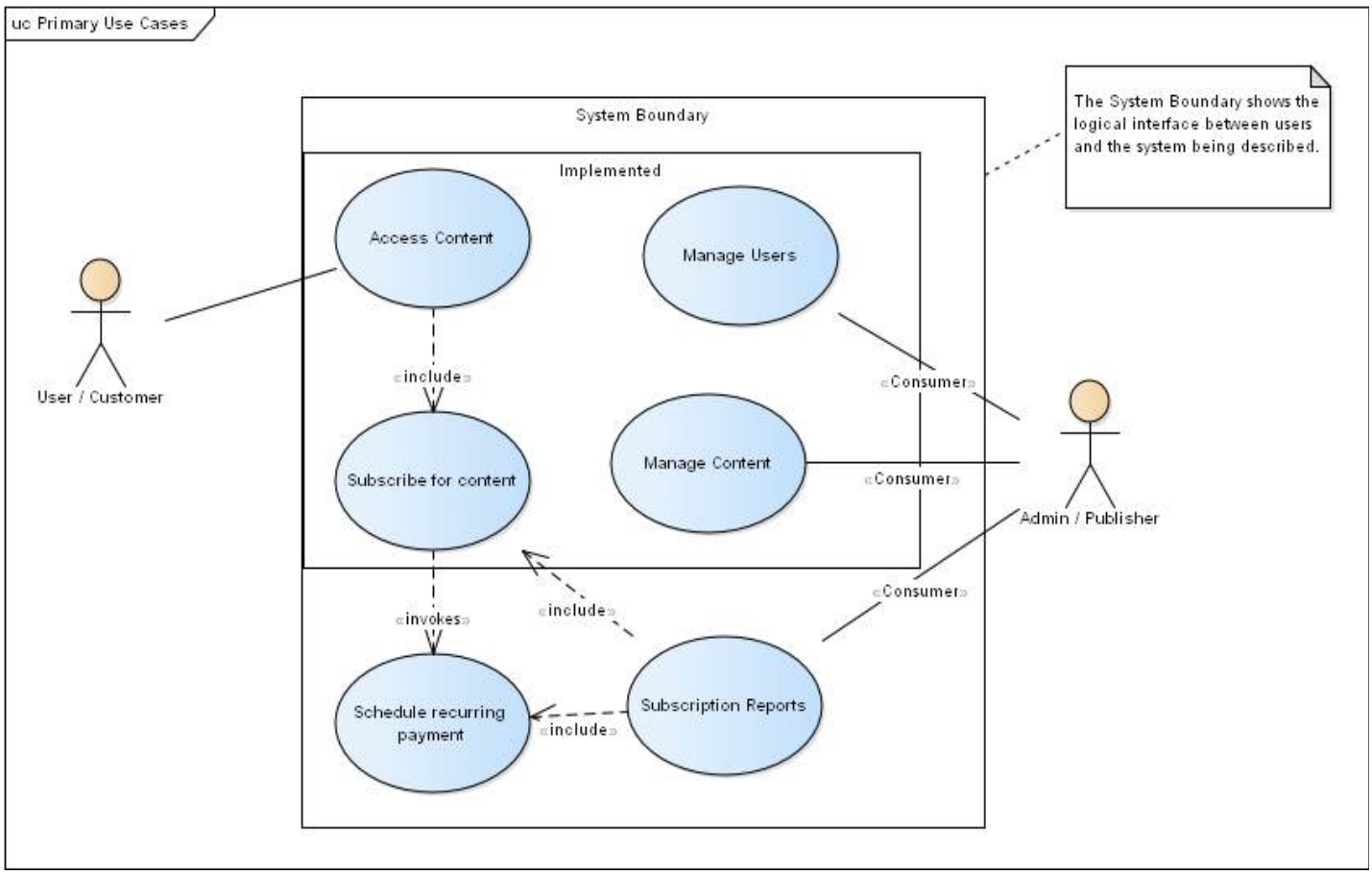
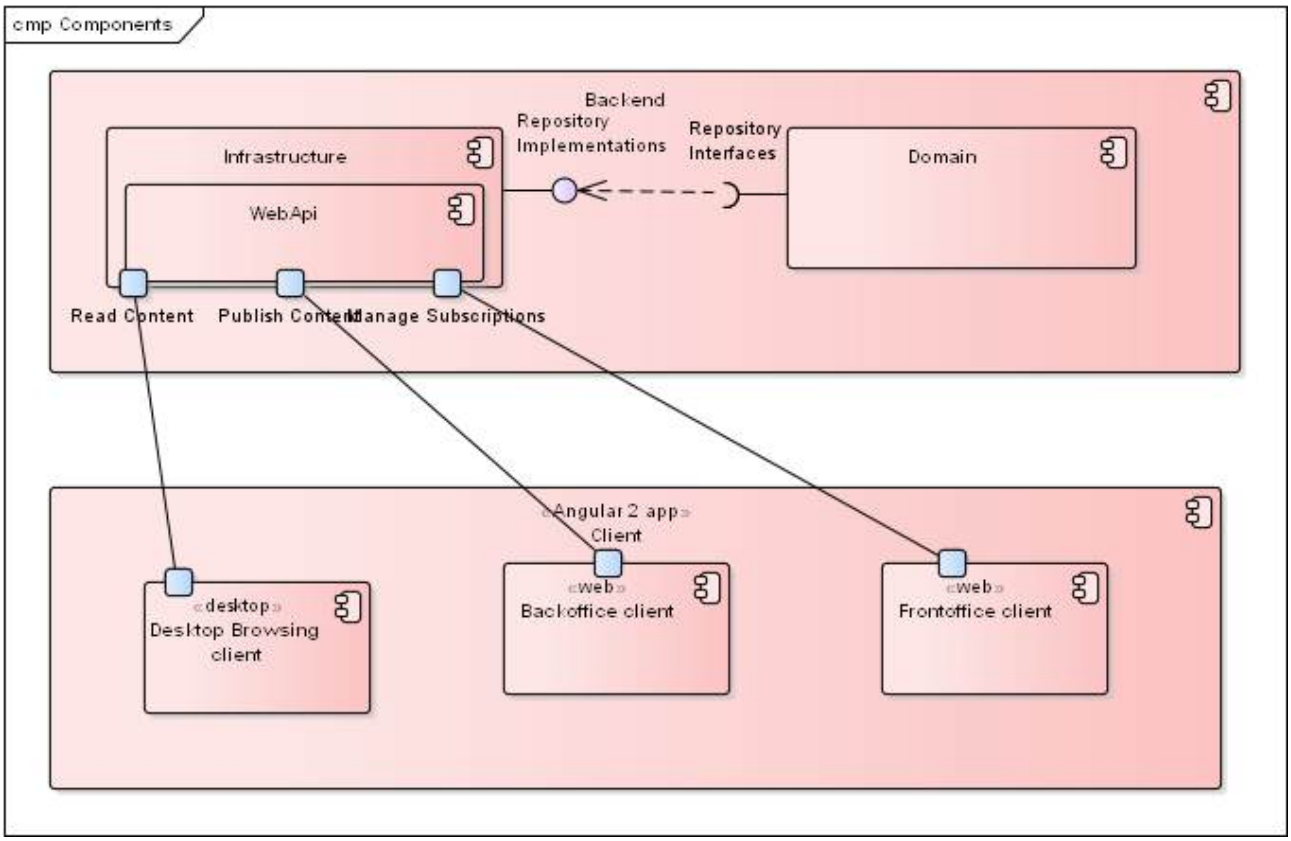
