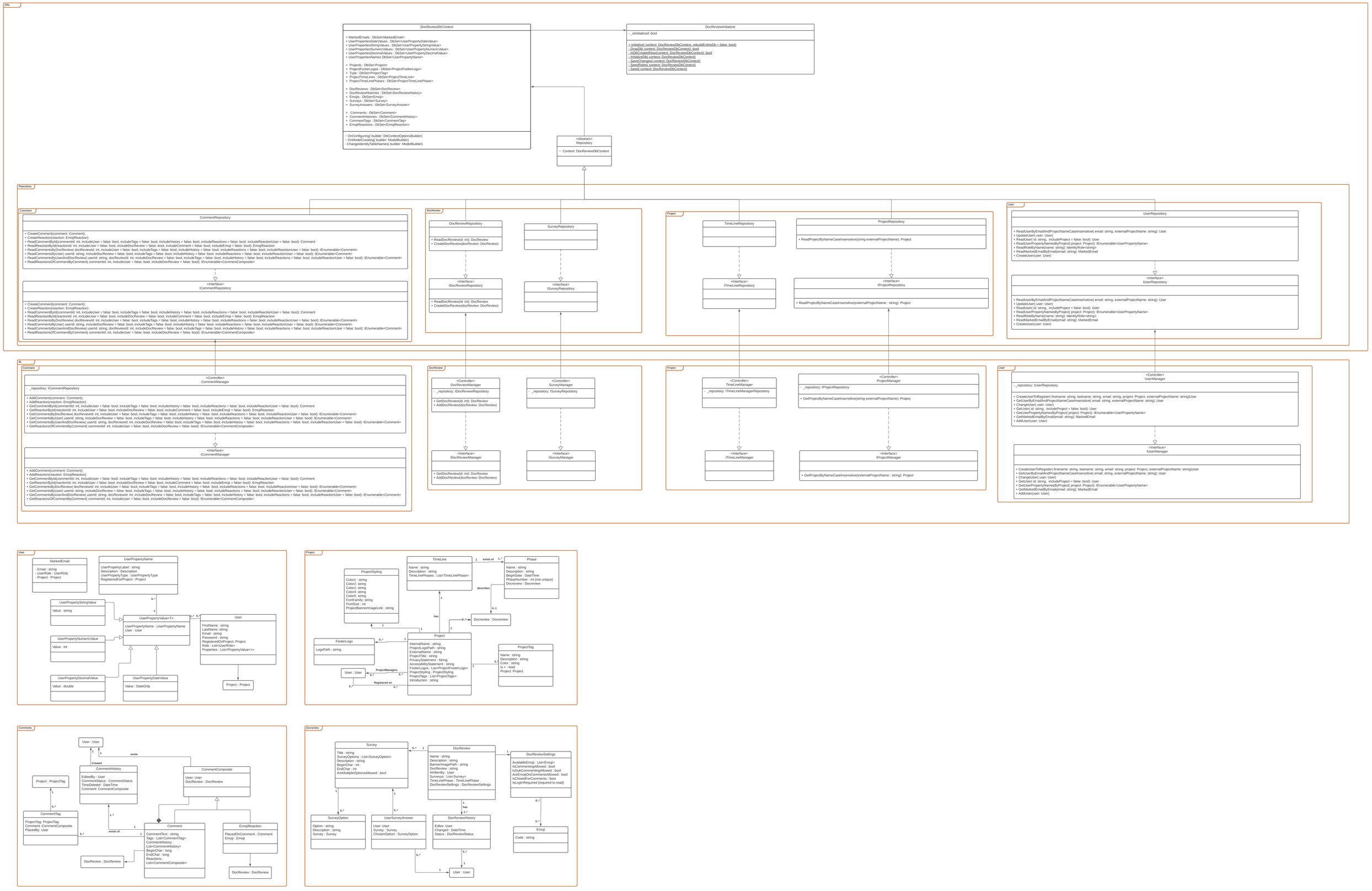
DAL
DocReviewDbContext
+ MarkedEmails : DbSet<MarkedEmail>
+ UserPropertiesDateValues : DbSet<UserPropertyDateValue>
+ UserPropertiesStringValues : DbSet<UserPropertyStringValue>
+ UserPropertiesNumericValues : DbSet<UserPropertyNumericValue>
+ UserPropertiesDecimalValues : DbSet<UserPropertyDecimalValue>
+ UserPropertiesNames DbSet<UserPropertyName>
+ Projects : DbSet<Project>
+ ProjectFooterLogos : DbSet<ProjectFooterLogo>
+ Type : DbSet<ProjectTag>
+ ProjectTimeLines : DbSet<ProjectTimeLine>
+ ProjectTimeLinePhases : DbSet<ProjectTimeLinePhase>
+ DocReviews : DbSet<DocReview>
+ DocReviewHistories : DbSet<DocReviewHistory>
+ Emojis : DbSet<Emoji>
+ Surveys : DbSet<Survey>
+ SurveyAnswers : DbSet<SurveyAnswer>
+ Comments : DbSet<Comment>
+ CommentHistories : DbSet<CommentHistory>
+ CommentTags : DbSet<CommentTag>
+ EmojiReactions : DbSet<EmojiReaction>
– OnConfiguring(builder: DbContextOptionsBuilder)
– OnModelCreating(builder: ModelBuilder)
- ChangeIdentityTableNames(builder: ModelBuilder)
DocReviewInitializer
- _isInitialized: bool
+ Initializer context: DocReviewDbContext, rebuildEntireDb = false: bool)
- DropDb(context: DocReviewDbContext): bool
- IsDbCreatedNow(context: DocReviewDbContext): bool
- InitializeDb(context: DocReviewDbContext)
- SaveChanges(context: DocReviewDbContext)
- SeedRoles(context: DocReviewDbContext)
- Seed(context: DocReviewDbContext)
<Abstract>
Repository
~ Context: DocReviewDbContext
Repository
Comment
CommentRepository
+ CreateComment(comment: Comment);
+ CreateReaction(reaction: EmojiReaction)
+ ReadCommentById(commentId: int, includeUser = false: bool, includeTags = false: bool, includeHistory = false: bool, includeReactions = false: bool, includeReactionUser = false: bool): Comment
+ ReadReactionById(reactionId: int, includeUser = false: bool, includeDocReview = false: bool, includeComment = false: bool, includeEmoji = false: bool): EmojiReaction
+ ReadCommentsByDocReview(docReviewId: int, includeUser = false: bool, includeTags = false: bool, includeHistory = false: bool, includeReactions = false: bool, includeReactionUser = false: bool): IEnumerable<Comment>
+ ReadCommentsByUser(userId: string, includeDocReview = false: bool, includeTags = false: bool, includeHistory = false: bool, includeReactions = false: bool, includeReactionUser = false: bool): IEnumerable<Comment>
+ ReadCommentsByUserAndDocReview(userId: string, docReviewId: int, includeDocReview = false: bool, includeTags = false: bool, includeHistory = false: bool, includeReactions = false: bool, includeReactionUser = false: bool): IEnumerable<Comment>
+ ReadReactionsOfCommentByComment(commentId: int, includeUser = false: bool, includeDocReview = false: bool): IEnumerable<CommentComposite>
<Interface>
ICommentRepository
+ CreateComment(comment: Comment);
+ CreateReaction(reaction: EmojiReaction)
+ ReadCommentById(commentId: int, includeUser = false: bool, includeTags = false: bool, includeHistory = false: bool, includeReactions = false: bool, includeReactionUser = false: bool): Comment
+ ReadReactionById(reactionId: int, includeUser = false: bool, includeDocReview = false: bool, includeComment = false: bool, includeEmoji = false: bool): EmojiReaction
