

[2025-02-19] hasta [2025-03-28]

Mejoras Significativas en la Gestión de Usuarios y Sistema de Agencias

Controladores y Endpoints

- **UserController:**
 - Implementado nuevo endpoint **AddUserToDb** para agregar usuarios directamente a la base de datos con validación completa de datos
 - Incorporada validación avanzada para datos de usuario con verificación de correo electrónico y formato de campos
 - Optimizado el manejo de roles y permisos con soporte para múltiples roles por usuario
 - Mejorada la gestión de contraseñas temporales con opciones específicas para modo DEBUG
 - Implementada verificación de campos obligatorios como **Roles** en las solicitudes
- **UploadController:**
 - Reestructurado el manejo de subida de archivos con soporte para diferentes tipos de contenido
 - Implementadas validaciones robustas para verificar tamaño, formato y contenido de archivos
 - Añadido soporte para subida de imágenes de avatar con procesamiento automatizado
 - Mejorada la seguridad en la gestión de archivos subidos con validación de extensiones
- **AgencyController:**
 - Optimizada la obtención de datos de agencias con nuevos filtros y ordenamientos
 - Incorporadas mejoras en la asignación de usuarios a agencias con validación de permisos
 - Actualizada la obtención de datos de agencias con información de programas asociados
 - Añadido soporte para filtrado por estado y búsqueda por texto en nombres de agencias
- **AgencyUserAssignmentController:**
 - Nuevo controlador especializado para manejar la relación entre usuarios y agencias
 - Implementados endpoints completos para:
 - Asignar usuarios a agencias con validación de roles y permisos
 - Desasignar usuarios de agencias manteniendo integridad referencial
 - Consultar asignaciones existentes con múltiples filtros
 - Modificar nivel de acceso de usuarios a agencias específicas
- **AuthController:**
 - Refactorizado para mejorar manejo de autenticación y registro
 - Optimizado flujo de trabajo para peticiones de login
 - Mejorado sistema de tokens y refresco de sesiones
 - Implementado soporte para desconexión segura y terminación de sesiones

Base de Datos

Procedimientos Almacenados

- **Nuevos Procedimientos:**

- **109_GeneratePasswordResetToken**: Procedimiento avanzado para generar tokens únicos de restablecimiento de contraseñas con expiración configurable
- **110_ValidatePasswordResetToken**: Sistema de validación de tokens con verificación de vigencia y estado
- **101_GetAllProgramInscriptions**: Consulta optimizada para obtener inscripciones a programas con información relacional completa
- **103_GetAgencyByIdAndUserId**: Obtención específica de agencia con verificación de permisos de usuario
- **104_GetAgencyByIdAndUserId**: Versión mejorada con soporte para múltiples roles y mayor detalle en la información retornada
- **108_GetAllUsersFromDb**: Procedimiento completamente reescrito para incluir información detallada de usuarios con filtrado avanzado

- **Actualizaciones y Optimizaciones:**

- **101_AssignAgencyToUser**: Mejorada la lógica de asignación para prevenir duplicados y mantener historial
- Renombrados múltiples procedimientos almacenados siguiendo la nueva convención numérica de versiones
- Optimizadas las consultas SQL con uso adecuado de índices y joins
- Mejorado el manejo de transacciones para garantizar integridad de datos
- Implementada consistencia en el manejo de parámetros opcionales y valores predeterminados

Modificaciones en Estructura de Tablas

- **Reorganización de Tablas:**

- Movimiento de archivos de definición de tablas siguiendo nueva estructura organizativa
- Renombrado de archivos para mejorar consistencia: **AgencyTable.sql**, **AgencyUsersTable.sql**
- Modificación en la tabla **AgencyUsers** para mejor relación entre entidades

- **Nuevos Campos y Tablas:**

- Añadidos campos en tablas de agencias para soporte de nuevas funcionalidades
- Creadas tablas para almacenar tokens de restablecimiento de contraseñas
- Modificada estructura para mejor soporte de asignaciones usuario-agencia
- Eliminado el campo **AgencyId** de la tabla **AspNetUsers** para mejor manejo de relaciones múltiples

Scripts de Migración y Control de Versiones

- **Sistema Flyway para Migraciones:**

- Implementados scripts Flyway para gestión controlada de cambios en la base de datos
- Creados scripts específicos para entornos:
 - **.flyway/migrations-azure-db/**: Migraciones específicas para Azure SQL

