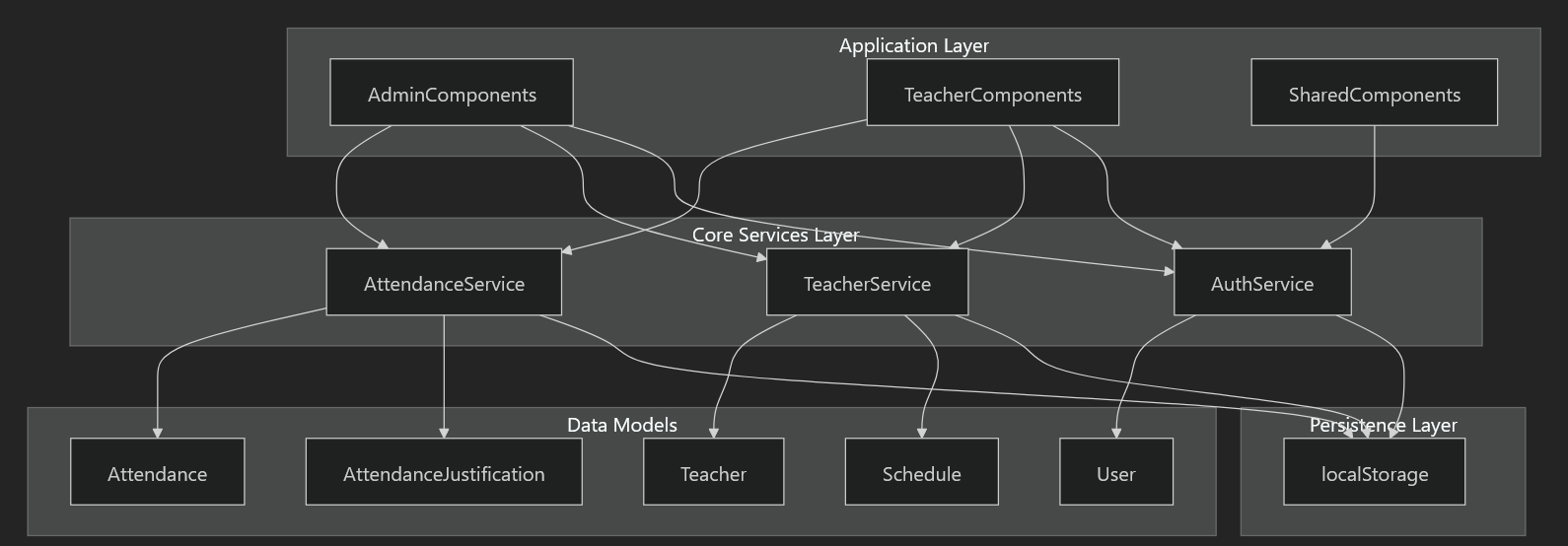
# Servicios básicos

Este documento cubre los tres servicios básicos que alimentan el sistema de gestión de la asistencia al profesorado: AttendanceService, TeacherService, y AuthService. Estos servicios forman la capa lógica de negocio y manejan todas las operaciones de datos, la gestión del estado y la funcionalidad básica.

## Resumen de la arquitectura de servicio

Los servicios básicos siguen una arquitectura reactiva y basada en señales con persistencia localStorage. Cada servicio gestiona su propio dominio de datos y ofrece operaciones sincrónicas y asincrónicas.

**Arquitectura del sistema de servicios básicos**



Fuentes: [src/app/core/services/attendance.service.ts1-234](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/attendance.service.ts#L1-L234) [src/app/core/services/teacher.serv.ts1-179](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/teacher.service.ts#L1-L179) [src/app/core/services/auth.service.ts1-96](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/auth.service.ts#L1-L96)

## Dependencias e Interacciones de Servicios

Los servicios tienen relaciones específicas de dependencia e interactúan a través de la inyección de dependencia y modelos de datos compartidos.

**Gráfico de Dependencia de Servicio**

A diagram of a software

AI-generated content may be incorrect.

Fuentes: [src/app/core/services/auth.service.ts6 a 12](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/auth.service.ts#L6-L12) [src/app/core/services/teacher.serv.ts15-17](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/teacher.service.ts#L15-L17)

## Servicio de asistencia

El AttendanceService gestiona los registros de asistencia, check-ins, check-outs y flujos de trabajo de justificación. Utiliza señales angulares para la gestión estatal reactiva.

### Características clave

|  |  |  |
| --- | --- | --- |
| Característica | Métodos | Objeto |
| Check-in/Check-out | checkIn(), checkOut() | Recordar presencia de maestro |
| Condición de asistencia | hasAttendanceToday(), getTodayAttendanceStatus() | Estado de asistencia a la consulta |
| Justificaciones | submitJustification(), reviewJustification() | Manipula justificaciones de ausencia |
| Recuperación de datos | getAttendancesByTeacher(), getAttendancesByDate() | Registros de asistencia a la consulta |

### Gestión estatal basada en la señal

  private attendancesSignal = signal<Attendance[]>([])

  private justificationsSignal = signal<AttendanceJustification[]>([])

  attendances = **this**.attendancesSignal.asReadonly()

  justifications = **this**.justificationsSignal.asReadonly()

### Operaciones críticas

**Corriente de proceso de facturación**

A screenshot of a diagram

AI-generated content may be incorrect.

El checkIn()el método maneja la prevención duplicada y crea registros de asistencia con marcas de tiempo:

Fuentes: [src/app/core/services/attendance.service.ts68-110](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/attendance.service.ts#L68-L110)

### Justificación Flujo de trabajo

El servicio gestiona un ciclo de vida de justificación completa, desde sumisión hasta revisión administrativa:

|  |  |  |
| --- | --- | --- |
| Situación | Descripción | Transiciones |
| pending | Presentación inicial | - approvedo o rejected |
| approved | Admin aceptado | Estado final |
| rejected | Admin negado | Estado final |

Fuentes: [src/app/core/services/attendance.service.ts197 a 224](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/attendance.service.ts#L197-L224)

## MaestroServicio

El TeacherService gestiona los registros de maestros, horarios y operaciones de la CRUD. Maneja tanto los datos del profesorado como sus horarios asociados.

### Operaciones de datos básicos

|  |  |  |
| --- | --- | --- |
| Operación | Método | Objeto |
| Creación | addTeacher() | Añadir nuevo disco de maestro |
| Lea | getTeacherById(), teachers() | Recuperar datos de los profesores |
| Actualización | updateTeacher() | Modificar registro de maestros |
| Supríma | deleteTeacher() | Eliminar registro de maestros |

### Gestión de los cuadros

addScheduleToTeacher(teacherId: string, schedule: Omit<Schedule, "id" | "teacherId">)  
removeScheduleFromTeacher(teacherId: string, scheduleId: string)

**Relación maestro-esculado**

A screenshot of a computer

AI-generated content may be incorrect.

### Inicialización de datos de simulacros

El servicio se inicializa con datos predefinidos del profesorado cuando no existen datos almacenados:

Fuentes: [src/app/core/services/teacher.serv.ts39 a 96](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/teacher.service.ts#L39-L96)

## AuthService

El AuthService maneja autenticación, sesiones de usuario y enrutamiento basado en roles. Se integra con ambos TeacherService y Router para flujos de autenticación completas.

### Estado de autenticación

El servicio utiliza señales calculadas para el estado de autenticación reactivo:

currentUser = **this**.currentUserSignal.asReadonly()

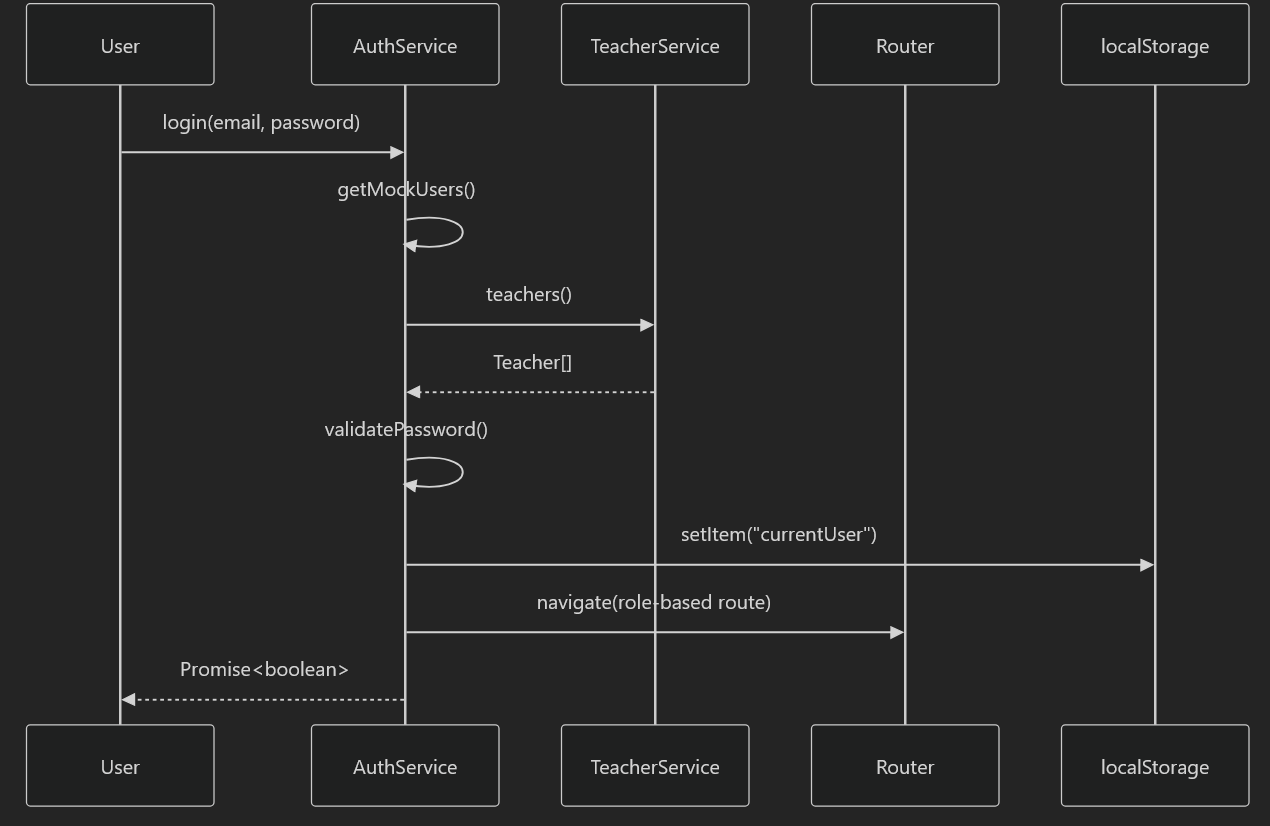
  isAuthenticated = computed(() => !!**this**.currentUserSignal())

  isAdmin = computed(() => **this**.currentUserSignal()?.role === "admin")

  isTeacher = computed(() => **this**.currentUserSignal()?.role === "teacher")

### Login Flow

**Proceso de autenticación**



### Navegación basada en el papel

El servicio redirige automáticamente a los usuarios en función de su papel después de una autenticación exitosa:

|  |  |  |
| --- | --- | --- |
| Papel | Ruta | Objeto |
| admin | /admin/dashboard | Interfaz administrativa |
| teacher | /teacher/dashboard | Interfaz de profesor |

Fuentes: [src/app/core/services/auth.service.ts42-70](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/auth.service.ts#L42-L70)

## Patrones de persistencia de datos

Los tres servicios siguen patrones de persistencia local consistentes de almacenamiento con controles conscientes de la plataforma para la compatibilidad de la SSR.

          if (isPlatformBrowser(**this**.platformId)) {

            localStorage.setItem("currentUser", JSON.stringify(user))

          }

## Cargada de datos y serialización

Cada servicio maneja la serialización/desierización de objetos de fecha para localStorage:

        const attendances = JSON.parse(storedAttendances).map((*a*: any) => ({

          ...*a*,

          date: new Date(*a*.date),

          checkIn: *a*.checkIn ? new Date(*a*.checkIn) : undefined,

          checkOut: *a*.checkOut ? new Date(*a*.checkOut) : undefined,

        }))

Fuentes: [src/app/core/services/attendance.service.ts27-33](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/attendance.service.ts#L27-L33) [src/app/core/services/teacher.serv.ts25 - 36](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/teacher.service.ts#L25-L36) [src/app/core/services/auth.service.ts27 - 40](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/auth.service.ts#L27-L40)

## Usage de servicio en componentes

Los componentes inyectan y utilizan estos servicios a través del sistema de inyección de dependencia de Angular. Así es como se consumen típicamente:

## Ejemplo de integración de componentes

  public teacherService: TeacherService = inject(TeacherService);

  private dialog: MatDialog = inject(MatDialog);

  deleteTeacher(*teacher*: Teacher): void {

    if (confirm(`¿Está seguro de eliminar al docente ${*teacher*.name}?`)) {

**this**.teacherService.deleteTeacher(*teacher*.id);

    }

  }

## Consumo de datos reactivos

Los componentes acceden a los datos reactivos mediante señales de servicio:

 if (**this**.teacher) {

**this**.teacherService.updateTeacher(**this**.teacher.id, formValue);

      } else {

**this**.teacherService.addTeacher({

          ...formValue,

          role: 'teacher',

          schedules: [],

        });

      }

Fuentes: [src/app/features/admin/teachers/teacher-management.component.ts49 - 65](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-management.component.ts#L49-L65) [src/app/features/maestro/justificaciones/justificación-form-dialog.s38 - 58](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/justifications/justification-form-dialog.component.ts#L38-L58)

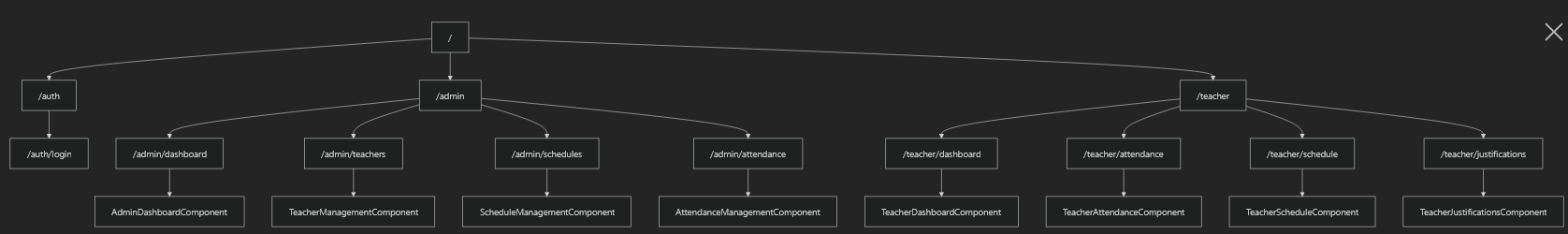
# Sistema de enrutamiento

Este documento cubre la arquitectura de encaminamiento del sistema de gestión de asistencia al profesorado, incluyendo navegación basada en roles, rutas de módulos de características y enrutamiento protegido por autenticación. El sistema de enrutamiento gestiona la navegación entre diferentes portales de usuarios (admin y profesor) y sus respectivas áreas de características.

## Arquitectura general de enrutamiento

La aplicación utiliza el router de Angular con una estructura jerárquica que separa las preocupaciones por el rol de usuario y el área de características. El sistema de enrutamiento implementa carga perezosa para la optimización del rendimiento y el control de acceso basado en roles.

### Diagrama de estructura de la ruta



Fuentes: [src/app/features/admin/admin.routes.ts1 a 29](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/admin.routes.ts#L1-L29) [src/app/features/teacher/teacher.routes.ts1 a 26](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/teacher.routes.ts#L1-L26)

## Sistema de tanteo encubierto

El sistema de enrutamiento implementa un control de acceso basado en el papel donde los usuarios se dirigen a diferentes áreas del portal en función de su papel autenticado. Después de una autenticación exitosa vía AuthService, los usuarios son enrutados al administrador o al portal del profesor.

### Autenticación de la Guardia Civil de Flujo y Rutas

A screenshot of a computer

AI-generated content may be incorrect.

Fuentes: [src/app/features/auth/login/login.component.ts86 a 111](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/auth/login/login.component.ts#L86-L111)

## Rotas del módulo de características

La aplicación organiza rutas en módulos de características con configuraciones de ruta específicas para cada rol de usuario.

### Rutas del Portal de Administración

El portal de administración ofrece capacidades de gestión integrales a través de rutas específicas:

|  |  |  |
| --- | --- | --- |
| Route Path | Component | Objeto |
| /admin/dashboard | AdminDashboardComponent | Administrative overview and statistics |
| /admin/teachers | TeacherManagementComponent | Teacher CRUD operations |
| /admin/schedules | ScheduleManagementComponent | Schedule management interface |
| /admin/attendance | AttendanceManagementComponent | Attendance oversight and justification review |

La configuración de las rutas de administración utiliza importaciones directas de componentes para carga inmediata:

  {

    path: "dashboard",

    component: AdminDashboardComponent

  },

Fuentes: [src/app/features/admin/admin.routes.ts7 a 28](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/admin.routes.ts#L7-L28)

### Rutas del Portal de Maestros

El portal del profesor se centra en la gestión de horarios personales y asistencia:

|  |  |  |
| --- | --- | --- |
| Ruta de ruta | Componente | Objeto |
| /teacher/dashboard | TeacherDashboardComponent | Tablero personal y resumen diario |
| /teacher/attendance | TeacherAttendanceComponent | Historia y situación de la asistencia |
| /teacher/schedule | TeacherScheduleComponent | Inscripción de horarios y asistencia |
| /teacher/justifications | TeacherJustificationsComponent | Justificación sumisión y seguimiento |

Las rutas del profesor implementan carga perezosa para la optimización del rendimiento:

  {

    path: "dashboard",

    loadComponent: () => import("./dashboard/teacher-dashboard.component").then((*m*) => *m*.TeacherDashboardComponent),

  },

Fuentes: [src/app/features/teacher/teacher.routes.ts3 a 25](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/teacher.routes.ts#L3-L25)

## Ejecución de carga perezosa

El sistema de enrutamiento implementa diferentes estrategias de carga basadas en el papel del usuario y los patrones de uso.

### Cargando la comparación de la estrategia



Fuentes: [src/app/features/admin/admin.routes.ts1 a 29](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/admin.routes.ts#L1-L29) [src/app/features/teacher/teacher.routes.ts1 a 26](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/teacher.routes.ts#L1-L26)

## Manejo de la ruta predeterminada

Tanto las configuraciones de la vía de administración como las de los profesores implementan redirecciones predeterminadas a sus respectivos componentes del salpicadero:

  {

    path: "",

    redirectTo: "dashboard",

    pathMatch: "full",

  },

Esto asegura que los usuarios se dirijan automáticamente al salpicadero apropiado al acceder a las URL del portal base ( /admino o /teacher).

Fuentes: [src/app/features/admin/admin.routes.ts24 a 28](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/admin.routes.ts#L24-L28) [src/app/features/teacher/teacher.routes.ts22 a 25](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/teacher.routes.ts#L22-L25)

## Integración de la ruta con Autenticación

El sistema de enrutamiento se integra con el AuthService para manejar los cambios de estado de autenticación y las rediciones basadas en roles. El LoginComponent demuestra esta integración llamando AuthService.login() y apoyándose en el servicio de autenticación para manejar la navegación posterior en función del papel del usuario.

const success = await **this**.authService.login(

          email as string,

          password as string

        );

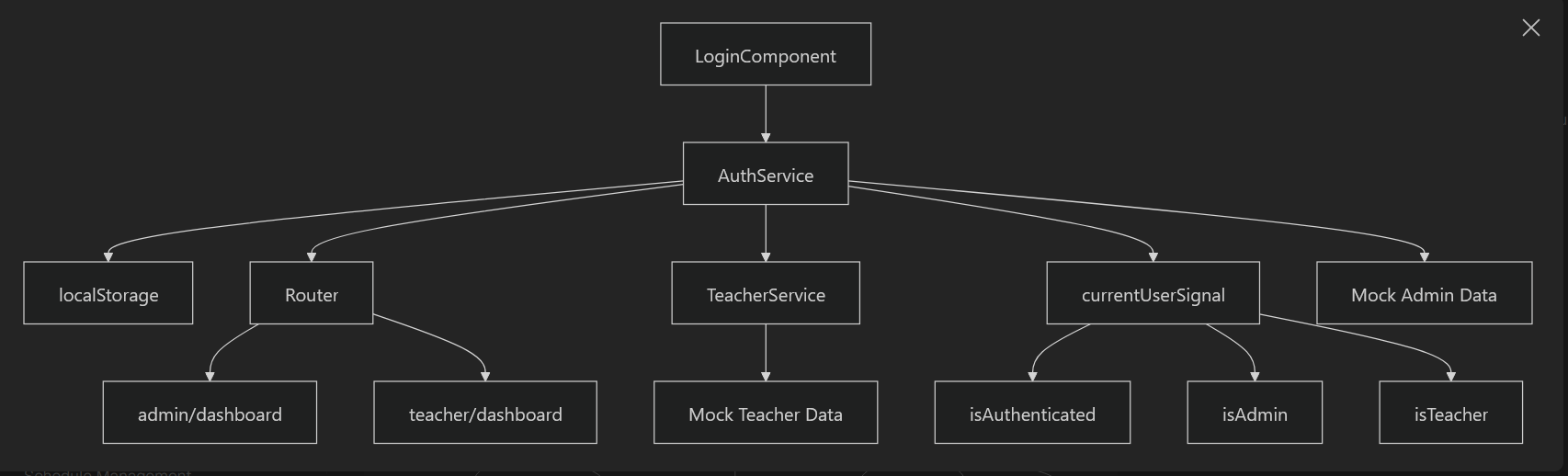
Fuentes: [src/app/features/auth/login/login.component.ts86 a 111](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/auth/login/login.component.ts#L86-L111)

# Autenticación

Este documento cubre el sistema de autenticación para la aplicación de gestión de asistencia al profesorado, incluyendo el inicio de sesión del usuario, el control de acceso basado en el rol y la gestión de sesiones. El sistema proporciona niveles de acceso separados para administradores y profesores con diferentes experiencias de panel.

## Resumen del sistema

El sistema de autenticación se construye utilizando las formas reactivas de Angular y las señales para la gestión estatal. Implementa un sistema de control de acceso basado en roles con dos tipos de usuarios: administradores y profesores. El estado de autenticación se insiste utilizando el navegador localStorage, y el sistema incluye credenciales simuladas de los usuarios para el desarrollo y la prueba.



Fuentes: [src/app/features/auth/login/login.component.ts1-112](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/auth/login/login.component.ts#L1-L112) [src/app/core/services/auth.service.ts1-96](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/auth.service.ts#L1-L96)

## Login de arquitectura de componentes

El LoginComponent proporciona la interfaz de usuario para la autenticación utilizando formas reactivas angulares. Incluye validación de correo electrónico y contraseña, estados de carga y manejo de errores.

### Estructura de componentes

|  |  |  |
| --- | --- | --- |
| Característica | Aplicación | Ubicación |
| Validación de la forma | Validators.required, Validators.email, Validators.minLength(6) | [src/app/features/auth/login/login.component.ts77-80](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/auth/login/login.component.ts#L77-L80) |
| Visibilidad de contraseña | hidePassword señal con función de conmutación | [src/app/features/auth/login/login.component.ts73 a 84](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/auth/login/login.component.ts#L73-L84) |
| Estado de carga | loading señal para operaciones de async | [src/app/features/auth/login/login.component.ts71 - 108](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/auth/login/login.component.ts#L71-L108) |
| Manejo de errores | errorMessage señal para fallos de inicio de sesión | [src/app/features/auth/login/login.component.ts72 a 106](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/auth/login/login.component.ts#L72-L106) |

### Login Flow



Fuentes: [src/app/features/auth/login/login.component.ts86 a 111](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/auth/login/login.component.ts#L86-L111) [src/app/core/services/auth.service.ts42-70](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/auth.service.ts#L42-L70)

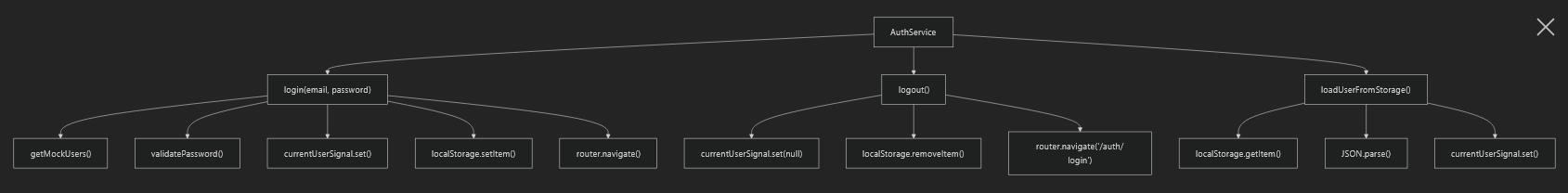
## Arquitectura de AuthService

El AuthServicegestiona el estado de autenticación utilizando señales angulares y proporciona acceso reactivo a la información del usuario a lo largo de la aplicación.

### Propiedades de servicio

|  |  |  |
| --- | --- | --- |
| Bienes | Tipo | Objeto |
| currentUserSignal | EstrictableSignal.Use Usuario | null. |
| currentUser | Firmar, Usuario | null. |
| isAuthenticated | ComputedSignal<boolean> | Estado de autenticación |
| isAdmin | ComputedSignal<boolean> | Control de rol de administrador |
| isTeacher | ComputedSignal<boolean> | Control de rol de maestro |

### Métodos básicos



Fuentes: [src/app/core/services/auth.service.ts11 a 96](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/auth.service.ts#L11-L96)

## Control de acceso basado en el papel

El sistema implementa un control de enrutamiento basado en el rol basado en los roles de usuario definidos en el modelo de usuario.