+ ReadCommentsByDocReview(docReviewId: int, includeUser = false: bool, includeTags = false: bool, includeHistory = false: bool, includeReactions = false: bool, includeReactionUser = false: bool): IEnumerable<Comment>
+ ReadCommentsByUser(userId: string, includeDocReview = false: bool, includeTags = false: bool, includeHistory = false: bool, includeReactions = false: bool, includeReactionUser = false: bool): IEnumerable<Comment>
+ ReadCommentsByUserAndDocReview(userId: string, docReviewId: int, includeDocReview = false: bool, includeTags = false: bool, includeHistory = false: bool, includeReactions = false: bool, includeReactionUser = false: bool): IEnumerable<Comment>
+ ReadReactionsOfCommentByComment(commentId: int, includeUser = false: bool, includeDocReview = false: bool): IEnumerable<CommentComposite>
DocReview
DocReviewRepository
+ ReadDocReview(id: int): DocReview
+ CreateDocReview(docReview: DocReview)
SurveyRepository
<Interface>
IDocReviewRepository
+ ReadDocReview(id: int): DocReview
+ CreateDocReview(docReview: DocReview)
<Interface>
ISurveyRepository
Project
TimeLineRepository
ProjectRepository
+ ReadProjectByNameCaseInsensitive(string externalProjectName): Project
<Interface>
ITimeLineRepository
<Interface>
IProjectRepository
+ ReadProjectByNameCaseInsensitive(externalProjectName : string): Project
User
UserRepository
+ ReadUserByEmailAndProjectNameCaseInsensitive(email: string, externalProjectName: string): User
+ UpdateUser(user: User)
+ ReadUser(id: string, includeProject = false: bool): User
+ ReadUserPropertyNamesByProject(project: Project): IEnumerable<UserPropertyName>
+ ReadRoleByName(name: string): IdentityRole<string>
+ ReadMarkedEmailByEmail(email: string): MarkedEmail
+ CreateUser(user: User)
<Interface>
IUserRepository
+ ReadUserByEmailAndProjectNameCaseInsensitive(email: string, externalProjectName: string): User
+ UpdateUser(user: User)
+ ReadUser(id: string, includeProject = false: bool): User
+ ReadUserPropertyNamesByProject(project: Project): IEnumerable<UserPropertyName>
+ ReadRoleByName(name: string): IdentityRole<string>
+ ReadMarkedEmailByEmail(email: string): MarkedEmail
+ CreateUser(user: User)
BL
Comment
<Controller>
CommentManager
- _repository: ICommentRepository
+ AddComment(comment: Comment);
+ AddReaction(reaction: EmojiReaction)
+ GetCommentById(commentId: int, includeUser = false: bool, includeTags = false: bool, includeHistory = false: bool, includeReactions = false: bool, includeReactionUser = false: bool): Comment
+ GetReactionById(reactionId: int, includeUser = false: bool, includeDocReview = false: bool, includeComment = false: bool, includeEmoji = false: bool): EmojiReaction
+ GetCommentsByDocReview(docReviewId: int, includeUser = false: bool, includeTags = false: bool, includeHistory = false: bool, includeReactions = false: bool, includeReactionUser = false: bool): IEnumerable<Comment>
+ GetCommentsByUser(userId: string, includeDocReview = false: bool, includeTags = false: bool, includeHistory = false: bool, includeReactions = false: bool, includeReactionUser = false: bool): IEnumerable<Comment>