- `.flyway/migrations-local-db/`: Migraciones para desarrollo local
- `.flyway/migrations-prod-db/`: Migraciones para entorno de producción
- Implementada numeración de versiones para secuencia controlada de aplicación

- **Documentación de Cambios en SQL:**

- Creados archivos de registro de cambios SQL (`sql-change-log-2025-03-13.md`, `sql-change-log-2024-03-25.md`)
- Implementado sistema de documentación histórica para facilitar seguimiento de evolución de base de datos

Interfaces y Repositorios

- **IUserRepository:**

- Actualizada la firma del método `RegisterUser` para utilizar DTOs completos en lugar de entidades User directas
- Añadidos métodos específicos para la gestión de contraseñas:
 - Generación de tokens de recuperación con configuración de expiración
 - Validación de tokens con múltiples criterios de seguridad
 - Cambio de contraseña con validación de token
- Incorporada documentación XML detallada para facilitar implementación
- Mejorada la estructura general de la interfaz con agrupación lógica de métodos

- **IEmailService:**

- Añadidos métodos especializados para envío de diferentes tipos de correos:
 - Notificaciones de recuperación de contraseña con tokens seguros
 - Plantillas HTML mejoradas para comunicaciones con usuarios
 - Soporte para adjuntos en correos electrónicos
- Mejorado el sistema de plantillas de correo con variables dinámicas
- Implementado manejo de errores específico para problemas de envío

- **IAgencyUsersRepository:**

- Reestructurada la interfaz para proveer métodos más específicos y cohesivos
- Mejorados los parámetros de entrada/salida para facilitar integración
- Añadida capacidad para consultas con múltiples criterios de filtrado

- **UserRepository:**

- Implementada lógica completa para registro de usuarios con validación avanzada de datos
- Mejorado el manejo de contraseñas temporales con procedimientos específicos de almacenamiento seguro
- Añadido soporte completo para:
 - Gestión de avatares con procesamiento de imágenes
 - Historial de cambios de contraseña y accesos
 - Validación de tokens de seguridad para recuperación de cuentas
- Optimizado el rendimiento mediante uso adecuado de caching

- **AgencyRepository:**
 - Implementado sistema de caché inteligente con invalidación selectiva
 - Mejorada la lógica de actualización de datos de agencias con validación de cambios
 - Optimizadas las consultas para reducir carga de base de datos
 - Añadido soporte para transacciones que involucran múltiples operaciones
 - Refinada la lógica para asignación de programas a agencias
 - Implementado manejo eficiente de relaciones entre agencias
- **AgencyUserAssignmentRepository:**
 - Nuevo repositorio especializado con métodos para:
 - Asignación eficiente de usuarios a agencias
 - Gestión de historial de asignaciones
 - Verificación de permisos cruzados
 - Desasignación segura manteniendo integridad referencial

Modelos y DTOs

- **DTOUser y DTOUserDB:**
 - Actualizados para incluir campos adicionales:
 - Información personal detallada con validación específica
 - Preferencias de usuario y configuración
 - Historial de acciones relevantes
 - Mejorada la validación de datos con atributos específicos
 - Optimizada la serialización/deserialización JSON
 - Añadidos campos para gestión de perfiles mejorados
- **UserAvatarRequest:**
 - Nuevo modelo especializado para gestionar la subida y procesamiento de avatares de usuario
 - Implementadas validaciones específicas para imágenes:
 - Verificación de dimensiones y resolución
 - Control de tamaño máximo
 - Validación de tipos MIME permitidos
 - Soporte para recorte y redimensionado de imágenes
- **DTOChangePasword:**
 - Nuevo modelo completo para el proceso de cambio de contraseñas con:
 - Validación de complejidad de contraseña
 - Verificación de token de seguridad
 - Prevención de reutilización de contraseñas antiguas
 - Implementadas reglas de validación de contraseñas según estándares de seguridad
- **QueryParameters:**
 - Ampliados los parámetros de consulta para soportar:
 - Filtrado avanzado por múltiples criterios

- Ordenamiento configurable
- Paginación optimizada
- Búsqueda por texto en múltiples campos
- **Role y Modelos de Autenticación:**
 - Reestructurados modelos de roles con mejor tipado
 - Refinados modelos de autenticación para mejor seguridad
 - Eliminado modelo `ResetPasswordRequest` y reemplazado por estructura más segura