### Roles de Usuario y enrutamiento

|  |  |  |
| --- | --- | --- |
| Papel | Ruta | Nivel de acceso |
| admin | /admin/dashboard | Acceso administrativo completo |
| teacher | /teacher/dashboard | Características específicas del profesor |

### Lógica de validación de roles

El sistema de autenticación utiliza señales calculadas para el control de acceso basado en el rol:

  isAdmin = computed(() => **this**.currentUserSignal()?.role === "admin")

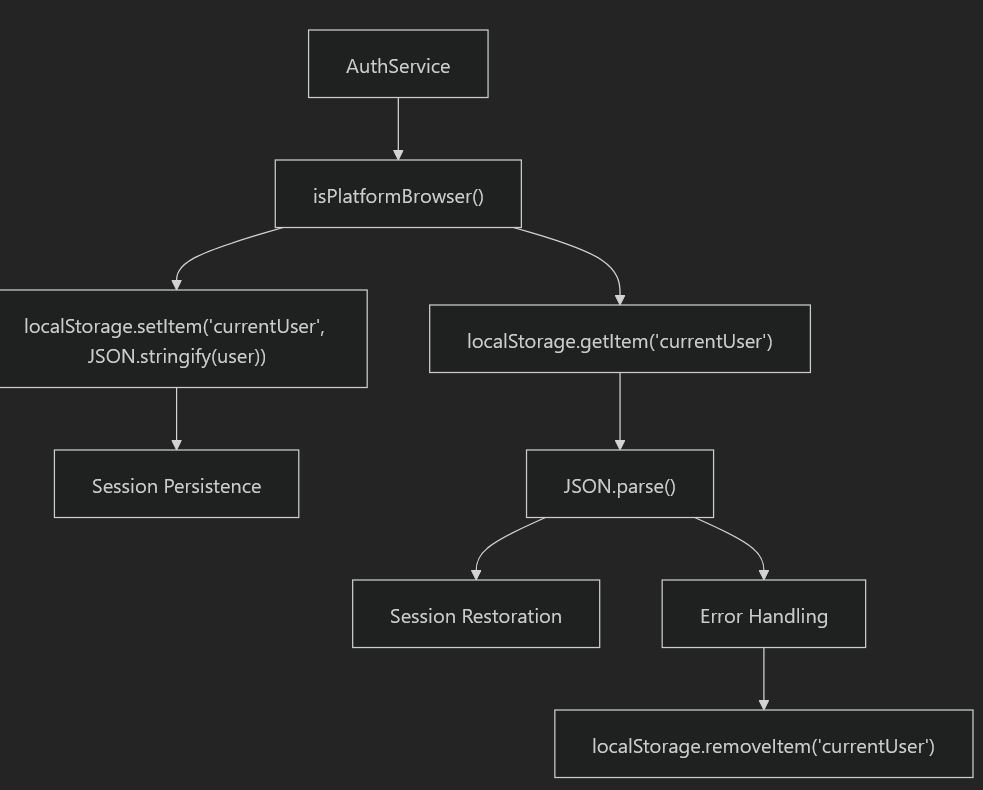
  isTeacher = computed(() => **this**.currentUserSignal()?.role === "teacher")

Fuentes: [src/app/core/services/auth.service.ts17 a 18](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/auth.service.ts#L17-L18) [src/app/core/services/auth.service.ts57-62](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/auth.service.ts#L57-L62)

## Persistencia de los datos

El estado de autenticación persiste usando el navegador localStorage para mantener las sesiones de los usuarios a través de actualizaciones de página y sesiones del navegador.

### Aplicación de la instalación



El servicio incluye la verificación de la plataforma para asegurar que localStorage solo sea accedida en entornos de navegador, apoyando escenarios de renderizado del lado del servidor.

Fuentes: [src/app/core/services/auth.service.ts27 - 40](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/auth.service.ts#L27-L40) [src/app/core/services/auth.service.ts53 - 55](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/auth.service.ts#L53-L55) [src/app/core/services/auth.service.ts74 - 76](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/auth.service.ts#L74-L76)

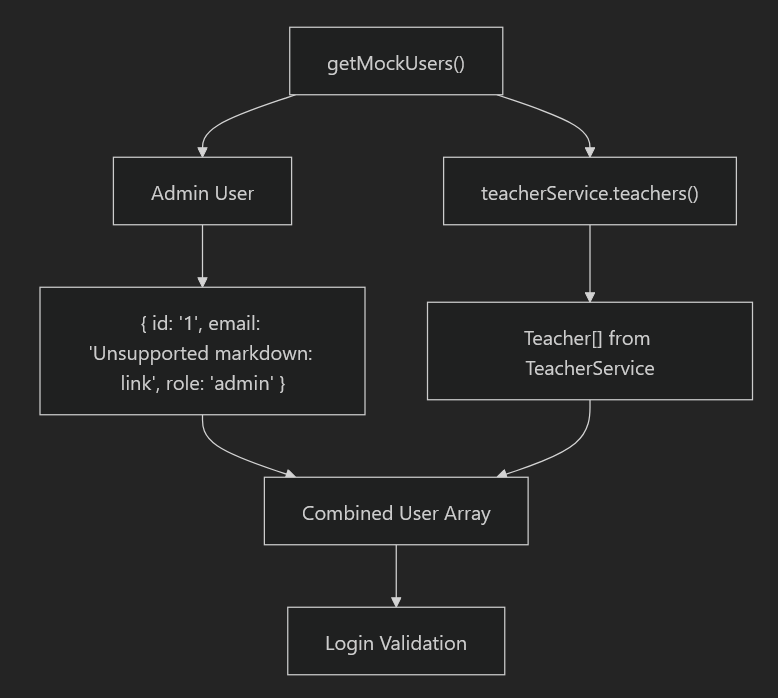
## Sistema de autenticación simulada

La aplicación incluye un sistema de autenticación simulada para fines de desarrollo y pruebas con credenciales de usuario predefinidas.

### Datos de Usuarios de Burla

|  |  |  |  |
| --- | --- | --- | --- |
| Papel | Correo electrónico | Contraseña | Nombre |
| Administración | admin@school.com | 123456 | Administrador |
| Maestro | teacher1@school.com | 123456 | María García |
| Maestro | teacher2@school.com | 123456 | Juan Pérez |

### Mock Integración de datos



El sistema de simulacro combina un usuario de administración con los datos del profesorado de la TeacherService, creando un sistema unificado de autenticación de usuario.

Fuentes: [src/app/core/services/auth.service.ts84-95](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/auth.service.ts#L84-L95) [src/app/features/auth/login/login.component.html126-149](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/auth/login/login.component.html#L126-L149) [src/app/core/services/teacher.serv.ts40 a 95](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/teacher.service.ts#L40-L95)

## Consideraciones de seguridad

La implementación actual incluye la validación básica de contraseña que requiere un mínimo de 6 caracteres. El sistema utiliza la validación del lado del cliente y la autenticación simulada, por lo que es adecuado para fines de desarrollo y demostración.

## Reglas de validación

* Email format validation using Angular's email validator
* Password minimum length of 6 characters
* Form-level validation preventing submission of invalid data

Sources: [src/app/features/auth/login/login.component.ts77-80](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/auth/login/login.component.ts#L77-L80) [src/app/core/services/auth.service.ts80-82](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/auth.service.ts#L80-L82)

# Portal de administración

El Portal de administración proporciona funcionalidad administrativa para la gestión de maestros, horarios y registros de asistencia en el sistema de gestión de asistencia al profesorado. Sirve como centro central para que los administradores realicen operaciones de CRUD sobre datos de maestros, revisen los registros de asistencia y aprueben o rechacen justificaciones de asistencia.

## Resumen del sistema

El Portal de administración se construye como una colección de componentes independientes angulares que comparten una estructura de navegación común e interactúan con los servicios básicos. El portal ofrece una experiencia basada en paneles con interfaces de gestión específicas para diferentes entidades.

### Arquitectura de Portal de Admin

A screenshot of a computer

AI-generated content may be incorrect.

Fuentes: [src/app/features/admin/dashboard/admin-dashboard.component.ts1-109](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/admin-dashboard.component.ts#L1-L109) [src/app/features/admin/teachers/teacher-management.component.ts1-66](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-management.component.ts#L1-L66) [src/app/features/admin/attendance/attendance-management.component.ts1-199](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/attendance/attendance-management.component.ts#L1-L199)

### Estructura de navegación

El Portal de administración utiliza un menú de navegación consistente en todos los componentes, definido por el MenuIteminterfaz y renderizado a través de la LayoutComponent:

A black and white screen

AI-generated content may be incorrect.

Fuentes: [src/app/features/admin/dashboard/admin-dashboard.component.ts84-89](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/admin-dashboard.component.ts#L84-L89) [src/app/features/admin/teachers/teacher-management.component.ts33 a 38](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-management.component.ts#L33-L38) [src/app/features/admin/attendance/attendance-management.component.ts50-55](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/attendance/attendance-management.component.ts#L50-L55)

## Componentes básicos

### AdminDashboardComponent

El AdminDashboardComponent sirve como la principal página de destino para los administradores, exhibiendo métricas clave y proporcionando acceso rápido a tareas administrativas comunes.

**Características clave**

|  |  |  |
| --- | --- | --- |
| Característica | Description | Implementation |
| Statistics Display | Shows total teachers, schedules, today's attendances, and pending justifications | Uses computed signals with service data |
| Quick Actions | Provides shortcuts to common admin tasks | Implemented via QuickActionsComponent |
| Stats Cards | Visual representation of key metrics | Implemented via StatsCardComponent |

**Encuadernaciones de datos**

El componente utiliza las señales computarizadas de Angular para mostrar de forma reactiva estadísticas:

* totalTeachers: Computado de teacherService.teachers().length
* totalSchedules: Agregado de todos los horarios de los maestros
* todayAttendances: Filtrado de attendanceService.getAttendancesByDate()
* pendingJustifications: De attendanceService.getPendingJustifications()

Fuentes:

[src/app/features/admin/dashboard/admin-dashboard.component.ts91 a 103](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/admin-dashboard.component.ts#L91-L103)

### MaestroGestiónComponento

El TeacherManagementComponentproporciona operaciones CRUD para los registros de maestros a través de una interfaz basada en tabla.

#### Estructura de la tabla

|  |  |  |
| --- | --- | --- |
| Column | Field | Description |
| Identificación de empleado | employeeId | Unique identifier for teacher |
| Nombre | name | Teacher's full name |
| Correo electrónico | email | Contact email |
| Departamento | department | Academic department |
| Horarios | schedules | Number of assigned schedules |
| Acciones | - - | Editar/Eliminar operaciones |

**Operaciones**

* **Crear** : Abrir TeacherFormDialogComponentsin datos
* **Actualización:** Abre TeacherFormDialogComponentcon los datos del profesor existentes
* **Eliminar** : Confirma eliminación y llamadas teacherService.deleteTeacher()

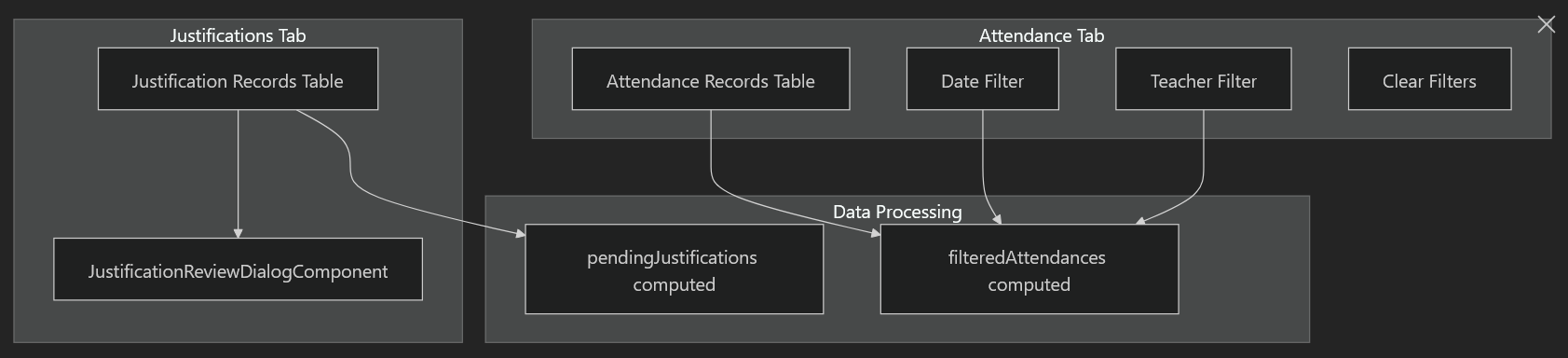
Fuentes:

[src/app/features/admin/teachers/teacher-management.component.ts40-65](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-management.component.ts#L40-L65)

### AsistenciaConcompleto de Gestión

El AttendanceManagementComponentgestiona registros de asistencia y revisiones de justificación a través de una interfaz de pestañas.

**Características de la gestión de asistencia**



**Sistema de filtrado**

El componente implementa filtrado reactivo usando señales angulares:

* selectedAttendanceDate: Señal para el filtrado de fechas
* selectedTeacherId: Signal para filtrado de profesor
* filteredAttendances: Señamento computarizada que aplica ambos filtros
* allAttendances: Señanza computal de base desde attendanceService.attendances()

Fuentes: [src/app/features/admin/attendance/attendance-management.component.ts76-125](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/attendance/attendance-management.component.ts#L76-L125)

## Arquitectura de flujo de datos

El Portal de administración sigue un patrón de flujo de datos unidireccional a través de capas de servicio:

A diagram of a service layer

AI-generated content may be incorrect.

Fuentes: [src/app/features/admin/dashboard/admin-dashboard.component.ts105 - 108](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/admin-dashboard.component.ts#L105-L108) [src/app/features/admin/teachers/teacher-management.component.ts49 - 51](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-management.component.ts#L49-L51) [src/app/features/admin/attendance/attendance-management.component.ts46 a 48](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/attendance/attendance-management.component.ts#L46-L48)

## Integración de servicios

### AsistenciaService Integración

El Portal de administración utiliza ampliamente AttendanceServicepara la gestión de la asistencia y justificación:

**Métodos clave utilizados**

|  |  |  |
| --- | --- | --- |
| Method | Purpose | Components |
| attendances() | Obtenga todos los registros de asistencia | AttendanceManagementComponent |
| justifications() | Consigue todas las justificaciones | AttendanceManagementComponent |
| getAttendancesByDate() | Filtrar por fecha | AdminDashboardComponent |
| getPendingJustifications() | Obtenga revisiones pendientes | AdminDashboardComponent |
| reviewJustification() | Aprobar/rechazar justificaciones | JustificationReviewDialogComponent |

**Persistencia de los datos**

El servicio se encarga automáticamente de la persistencia local de almacenamiento:

* saveAttendances(): Persiste récords de asistencia
* saveJustifications(): Persiste en los registros de justificación
* loadData(): Inicializa datos de localStorage sobre construcción de servicios

Fuentes: [src/app/core/services/attendance.service.ts56-66](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/attendance.service.ts#L56-L66) [src/app/core/services/attendance.service.ts18 - 54](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/attendance.service.ts#L18-L54)

### Integración docente

El Portal de administración utiliza TeacherServicepara las operaciones de gestión del profesorado:

**Operaciones de la CRUD**

* **Crear** : addTeacher()- Llamada de TeacherFormDialogComponent
* **Lea** : teachers()- Acceso reactivo basado en la señal
* **Actualización** : updateTeacher()- Llamada de TeacherFormDialogComponent
* Suprímase **Delete**: deleteTeacher()- Llamada de TeacherManagementComponent

**Maestro Busto**

El servicio proporciona métodos de utilidad para el acceso a los datos del maestro:

* getTeacherById(): Utilizado en todos los componentes de administración para la visualización del nombre del maestro
* getTeacherByEmployeeId(): Método de búsqueda alternativa

Fuentes: [src/app/features/admin/teachers/teacher-management.component.ts61 a 65](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-management.component.ts#L61-L65) [src/app/features/admin/attendance/attendance-management.component.ts126-134](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/attendance/attendance-management.component.ts#L126-L134)

## Patrones de flujo de trabajo de administración

### Justificación Revisión Flujo de trabajo

A diagram of a company

AI-generated content may be incorrect.

Fuentes: [src/app/features/admin/attendance/attendance-management.component.ts188-198](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/attendance/attendance-management.component.ts#L188-L198) [src/app/core/services/attendance.service.ts209 a 216](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/attendance.service.ts#L209-L216)

### Desplazamiento de trabajo de la Gerencia de

A diagram of a software project

AI-generated content may be incorrect.

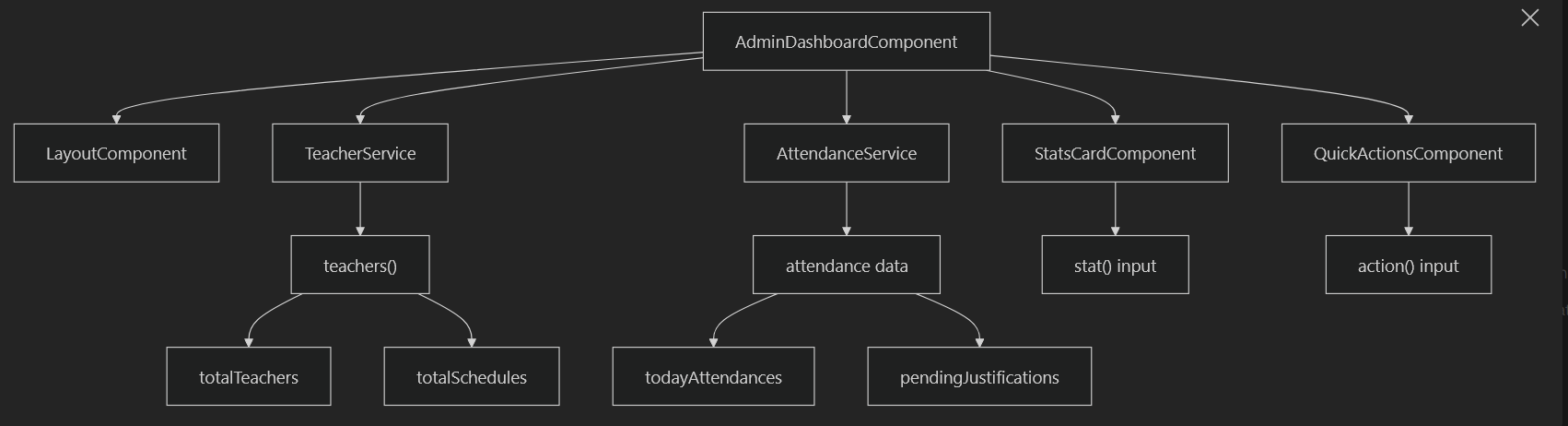
Fuentes: [src/app/features/admin/teachers/teacher-management.component.ts54-65](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-management.component.ts#L54-L65)

# Admin Dashboard

El panel de administración sirve como interfaz central de control para los administradores, proporcionando una visión general de las estadísticas del sistema y el rápido acceso a funciones administrativas clave. Este documento cubre la implementación de los componentes principales del salpicadero, la computación de datos y la estructura de interfaz de usuario.

## Arquitectura de componentes

El AdminDashboardComponentes el principal componente contenedor que orquesta la pantalla del salpicadero. Aprovecha la programación reactiva basada en señales de Angular para calcular estadísticas en tiempo real y renderiza dos subcomponentes clave para mostrar información.

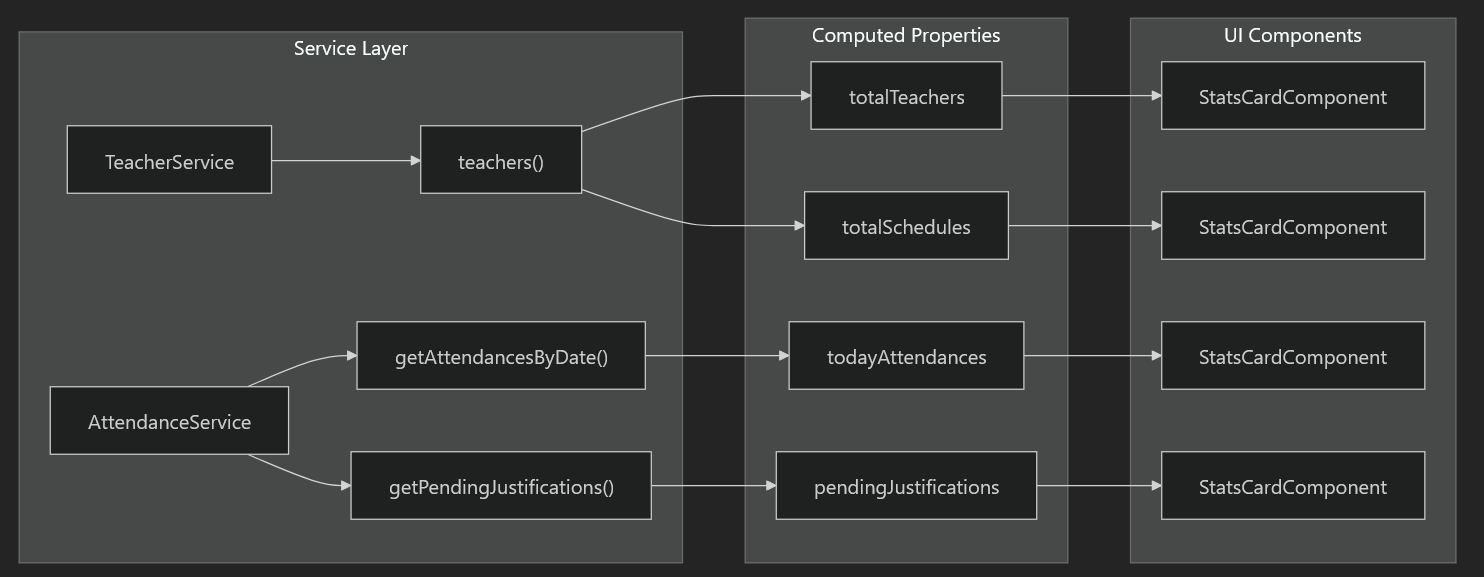


**Fuentes:** [src/app/features/admin/dashboard/admin-dashboard.component.ts1-109](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/admin-dashboard.component.ts#L1-L109)

## Cálculo de las estadísticas

El salpicadero usa Angular's computed()señales para calcular métricas del sistema de claves. Estos cálculos se actualizan automáticamente cuando los datos subyacentes cambian.

|  |  |  |
| --- | --- | --- |
| Estatística | Fuente de computación | Descripción |
| totalTeachers | teacherService.teachers().length | Número total de profesores inscritos |
| totalSchedules | Suma de todos los horarios de los maestros | Número total de clases programadas en todos los maestros |
| todayAttendances | attendanceService.getAttendancesByDate() | Número de registros de asistencia correspondientes a la fecha actual |
| pendingJustifications | attendanceService.getPendingJustifications() | Número de justificaciones a la espera de revisión administrativa |



**Fuentes:** [src/app/features/admin/dashboard/admin-dashboard.component.ts91 a 103](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/admin-dashboard.component.ts#L91-L103)

## Estructura del menú de navegación

El salpicadero define una estructura de menús que proporciona navegación a todas las principales funciones administrativas:

  menuItems: MenuItem[] = [

    { label: 'Dashboard', route: '/admin/dashboard', icon: 'dashboard' },

    { label: 'Docentes', route: '/admin/teachers', icon: 'people' },

    { label: 'Horarios', route: '/admin/schedules', icon: 'schedule' },

    { label: 'Asistencias', route: '/admin/attendance', icon: 'assignment' },

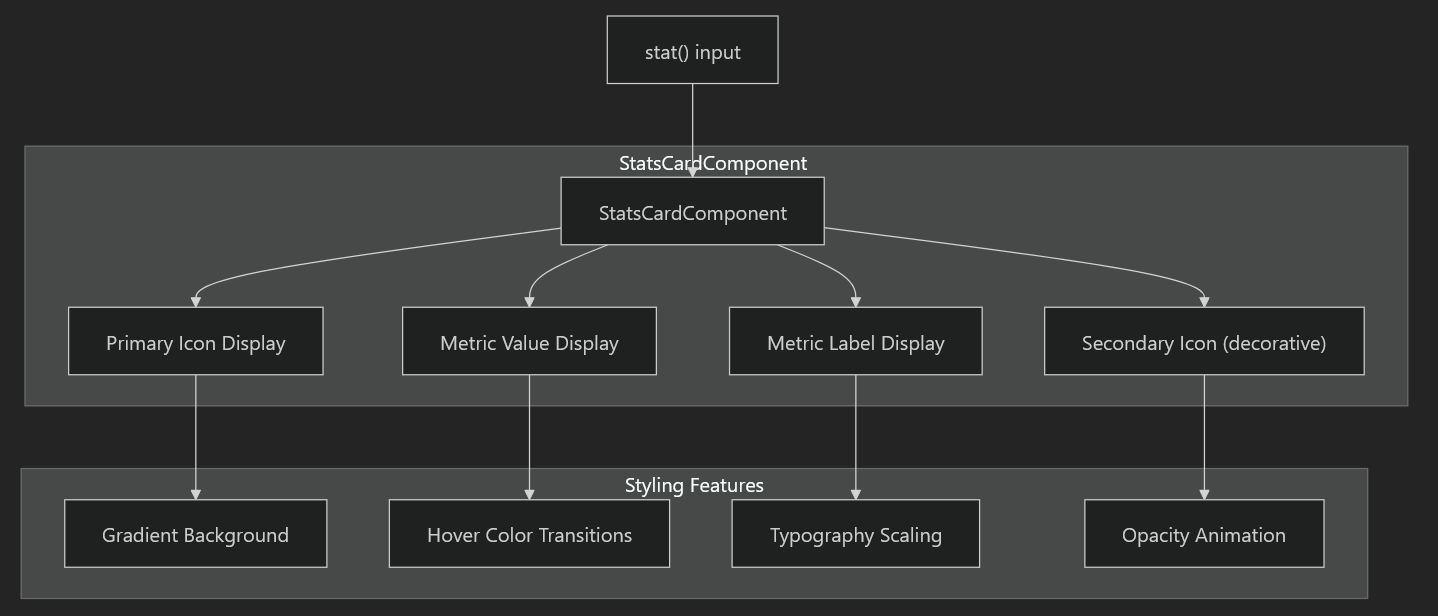
  ];

**Fuentes:** [src/app/features/admin/dashboard/admin-dashboard.component.ts84-89](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/admin-dashboard.component.ts#L84-L89)

## Componente de tarjeta de estadísticas

El StatsCardComponentes un componente reutilizable que muestra estadísticas individuales con mejoras visuales. Cada carta recibe un stat()entradas que contengan los datos métricos y la información de formateo.