+ GetCommentsByUserAndDocReview(userId: string, docReviewId: int, includeDocReview = false: bool, includeTags = false: bool, includeHistory = false: bool, includeReactions = false: bool, includeReactionUser = false: bool): IEnumerable<Comment>
+ GetReactionsOfCommentByComment(commentId: int, includeUser = false: bool, includeDocReview = false: bool): IEnumerable<CommentComposite>
<Interface>
ICommentManager
+ AddComment(comment: Comment);
+ AddReaction(reaction: EmojiReaction)
+ GetCommentById(commentId: int, includeUser = false: bool, includeTags = false: bool, includeHistory = false: bool, includeReactions = false: bool, includeReactionUser = false: bool): Comment
+ GetReactionById(reactionId: int, includeUser = false: bool, includeDocReview = false: bool, includeComment = false: bool, includeEmoji = false: bool): EmojiReaction
+ GetCommentsByDocReview(docReviewId: int, includeUser = false: bool, includeTags = false: bool, includeHistory = false: bool, includeReactions = false: bool, includeReactionUser = false: bool): IEnumerable<Comment>
+ GetCommentsByUser(userId: string, includeDocReview = false: bool, includeTags = false: bool, includeHistory = false: bool, includeReactions = false: bool, includeReactionUser = false: bool): IEnumerable<Comment>
+ GetCommentsByUserAndDocReview(userId: string, docReviewId: int, includeDocReview = false: bool, includeTags = false: bool, includeHistory = false: bool, includeReactions = false: bool, includeReactionUser = false: bool): IEnumerable<Comment>
+ GetReactionsOfCommentByComment(commentId: int, includeUser = false: bool, includeDocReview = false: bool): IEnumerable<CommentComposite>
DocReview
<Controller>
DocReviewManager
- _repository: IDocReviewRepository
+ GetDocReview(id: int): DocReview
+ AddDocReview(docReview: DocReview)
<Controller>
SurveyManager
- _repository: ISurveyRepository
<Interface>
IDocReviewManager
+ GetDocReview(id: int): DocReview
+ AddDocReview(docReview: DocReview)
<Interface>
ISurveyManager
Project
<Controller>
TimeLineManager
- _repository: ITimeLineManagerRepository
<Controller>
ProjectManager
- _repository: IProjectRepository
+ GetProjectByNameCaseInsensitive(string externalProjectName): Project
<Interface>
ITimeLineManager
<Interface>
IProjectManager
+ GetProjectByNameCaseInsensitive(externalProjectName : string): Project
User
<Controller>
UserManager
- _repository: IUserRepository
+ CreateUserToRegister(firstname: string, lastname: string, email: string, project: Project, externalProjectName: string)User
+ GetUserByEmailAndProjectNameCaseInsensitive(email: string, externalProjectName: string): User
+ ChangeUser(user: User)
+ GetUser(id: string, includeProject = false: bool): User
+ GetUserPropertyNamesByProject(project: Project): IEnumerable<UserPropertyName>
+ GetMarkedEmailByEmail(email: string): MarkedEmail
+ AddUser(user: User)
<Interface>
IUserManager
+ CreateUserToRegister(firstname: string, lastname: string, email: string, project: Project, externalProjectName: string)User
+ GetUserByEmailAndProjectNameCaseInsensitive(email: string, externalProjectName: string): User
+ ChangeUser(user: User)
+ GetUser(id: string, includeProject = false: bool): User
+ GetUserPropertyNamesByProject(project: Project): IEnumerable<UserPropertyName>
+ GetMarkedEmailByEmail(email: string): MarkedEmail