Mejoras en Infraestructura y Rendimiento

- **Configuración CORS:**
 - Actualización completa de políticas para mejor control de acceso con:
 - Soporte para múltiples orígenes configurables por ambiente
 - Control granular de métodos HTTP permitidos
 - Configuración específica de cabeceras permitidas
 - Añadido soporte para nuevos dominios de desarrollo y producción:
 - Actualizado para incluir `http://localhost:4202`
 - Añadido soporte para `https://nutre-dev.local:4202`
 - Implementada configuración separada para ambientes de desarrollo, pruebas y producción
- **Caché y Rendimiento:**
 - Implementado sistema de caché multinivel con:
 - Caché en memoria para accesos frecuentes
 - Caché distribuida para datos compartidos entre instancias
 - Estrategias de invalidación inteligente
 - Optimizadas consultas a base de datos mediante:
 - Uso efectivo de índices
 - Reducción de joins innecesarios
 - Paginación eficiente
 - Implementada compresión de respuestas HTTP para reducir tráfico de red
 - Extensión del sistema de caché a múltiples repositorios:
 - `AlternativeCommunicationRepository`
 - `EducationLevelRepository`
 - `FacilityRepository`
 - `FederalFundingCertificationRepository`
 - `FoodAuthorityRepository`
 - `MealTypeRepository`
 - `OperatingPeriodRepository`
 - `OperatingPolicyRepository`
 - `OrganizationTypeRepository`
 - `ProgramRepository`
 - `SchoolRepository`
- **Seguridad:**

- Mejorada la gestión de tokens JWT con:
 - Rotación de claves de firma
 - Validación avanzada de audiencia e emisor
 - Configuración de tiempos de expiración por tipo de operación
- Implementado sistema completo de recuperación de contraseñas con:
 - Tokens de un solo uso con tiempo limitado
 - Verificación de correo electrónico
 - Protección contra ataques de fuerza bruta
- Incorporadas validaciones robustas para datos sensibles con:
 - Sanitización de entrada para prevenir XSS
 - Validación de formato para datos críticos
 - Protección contra CSRF

- **Telemetría y Logging:**

- Implementado servicio de logging estructurado con:
 - Niveles configurables de detalle
 - Rotación automática de archivos de log
 - Formateo JSON para mejor análisis
- Integración completa con Application Insights para:
 - Monitoreo de rendimiento y disponibilidad
 - Seguimiento de transacciones distribuidas
 - Detección de anomalías
 - Alertas automáticas
- Añadido inicializador de telemetría para enriquecimiento de datos con información de entorno:
 - `EnvironmentTelemetryInitializer` para agregar datos contextuales
 - Identificación automática de entorno (desarrollo/producción)
 - Enriquecimiento de telemetría con datos de solicitud

- **Gestión de Archivos:**

- Mejorado sistema de subida y almacenamiento con:
 - Validación previa de archivos
 - Almacenamiento seguro con nombres aleatorios
 - Gestión adecuada de MIME types
- Implementado soporte específico para avatares de usuario con:
 - Procesamiento automático de imágenes
 - Generación de múltiples tamaños para diferentes contextos
 - Eliminación de metadatos sensibles

Configuración y Despliegue

- **Automatización de Despliegue:**

- Configurado flujo de trabajo de Azure App Service para CI/CD
- Mejorada configuración de entornos con variables específicas
- Implementada separación clara de configuraciones por ambiente
- Agregado soporte para despliegue automatizado a través de GitHub Actions

- **Mejoras en el Entorno de Desarrollo:**

- Implementada configuración `.csharpierrc.json` para formato consistente de código
- Actualizadas configuraciones de VS Code para mejor experiencia de desarrollo
- Simplificadas configuraciones de lanzamiento y depuración
- Añadidas reglas y guías específicas para distintos aspectos del código:
 - `code-format-corenet.mdc`: Reglas para Core .NET
 - `dapper-best-practices.mdc`: Mejores prácticas para Dapper
 - `jwt-bearer-best-practices.mdc`: Mejores prácticas para JWT
 - `sql-server-best-practices.mdc`: Guía para SQL Server

- **Seguridad en Producción:**

- Actualizada configuración para entornos productivos
- Implementadas mejores prácticas de seguridad para despliegue
- Configuración de certificados y HTTPS
- Optimización de configuraciones para producción

- **Migraciones de Base de Datos:**

- Añadidos scripts de migración Flyway para actualizaciones controladas
- Implementada estrategia de versionamiento de base de datos
- Configuración separada para bases de datos locales, de desarrollo y producción
- Documentación detallada de cambios en base de datos en archivos de changelog SQL