El componente implementa efectos flotadores y fondos de gradiente para una mejor experiencia de usuario:



**Fuentes:** [src/app/features/admin/dashboard/stats-card.component.html1-33](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/stats-card.component.html#L1-L33)

### Componente de acción rápida

El QuickActionsComponentproporciona atajos a las funciones administrativas de acceso frecuente. Cada acción contiene información de enrutamiento y contenido descriptivo.

La estructura de los componentes incluye:

* **Cabeza de acción:** Icono y título con efectos flotantes
* **Descripción** :Texto explicativo sobre la acción
* **Botón de acción** : Botón de racha con efectos de animación

A screenshot of a computer

AI-generated content may be incorrect.

**Fuentes:** [src/app/features/admin/dashboard/quick-actions.component.html1-52](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/quick-actions.component.html#L1-L52)

## Dependencias de servicios

El salpicadero depende de dos servicios básicos para operaciones de datos:

### Integración docente

* Ofrece acceso a los datos del profesorado a través de teachers()señal
* Permite calcular las estadísticas relacionadas con el profesorado
* Suministros de la lista de información para cálculos agregados

### AsistenciaService Integración

* Suministros de datos de asistencia filtrados por fecha
* Proporciona conteos de justificación pendientes
* Permite seguimiento de asistencia en tiempo real

**Fuentes:** [src/app/features/admin/dashboard/admin-dashboard.component.ts105 - 108](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/admin-dashboard.component.ts#L105-L108)

## Estilo y Animaciones

El salpicadero implementa animaciones y transiciones CSS completas para una mejor experiencia de usuario:

* **Efectos de la sabadura:** Transformar animaciones en tarjetas y botones
* **Animaciones superiores** : Transtransmisiones de gradiente de fondo
* **Transformaciones de escalas:** Icono y escalamiento de elementos en la interacción
* **Efectos de la sombra:** Cambios dinámicos de sombra en el flotador

El estilo utiliza clases de Tailwind CSS combinadas con CSS personalizadas para animaciones avanzadas, incluyendo una animación de gradiente basada en fotos clave que cicloja posiciones de fondo.

**Fuentes:** [src/app/features/admin/dashboard/admin-dashboard.component.ts30 a 81](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/admin-dashboard.component.ts#L30-L81)

# Gestión del profesorado

Este documento cubre el sistema administrativo de Gestión Docente, que proporciona las operaciones de la CRUD para la gestión de los registros de los maestros dentro de la aplicación de gestión de asistencia. Este sistema permite a los administradores crear, ver, editar y eliminar perfiles de maestros, incluyendo su información personal, asignaciones del departamento e identificación de empleados.

## Finalidad y alcance

El sistema de Gestión Docente permite a los administradores:

* Vea una lista completa de todos los profesores en un formato tabular
* Añadir nuevos profesores con información completa de perfil
* Editar los registros de los maestros existentes
* Eliminar registros de profesores con confirmación
* Gestionar asignaciones de departamentos e identificaciones de empleados
* Validar los datos del profesor a través de controles de forma

## Arquitectura de sistemas

### Estructura de componentes

**Flujo de Componente de Gestión Docente** Fuentes: [src/app/features/admin/teachers/teacher-management.component.ts1-67](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-management.component.ts#L1-L67) [src/app/features/admin/maachers/teacher-form-dialog.component.ts1-85](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-form-dialog.component.ts#L1-L85)

### Arquitectura de flujo de datos

**Flujo de datos de gestión docente** Fuentes: [src/app/features/admin/teachers/teacher-management.component.ts49 a 66](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-management.component.ts#L49-L66) [src/app/features/admin/maachers/teacher-form-dialog.component.ts60-84](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-form-dialog.component.ts#L60-L84)

## Componentes básicos

### MaestroGestiónComponento

El principal componente responsable de mostrar la lista de maestros y gestionar las operaciones del profesorado.

|  |  |  |
| --- | --- | --- |
| Bienes | Tipo | Descripción |
| teacher | El maestro de correos. | null. |
| menuItems | MenuItem[] | Configuración del menú de navegación |
| displayedColumns | string[] | Definiciones de la columna |
| teacherService | TeacherService | Inyectó servicio para operaciones de maestros |
| dialog | MatDialog | Servicio de diálogo de materiales angulares |

**Métodos clave**

* openTeacherDialog(teacher?: Teacher)- Abre el diálogo de forma del profesor para crear/editar operaciones
* deleteTeacher(teacher: Teacher)- Elimina un acta de profesor con confirmación

Fuentes:

[src/app/features/admin/teachers/teacher-management.component.ts30-66](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-management.component.ts#L30-L66)

### MaestroFormDialogComponent

Componente de diálogo modificado para la creación y edición de registros del profesorado con validación de forma.

|  |  |  |
| --- | --- | --- |
| Bienes | Tipo | Descripción |
| teacherForm | FormGroup | Formulario reactivo para datos del profesorado |
| teacher | Teacher | Datos del profesor pasados por inyección de diálogo |
| fb | FormBuilder | Servicio de constructora de formularios |
| teacherService | TeacherService | Servicio de operaciones de los maestros |
| dialogRef | MatDialogRef | Dialog de referencia para el cierre |

**Forma campos**

El formulario del profesor incluye los siguientes campos validados:

|  |  |  |
| --- | --- | --- |
| Campo | Reglas de validación | Descripción |
| name | Requerido | El nombre completo del profesor |
| email | Se requiere, formato de correo electrónico | Dirección de correo electrónico del profesor |
| employeeId | Requerido | Identificado único del empleado |
| department | Requerido | Asignación del Departamento |
| phone | Opcional | Número de teléfono de contacto |

**Métodos clave**

* onSave()- Manijas forma de presentación para operaciones de creación/actualización
* close()- Cierra el diálogo sin guardar
* ngOnInit()- Inicializa la forma con los datos del profesorado existentes para el modo de edición

Fuentes:

[src/app/features/admin/maachers/teacher-form-dialog.component.ts33 - 84](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-form-dialog.component.ts#L33-L84)

## Interfaz de usuario

### Vista de lista de maestros

La interfaz de gestión del profesora muestra a los profesores en un formato de tabla completo con las siguientes columnas:

* **ID Empleado** - ID de empleado con icono de placa
* **Nombre** - Nombre completo del profesor
* **Correo** - Dirección de correo electrónico
* **Departamento** con placas codificadas por color
* **Horarios** - Número de horarios asignados
* **Acciones** - Editar y eliminar botones de acción

**Departa de Color Coding**

El sistema utiliza insignias codificadas por color para diferentes departamentos:

  'bg-green-100 text-green-800': teacher.department === 'Matemáticas',

                                        'bg-blue-100 text-blue-800': teacher.department === 'Ciencias',

                                        'bg-purple-100 text-purple-800': teacher.department === 'Humanidades',

                                        'bg-yellow-100 text-yellow-800': teacher.department === 'Tecnología',

                                        'bg-gray-100 text-gray-800'

Fuentes: [src/app/features/admin/teachers/teacher-management.component.html24 a 121](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-management.component.html#L24-L121)

### Diálogo de formación de profesores

El diálogo de forma del profesor proporciona una interfaz limpia para la entrada de datos con:

* Etiquetas de campo basadas en iconos
* Validación de la forma en tiempo real
* Mostrar mensaje de error
* Diseño sensible
* Botones de acción para operaciones de guardar/cancelar

**Departamentos disponibles**

El sistema soporta los siguientes departamentos predefinidos:

* Matemáticas
* Historia
* Ciencias
* Literatura
* Inglés
* Educación Física
* Arte

Fuentes: [src/app/features/admin/teachers/teacher-form-dialog.component.html68-78](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-form-dialog.component.html#L68-L78)

## Puntos de integración

### Dependencias de servicios

**Arquitectura de Integración de Servicios** Fuentes: [src/app/features/admin/teachers/teacher-management.component.ts49-52](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-management.component.ts#L49-L52) [src/app/features/admin/maachers/teacher-form-dialog.component.ts34 - 40](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-form-dialog.component.ts#L34-L40)

### Integración de la navegación

El componente de Gestión del Profesor se integra con el sistema de navegación administrativa a través de la LayoutComponent:

  menuItems: MenuItem[] = [

    { label: 'Dashboard', route: '/admin/dashboard', icon: 'dashboard' },

    { label: 'Docentes', route: '/admin/teachers', icon: 'people' },

    { label: 'Horarios', route: '/admin/schedules', icon: 'schedule' },

    { label: 'Asistencias', route: '/admin/attendance', icon: 'assignment' },

  ];

Fuentes: [src/app/features/admin/teachers/teacher-management.component.ts33 a 38](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-management.component.ts#L33-L38)

## Operaciones de datos

### Maestros Operaciones de la CRUD

**Maestra CRUD operación Flujo** Fuentes: [src/app/features/admin/maachers/teacher-form-dialog.component.ts60-79](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-form-dialog.component.ts#L60-L79) [src/app/features/admin/teachers/teacher-management.component.ts61 a 65](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-management.component.ts#L61-L65)

### Normas de validación de la formación

El formulario de profesorado implementa una validación integral:

  teacherForm = **this**.fb.group({

    name: ['', [Validators.required]],

    email: ['', [Validators.required, Validators.email]],

    employeeId: ['', Validators.required],

    department: ['', Validators.required],

    phone: [''],

  });

Fuentes: [src/app/features/admin/maachers/teacher-form-dialog.component.ts40 a 46](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-form-dialog.component.ts#L40-L46)

## Manejo de errores

El sistema implementa el manejo de errores a múltiples niveles:

1. **Validación** de **formulario** - Validación del lado del cliente con retroalimentación inmediata
2. **Integración de servicios** - Manejo de errores dentro de las operaciones de servicio
3. **Confirmación del usuario** - Diálogos de confirmación para operaciones destructivas

### Muestra de error de validación

Los errores de validación de formularios se muestran con iconos y mensajes descriptivos:

      @if (teacherForm.get('employeeId')?.hasError('required') && teacherForm.get('employeeId')?.touched) {

      <div *class*="text-sm text-red-600 mt-2 flex items-center">

        <mat-icon *class*="!text-sm mr-1">error</mat-icon>

        El ID de empleado es requerido

      </div>

      }

Fuentes: [src/app/features/admin/teachers/teacher-form-dialog.component.html20 a 24](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-form-dialog.component.html#L20-L24)

## Consideraciones de seguridad

El sistema de gestión del profesorado incluye medidas básicas de seguridad:

* **Acceso basado en el papel** - Sólo accesible a través del portal de administración
* **Validación de datos** - Validación de lado del cliente y del lado del servidor
* **Diálogos de confirmación** - Previene la eliminación accidental de datos
* **Form Sanitization** - Angular's incorporado de la forma desinfectación

Fuentes: [src/app/features/admin/teachers/teacher-management.component.ts1-67](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-management.component.ts#L1-L67) [src/app/features/admin/maachers/teacher-form-dialog.component.ts1-85](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/teachers/teacher-form-dialog.component.ts#L1-L85)

# Gestión de los horarios

Este documento cubre la interfaz administrativa para la gestión de los horarios de los maestros y las asignaciones de clases dentro del portal administrativo. El sistema de gestión de horarios permite a los administradores crear, ver, editar y eliminar los horarios de los maestros en todos los días de la semana.

## Resumen

El sistema de gestión de horarios proporciona a los administradores un control centralizado sobre todos los horarios de los maestros. Consiste en una interfaz de gestión principal que muestra todos los horarios en un formato tabular, y un sistema de formulario basado en diálogos para crear y editar entradas de horarios.

El sistema se integra con la TeacherService gestionar los datos del calendario y proporciona actualizaciones en tiempo real de la pantalla del calendario cuando se realicen cambios.

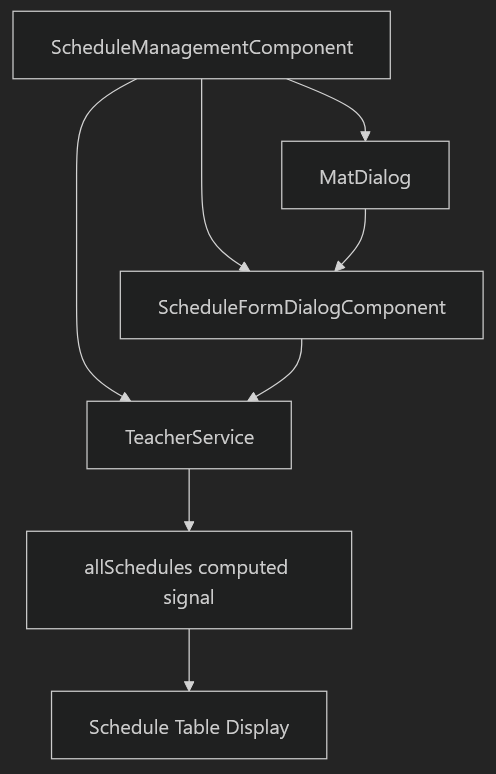
**Fuentes:**

[src/app/features/admin/schedule/schedule-management.component.ts1-107](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/schedule/schedule-management.component.ts#L1-L107)

## Componentes básicos

### ScheduleManagementComponent

El ScheduleManagementComponent sirve como interfaz principal para la administración de horarios. Muestra todos los horarios de todos los maestros en una vista de mesa unificada y proporciona acceso a las funciones de creación y edición de horarios.



**Fuentes:** [src/app/features/admin/schedule/schedule-management.component.ts32 - 107](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/schedule/schedule-management.component.ts#L32-L107)

### ScheduleFormDialogComponent

El ScheduleFormDialogComponent maneja tanto la creación de horarios como la edición a través de una interfaz de diálogo modal. Utiliza formularios reactivos para la validación de datos e se integra directamente con la TeacherService para operaciones de persistencia.

**Fuentes:** [src/app/features/admin/schedule/schedule-form-dialog.component.ts29 - 107](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/schedule/schedule-form-dialog.component.ts#L29-L107)

## Modelo y estructura de datos

### Programar Estructura de datos

El sistema de horarios funciona con objetos mejorados que incluyen información del profesorado con fines de visualización:

|  |  |  |
| --- | --- | --- |
| Field | Tipo | Descripción |
| teacherId | string | Reference to teacher ID |
| dayOfWeek | número | Day of week (0-6, Sunday-Saturday) |
| startTime | cuerdas | Start time in HH:MM format |
| endTime | cuerdas | End time in HH:MM format |
| subject | cuerdas | Subject or course name |
| classroom | cuerdas | Ubicación en el aula |
| teacherName | cuerdas | Mostrar nombre del profesor asignado |
| teacherEmployeeId | cuerdas | Identificación de empleado para identificación de maestro |

**Fuentes:** [src/app/features/admin/schedule/schedule-management.component.ts49-72](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/schedule/schedule-management.component.ts#L49-L72)

### Agregación de la Lista

El sistema agrega los horarios de todos los maestros en una sola señal computada llamada allSchedules(). Esta señal calculada:

* Iterates through all teachers from TeacherService
* Extracts each teacher's schedules
* Enriches schedule data with teacher name and employee ID
* Sorts schedules by day of week, then by start time

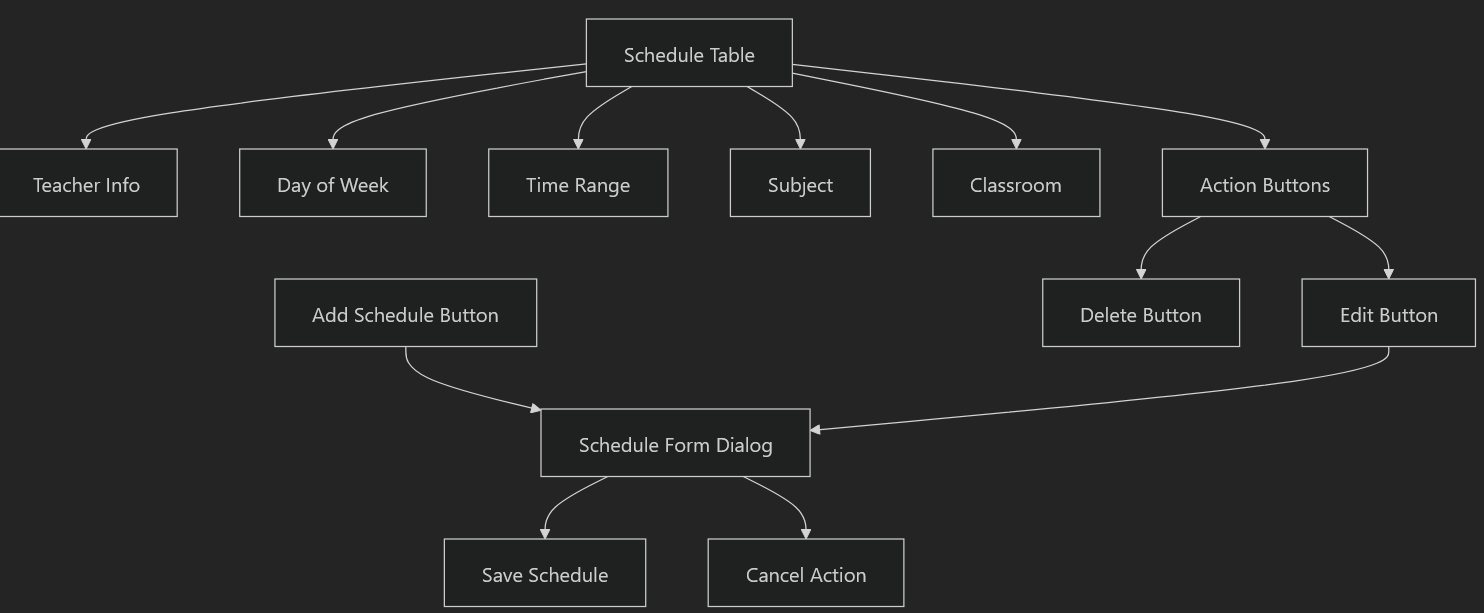
**Fuentes:**

[src/app/features/admin/schedule/schedule-management.component.ts49-72](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/schedule/schedule-management.component.ts#L49-L72)

## Interfaz de uso y flujos de trabajo

### Interfaz de gestión de cuadros

La interfaz principal de gestión de horarios ofrece una visión completa de todas las clases programadas:



**Fuentes:** [src/app/features/admin/schedule/schedule-management.component.html25 - 127](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/schedule/schedule-management.component.html#L25-L127)

### Calendario de formulario Diálogo

El diálogo de formulario de calendario proporciona una interfaz estructurada para la entrada de datos de calendario:

A black screen with white text

AI-generated content may be incorrect.

**Fuentes:** [src/app/features/admin/schedule/schedule-form-dialog.html10 a 129](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/schedule/schedule-form-dialog.component.html#L10-L129)

## Integración de servicios

### Integración docente

El sistema de gestión de horarios se integra estrechamente con la TeacherService para todas las operaciones de datos:

|  |  |  |
| --- | --- | --- |
| Operation | Method | Descripción |
| Lea | teachers() | Retrieves all teachers and their schedules |
| Creación | addScheduleToTeacher() | Adds new schedule to specified teacher |
| Actualización | updateTeacher() | Updates teacher's schedule array |
| Supríma | removeScheduleFromTeacher() | Removes schedule from teacher |

**Fuentes:** [src/app/features/admin/schedule/schedule-management.component.ts73 - 74](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/schedule/schedule-management.component.ts#L73-L74) [src/app/features/admin/schedule/schedule-form-dialog.component.ts33 a 34](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/schedule/schedule-form-dialog.component.ts#L33-L34)

### Validación de la forma y encuadernación de datos

El formulario de programación utiliza formularios reactivos angulares con una validación completa:

  scheduleForm = **this**.fb.group({

    teacherId: ['', Validators.required],

    dayOfWeek: ['', Validators.required],

    startTime: ['', Validators.required],

    endTime: ['', Validators.required],

    subject: ['', Validators.required],

    classroom: ['', Validators.required],

  });

**Fuentes:** [src/app/features/admin/schedule/schedule-form-dialog.component.ts37 a 44](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/schedule/schedule-form-dialog.component.ts#L37-L44)

## Operaciones de la CRUD

### Crear horario

La creación de un nuevo calendario sigue este flujo de trabajo:

1. El usuario hace clics "Agregar Horario" botón
2. ScheduleFormDialogComponent se abre en modo de creación
3. Usuario selecciona al profesor y rellena campos de formulario
4. Validación de formularios asegura que todos los campos requeridos están presentes
5. addScheduleToTeacher()Persiste el nuevo calendario
6. Diálogo se cierra y la mesa se refresca automáticamente

**Fuentes:**

[src/app/features/admin/schedule/schedule-form-dialog.component.ts89-97](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/schedule/schedule-form-dialog.component.ts#L89-L97)

### Editar el calendario

La edición del horario sigue este flujo de trabajo:

1. Usuario hace clics botón de editar para un horario específico
2. ScheduleFormDialogComponentse abre en modo de edición
3. La forma está pre-poblada con los datos de los horarios existentes
4. Desactivan selección de maestros para evitar reasignación
5. Los datos actualizados se guardan a través de updateTeacher()método de método
6. Diálogo se cierra y tabla refleja cambios

**Fuentes:**

[src/app/features/admin/schedule/schedule-form-dialog.component.ts71-88](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/schedule/schedule-form-dialog.component.ts#L71-L88)

### Borrar el calendario

Eliminación de la agenda utiliza un diálogo de confirmación:

1. El usuario hace clics botones borrados
2. Diálogo de confirmación aparece con sujeto de horario
3. De confirmarse, removeScheduleFromTeacher()método se llama
4. Se elimina el horario y se actualiza la tabla

**Fuentes:**

[src/app/features/admin/schedule/schedule-management.component.ts84-93](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/schedule/schedule-management.component.ts#L84-L93)

## Navegación y diseño

La interfaz de gestión de horarios se integra con el sistema de navegación del portal de administración a través de la LayoutComponent. Los elementos del menú incluyen navegación a otras funciones de administración:

* - Abre el diálogo de forma del profesor para crear/editar operacionesDashboard (/admin/dashboard)
* - Elimina un acta de profesor con confirmaciónTeachers (/admin/teachers)
* Calendarios (a)/admin/schedules) - página actual
* Asistencia (/admin/attendance)

**Fuentes:**

[src/app/features/admin/schedule/schedule-management.component.ts33 a 38](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/schedule/schedule-management.component.ts#L33-L38)

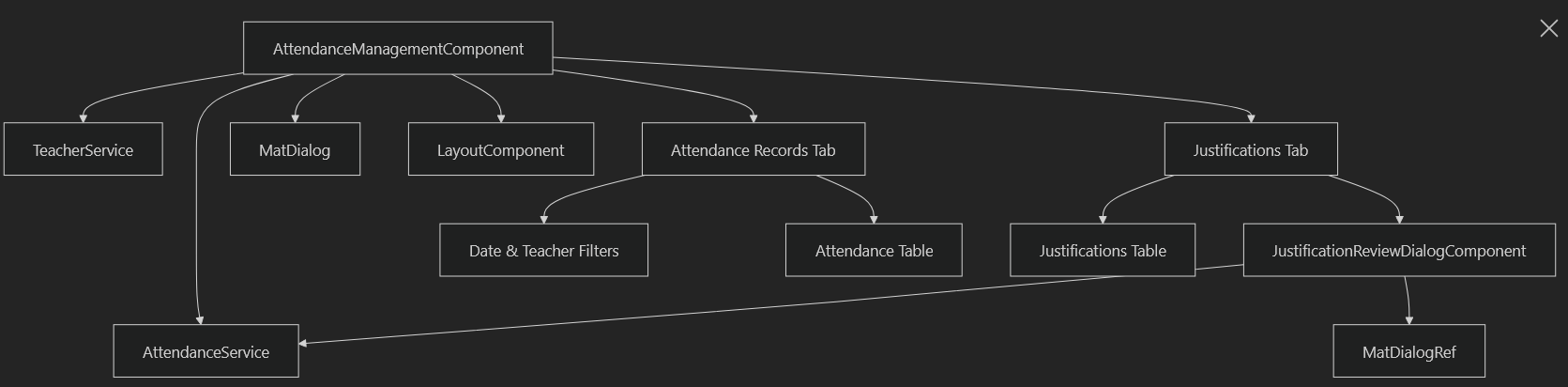
# Gestión de asistencia

Este documento cubre la interfaz administrativa para gestionar los registros de asistencia al profesorado y revisar justificaciones. El sistema de gestión de asistencia proporciona a los administradores herramientas para monitorear la asistencia de maestros, los registros de filtros y revisar las justificaciones de ausencia presentadas por los maestros.