+ AddUser(user: User)
User
MarkedEmail
- Email : string
- UserRole : UserRole
- Project : Project
UserPropertyName
UserPropertyLabel : string
Description : Description
UserPropertyType : UserPropertyType
RegisteredForProject : Project
UserPropertyStringValue
Value : string
UserPropertyValue<T>
UserPropertyName : UserPropertyName
User : User
User
FirstName : string
LastName: string
Email : string
Password : string
RegisteredOnProject: Project
Role : List<UserRole>
Properties : List<PropertyValue<>>
UserPropertyNumericValue
Value : int
UserPropertyDecimalValue
Value : double
UserPropertyDateValue
Value : DateOnly
Project : Project
0..*
1
0..*
0..*
Project
TimeLine
Name : string
Description : string
TimeLinePhases : List<TimeLinePhase>
1
exists of
1..*
Phase
Name : string
Description : string
BeginDate : DateTime
PhaseNumber : int {not unique}
Docreview : Docreview
describes
0..1
ProjectStyling
Color1 : string
Color2: string
Color2: string
Color4: string
Color5: string
FontFamily: string
FontSize : int
ProjectBannerImageLink : string
1
has
1
0..*
Docreview : Docreview
1
1
1
FooterLogo
LogoPath : string
0..*
1
Project
InternalName : string
ProjectLogoPath : string
ExternalName : string
ProjectTitle : string
PrivacyStatement : String
AccessibilityStatement : string
FooterLogos : List<ProjectFooterLogo>
ProjectStyling : ProjectStyling
ProjectTags : List<ProjectTags>
Introduction : string
0..1
1
ProjectTag
Name : string
Description : string
Color : string
Is + : bool
Project: Project
User : User
ProjectManagers
0..*
0..*
Registered on
0..*
0..*
Comments
User : User
1
1
wrote
Created
CommentHistory
EditedBy : User
CommentStatus : CommentStatus
TimeDeleted : DateTime
Comment: CommentComposite
CommentComposite
User: User
DocReview : DocReview
Project : ProjectTag
1
0..*
CommentTag
ProjectTag: ProjectTag
Comment: CommentComposite
PlacedBy: User
1..*
exists of
1
0..*
1
Comment
CommentText : string
Tags : List<CommentTag>
CommentHistory :
List<CommentHistory>
BeginChar : long
EndChar : long
Reactions :
List<CommentComposite>
EmojiReaction
PlacedOnComment : Comment
Emoji : Emoji
DocReview : DocReview
DocReview : DocReview
Docreview
Survey
Title : string
SurveyOptions : List<SurveyOption>
Description : string
BeginChar : int
EndChar : int
AreMultipleOptionsAllowed : bool
0..*
1
DocReview
Name : string
Description : string
BannerImagePath : string
DocReview : string
WritenBy : User
Surveys : List<Survey>
TimeLinePhase : TimeLinePhase
DocReviewSettings : DocReviewSettings
1
DocReviewSettings
AvailableEmoji : List<Emoji>
IsCommentingAllowed : bool
IsSubCommentingAllowed : bool
AreEmojiOnCommentsAllowed : bool
IsClosedForComments : bool
IsLoginRequired {required to read}
1
1
1
has
1..*
0..*
0..*
0..*
SurveyOption
Option : string
Description : string
Survey : Survey
UserSurveyAnswer
User: User
Survey : Survey
ChosenOption : SurveyOption
DocReviewHistory
Editor: User
Changed : DateTime
Status : DocReviewStatus
Emoji
Code : string
0..*
0..*
1
1
User : User
UM
<enum>
CommentStatus
Created,
Published,
Edited,
Removed,
Marked,
Inappropriate
<enum>
UserPropertyType
String,
Integer,
Double,
Date
<enum>
UserRole
Admin,
ProjectManager,
User
<enum>
DocReviewStatus
Created,
Published,
Archived,
Deleted