# Full Project Reference - Domain & Services

This document contains a complete reference of all Domain Entities, Enums, Builders, and Services (with their Interfaces, Methods, and Parameters). It summarizes everything implemented so far, and can be used as a reference for further development.

## 1. Domain Entities & Enums

### User

* - string Username
* - string Email
* - string PasswordHash
* - string PasswordSalt
* - int PasswordFormatId
* - PasswordFormat PasswordFormat
* - bool IsActive
* - bool IsLockedOut
* - int FailedLoginAttempts
* - DateTime? LockoutEndUtc
* - DateTime CreatedOnUtc
* - DateTime? UpdatedOnUtc
* - DateTime? LastLoginDateUtc
* - DateTime? DeactivatedOnUtc
* - int? InstitutionId
* - int? RegistrationId
* - string PasswordResetToken
* - DateTime? PasswordResetTokenExpiry
* - string TwoFactorSecret
* - bool TwoFactorEnabled

### Registration

* - string InstitutionName
* - string LicenseNumber
* - int LicenseSectorId
* - LicenseSector LicenseSector
* - int LicenseTypeId
* - LicenseType LicenseType
* - int FinancialDomainId
* - FinancialDomain FinancialDomain
* - DateTime IssueDate
* - DateTime ExpiryDate
* - int CountryId
* - string Address
* - RegistrationStatus Status
* - int CreatedByUserId
* - int? UpdatedByUserId
* - int? SubmittedToUserId
* - DateTime CreatedOnUtc
* - DateTime? SubmittedDateUtc
* - DateTime? ApprovedDateUtc
* - DateTime? AuditedDateUtc
* - int? BusinessScaleRangeId
* - int? EmployeeRangeId

### Institution

* - string Name
* - string LicenseNumber
* - int LicenseTypeId
* - LicenseType LicenseType
* - int LicenseSectorId
* - LicenseSector LicenseSector
* - int FinancialDomainId
* - FinancialDomain FinancialDomain
* - DateTime LicenseIssueDate
* - DateTime LicenseExpiryDate
* - int CountryId
* - string Address
* - bool IsActive
* - DateTime CreatedOnUtc
* - DateTime? UpdatedOnUtc
* - ICollection<Department> Departments
* - ICollection<Job> Jobs
* - ICollection<Contract> Contracts
* - ICollection<Duty> Duties

### Notification

* - int EventTypeId
* - NotificationEvent EventType
* - int RecipientUserId
* - int RegistrationId
* - int TriggeredByUserId
* - string Message
* - string Channel
* - string Status
* - DateTime CreatedOnUtc

### AuditTrail

* - string TableName
* - string ColumnName
* - string PrimaryKeyValue
* - string OldValue
* - string NewValue
* - int ChangedByUserId
* - DateTime ChangedOnUtc
* - string ActionType
* - string ClientIp
* - string Note
* - string EntityName
* - int EntityId
* - string Action
* - string Comment
* - string Field

### Enums

* - LicenseType
* - LicenseSector
* - FinancialDomain
* - RegistrationStatus
* - NotificationEvent
* - AuditActionType
* - PasswordFormat

## 2. Services & Interfaces

### IUserService

* - GetByIdAsync(int id) → 1 param
* - GetAllAsync(bool onlyActive = true) → 1 param
* - InsertAsync(User user, string password, int createdByUserId) → 3 params
* - UpdateAsync(User user) → 1 param
* - DeleteAsync(int id) → 1 param
* - GetByEmailAsync(string email) → 1 param
* - GetByUsernameAsync(string username) → 1 param
* - ActivateAsync(int userId) → 1 param
* - DeactivateAsync(int userId) → 1 param
* - ResetPasswordAsync(int userId, string newPassword, int performedByUserId) → 3 params
* - LockUserAsync(int userId, int? minutes = null) → 2 params
* - UnlockUserAsync(int userId) → 1 param
* - GetRolesAsync(int userId) → 1 param
* - AddToRoleAsync(int userId, int roleId) → 2 params
* - RemoveFromRoleAsync(int userId, int roleId) → 2 params

### IRegistrationService

* - GetByIdAsync(int id)
* - GetAllAsync()
* - InsertAsync(Registration registration)
* - UpdateAsync(Registration registration)
* - DeleteAsync(int id)
* - SubmitAsync(int registrationId, int performedByUserId, string remarks = null)
* - ValidateAsync(int registrationId, int performedByUserId, ValidationStatus status, string remarks = null)
* - ApproveAsync(int registrationId, int performedByUserId, ApprovalStatus status, string remarks = null)
* - AuditAsync(int registrationId, int performedByUserId, AuditStatus status, string remarks = null)
* - ReturnForEditAsync(int registrationId, int performedByUserId, string remarks = null)
* - RejectAsync(int registrationId, int performedByUserId, string remarks = null)
* - ArchiveAsync(int registrationId, int performedByUserId, string remarks = null)
* - FinalSubmissionAsync(int registrationId, int performedByUserId, string remarks = null)
* - GetContactsByRegistrationIdAsync(int registrationId)
* - GetDocumentsByRegistrationIdAsync(int registrationId)
* - AddContactAsync(int registrationId, FIContact contact)
* - AddDocumentAsync(int registrationId, FIDocument document)
* - CheckDuplicateAsync(string institutionName, string licenseNumber)
* - GetStatusHistoryAsync(int registrationId)
* - NotifyAsync(int registrationId, NotificationEvent eventType, int triggeredByUserId, string customMessage = null)

### IInstitutionService

* - GetByIdAsync(int id)
* - GetAllAsync(bool onlyActive = true)
* - InsertAsync(Institution institution)
* - UpdateAsync(Institution institution)
* - DeleteAsync(int id)
* - SearchAsync(string keyword, int? countryId, int? sectorId)
* - GetSummaryAsync()

### INotificationService

* - SendAsync(int registrationId, NotificationEvent eventType, int triggeredByUserId, string message = null, int recipientUserId = 0, NotificationChannel channel = InApp)

### IAuditTrailService

* - LogAsync(string tableName, string columnName, string pk, string oldValue, string newValue, int userId, string actionType, string entityName, int entityId, string clientIp = null, string note = null)
* - GetLogsAsync(string entityName, int entityId)

### IPermissionService

* - GetByIdAsync(int id)
* - GetBySystemNameAsync(string systemName)
* - GetAllAsync()
* - InsertAsync(Permission permission)
* - UpdateAsync(Permission permission)
* - DeleteAsync(int id)
* - AuthorizeAsync(int userId, string permissionSystemName)
* - GrantPermissionToRoleAsync(int roleId, int permissionId)
* - RevokePermissionFromRoleAsync(int roleId, int permissionId)
* - GetPermissionsByRoleIdAsync(int roleId)
* - GrantPermissionToUserAsync(int userId, int permissionId)
* - RevokePermissionFromUserAsync(int userId, int permissionId)
* - GetPermissionsByUserIdAsync(int userId)

### IRoleService

* - GetByIdAsync(int roleId)
* - GetAllAsync(bool onlyActive = true)
* - InsertAsync(Role role)
* - UpdateAsync(Role role)
* - DeleteAsync(int roleId)
* - GetRolesByUserIdAsync(int userId)
* - AddUserToRoleAsync(int userId, int roleId)
* - RemoveUserFromRoleAsync(int userId, int roleId)

### IAccessControlService

* - CanAsync(int userId, string entity, CrudAction action)
* - CanDoAsync(int userId, string permissionSystemName)