## Arquitectura de componentes

El sistema de gestión de asistencia se construye en torno a la AttendanceManagementComponent que proporciona una interfaz de pestañas para ver los registros de asistencia y la gestión de justificaciones.

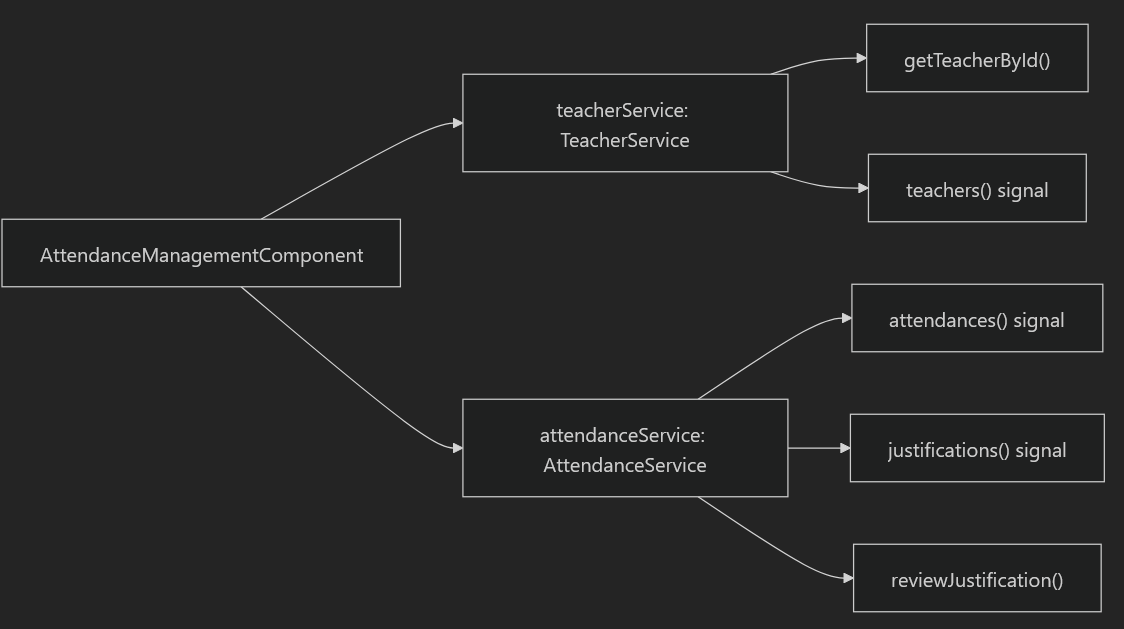
### Reseña del componente



Fuentes: [src/app/features/admin/attendance/attendance-management.component.ts1-199](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/attendance/attendance-management.component.ts#L1-L199) [src/app/features/adminance/attendance/justification-review-dialog.component.ts1-36](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/attendance/justification-review-dialog.component.ts#L1-L36)

### Integración de los servicios básicos

El componente se integra con los servicios básicos para recuperar y manipular los datos de asistencia:



Fuentes: [src/app/features/admin/attendance/attendance-management.component.ts46 a 48](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/attendance/attendance-management.component.ts#L46-L48) [src/app/features/admin/attendance/attendance-management.component.ts126-143](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/attendance/attendance-management.component.ts#L126-L143)

## Estructura de interfaz de usuario

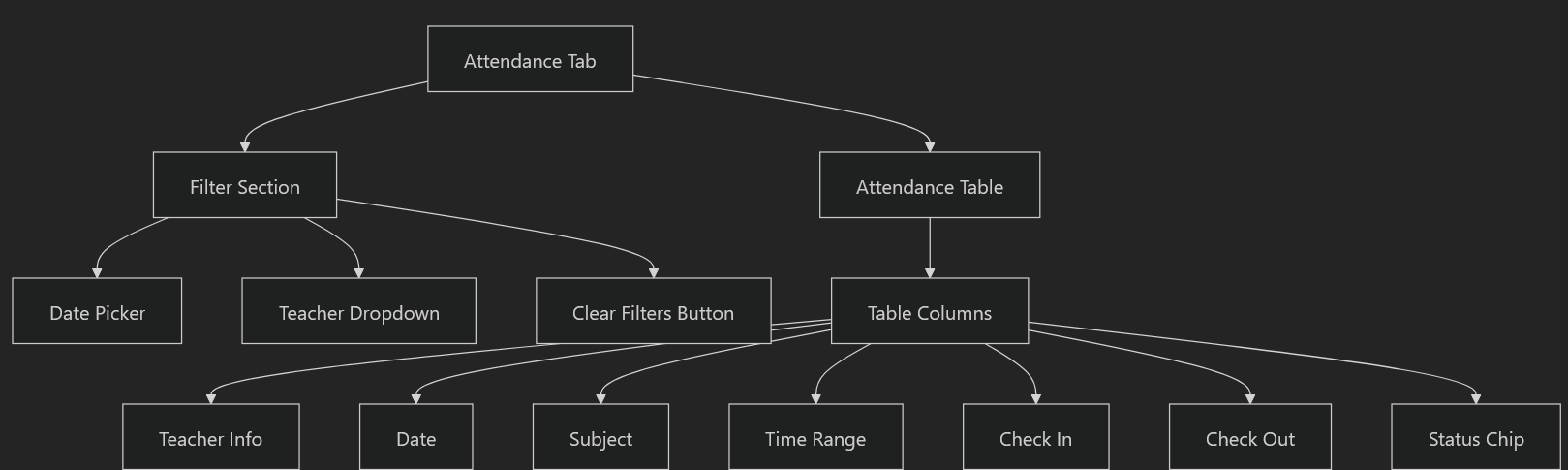
La interfaz se organiza en dos pestañas principales utilizando el componente de pestañas de Angular Material.

### Organización de la Tab

|  |  |  |
| --- | --- | --- |
| Tab | Purpose | Key Features |
| Registro de Asistencias | View attendance records | Date/teacher filtering, status display |
| Sólo unas penas. | Review justifications | Approval/rejection workflow |

### Asistencia Records Tab

La primera pestaña muestra los registros de asistencia con capacidades de filtrado completas:



Fuentes: [src/app/features/adminance/attendance-management.component.html19-47](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/attendance/attendance-management.component.html#L19-L47) [src/app/features/adminance/attendance-management.component.html49 a 125](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/attendance/attendance-management.component.html#L49-L125)

### Justificaciones Tab

La segunda ficha se centra en la gestión de la justificación:

A screenshot of a computer

AI-generated content may be incorrect.

Fuentes: [src/app/features/adminance/attendance-management.component.html127-206](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/attendance/attendance-management.component.html#L127-L206)

## Flujo de datos y filtrado

El componente utiliza señales angulares para la gestión de datos reactivos con propiedades calculadas para filtrado.

### Asistencia Data Flow

A diagram of a computer

AI-generated content may be incorrect.

Fuentes: [src/app/features/admin/attendance/attendance-management.component.ts79-105](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/attendance/attendance-management.component.ts#L79-L105) [src/app/features/admin/attendance/attendance-management.component.ts76-77](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/attendance/attendance-management.component.ts#L76-L77)

### Aplicación de la filtración

El sistema de filtrado utiliza dos señales reactivas:

|  |  |  |
| --- | --- | --- |
| Signal | Type | Purpose |
| selectedAttendanceDate | \*Fecha \* | null` |
| selectedTeacherId | string | Filters by specific teacher |

A diagram of a computer

AI-generated content may be incorrect.

Fuentes: [src/app/features/admin/attendance/attendance-management.component.ts113-124](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/attendance/attendance-management.component.ts#L113-L124)

## Gestión de la situación

El sistema maneja múltiples estados de asistencia y justificación con los indicadores visuales correspondientes.

### Tipos de situación de asistencia

|  |  |  |  |
| --- | --- | --- | --- |
| Situación | Ebel española | Color | Descripción |
| present | Presente | primaria | Maestra asistió a clase |
| absent | Ausente | Avisar | Maestra no asistió |
| late | Tardanza | Avisar | Maestro llegó tarde |
| justified | Justificado | acento | La ausencia estaba justificada |

### Tipos de estado de Justificación

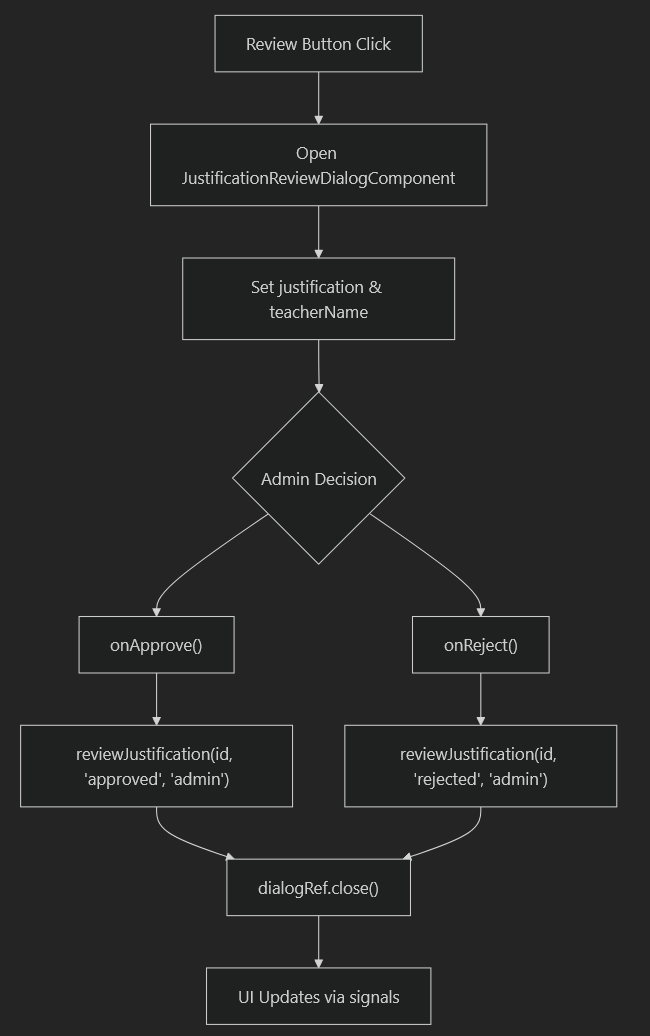
|  |  |  |  |
| --- | --- | --- | --- |
| Situación | Ebel española | Color | Descripción |
| pending | Pendiente | acento | A la espera de revisión de admin |
| approved | Aprobada | primaria | Justificación aceptada |
| rejected | Rechazada | Avisar | Denegada justificación |

Fuentes: [src/app/features/admin/attendance/attendance-management.component.ts145-186](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/attendance/attendance-management.component.ts#L145-L186)

## Justificación Revisión Flujo de trabajo

El proceso de revisión de la justificación utiliza un diálogo modal para las decisiones administrativas.

### Flujo de proceso de revisión



Fuentes: [src/app/features/admin/attendance/attendance-management.component.ts188-199](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/attendance/attendance-management.component.ts#L188-L199) [src/app/features/adminance/attendance/justification-review-dialog.component.ts22-34](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/attendance/justification-review-dialog.component.ts#L22-L34)

### Estructura de componentes de diálogo

El JustificationReviewDialogComponentproporciona una simple interfaz de aprobación/rejeción:

A diagram of a company

AI-generated content may be incorrect.

Fuentes: [src/app/features/adminance/attendance/justification-review-dialog.component.ts](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/attendance/justification-review-dialog.component.ts#L16-L34)

## Definiciones de columna

El componente define las columnas de tabla como matrices de cuerda para las dos vistas principales de datos:

### Columnas de la Mesa de asistencia

  attendanceColumns: string[] = [

    'teacher',

    'date',

    'subject',

    'time',

    'checkIn',

    'checkOut',

    'status',

  ];

### Justificación de columnas de mesa

justificationColumns: string[] = [

    'teacher',

    'date',

    'reason',

    'createdAt',

    'status',

    'actions',

  ];

Fuentes: [src/app/features/admin/attendance/attendance-management.component.ts57-74](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/attendance/attendance-management.component.ts#L57-L74)

## Métodos de ayudantes

El componente incluye varios métodos de utilidad para la presentación de datos y búsquedas cruzadas de servicios:

|  |  |  |
| --- | --- | --- |
| Método | Objeto | Tipo de retorno |
| getTeacherName() | Conseguir nombre del profesor por identificación | string |
| getTeacherEmployeeId() | Obtenga identificación de empleado por identificación de profesor | string |
| getScheduleInfo() | Obtenga los detalles del horario por ID | \*Schedule \* |
| getStatusText() | Convertir estado en Etiqueta española | string |
| getStatusColor() | Obtenga color de material para el estado | Primaria |

Fuentes: [src/app/features/admin/attendance/attendance-management.component.ts126-186](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/attendance/attendance-management.component.ts#L126-L186)

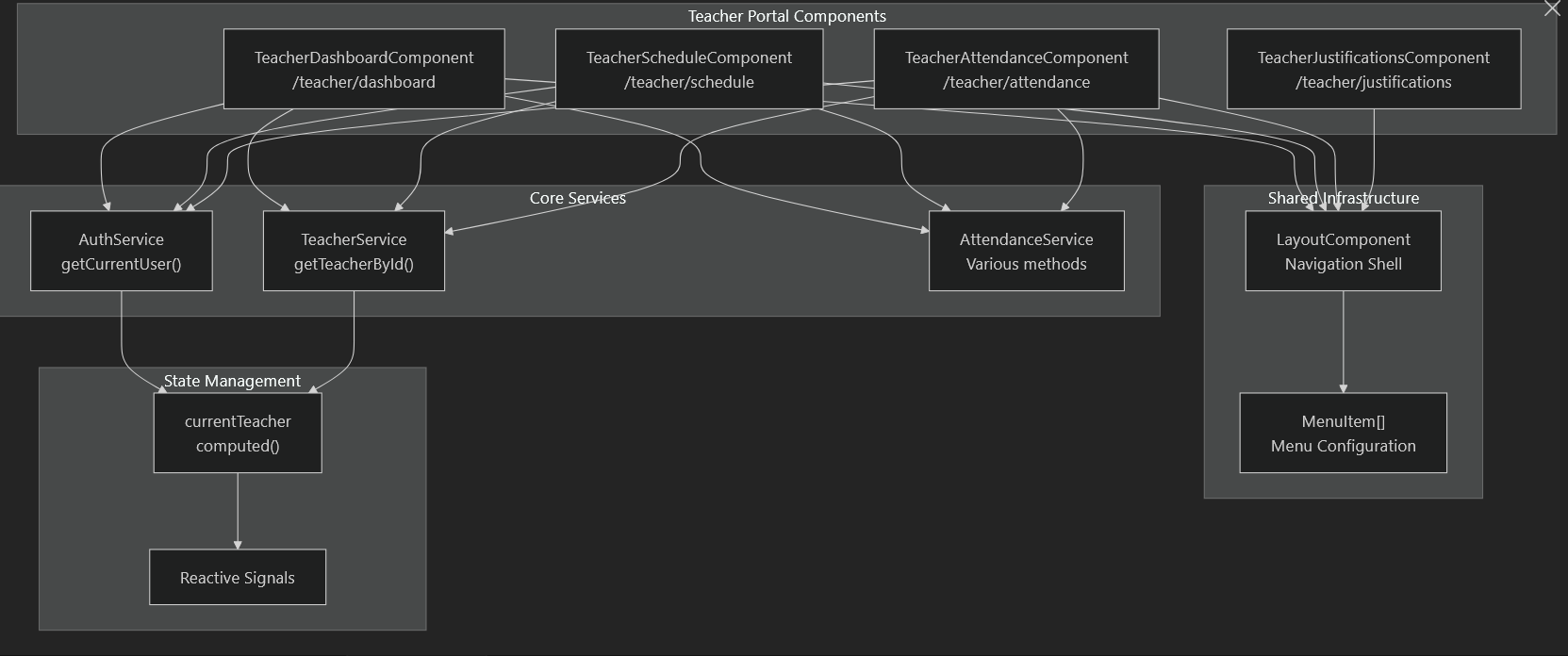
# Maestro Portal

El Portal de Maestro proporciona la interfaz completa orientada al profesorado para el sistema de gestión de asistencia. Permite a los profesores ver sus horarios, registrar la asistencia, rastrear su historial de asistencia y gestionar justificaciones de ausencias. El portal se organiza en distintas áreas funcionales accesibles a través de una estructura de navegación consistente.

## Arquitectura Portal

El Portal de Maestro se construye como una colección de componentes autónomos Angulares que comparten dependencias comunes de navegación y servicio. Todos los componentes siguen el mismo patrón arquitectónico con estructuras de menú consistentes y gestión estatal reactiva.

## Estructura de componentes de Portal de Maestro



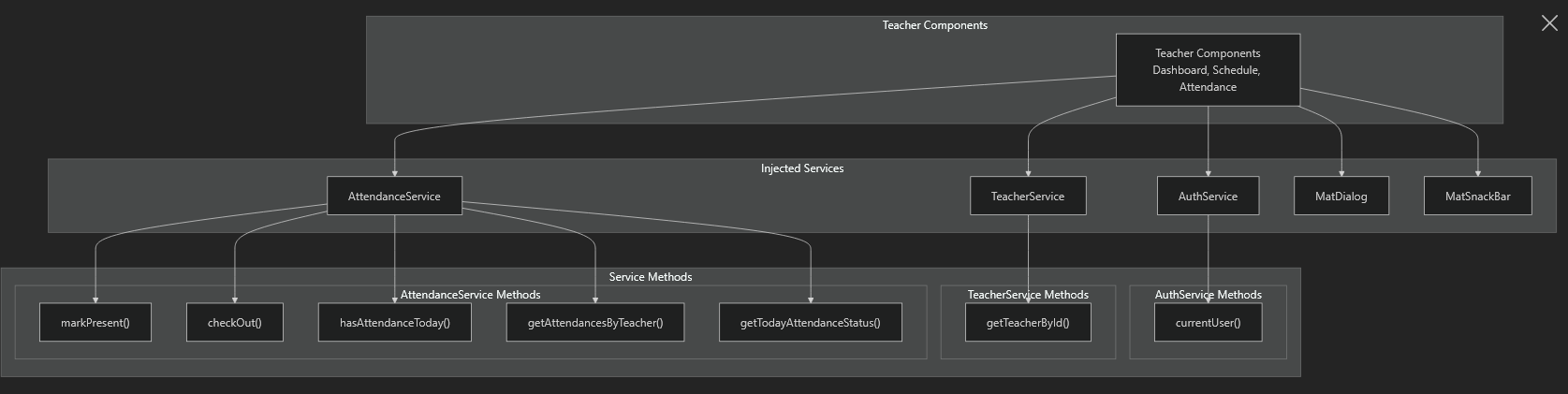
Fuentes: [src/app/features/teacher/dashboard/teacher-dashboard.component.ts1 a 42](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/dashboard/teacher-dashboard.component.ts#L1-L42) [src/app/features/teacher/schedule/teacher-schedule.component.ts51 - 67](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/teacher-schedule.component.ts#L51-L67) [src/app/features/teatracher/aconceance/teatra-attendance.component.ts46-56](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/attendance/teacher-attendance.component.ts#L46-L56)

## Sistema de navegación

Todos los componentes del portal del profesor comparten una estructura de menú de navegación consistente definida en su menuItems propiedad. La navegación proporciona acceso a cuatro áreas funcionales principales.

|  |  |  |  |
| --- | --- | --- | --- |
| Ruta | Etiqueta | Icono | Objeto |
| /teacher/dashboard | Dashboard | salpicadero | Resumen y acciones rápidas |
| /teacher/attendance | Asistencias | Asignación | Historial de asistencia y gestión |
| /teacher/justifications | Sólo unas penas. | descripción | Justificación sumisión y seguimiento |
| /teacher/schedule | Horarios | horario | Inscripción de horarios y asistencia |

## Dependencias de servicios



Fuentes: [src/app/features/teacher/dashboard/teacher-dashboard.component.ts32-35](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/dashboard/teacher-dashboard.component.ts#L32-L35) [src/app/features/teacher/schedule/teacher-schedule.component.ts52-56](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/teacher-schedule.component.ts#L52-L56) [src/app/features/teatracher/aconceance/teatra-attendance.component.ts101-103](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/attendance/teacher-attendance.component.ts#L101-L103)

## Resolución actual de maestros

Todos los componentes del portal del profesor utilizan el mismo patrón para resolver al actual maestro del usuario autenticado. Esto se implementa como una señal calculada que reacciona a los cambios de estado de autenticación.

A screenshot of a computer

AI-generated content may be incorrect.

Fuentes: [src/app/features/teacher/dashboard/teacher-dashboard.component.ts47 - 53](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/dashboard/teacher-dashboard.component.ts#L47-L53) [src/app/features/teacher/schedule/teacher-schedule.component.ts89-95](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/teacher-schedule.component.ts#L89-L95) [src/app/features/teatracher/aconceance/teatra-attendance.component.ts69-75](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/attendance/teacher-attendance.component.ts#L69-L75)

## Registro de asistencia Flujo de Registro

El portal del profesor implementa un flujo de registro de asistencia consistente a través de múltiples componentes. Los profesores pueden registrar la asistencia tanto en el salpicadero como en las vistas del horario.

A screenshot of a computer

AI-generated content may be incorrect.

Fuentes: [src/app/features/teacher/dashboard/teacher-dashboard.component.ts127-155](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/dashboard/teacher-dashboard.component.ts#L127-L155) [src/app/features/teacher/schedule/teacher-schedule.component.ts148 a 182](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/teacher-schedule.component.ts#L148-L182)

## Gestión estatal reactiva

Los componentes del portal del profesor utilizan señales angulares para la gestión estatal reactiva, asegurando que la interfaz de usuario se actualice automáticamente cuando los datos subyacentes cambian.

|  |  |  |
| --- | --- | --- |
| Componente | Señales clave computarizadas | Purpose |
| TeacherDashboardComponent | totalSchedules, monthlyAttendances, todaySchedules | Estadísticas del tablero y el horario de hoy |
| TeacherScheduleComponent | teacherSchedules, currentTeacher | Estado de visualización y asistencia |
| TeacherAttendanceComponent | teacherAttendances, filteredAttendances | Historial de asistencia con filtrado |

## Integración de componentes de la interfaz de usuario

Todos los componentes del portal del profesor se integran con los LayoutComponentpara proporcionar una navegación consistente y un diseño visual. También utilizan componentes de Diseño de Materiales para interacciones consistentes del usuario.

### Componentes de materiales comunes

|  |  |  |
| --- | --- | --- |
| Componente | Uso | Objeto |
| MatCardModule | Contenedores de contenido | Organizar información en tarjetas |
| MatButtonModule | Botones de acción | Registro de asistencia de desencadenante, navegación |
| MatIconModule | Indicadores visuales | iconos del estado, iconos de navegación |
| MatSnackBarModule | Retroalimentación del usuario | Mensajes de éxito/error |
| MatChipsModule | Indicadores de situación | Estado de asistencia, situación del calendario |

Fuentes: [src/app/features/teacher/dashboard/teacher-dashboard.component.ts18-27](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/dashboard/teacher-dashboard.component.ts#L18-L27) [src/app/features/teacher/schedule/teacher-schedule.component.ts37-46](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/teacher-schedule.component.ts#L37-L46) [src/app/features/teatracher/aconceance/teatra-attendance.component.ts30 a 42](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/attendance/teacher-attendance.component.ts#L30-L42)

## Manipulación de errores y alimentación de usuario

El portal del profesor implementa patrones de manejo de errores y retroalimentación del usuario en todos los componentes. Las acciones proporcionan información inmediata a través de MatSnackBarnotificaciones con estilo adecuado.



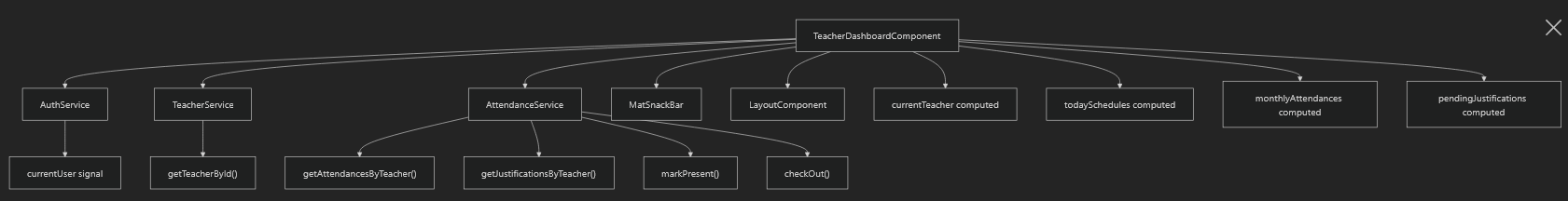
Fuentes: [src/app/features/teacher/dashboard/teacher-dashboard.component.ts129-155](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/dashboard/teacher-dashboard.component.ts#L129-L155) [src/app/features/teacher/schedule/teacher-schedule.component.ts148 a 182](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/teacher-schedule.component.ts#L148-L182)

# Maestro Dashboard

El Panel de Maestro sirve como interfaz primaria para que los maestros administren su asistencia diaria, horarios y acceso a características específicas del profesor. Este componente ofrece una visión completa del estado actual del maestro, las clases de hoy y el acceso rápido a otras características del portal de maestros.

## Arquitectura de componentes

El TeacherDashboardComponent es un componente independiente Angular que sirve como la principal página de aterrizaje para los maestros autenticados. Se integra con múltiples servicios básicos para proporcionar una experiencia unificada del salpicadero.



**Fuentes:** [src/app/features/teacher/dashboard/teacher-dashboard.component.ts1 a 42](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/dashboard/teacher-dashboard.component.ts#L1-L42)

## Características clave

### Secretarías Dashboard

El salpicadero muestra tres tarjetas de estadísticas clave que proporcionan a los maestros una visión de un vistazo de su estado actual:

|  |  |  |
| --- | --- | --- |
| Estatística | Descripción | Propiedad computada |
| Cuadros totales | Número total de horarios asignados | totalSchedules |
| Asistencias mensuales | Número de asistencias actuales este mes | monthlyAttendances |
| A la espera de justificaciones | Número de justificaciones pendientes de aprobación | pendingJustifications |

### La gestión de horarios de hoy

El salpicadero muestra de manera prominente las clases de hoy con plenas capacidades de gestión de asistencia:

* **Horario Mostrar descripción:** Especenas de la materia, el tiempo y el aula para cada clase
* **Registro de asistencia** : Marcado de asistencia de un solo clic para clases individuales
* **Registro a granel** : Opción para registrar la asistencia a todas las clases de hoy
* **Gestión de salida** : Capacidad para marcar la finalización de la clase
* **Seguimiento de estado** : Muestra en tiempo real de estado de asistencia con marcas de tiempo

### Panel de medidas rápidas

El salpicadero proporciona una navegación rápida a otras características del portal del profesor:

* Ver todos los registros de asistencia
* Acceso a información completa del calendario
* Gestionar justificaciones
* Toggle debug información

**Fuentes:**

[src/app/features/teacher/dashboard/teacher-dashboard.component.html53 a 228](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/dashboard/teacher-dashboard.component.html#L53-L228)

## Arquitectura de flujo de datos

El salpicadero implementa un flujo de datos reactivo usando señales angulares para asegurar actualizaciones en tiempo real:



**Fuentes:** [src/app/features/teacher/dashboard/teacher-dashboard.component.ts47 - 102](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/dashboard/teacher-dashboard.component.ts#L47-L102) [src/app/features/teacher/dashboard/teacher-dashboard.component.ts127 - 174](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/dashboard/teacher-dashboard.component.ts#L127-L174)

## Estructura de interfaz de usuario

### Integración de diseño

El salpicadero utiliza el compartido LayoutComponentcon una estructura de menú predefinida:

  menuItems: MenuItem[] = [

    { label: "Dashboard", route: "/teacher/dashboard", icon: "dashboard" },

    { label: "Asistencias", route: "/teacher/attendance", icon: "assignment" },

    { label: "Justificaciones", route: "/teacher/justifications", icon: "description" },

    { label: "Horarios", route: "/teacher/schedule", icon: "schedule" },

  ]

### Arquitectura de la plantilla

La plantilla se organiza en distintas secciones:

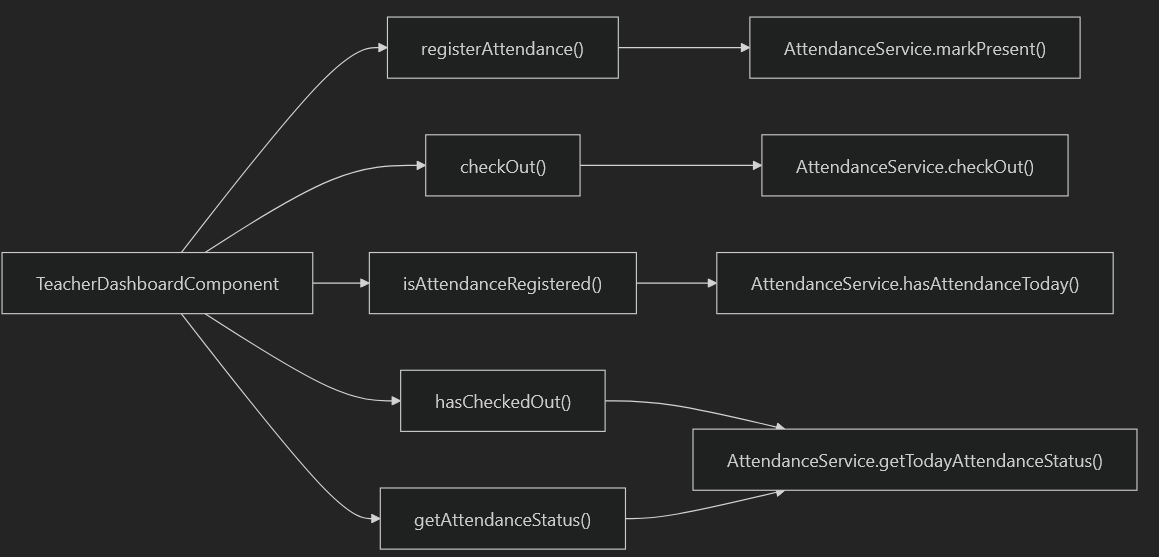
1. **Sección de encabezado:** Maestro bienvenido con información de perfil
2. **Panel de depuración** : Visual información de depuración condicional
3. **Tarjetas Estadísticas** : Rejilla de tres columnas de métricas clave
4. **Área de Contenido Principal** : Diseño de dos columnas con el horario de hoy y acciones rápidas
5. **Horario semanal** : Pantalón de ancho completo de todos los horarios semanales

**Fuentes:**

[src/app/features/teacher/dashboard/teacher-dashboard.component.ts37-42](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/dashboard/teacher-dashboard.component.ts#L37-L42) [src/app/features/teacher/dashboard/teacher-dashboard.component.html1 a 21](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/dashboard/teacher-dashboard.component.html#L1-L21)

## Integración de la gestión de asistencia

El salpicadero ofrece una gestión integral de asistencia a través de la integración directa con la AttendanceService:



### Actuación de Estado de asistencia

El salpicadero proporciona actualizaciones de estado de asistencia en tiempo real con los siguientes estados:

* **No registrado** : Estado por defecto, permite registro de asistencia
* **Presente** : Aseo registrado, muestra tiempo de check-in, permite el check-out
* **Completado** : Tanto el check-in y check-out registrado, muestra el estado de finalización

**Fuentes:** [src/app/features/teacher/dashboard/teacher-dashboard.component.ts127 a 213](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/dashboard/teacher-dashboard.component.ts#L127-L213)

## Características de depuración y desarrollo

El salpicadero incluye capacidades de depuración integradas controladas por el showDebugpropiedad:

* **Información del profesorado:** Muestra la identificación de profesor actual y la información básica
* **Análisis del horario:** Muestra el día y el número de horarios actuales
* **Asistencia Depuración de asistencia** : Proporciona producción de consola para los registros de asistencia
* **Control de movimiento** : Fácil habilitar/desactivación de la información de depuración

**Fuentes:**

[src/app/features/teacher/dashboard/teacher-dashboard.component.ts115 a 125](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/dashboard/teacher-dashboard.component.ts#L115-L125) [src/app/features/teacher/dashboard/teacher-dashboard.component.html22 a 51](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/dashboard/teacher-dashboard.component.html#L22-L51)

## Detalles de la ejecución técnica

### Estrategia de detección de cambios

El componente utiliza OnPushestrategia de detección de cambios para un rendimiento óptimo:

  changeDetection: ChangeDetectionStrategy.OnPush,

### Inyección de dependencia

Todos los servicios se inyectan usando Angular's inject()función:

  private authService: AuthService = inject(AuthService);

  private teacherService: TeacherService = inject(TeacherService);

  private attendanceService: AttendanceService = inject(AttendanceService);

  private snackBar: MatSnackBar = inject(MatSnackBar);

### Propiedades computadas

El salpicadero aprovecha las señales angulares con propiedades calculadas para actualizaciones de datos reactivas, asegurando que la interfaz de usuario se actualice automáticamente cuando los datos subyacentes cambian.

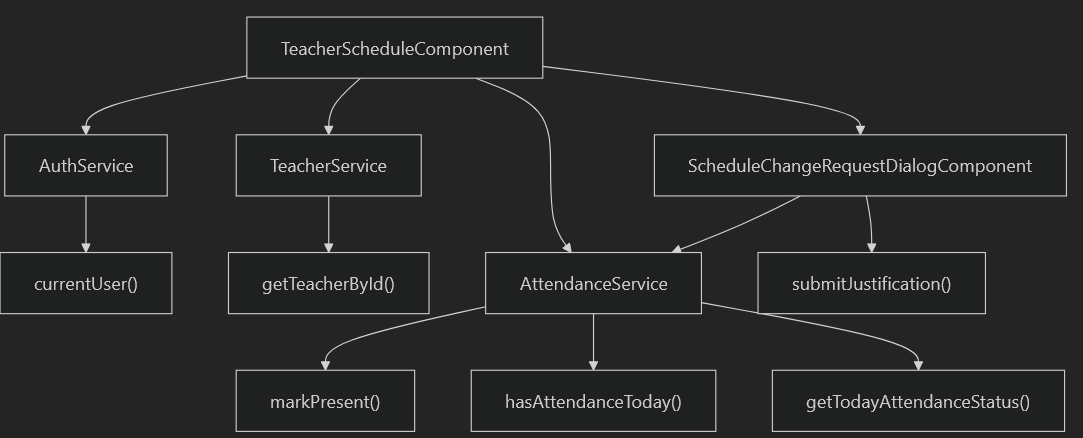
**Fuentes:** [src/app/features/teacher/dashboard/teacher-dashboard.component.ts15 a 35](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/dashboard/teacher-dashboard.component.ts#L15-L35)

# Gestión de los horarios

Este documento cubre el sistema de gestión de horarios orientado al profesorado que permite a los maestros ver sus horarios asignados, registrar asistencia y solicitar cambios en los horarios. Este sistema forma parte del portal del profesorado y proporciona la interfaz primaria para que los profesores interactúen con sus horarios de clase.

## Resumen del sistema

El sistema de gestión de horarios proporciona a los profesores dos puntos de vista principales de sus datos del calendario: una vista semanal del calendario y una visión detallada de la tabla. Los maestros pueden registrar la asistencia para sus clases, ver estadísticas de asistencia y presentar solicitudes de cambios de horario a través de esta interfaz.



**Fuentes:** [src/app/features/teacher/schedule/teacher-schedule.component.ts26-28](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/teacher-schedule.component.ts#L26-L28) [src/app/features/teacher/schedule/teacher-schedule.component.ts52 - 56](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/teacher-schedule.component.ts#L52-L56)

## Componentes básicos

### MaestroScheduleComponent

El componente principal que orquesta la interfaz de gestión del horario. Proporciona propiedades calculadas para la visualización de datos reactivos y métodos para el registro de asistencia y solicitudes de cambio de horario.

|  |  |  |
| --- | --- | --- |
| Bienes | Tipo | Descripción |
| currentTeacher | Computado, maestro. | null. |
| teacherSchedules | computed<Schedule[]> | Clasificación de los horarios de los maestros |
| displayedColumns | string[] | Configuración de columnas de tabla |
| weekDays | {name: string, value: number}[] | Definiciones del día de la semana |

**Fuentes:** [src/app/features/teacher/schedule/teacher-schedule.component.ts89 a 106](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/teacher-schedule.component.ts#L89-L106) [src/app/features/teacher/schedule/teacher-schedule.component.ts69 - 87](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/teacher-schedule.component.ts#L69-L87)

### ScheduleChangeRequestDialogComponent

Componente de diálogo modificador para la presentación de solicitudes de cambio de horario. Las solicitudes se tramitan como justificaciones especiales con un prefijo "CAMBIO DE HORARIO".

|  |  |  |
| --- | --- | --- |
| Bienes | Tipo | Descripción |
| schedule | Schedule | Calendario que se modifica |
| teacherId | string | DNI de solicitar profesor |
| requestForm | FormGroup | Forma con fecha, solicitudTipo, y campos de razón |

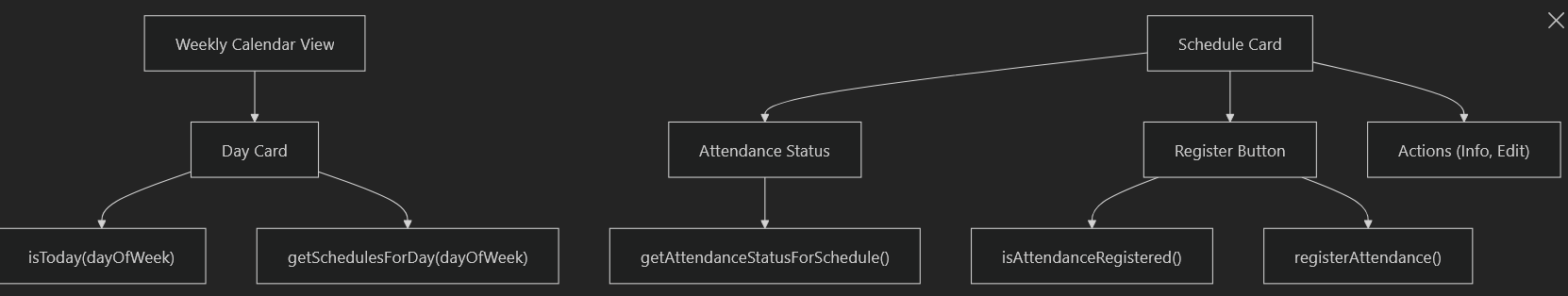
**Fuentes:** [src/app/features/catracado/gramedule/schedule-change-request-dialog.component.ts70-79](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/schedule-change-request-dialog.component.ts#L70-L79)

## Sistema de visualización de horarios

El sistema ofrece dos puntos de vista complementarios de los datos del calendario:

### Vista semanal del calendario

Muestra horarios organizados por día de la semana con énfasis visual en el día actual. Cada tarjeta programa muestra el estado de la materia, el tiempo, el aula y el estado de asistencia.



**Fuentes:** [src/app/features/catracado/escuela/teatracher-schedule.component.html1498](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/teacher-schedule.component.html#L14-L98) [src/app/features/teacher/schedule/teacher-schedule.component.ts108-118](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/teacher-schedule.component.ts#L108-L118)

### Vista de mesa detallada

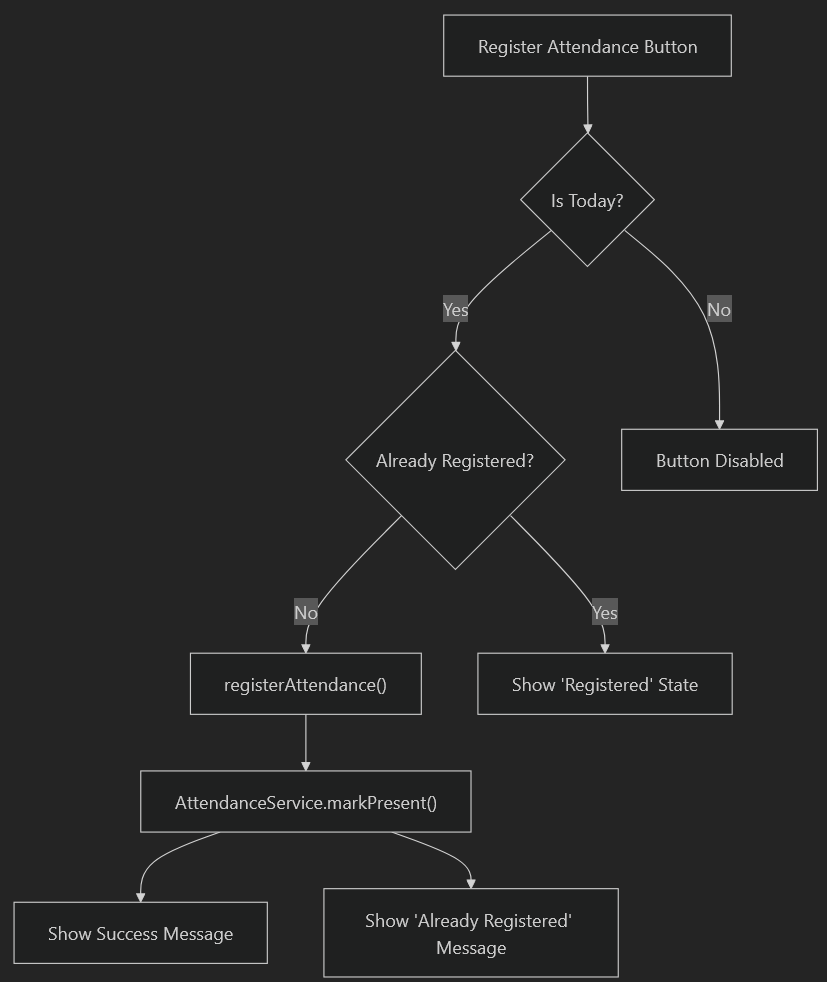
Proporciona una tabla completa con todas las acciones de información y gestión del calendario. Utiliza componentes de la mesa de diseño de material con capacidades de clasificación y filtrado.

**Fuentes:** [src/app/features/catracado/escuela/teatracher-schedule.component.html112 a 234](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/teacher-schedule.component.html#L112-L234) [src/app/features/teacher/schedule/teacher-schedule.component.ts69-77](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/teacher-schedule.component.ts#L69-L77)

## Registro de asistencia

El sistema se integra con AttendanceService para proporcionar inscripción de asistencia de un clic para las clases de hoy.

### Flujo de registro



**Fuentes:** [src/app/features/teacher/schedule/teacher-schedule.component.ts148 a 182](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/teacher-schedule.component.ts#L148-L182)

### Actuación de Estado de asistencia

El sistema muestra el estado de asistencia en tiempo real usando propiedades calculadas:

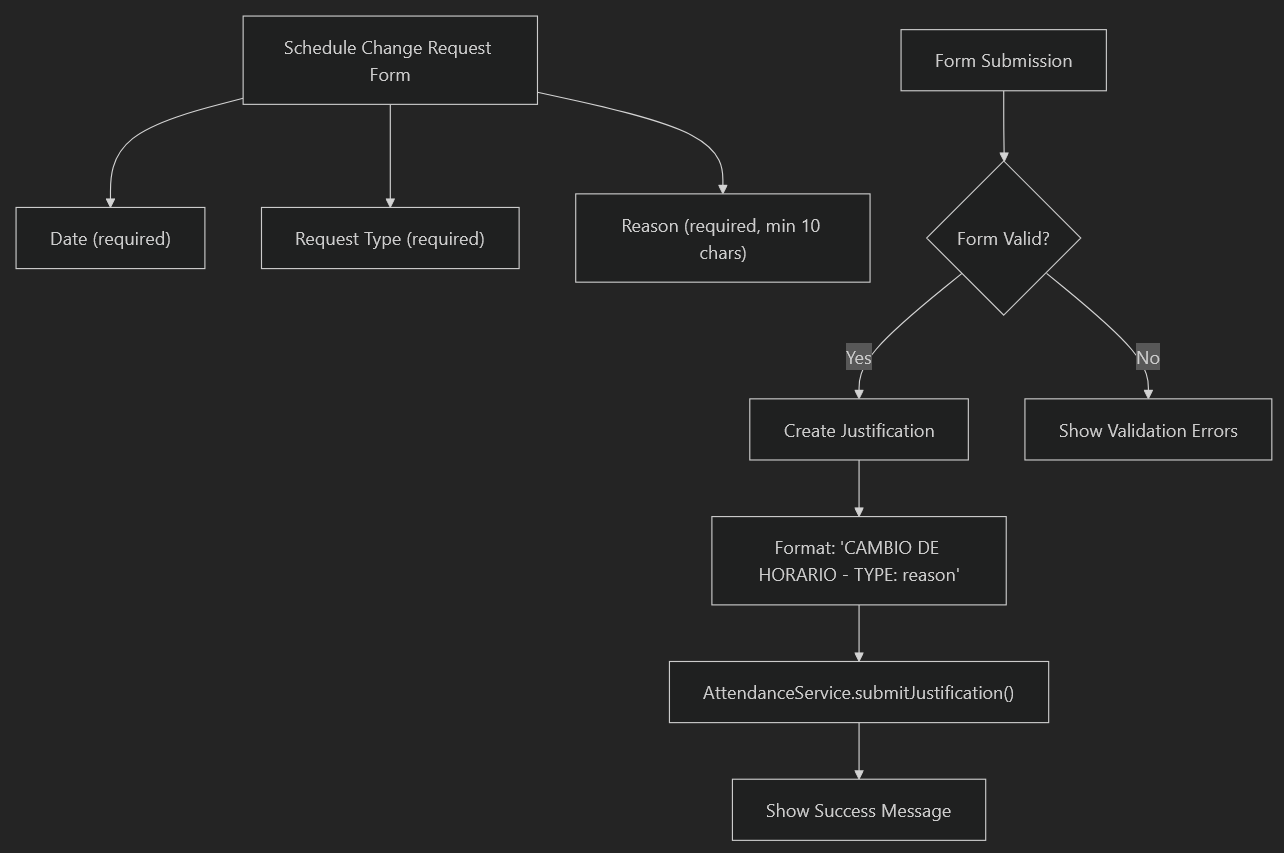
|  |  |  |
| --- | --- | --- |
| Método | Tipo de retorno | Descripción |
| isAttendanceRegistered() | boolean | Comprobar si la asistencia se registra para hoy |
| getAttendanceStatusForSchedule() | {text: string, color: string, icon: string} | Devuelve indicador de estado visual |
| getAttendanceCount() | number | Devuelve el conteo total de asistencia para el horario |

**Fuentes:** [src/app/features/teacher/schedule/teacher-schedule.component.ts141 a 146](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/teacher-schedule.component.ts#L141-L146) [src/app/features/teacher/schedule/teacher-schedule.component.ts214-240](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/teacher-schedule.component.ts#L214-L240) [src/app/features/teacher/schedule/teacher-schedule.component.ts133-139](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/teacher-schedule.component.ts#L133-L139)

## Sistema de Solicitud de Cambio de Horario

Los maestros pueden solicitar cambios de horario a través de un diálogo modal que crea entradas de justificación especiales.

### Solicitar estructura de formulario



**Fuentes:** [src/app/features/catracado/gramedule/schedule-change-request-dialog.component.ts75 a 79](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/schedule-change-request-dialog.component.ts#L75-L79) [src/app/features/catracado/gramedule/schedule-change-request-dialog.component.ts94-118](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/schedule-change-request-dialog.component.ts#L94-L118)

### Solicitud de procesamiento

Las solicitudes de cambio de calendario se tramitan como justificaciones especiales con un formato estandarizado:

      const reason = `CAMBIO DE HORARIO - ${formValue?.requestType?.toUpperCase()}: ${

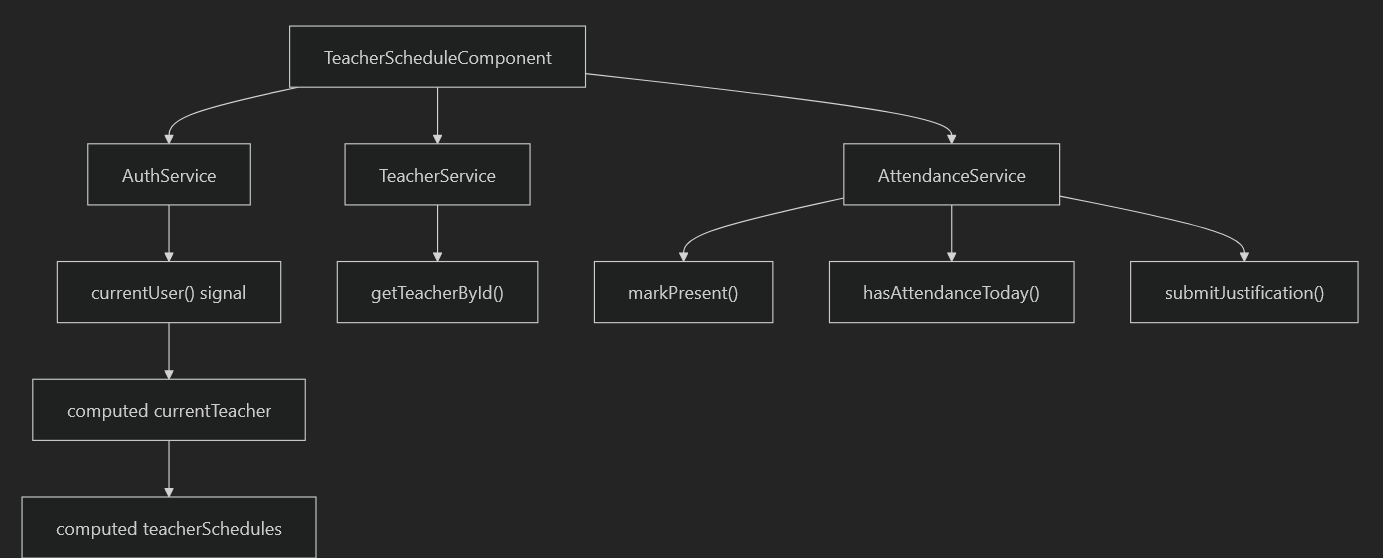
        formValue.reason

      }`;

## Integración de flujo de datos

El sistema de gestión de horarios se integra con los servicios básicos mediante patrones reactivos:

### Dependencias de servicios



**Fuentes:** [src/app/features/teacher/schedule/teacher-schedule.component.ts52 - 56](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/teacher-schedule.component.ts#L52-L56) [src/app/features/teacher/schedule/teacher-schedule.component.ts89 a 106](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/teacher-schedule.component.ts#L89-L106)

## Componentes de interfaz de usuario

### Integración de la navegación

La interfaz de gestión de horarios se integra con el trazado del portal del profesor a través de la LayoutComponent:

|  |  |  |  |
| --- | --- | --- | --- |
| Menú artículo | Ruta | Icono | Descripción |
| Dashboard | /teacher/dashboard | dashboard | Página de inicio de profesor |
| Asistencias | /teacher/attendance | assignment | Historia de asistencia |
| Sólo unas penas. | /teacher/justifications | description | Gestión de la justificación |
| Horarios | /teacher/schedule | schedule | Gestión de la lista (actual) |

**Fuentes:** [src/app/features/teacher/schedule/teacher-schedule.component.ts58 - 67](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/teacher-schedule.component.ts#L58-L67)

### Componentes de diseño de materiales

La interfaz utiliza componentes de Material angular para un estilo consistente:

* MatCardModule for content containers
* MatTableModule for detailed schedule listing
* MatButtonModule for actions
* MatDialogModule for schedule change requests
* MatChipsModule for status indicators

**Fuentes:**

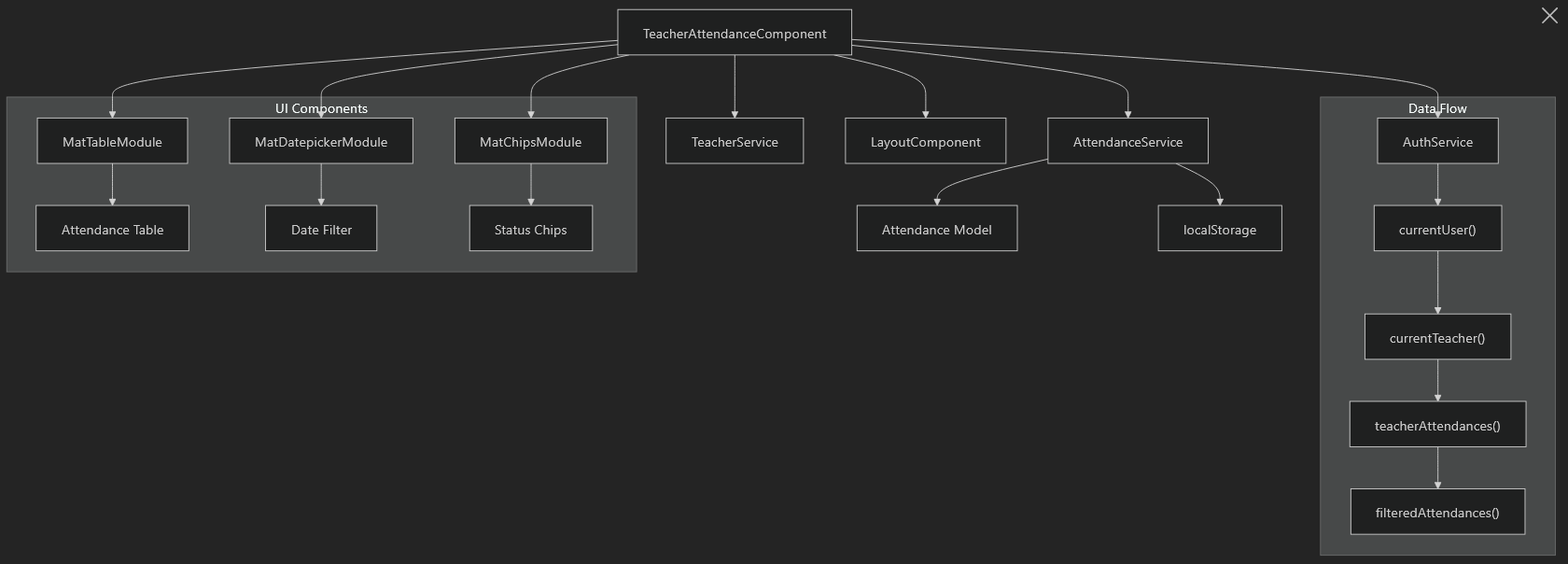
[src/app/features/teacher/schedule/teacher-schedule.component.ts37 a 46](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/teacher-schedule.component.ts#L37-L46)

# Seguimiento de la asistencia

Este documento cubre la interfaz de seguimiento de asistencia orientada al maestro dentro del Portal del Maestro. El sistema permite a los maestros ver su historial de asistencia, filtrar registros por fecha y gestionar acciones de check-in/check-out para sus clases programadas.

## Arquitectura de componentes

El sistema de seguimiento de asistencia se construye alrededor de la TeacherAttendanceComponentque sirve de interfaz primaria para que los profesores vean y gestionen sus registros de asistencia.[Asistencia](https://app.devin.ai/wiki/lesquel/proyecto-asistencia-docentes-ng#5.4).



**Arcotectura de Componentes de Asistencia al Maestro**

El componente utiliza el enfoque reactivo de Angular con señales calculadas para gestionar el flujo de datos y actualizar automáticamente la interfaz de usuario cuando los datos subyacentes cambian.

Fuentes:

[src/app/features/teatracher/aconceance/teatra-attendance.component.ts1 a 1511](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/attendance/teacher-attendance.component.ts#L1-L151)

## Flujo de datos y gestión del Estado

El sistema de seguimiento de asistencia sigue un patrón de flujo de datos reactivo usando señales angulares y propiedades calculadas.

A screenshot of a computer

AI-generated content may be incorrect.

**Asistencia Data Flow**

El sistema mantiene el estado reactivo a través de señales calculadas que se recalculan automáticamente cuando las dependencias cambian, asegurando que la interfaz de usuario permanece sincronizada con los datos subyacentes.

Fuentes:

[src/app/features/teatracher/aconceance/teatra-attendance.component.ts69-99](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/attendance/teacher-attendance.component.ts#L69-L99)

## Características de Interfaz de Usuario

### Lista de asistencia

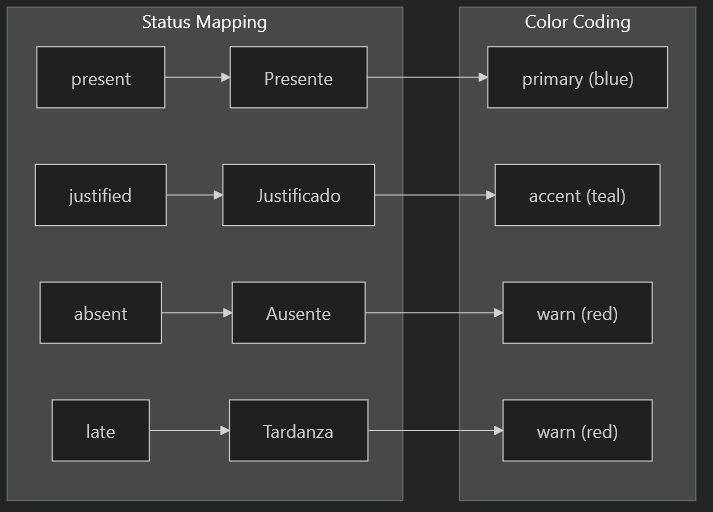
La interfaz principal presenta registros de asistencia en un formato tabular con las siguientes columnas:

|  |  |  |
| --- | --- | --- |
| Column | Purpose | Data Source |
| date | Class date | attendance.date |
| subject | Subject name | schedule.subject via getScheduleInfo() |
| time | Class time | schedule.startTime - schedule.endTime |
| checkIn | Marca de check-in | attendance.checkIn |
| checkOut | Temporada de check-out | attendance.checkOut |
| status | Situación de asistencia | attendance.status |
| actions | Acciones disponibles | Aviso de salida condicional |

Fuentes: [src/app/features/teatracher/aconceance/teatra-attendance.component.ts58-66](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/attendance/teacher-attendance.component.ts#L58-L66)

### Sistema de visualización de estado

El componente implementa un sistema de mapeo de estado que traduce códigos de estado internos a texto español fácil de usar:



**Muestra de estado Lúbrica de la Lí**

Fuentes: [src/app/features/teatracher/aconceance/teatra-attendance.component.ts119 a 138](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/attendance/teacher-attendance.component.ts#L119-L138)

### Fecha de filtración

El sistema proporciona funcionalidad de filtrado basada en fechas:

A diagram of a computer

AI-generated content may be incorrect.

**Corriente de filtro de fecha**

El filtro compara las fechas normalizando tanto la fecha del filtro como las fechas de asistencia hasta la medianoche (00:00:00) para asegurar una coincidencia precisa a nivel de día.

Fuentes:

[src/app/features/teatracher/aconceance/teatra-attendance.component.ts85-111](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/attendance/teacher-attendance.component.ts#L85-L111)

## Funcionalidad de check-Out

Los maestros pueden salir de las clases en las que ya han registrado, proporcionando un ciclo de vida completo de asistencia:

A screenshot of a computer

AI-generated content may be incorrect.

**Flujo de proceso de check-Out**

Fuentes: [src/app/features/teatracher/aconceance/teatra-attendance.component.ts140-149](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/attendance/teacher-attendance.component.ts#L140-L149)

## Integración con Servicios Básicos

El componente de seguimiento de la asistencia se integra con tres servicios básicos:

### AsistenciaService Integración

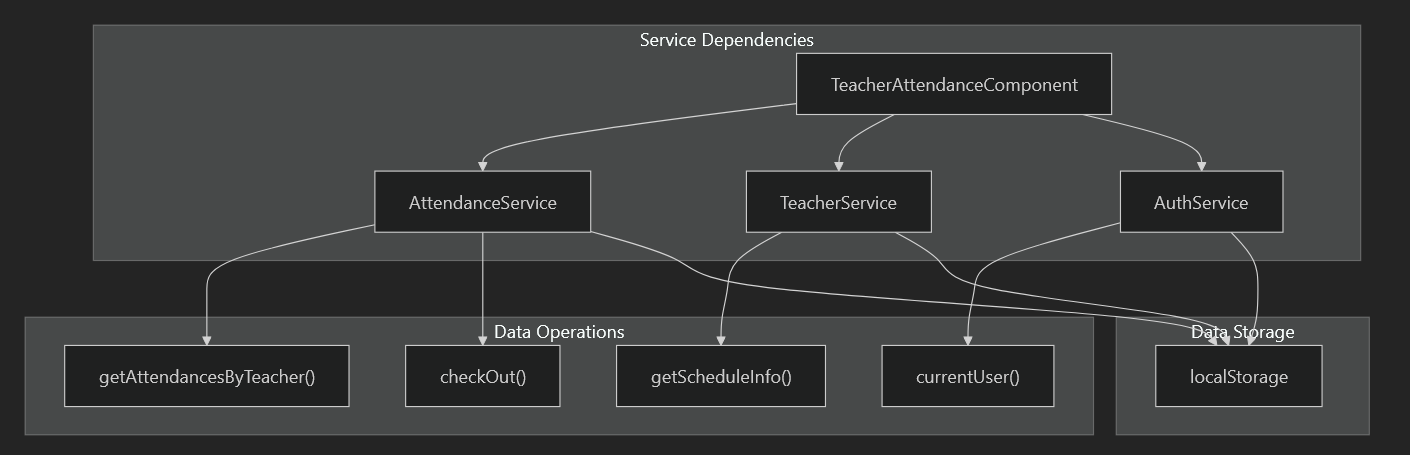
* **Recuperación de datos**: getAttendancesByTeacher() Atrapa registros específicos de maestros
* **Operaciones de salida**: checkOut() actualizaciones de los registros de asistencia
* **Gestión de la situación**: Proporciona información sobre el estado de asistencia

### Integración docente

* **Información del calendario** getScheduleInfo()recupera detalles de la clase
* **Datos del** Maestro: Accede a asignaciones de horarios de maestros

### Integración de austro vertebración

* **Contexto de Usuario:** currentUser()proporciona información autenticada del profesor
* **Control de acceso** : Garantiza que los maestros solo accedan a sus propios datos de asistencia



**Arquitectura de Integración de Servicios**

Fuentes:

[src/app/features/teatracher/aconceance/teatra-attendance.component.ts101-103](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/attendance/teacher-attendance.component.ts#L101-L103) [src/app/core/services/attendance.service.ts1-234](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/attendance.service.ts#L1-L234)

## Navegación y diseño

El componente utiliza el LayoutComponentcon una estructura de menú específica del profesor:

* **Dashboard:** /teacher/dashboard
* **Asistencias:** /teacher/attendance(página corriente)
* **Justificaciones** : /teacher/justifications
* **Horarios:** /teacher/schedule

Esto proporciona una navegación consistente en todo el portal del profesorado manteniendo el contexto actual de la página.

Fuentes: [src/app/features/teatracher/aconceance/teatra-attendance.component.ts47 - 56](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/attendance/teacher-attendance.component.ts#L47-L56)

# Justifications

Esta sección cubre el sistema de justificaciones del profesorado, que permite a los maestros presentar justificaciones de ausencias y rastrear su estatus. El sistema proporciona un flujo de trabajo completo desde la presentación a la revisión administrativa.

## Resumen del sistema

El sistema de justificaciones permite a los profesores:

* Presentar justificaciones de ausencias con fecha y razón
* Ver todas sus justificaciones y su estado actual
* Rastrear el proceso de revisión incluyendo marcas de tiempo de aprobación/reyección

A diagram of a teacher

AI-generated content may be incorrect.

Fuentes: [src/app/features/caestor/justificaciones/caestro/caestro.componente.ts1-104](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/justifications/teacher-justifications.component.ts#L1-L104) [src/app/features/maestro/justificaciones/justificación-form-dialog.s1 a 64](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/justifications/justification-form-dialog.component.ts#L1-L64) [src/app/core/services/attendance.service.ts197 a 224](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/attendance.service.ts#L197-L224)

## Interfaz de maestro

El TeacherJustificationsComponentproporciona la interfaz principal para que los profesores gestionen sus justificaciones. Muestra una tabla de todas las justificaciones presentadas con su estado actual.

### Estructura de componentes

|  |  |
| --- | --- |
| Feature | Implementation |
| Component | TeacherJustificationsComponent |
| Template | teacher-justifications.component.html |
| Navegación | Integrated with LayoutComponent |
| Fuente de datos | AttendanceService.getJustificationsByTeacher() |

### Mostrar tabla

El componente muestra justificaciones en una tabla con las siguientes columnas:

|  |  |  |
| --- | --- | --- |
| Columna | Descripción | Fuente de datos |
| date | Fecha de ausencia | AttendanceJustification.date |
| reason | Justificación razón | AttendanceJustification.reason |
| status | Situación actual | AttendanceJustification.status |
| createdAt | Timestamp de envío | AttendanceJustification.createdAt |
| reviewedAt | Calendario de revisión | AttendanceJustification.reviewedAt |

### Gestión de la situación

El componente incluye métodos para mostrar el estado de justificación:

* getStatusText() - Convierte códigos de estado al texto en español
* getStatusColor() - Mapas de esquemas de color de Diseño de Material

A diagram of a process

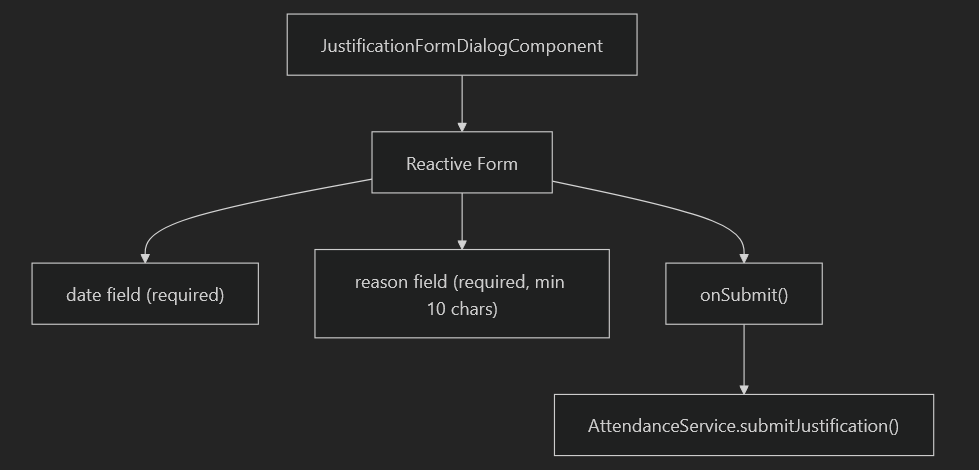
AI-generated content may be incorrect.

Fuentes: [src/app/features/caestor/justificaciones/caestro/caestro.componente.ts83 - 103](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/justifications/teacher-justifications.component.ts#L83-L103)

## Presentación de justificación

El JustificationFormDialogComponentmaneja la creación de nuevas justificaciones a través de un diálogo de forma reactiva.

### Estructura de la forma



### Validación de la forma

|  |  |
| --- | --- |
| **Campo** | **Reglas de validación** |
| date | Requerido |
| reason | Requerido, mínimo 10 caracteres |

### Proceso de presentación

El proceso de presentación de formularios implica:

1. Validación de formularios con formularios reactivos angulares
2. Transformación de datos para crear objeto de justificación
3. Llamadas AttendanceService.submitJustification()
4. Cierre de diálogo sobre presentación exitosa

Fuentes:

[src/app/features/maestro/justificaciones/justificación-form-dialog.s42 a 59](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/justifications/justification-form-dialog.component.ts#L42-L59)

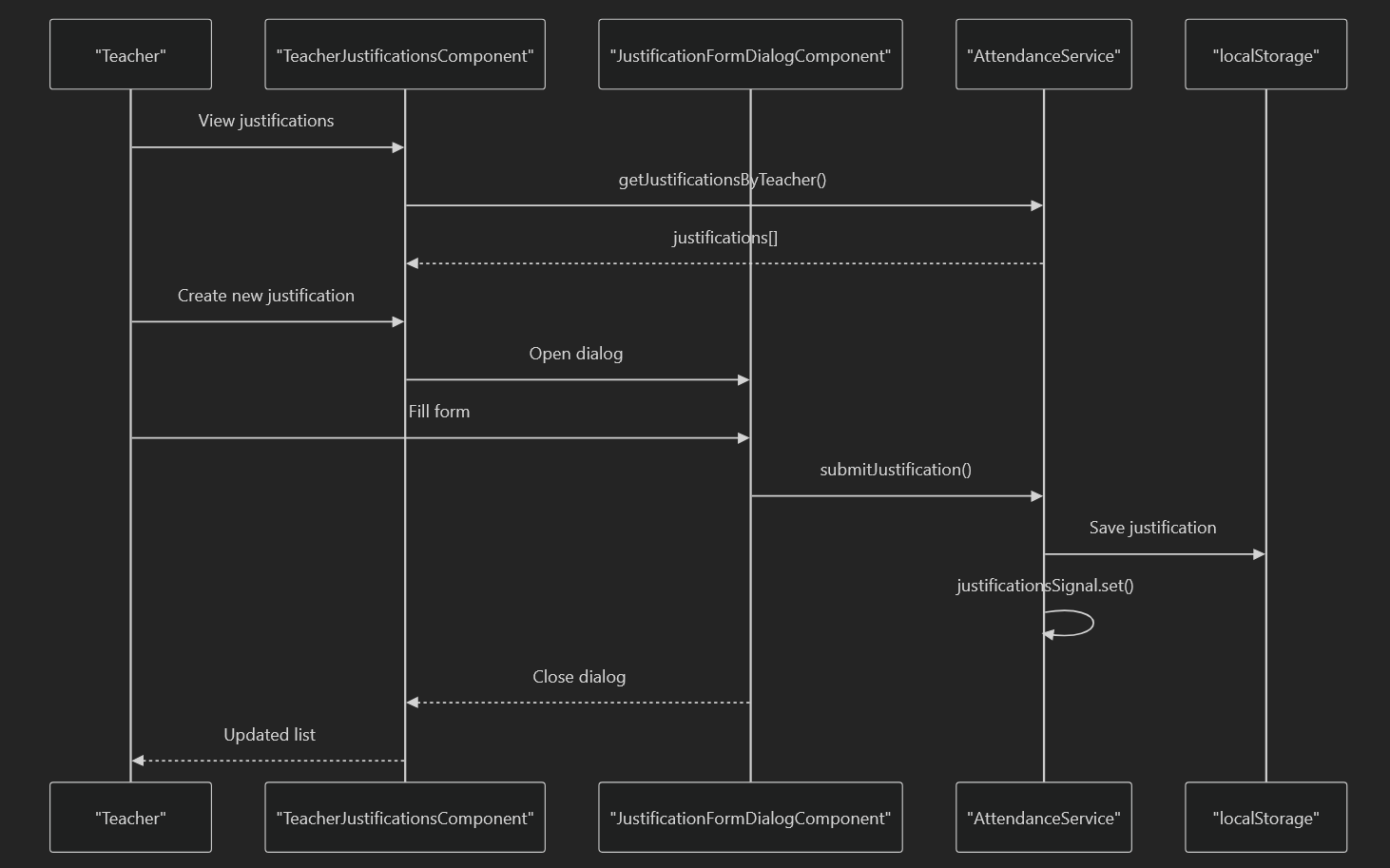
## Integración de servicios de apoyo

El AttendanceServiceproporciona la funcionalidad básica para la gestión de la justificación a través de varios métodos clave.

### Métodos de justificación

|  |  |  |
| --- | --- | --- |
| Método | Objeto | Parámetros |
| submitJustification() | Crear una nueva justificación | teacherId, date, reason |
| getJustificationsByTeacher() | Recuperar justificaciones de profesor | teacherId |
| reviewJustification() | Medidas de revisión administrativa | id, status, reviewedBy |
| getPendingJustifications() | Obtenga revisiones pendientes | Ninguno |

### Flujo de datos



Fuentes: [src/app/core/services/attendance.service.ts197 a 224](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/attendance.service.ts#L197-L224)

## Modelo de datos

El sistema de justificación utiliza la AttendanceJustificationinterfaz con la estructura siguiente:

### AsistenciaSustificación Propiedades

|  |  |  |  |
| --- | --- | --- | --- |
| Bienes | Tipo | Descripción | Requerido |
| id | string | Identificado único | Autogenerado |
| teacherId | string | Identificador del profesor | Sí, |
| date | Date | Fecha de ausencia | Sí, |
| reason | string | Justificación razón | Sí, |
| status | Dependiendo | "aprobado". | "rechazado". |
| createdAt | Date | Timestamp de envío | Autogenerado |
| reviewedAt | Date | Calendario de revisión | Opcional |
| reviewedBy | string | Identificación del revisor administrativo | Opcional |

### Estado ciclo de vida

Fuentes: [src/app/core/models/user.model.ts](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/models/user.model.ts) [src/app/core/services/attendance.service.ts197 a -18](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/attendance.service.ts#L197-L216)

## Persistencia de los datos

El sistema de justificación utiliza LocalStorage para la persistencia de datos con serialización/desiernización automática:

### Operaciones de almacenamiento

|  |  |  |
| --- | --- | --- |
| Operación | Método | Clave de almacenamiento |
| Salvar | saveJustifications() | "justifications" |
| Carga | loadData() | "justifications" |
| Actualización | Actualizaciones reactivas basadas en señales | Automático |

### Serialización de datos

El servicio maneja la serialización de fecha automáticamente al cargar desde localStorage, la conversión de la cadena se remonta a los objetos de Fecha para:

* date
* createdAt
* reviewedAt(opcional)

Fuentes: [src/app/core/services/attendance.service.ts39 a 66](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/attendance.service.ts#L39-L66)

## Puntos de integración

### Navegación portal docente

La función de justificaciones se integra con la navegación del portal del profesor a través de elementos de menú:

  menuItems: MenuItem[] = [

    { label: 'Dashboard', route: '/teacher/dashboard', icon: 'dashboard' },

    { label: 'Asistencias', route: '/teacher/attendance', icon: 'assignment' },

    {

      label: 'Justificaciones',

      route: '/teacher/justifications',

      icon: 'description',

    },

    { label: "Horarios", route: "/teacher/schedule", icon: "schedule" },

  ];

### Integración de autenticación

El componente utiliza AuthServiceidentificar adecuadamente al profesor actual y filtrar justificaciones adecuadas a través de la currentTeacherPropiedad calculada.

Fuentes:

[src/app/features/caestor/justificaciones/caestro/caestro.componente.ts33-43](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/justifications/teacher-justifications.component.ts#L33-L43) [src/app/features/caestor/justificaciones/caestro/caestro.componente.ts53 - 56](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/justifications/teacher-justifications.component.ts#L53-L56)

# Componentes compartidos

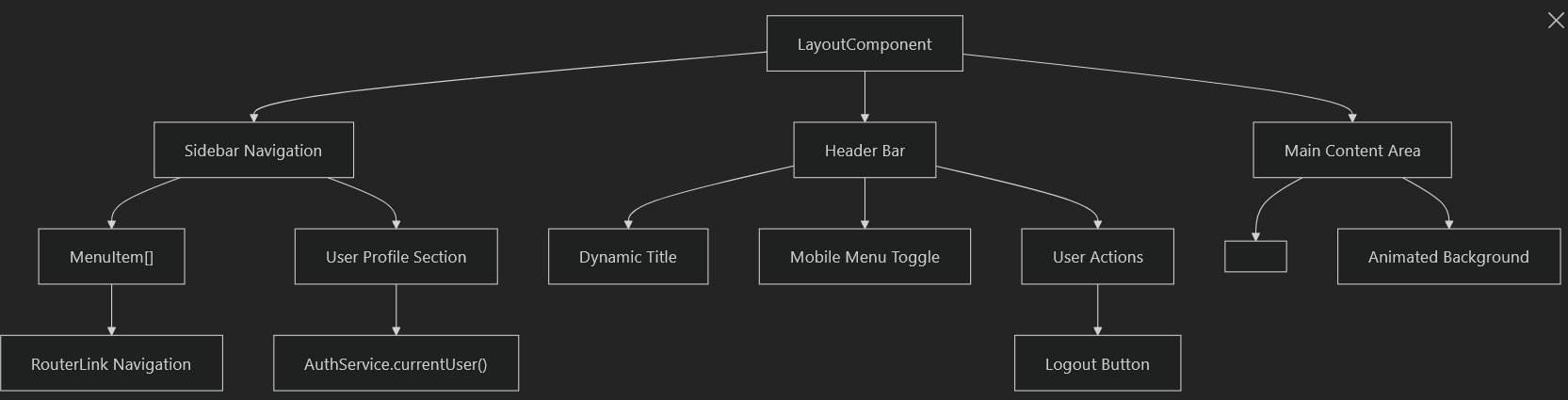
Este documento cubre los componentes de IU reutilizables y el sistema de diseño utilizado en toda la aplicación de gestión de la asistencia al profesorado. Estos componentes proporcionan un diseño visual consistente, funcionalidad común y patrones de código mantenibles en los portales de administración y de los profesores.

## Sistema de diseño

La aplicación utiliza un sistema de diseño centralizado construido alrededor de la LayoutComponent que proporciona la principal estructura de shell de aplicación, navegación y interfaz de usuario.

### LayoutComponent Architecture

El LayoutComponent sirve como la cáscara de aplicación principal con la navegación, encabezado y área de contenido de la barra lateral sensible. Acepta artículos de menú configurables y títulos a través de entradas angulares.



El componente de diseño implementa un diseño sensible con:

* Fixed sidebar for desktop, collapsible for mobile
* Gradient backgrounds and backdrop blur effects
* Animated background patterns
* Integración con componentes de materiales angulares

Fuentes:

[src/app/shared/components/layout/layout.component.ts1-74](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/shared/components/layout/layout.component.ts#L1-L74) [src/app/shared/components/layout/layout.component.html1 a 116](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/shared/components/layout/layout.component.html#L1-L116)

### Interfaz de MenuItem

El sistema de diseño utiliza un sistema estandarizado MenuItem interfaz para la configuración de navegación:

|  |  |  |
| --- | --- | --- |
| Property | Type | Descripción |
| label | string | Mostrar texto para el elemento del menú |
| route | string | Ruta angular |
| icon | string | Nombre del icono de material |

Fuentes: [src/app/shared/components/layout/layout.component.ts11 a 15](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/shared/components/layout/layout.component.ts#L11-L15)

### Patrón de integración de diseño

Los componentes se integran con el sistema de diseño envolviendo el contenido en el app-layout componente:

<app-layout *[title]*="'Mis Justificaciones'" *[menuItems]*="menuItems">

</app-layout>

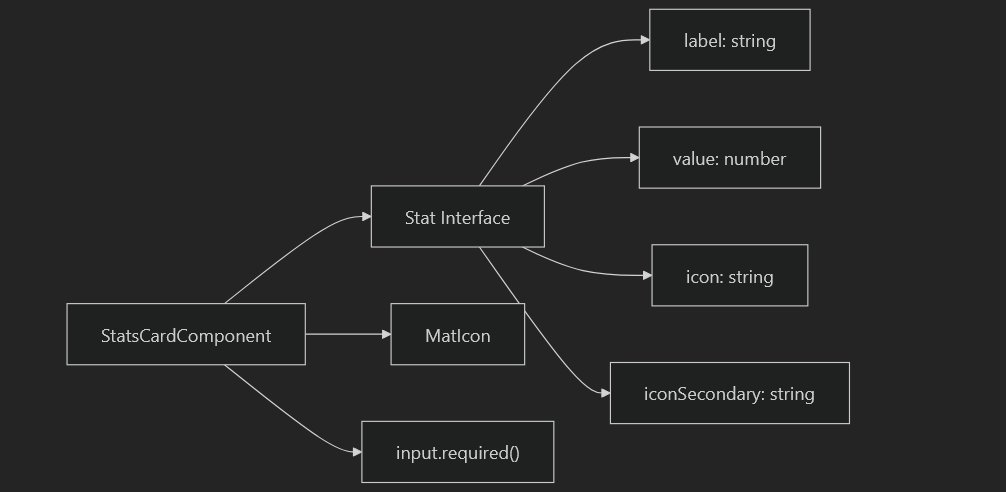
Fuentes: [src/app/features/admin/dashboard/admin-dashboard.component.html1-2](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/admin-dashboard.component.html#L1-L2)

## Componentes de tablero

La aplicación incluye componentes especializados para interfaces de salpicadero que muestran estadísticas y acciones rápidas.

### StatsCardComponent

El StatsCardComponent muestra información estadística en un formato de tarjeta consistente a través de los paneles.



El componente acepta una Stat objeto con datos estadísticos y lo convierte en estilo Material Design.

Fuentes: [src/app/features/admin/dashboard/stats-card.component.ts1 a 18](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/stats-card.component.ts#L1-L18)

### QuickActionsComponent

El QuickActionsComponent proporciona botones procesables para la navegación y tareas comunes.

A screenshot of a computer

AI-generated content may be incorrect.

El componente maneja el enrutamiento y proporciona un estilo de botón consistente a través de diferentes vistas del salpicadero.

Fuentes: [src/app/features/admin/dashboard/quick-actions.component.ts1-20](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/quick-actions.component.ts#L1-L20)

### Patrón de uso del tablero

Los componentes del tablero se utilizan en un patrón de diseño de la rejilla:

<app-stats-card *[stat]*="{

          label: 'Docentes Registrados',

          value: totalTeachers(),

          icon: 'people',

          iconSecondary: 'trending\_up'

         }"></app-stats-card>

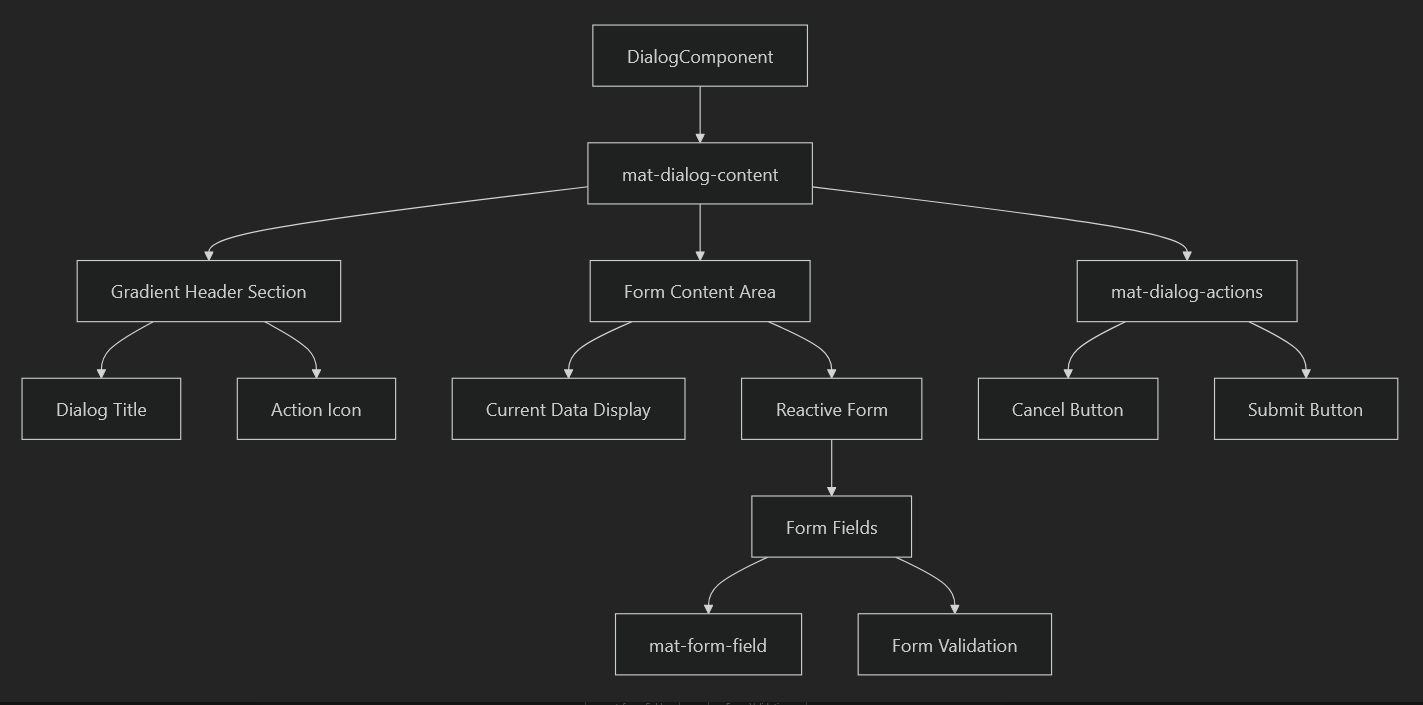
Fuentes: [src/app/features/admin/dashboard/admin-dashboard.component.html23 a 30](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/admin-dashboard.component.html#L23-L30)

## Componentes de diálogo

La aplicación utiliza diálogos de Material angular para interacciones modales, con patrones de estilo y forma.

### Patrón de estructura de diálogo

Los componentes de diálogo siguen una estructura estandarizada con encabezado, contenido y secciones de acción:



Fuentes: [src/app/features/catracado/gramedule/schedule-change-request-dialog.html1-104](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/schedule-change-request-dialog.component.html#L1-L104)

### Patrones de estilo de diálogo

Los diálogos utilizan estilo consistente con:

* Gradient headers with white text and icons
* Structured content areas with background colors
* Validación de formularios con mensajes de error de material
* Botones de acción con efectos flotantes

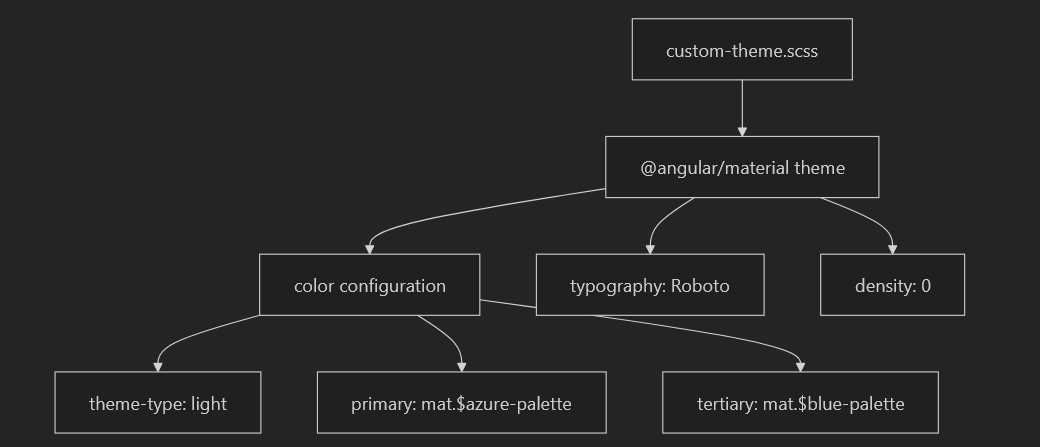
La estructura de la plantilla de diálogo incluye:

* Header section with gradient background [src/app/features/teacher/schedule/schedule-change-request-dialog.component.html2-7](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/schedule-change-request-dialog.component.html#L2-L7)
* Content area with form fields [src/app/features/teacher/schedule/schedule-change-request-dialog.component.html9-92](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/schedule-change-request-dialog.component.html#L9-L92)
* (opcional)Footer with action buttons [src/app/features/teacher/schedule/schedule-change-request-dialog.component.html94-104](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/schedule-change-request-dialog.component.html#L94-L104)

## Sistema de catalogación

La aplicación utiliza Material Angular theming con paletas de colores personalizadas y configuración de tipografía.

### Configuración del tema



La configuración del tema define:

* Tema de luz con paletas de color azul y azul
* tipografía de Roboto
* Ajustes de densidad estándar

Fuentes: [src/costom-theme.scss1-17](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/custom-theme.scss#L1-L17)

### Patternes de estilo

Los componentes utilizan patrones de estilismo consistentes:

* Fondos de gracia para encabezados y elementos de acento
* Efectos desenfoque de retroceso para el diseño moderno de amorfimorfismo de vidrio
* Espaciamiento constante y radio de frontera
* Arriba de diseño de material y sombras

## Arquitectura de componentes

Los componentes compartidos siguen los patrones modernos de Angular con componentes independientes y señales de entrada.

A screenshot of a computer

AI-generated content may be incorrect.

### Patrones de componentes

Todos los componentes compartidos siguen estos patrones:

* Componentes independientes con importaciones explícitas
* Señales de entrada para la encuadernación de datos
* interfaces descripto de tipo para la seguridad de tipo
* Integración con componentes de materiales angulares
* Estilo consistente con clases de Tailwind CSS

Fuentes: [src/app/shared/components/layout/layout.component.ts16 a 28](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/shared/components/layout/layout.component.ts#L16-L28) [src/app/features/admin/dashboard/stats-card.component.ts11 a 18](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/stats-card.component.ts#L11-L18) [src/app/features/admin/dashboard/quick-actions.component.ts13 a 20](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/quick-actions.component.ts#L13-L20)

## Directrices de uso

### Uso de componentes de diseño

El componente de diseño debe utilizarse como componente raíz para todas las vistas principales de aplicación:

<app-layout *[title]*="'Mis Justificaciones'" *[menuItems]*="menuItems">

</app-layout>

### Usage de los componentes del tablero

Los componentes del tablero deben utilizarse en diseños de red con clases sensibles:

        <app-quick-actions *[action]*="{

          label: 'Gestionar Docentes',

          route: '/admin/teachers',

          description: 'Administra información completa de docentes, perfiles y asignaciones',

          icon: 'people'

        }"></app-quick-actions>

### Integración de diálogo

Los diálogos deben seguir el patrón de estructura establecido con encabezados de gradiente y estilo de forma consistente para mantener la aplicación en toda la aplicación.

Fuentes:

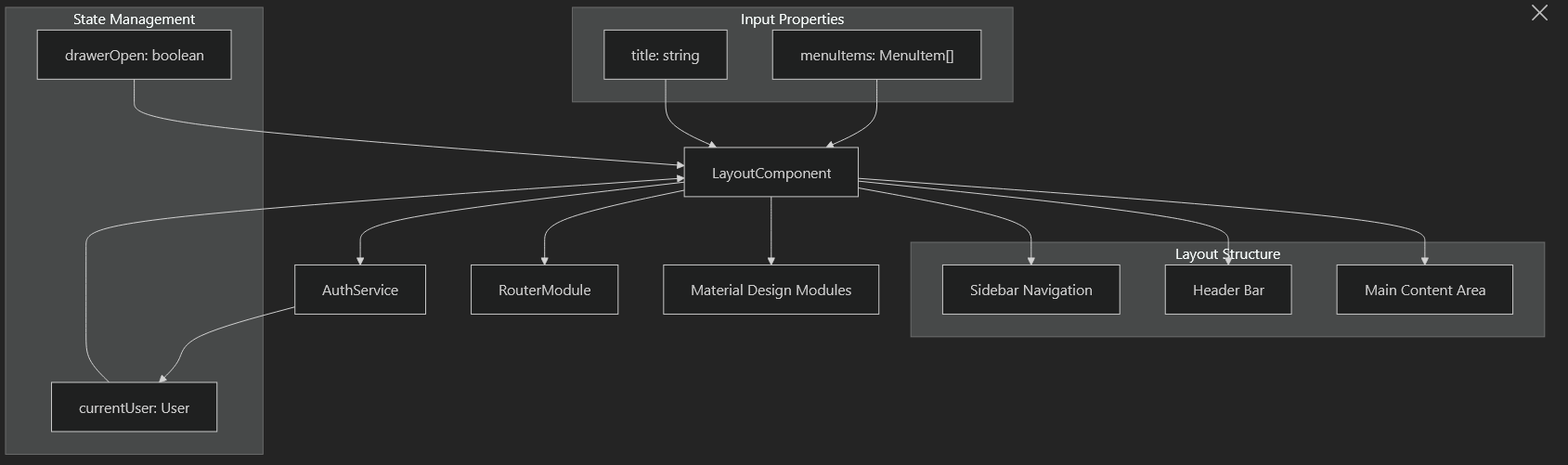
[src/app/features/admin/dashboard/admin-dashboard.component.html69 a 97](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/admin-dashboard.component.html#L69-L97)

# Sistema de diseño

El Sistema de Diseño proporciona la infraestructura de fijación de aplicaciones y navegación para el sistema de gestión de asistencia al profesorado. Este sistema abarca el componente de diseño principal, menús de navegación, patrones de diseño sensibles e integración con el sistema de autenticación para crear una interfaz de usuario consistente en todas las características de la aplicación.

## Arquitectura de componentes

El sistema de diseño se construye alrededor de un único componente principal que sirve como shell de aplicación para los portales de administración y de profesores.



**Fuentes:** [src/app/shared/components/layout/layout.component.ts1-73](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/shared/components/layout/layout.component.ts#L1-L73) [src/app/shared/components/layout/layout.component.html1 a 116](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/shared/components/layout/layout.component.html#L1-L116)

## Componente de diseño de núcleo

El LayoutComponent es un componente independiente Angular que proporciona la estructura fundacional para todas las páginas de la aplicación.

|  |  |  |
| --- | --- | --- |
| Property | Type | Objeto |
| title | input<string> | Título de la página exhibida en la cabecera |
| menuItems | input<MenuItem[]> | Configuración del menú de navegación |
| drawerOpen | boolean | Controla la visibilidad de la barra lateral en el móvil |
| authService | AuthService | Proporciona datos de autenticación de los usuarios |

### Interfaz de MenuItem

export interface MenuItem {

  label: string;

  route: string;

  icon: string;

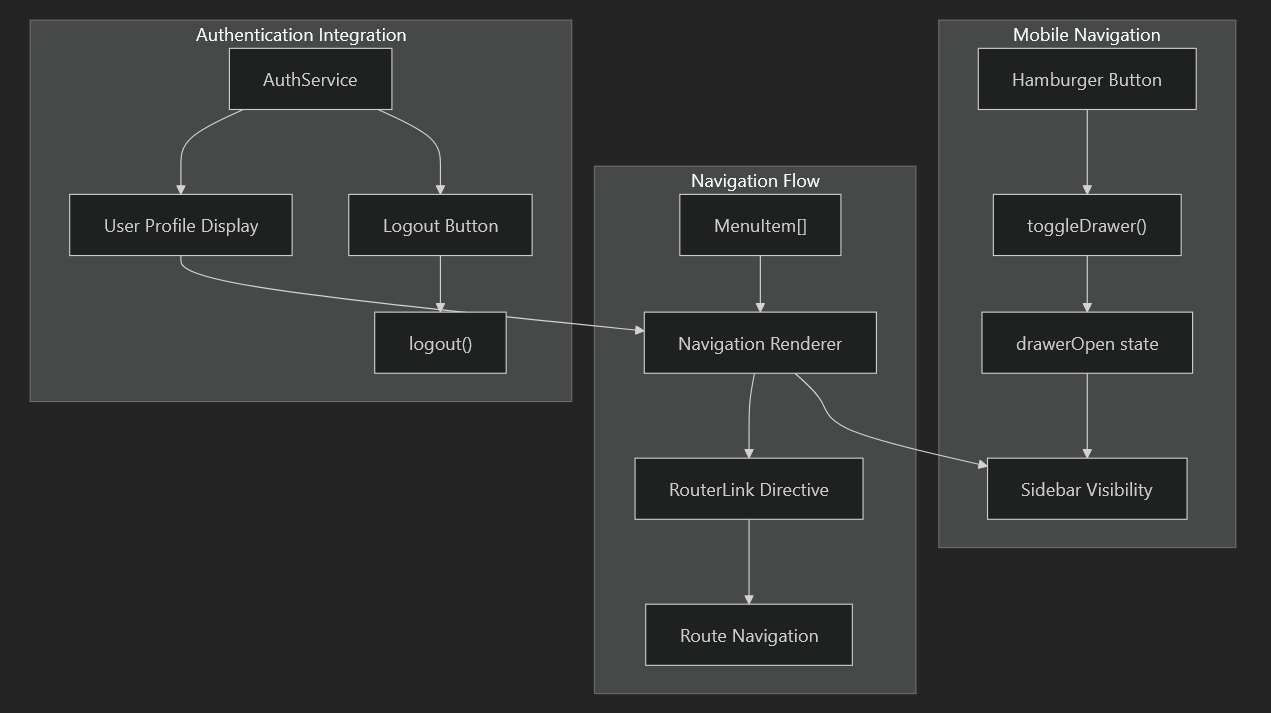
}

Esta interfaz define la estructura para los elementos de navegación pasada al componente de diseño.

**Fuentes:** [src/app/shared/components/layout/layout.component.ts11 a 15](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/shared/components/layout/layout.component.ts#L11-L15)

## Sistema de navegación

El diseño implementa un sistema de navegación jerárquica con elementos de menú basados en roles.



### Aplicación de la navegación

El menú de navegación se renderiza dinámicamente en función de la menuItemsentrada:

* Each menu item includes an icon, label, and route
* Navigation uses Angular Router's routerLink directive
* Mobile navigation is handled through a collapsible drawer
* Active route highlighting is managed by Angular Router

**Fuentes:** [src/app/shared/components/layout/layout.component.html15-31](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/shared/components/layout/layout.component.html#L15-L31)

## Arquitectura de diseño sensible

El sistema de diseño implementa un patrón de diseño sensible que se adapta a diferentes tamaños de pantalla.

A screenshot of a computer

AI-generated content may be incorrect.

### Responsiva de las eliminatorias

|  |  |
| --- | --- |
| Punto de inflexión | Comportamiento |
| lgy arriba | Diseño fijo de la barra lateral |
| Abajo lg | Cajón de sobrecapas con botón de conmutación |

El comportamiento sensible se controla a través de las clases de Tailwind CSS y la gestión del estado componente.

**Fuentes:** [src/app/shared/components/layout/layout.component.html3 a 5](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/shared/components/layout/layout.component.html#L3-L5) [src/app/shared/components/layout/layout.component.html56-60](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/shared/components/layout/layout.component.html#L56-L60)

## Integración de autenticación

El sistema de diseño se integra profundamente con el servicio de autenticación para proporcionar funcionalidad específica del usuario.

A screenshot of a computer

AI-generated content may be incorrect.

### Métodos de autenticación

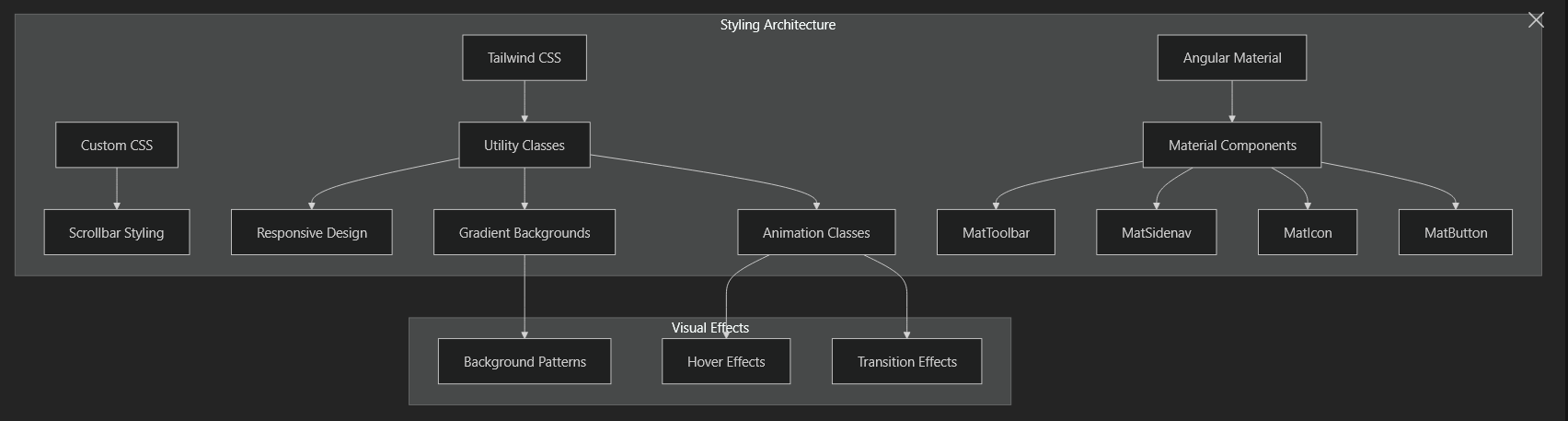
|  |  |
| --- | --- |
| Método | Objeto |
| currentUser() | Proporciona datos reactivos del usuario |
| logout() | Administradora de acceso y navegación del usuario |

El componente de diseño accede a la información del usuario a través de señales angulares y actualiza la interfaz de usuario de forma reactiva.

**Fuentes:** [src/app/shared/components/layout/layout.component.ts62](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/shared/components/layout/layout.component.ts#L62-L62) [src/app/shared/components/layout/layout.component.ts70-72](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/shared/components/layout/layout.component.ts#L70-L72)

### Sistema de estilo

El diseño utiliza un amplio sistema de estilismo que combina Tailwind CSS con Material Angular.



### Características de atipleje personalizado

El diseño incluye varias características de estilo personalizadas:

* Agrietamiento de barra de desplazamiento personalizada con colores de gradiente
* Patrones de fondo animados con efectos borrosos
* Transiciones de la sobrepadura y efectos de escala
* Efectos de desenfocar retroceder para la apariencia moderna

**Fuentes:** [src/app/shared/components/layout/layout.component.ts29-57](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/shared/components/layout/layout.component.ts#L29-L57) [src/app/shared/components/layout/layout.component.html92-102](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/shared/components/layout/layout.component.html#L92-L102)

## Patrones de uso

### Ejemplo de aplicación

Los componentes de administración utilizan la distribución con elementos específicos del menú:

<app-layout *[title]*="'Mis Justificaciones'" *[menuItems]*="menuItems">

</app-layout>

El componente de diseño utiliza la proyección de contenido de Angular para mostrar contenido específico de la página dentro del área de contenido principal.

**Fuentes:** [src/app/features/admin/dashboard/admin-dashboard.component.html1](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/admin-dashboard.component.html#L1-L1) [src/app/shared/components/layout/layout.component.html110](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/shared/components/layout/layout.component.html#L110-L110)

## Integración de diseño de materiales

El diseño aprovecha los componentes de Material angular para patrones de diseño consistentes.

|  |  |
| --- | --- |
| Componente de materiales | Uso |
| MatToolbarModule | Barra de herramientas de encabezado |
| MatSidenavModule | Camiseta de navegación |
| MatIconModule | Iconos en toda la interfaz |
| MatButtonModule | Botones de acción |
| MatListModule | Artículos de menú de navegación |

La integración proporciona características de accesibilidad y un comportamiento consistente entre los navegadores.

**Fuentes:**

[src/app/shared/components/layout/layout.component.ts19 a 27](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/shared/components/layout/layout.component.ts#L19-L27)

# Componentes de la interfaz de usuario

Este documento cubre el sistema de componentes de IU compartido utilizado durante toda la aplicación de gestión de la asistencia al profesorado. Estos componentes proporcionan elementos visuales consistentes e interacciones tanto en los portales de administración como en los de profesores.

## Arquitectura de componentes

Los componentes de interfaz de usuario compartidos se construyen utilizando componentes de diseño de materiales angulares combinados con Tailwind CSS para el estilo. Siguen un patrón consistente de aceptar señales de entrada y proporcionar elementos visuales estandarizados que puedan ser reutilizados en diferentes módulos de características.



Fuentes: [src/app/features/admin/dashboard/stats-card.component.ts1 a 18](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/stats-card.component.ts#L1-L18) [src/app/features/admin/dashboard/quick-actions.component.ts1-20](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/quick-actions.component.ts#L1-L20) [src/costom-theme.scss1 a 18](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/custom-theme.scss#L1-L18)

## Tipos de componentes básicos

### Componentes de visualización de estadísticas

El StatsCardComponentproporciona una manera estandarizada de mostrar estadísticas numéricas con iconos y etiquetas. Acepta un Statinterfaz a través de señales angulares.

A screenshot of a computer

AI-generated content may be incorrect.

El componente utiliza un patrón de interacción basado en flotas con fondos de gradiente y animaciones de escalada. La plantilla incluye estilo de temática verde con iconos materiales para elementos visuales primarios y secundarios.

Fuentes:

[src/app/features/admin/dashboard/stats-card.component.ts4 a 18](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/stats-card.component.ts#L4-L18)

[src/app/features/admin/dashboard/stats-card.component.html1-33](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/stats-card.component.html#L1-L33)

### Componentes de acción

El QuickActionsComponent proporciona botones de acción basados en la navegación con capacidades de estilo y enrutamiento consistentes. Acepta un Action interfaz que define la apariencia y el destino del botón.

A screenshot of a computer

AI-generated content may be incorrect.

El componente utiliza gradientes de temática azul e incluye animaciones hover con efectos de escalado. Se integra con Angular Router para la navegación a través de la RouterLink Directiva.

Fuentes:

[src/app/features/admin/dashboard/quick-actions.component.ts6 a 20](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/quick-actions.component.ts#L6-L20) [src/app/features/admin/dashboard/quick-actions.component.html1-52](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/quick-actions.component.html#L1-L52)

## Componentes de diálogo

### Programar Cambiar diálogo de solicitud

La aplicación incluye sofisticadas plantillas de diálogo para interacciones complejas de los usuarios. El diálogo de solicitud de cambio de calendario demuestra el patrón utilizado para los diálogos basados en forma en todo el sistema.

A screenshot of a computer

AI-generated content may be incorrect.

El diálogo utiliza un patrón consistente de:

* Gradient header with icon and title
* Information display section for context
* Form section with Material form fields
* Action buttons with validation state binding

Fuentes: [src/app/features/catracado/gramedule/schedule-change-request-dialog.html1 a 105](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/schedule-change-request-dialog.component.html#L1-L105)

## Sistema Temático

### Personalización de diseño de material

La aplicación utiliza un tema de material angular personalizado con paletas de colores Azure y azul. La configuración del tema proporciona colores consistentes en todos los componentes del Material.

|  |  |  |
| --- | --- | --- |
| Property | Value | Usage |
| theme-type | light | Base theme mode |
| primary | mat.$azure-palette | Primary actions and highlights |
| tertiary | mat.$blue-palette | Secondary elements |
| typography | Roboto | Font family |
| density | 0 | Nivel de densidad de componentes |

El tema se aplica globalmente a través de la html elemento y afecta a todos los componentes Materiales en toda la aplicación.

Fuentes: [src/costom-theme.scss5-17](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/custom-theme.scss#L5-L17)

## Patrones de uso de componentes

### Patrón de señal de entrada

Todos los componentes compartidos siguen el patrón de señales angulares para la encuadernación de datos reactivos:

  stat = input.required<Stat>();

  action = input.required<Action>();

### Integración de Plantilla

Los componentes se integran en plantillas utilizando el patrón de componentes independientes de Angular:

  imports: [MatIcon, RouterLink, MatButton],

### Enfoque de estilo

Los componentes utilizan una combinación de:

* Tailwind CSS utility classes for layout and spacing
* Material Design tokens for semantic colors
* Custom gradient backgrounds for visual enhancement
* Hover states and animations for interactivity

Fuentes: [src/app/features/admin/dashboard/stats-card.component.ts1 a 18](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/stats-card.component.ts#L1-L18) [src/app/features/admin/dashboard/quick-actions.component.ts1-20](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/quick-actions.component.ts#L1-L20)

## Integración con la capa de servicio

Los componentes de IU reciben datos a través de la jerarquía de componentes de los servicios básicos. El flujo de datos típico implica:

1. Servicios básicos (como AuthService, TeacherService) proporcionar datos a través de señales
2. Componentes de características consumen datos de servicio y los transforman para visualización
3. Componentes de interfaz de usuario compartidos reciben datos tratados a través de señales de entrada
4. Los componentes se vuelven usando Diseño de Material y estilo personalizado

Este patrón garantiza un flujo de datos consistente y reutilizabilidad de componentes a través de la aplicación.

Fuentes: [src/app/core/services/auth.service.ts1-96](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/auth.service.ts#L1-L96) [src/app/core/services/teacher.serv.ts1-179](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/core/services/teacher.service.ts#L1-L179)

# Guía para el desarrollo

Este documento proporciona una documentación técnica completa para los desarrolladores que trabajan en la base de código del sistema de gestión de asistencia al profesorado. Abarca la configuración del entorno de desarrollo, la configuración de la construcción, los patrones de organización de códigos y los flujos de trabajo de desarrollo.

## Configuración del entorno para el desarrollo

La aplicación utiliza Angular 20 con herramientas de desarrollo modernas, incluyendo soporte de renderizado del lado del servidor (SSR) y Tailwind CSS para el estilo.

### Prerrequisitos y Dependencias

El proyecto requiere Node.js y utiliza CLI angular para operaciones de desarrollo. Las dependencias clave incluyen:

|  |  |
| --- | --- |
| Categoría | Dependencias |
| Marco básico | Angular 20.0.0, Angular Material 20.0.2, CDK Angular 20.0.2 |
| Estilo | Tailwind CSS 4.1.10, Material angular, fuentes de Roboto |
| Gestión estatal | RxJS 7.8.0, Señales angulares |
| Server-Side Rendering | Angular SSR 20.0.1, Express 5.1.0 |
| Herramientas de desarrollo | Angular CLI 20.0.1, TypeScript 5.8.2, Pruebas de Karma/Jasmine |

### Guantes de desarrollo

**# Start development server**

npm run start

**# Build for production**

npm run build

**# Run tests**

npm run test

**# Build and watch for changes**

npm run watch

**# Serve SSR build**

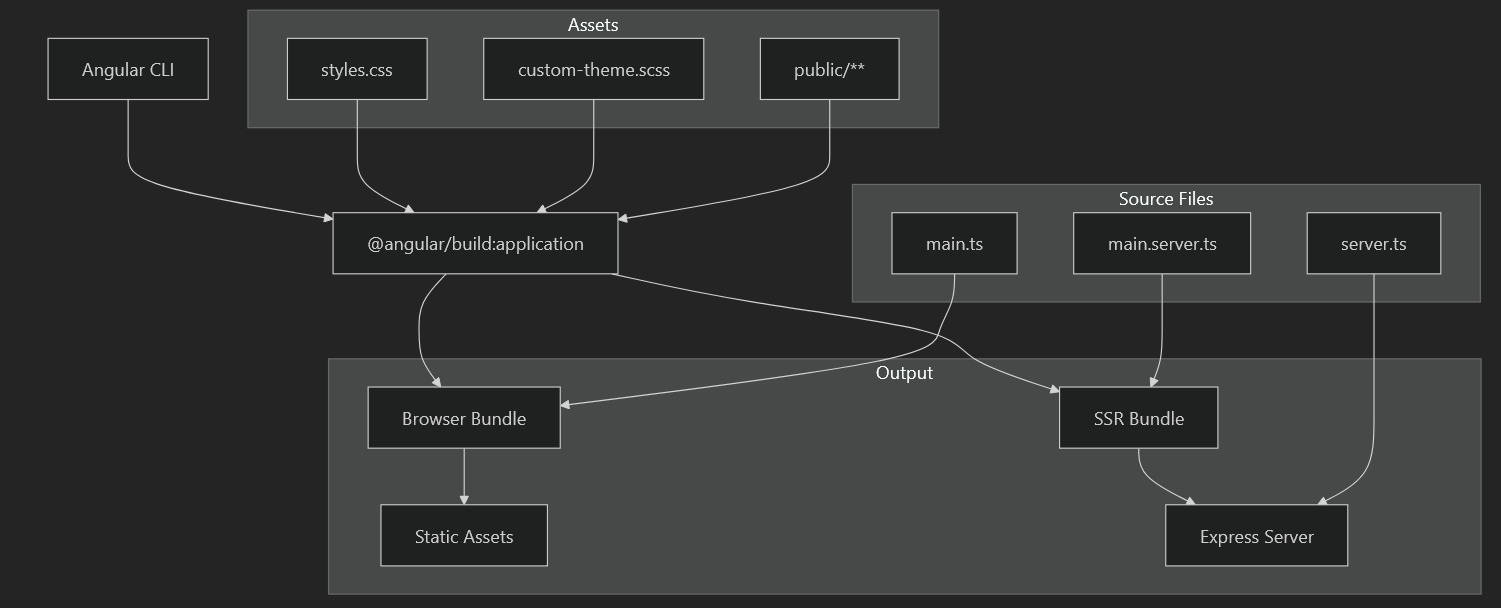
npm run serve:ssr:proyecto-asistencia-docentes-ng

**Fuentes:** [paquete.json4-11](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L4-L11)

## Sistema de construcción y configuración

La aplicación utiliza el moderno sistema de compilación de Angular con soporte para la representación tanto del lado del cliente como del lado del servidor.

### Construir arquitectura de configuración



### Configuraciones de construcción

El sistema de compilación soporta múltiples configuraciones:

|  |  |  |
| --- | --- | --- |
| Configuración | Objeto | Optimizaciones |
| production | Construcción de producción | Presupuestos de paquetes (atención de 500kB, error de 1MB), hashing de salida |
| development | Construyemientos de desarrollo | Sin optimización, mapas de origen, reconstrucciones rápidas |

**Fuentes:** [angular.json13 - 56](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/angular.json#L13-L56)

## Organización de código y estructura de módulos

La aplicación sigue la arquitectura de módulo de características recomendada de Angular con carga perezosa y componentes independientes.

### Estructura y enrutamiento del módulo

A screenshot of a computer

AI-generated content may be incorrect.

### Patrones de configuración de ruta

La aplicación utiliza carga perezosa para los módulos de características:

**Rutas de administración:**

* /admin/dashboard → AdminDashboardComponent
* /admin/teachers → TeacherManagementComponent
* /admin/schedules → ScheduleManagementComponent
* /admin/attendance- AttendanceManagementComponent

**Rutas maestras:**

* /teacher/dashboard → Lazy loaded TeacherDashboardComponent
* /teacher/attendance → Lazy loaded TeacherAttendanceComponent
* /teacher/schedule → Lazy loaded TeacherScheduleComponent
* /teacher/justificationsPerezoso cargado TeacherJustificationsComponent

**Fuentes:** [src/app/features/admin/admin.routes.ts7-29](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/admin.routes.ts#L7-L29) [src/app/features/teacher/teacher.routes.ts3 a 26](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/teacher.routes.ts#L3-L26)

## Desarrollo Flujo de Trabajo y Deatación

### Configuración de servidor de desarrollo

El servidor de desarrollo utiliza el servidor de dev incorporado de Angular con soporte de reemplazo de módulos de hot. La configuración predeterminada sirve a la aplicación en http://localhost:4200.



### Gestión de activos

Los activos se gestionan a través del sistema de construcción de Angular:

* **Global Styles:** Combined from custom-theme.scss and styles.css
* **Static Assets:** Served from public/ directory
* **External Dependencies:** Google Fonts (Roboto), Material Icons

**Fuentes:**

[angular.json18-27](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/angular.json#L18-L27) [angular.json77 a 85](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/angular.json#L77-L85)

## Sistema de estilo y configuración temática

### Arquitectura CSS

La aplicación utiliza un enfoque híbrido que combina la temática de Material Angular con servicios de Táilwind CSS:

A screenshot of a computer

AI-generated content may be incorrect.

### Configuración de fuente e icono

La aplicación utiliza Google Fonts y Material Icons cargados externamente:

* **Typography:** Roboto font family (300, 400, 500 weights)
* **Icons:** Material Icons font
* **Base Styles:** Full height layout with zero margins

**Fuentes:** [src/styles.css1-6](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/styles.css#L1-L6) [src/index.html9-10](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/index.html#L9-L10)

## Pruebas y aseguramiento de la calidad

### Configuración de prueba

La aplicación utiliza Karma y Jasmine para pruebas de unidad:

A screenshot of a computer

AI-generated content may be incorrect.

### Monitoreo de tamaño de Bundle

Las creaciones de producción incluyen la supervisión presupuestaria:

* **Initial Bundle:** 500kB warning, 1MB error threshold
* **Component Styles:** 4kB warning, 8kB error threshold per component

**Fuentes:**

[angular.json73 - 87](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/angular.json#L73-L87) [angular.json36-47](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/angular.json#L36-L47)

## Configuración de representación de servidor-side

### Arquitectura de la SSR

La aplicación es compatible con renderizado del lado del servidor usando Angular Universal y Express:

A screenshot of a computer

AI-generated content may be incorrect.

### Flujo de trabajo para el desarrollo de la RSS

**Development:** Use ng serve for client-side development

**SSR Testing:** Build with ng build then serve with npm run serve:ssr:proyecto-asistencia-docentes-ng

**Producción:** Implementar tanto activos estáticos como Servidor Express

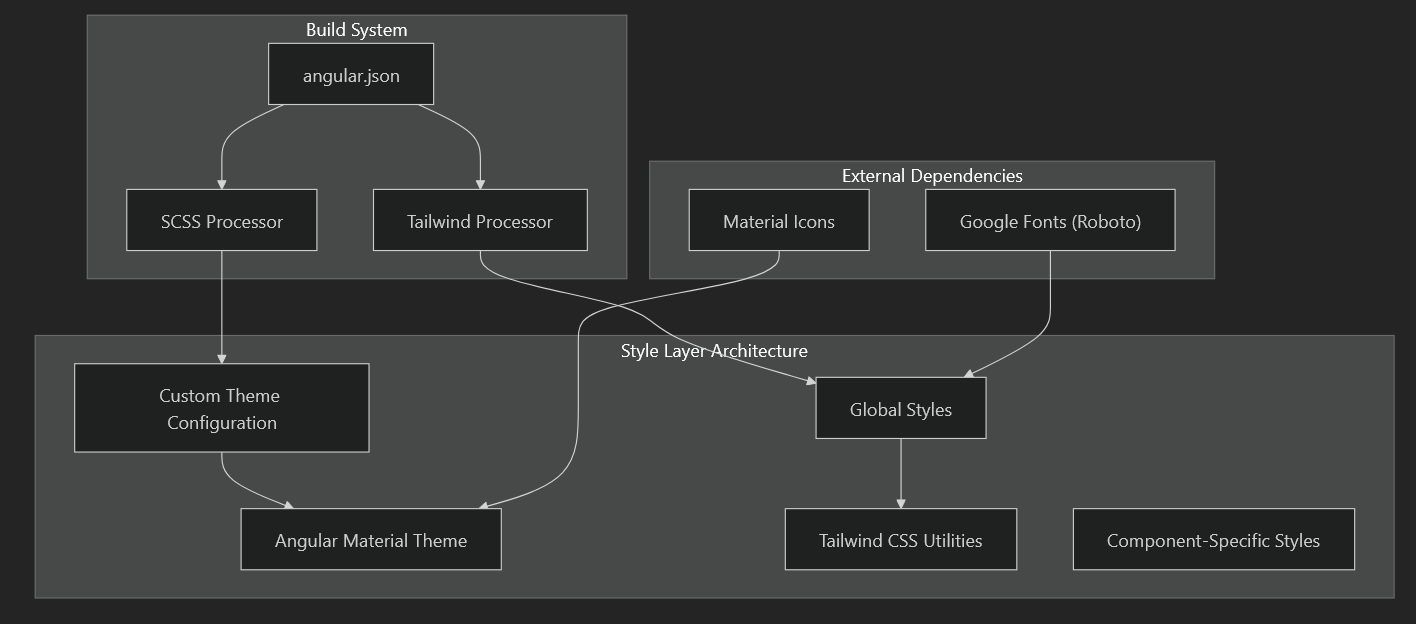
**Fuentes:** [paquete.json10](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/package.json#L10-L10) [angular.json28-32](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/angular.json#L28-L32)

# Sistema de estilo

Este documento cubre la arquitectura de estilo y la implementación del sistema de diseño para la aplicación de gestión de asistencia al profesorado. El sistema de estilismo utiliza un enfoque híbrido que combina el sistema de caracterización de Angular Material con clases de utilidad Tailwind CSS para un diseño flexible y consistente.

## Resumen de la arquitectura

El sistema de estilo sigue una arquitectura en capas que combina múltiples marcos y metodologías de CSS:



**Fuentes:** [angular.json24 a 26](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/angular.json#L24-L26) [src/costom-theme.scss1 a 18](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/custom-theme.scss#L1-L18) [src/styles.css1 a 6](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/styles.css#L1-L6) [src/index.html9 a 10](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/index.html#L9-L10)

## Configuración del tema

La aplicación utiliza el sistema de mestizaje de Angular Material con paletas de colores personalizadas y configuración de tipografía:

### Configuración de tema de materiales

La configuración del tema se define en custom-theme.scss utilizando la API de abordaje de Angular Material:

|  |  |  |
| --- | --- | --- |
| Property | Value | Descripción |
| theme-type | light | Establece el modo temático general |
| primary | mat.$azure-palette | Paleta de colores primarios (azure) |
| tertiary | mat.$blue-palette | Paleta de colores terciarios (azul) |
| typography | Roboto | Familia de tipografía |
| density | 0 | Nivel de densidad de componentes |

A screenshot of a computer

AI-generated content may be incorrect.

**Fuentes:** [src/costom-theme.scss6 a 16](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/custom-theme.scss#L6-L16)

### Sistema de tipografía

El sistema de tipografía utiliza la familia de fuentes Roboto cargada de Google Fonts:

* **Familia de la fuente**: Roboto con retrocesos a "Helvetica Neue" y sans-serif
* **Pesas de la fuente**: 300, 400, 500 (luz, regular, medio)
* **Cargando**: CDN externo a través de Google Fonts API

**Fuentes:** [src/index.html9](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/index.html#L9-L9) [src/styles.css5](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/styles.css#L5-L5)

## Estilos globales

Los estilos globales se configuran en styles.cssy establecer la base para la aparición de la solicitud:

### Integración de viento en mano

La solicitud importa Tailwind CSS utilizando la @import "tailwindcss"Directiva, que proporciona acceso a las clases de servicios públicos durante toda la solicitud.

### Estilos de base

html, body { height: 100%; }

body { margin: 0; font-family: Roboto, "Helvetica Neue", sans-serif; }

Estos estilos aseguran:

* Altura completa del puerto para la aplicación
* Restablecer los márgenes de navegador predeterminados
* tipografía consistente en toda la aplicación

**Fuentes:** [src/styles.css2 a 5](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/styles.css#L2-L5)

## Patrones de estilo de componentes

La aplicación utiliza patrones de estilismo consistentes en todos los componentes, combinando componentes de Material angular con servicios de Tailwind:

### Enfoque de estilo híbrido

Los componentes utilizan una combinación de:

* Componentes de materiales angulares para la estructura y el comportamiento
* Servicios públicos Tailwind CSS para espaciar, colores y diseño
* Clases CSS personalizadas para el estilo de componente específico

A screenshot of a computer

AI-generated content may be incorrect.

**Fuentes:** [src/app/features/catracado/gramedule/schedule-change-request-dialog.html1-104](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/schedule-change-request-dialog.component.html#L1-L104)

### Patrones de estilo común

**Estilo de diálogo**

* Las secciones de encabezado utilizan fondos de gradiente: bg-gradient-to-r from-blue-600 to-blue-700
* Las áreas de contenido utilizan acolchado consistente: !p-0, !px-6, !py-4
* Las áreas de acción utilizan colores de fondo: !bg-gray-50

**Forma Estilo**

* Campos de forma de ancho completo: w-full
* Espaciado consistente: space-y-4, space-y-6
* Estilo de error con iconos: flex items-center

**Sistema de color**

* Blues primarios: blue-600, blue-700, blue-800
* Variaciones de fondo: blue-50, gray-50
* Colores de texto: text-gray-700, text-gray-800

**Fuentes:** [src/app/features/catracado/gramedule/schedule-change-request-dialog.html2 a 3](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/schedule-change-request-dialog.component.html#L2-L3) [src/app/features/catracado/gramedule/schedule-change-request-dialog.html35 a 1945](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/schedule-change-request-dialog.component.html#L35-L45) [src/app/features/catracado/gramedule/schedule-change-request-dialog.html94-104](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/schedule-change-request-dialog.component.html#L94-L104)

### Utilidad prevalece patrones

La aplicación utiliza Tailwind's !prefijo para anular los estilos predeterminados de Angular Material:

|  |  |  |
| --- | --- | --- |
| Pattern | Usage | Example |
| !p-0 | Remove default padding | Dialog content |
| !text-xl | Override text size | Dialog titles |
| !font-bold | Sobredimensión de peso de la fuente | Cabeceras |
| !mb-0 | Eliminar los márgenes por defecto | Títulos |

**Fuentes:** [src/app/features/catracado/gramedule/schedule-change-request-dialog.html1](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/schedule-change-request-dialog.component.html#L1-L1) [src/app/features/catracado/gramedule/schedule-change-request-dialog.html3](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/schedule-change-request-dialog.component.html#L3-L3)

## Configuración de construcción

El sistema de estilismo se integra en el proceso de compilación Angular a través de la configuración específica:

### Orden de carga de estilo

El angular.jsonla configuración define el orden de carga de estilo:

1. src/custom-theme.scss - Tema de material angular
2. src/styles.css - Estilos globales con Tailwind

Este orden asegura que:

* Las variables de los temas materiales están disponibles a nivel mundial
* Las utilidades de Tailwind pueden anular los valores por defecto del material
* Los estilos específicos de los componentes tienen prioridad

### Presupuestos de estilo

El sistema de construcción hace cumplir los límites presupuestarios de estilo:

|  |  |  |
| --- | --- | --- |
| Tipo de presupuesto | Urea de advertencia | Urea de error |
| anyComponentStyle | 4kB | 8kB |
| initial | 500kB | 1MB |

**Fuentes:** [angular.json24 a 26](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/angular.json#L24-L26) [angular.json42 - 46](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/angular.json#L42-L46)

## Sistema de iconos

La aplicación utiliza Iconos de Material para una iconografía consistente:

### Integración de Iconos de Material

Los iconos se cargan de Google Fonts CDN y se utilizan en toda la aplicación:

        <mat-icon *matPrefix* *class*="!text-gray-500">event</mat-icon>

### Patrones de estilo icono

Servicios de gran tamaño: !text-base, !text-sm

Servicios de escapada: mr-1, mr-2

Servicios de colores: !text-gray-500, text-gray-600

Servicios de asistencia: align-middle

**Fuentes:** [src/index.html10](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/index.html#L10-L10) [src/app/features/catracado/gramedule/schedule-change-request-dialog.html4](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/schedule-change-request-dialog.component.html#L4-L4) [src/app/features/catracado/gramedule/schedule-change-request-dialog.html40](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/teacher/schedule/schedule-change-request-dialog.component.html#L40-L40)

## Estilo específico de componente

### Componente de tarjeta de estadísticas

El StatsCardComponentsigue las convenciones de estilo de la aplicación con un estilo mínimo basado en plantillas, confiando en servicios públicos Tailwind para el diseño y la apariencia.

### Componente de acción rápida

El QuickActionsComponentutiliza estilo de botón consistente con principios de diseño de materiales mejorados por las utilidades Tailwind.

**Fuentes:** [src/app/features/admin/dashboard/stats-card.component.ts1 a 18](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/stats-card.component.ts#L1-L18) [src/app/features/admin/dashboard/quick-actions.component.ts1-20](https://github.com/lesquel/proyecto-asistencia-docentes-ng/blob/1796b333/src/app/features/admin/dashboard/quick-actions.component.ts#L1-L20)