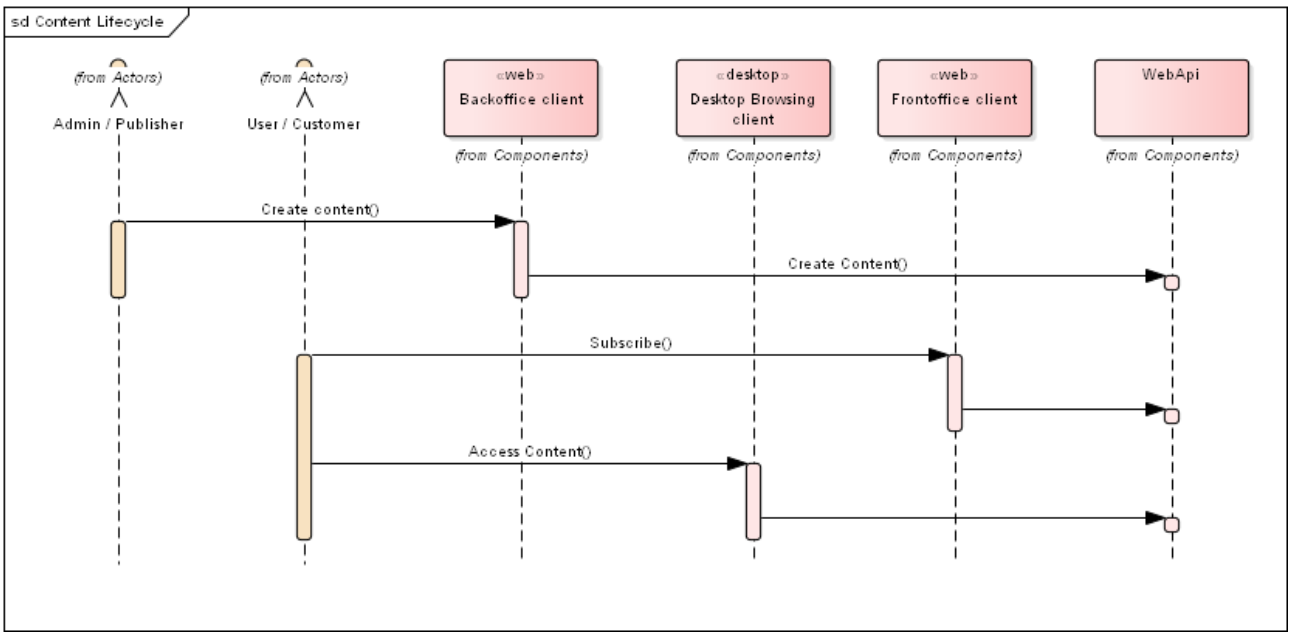
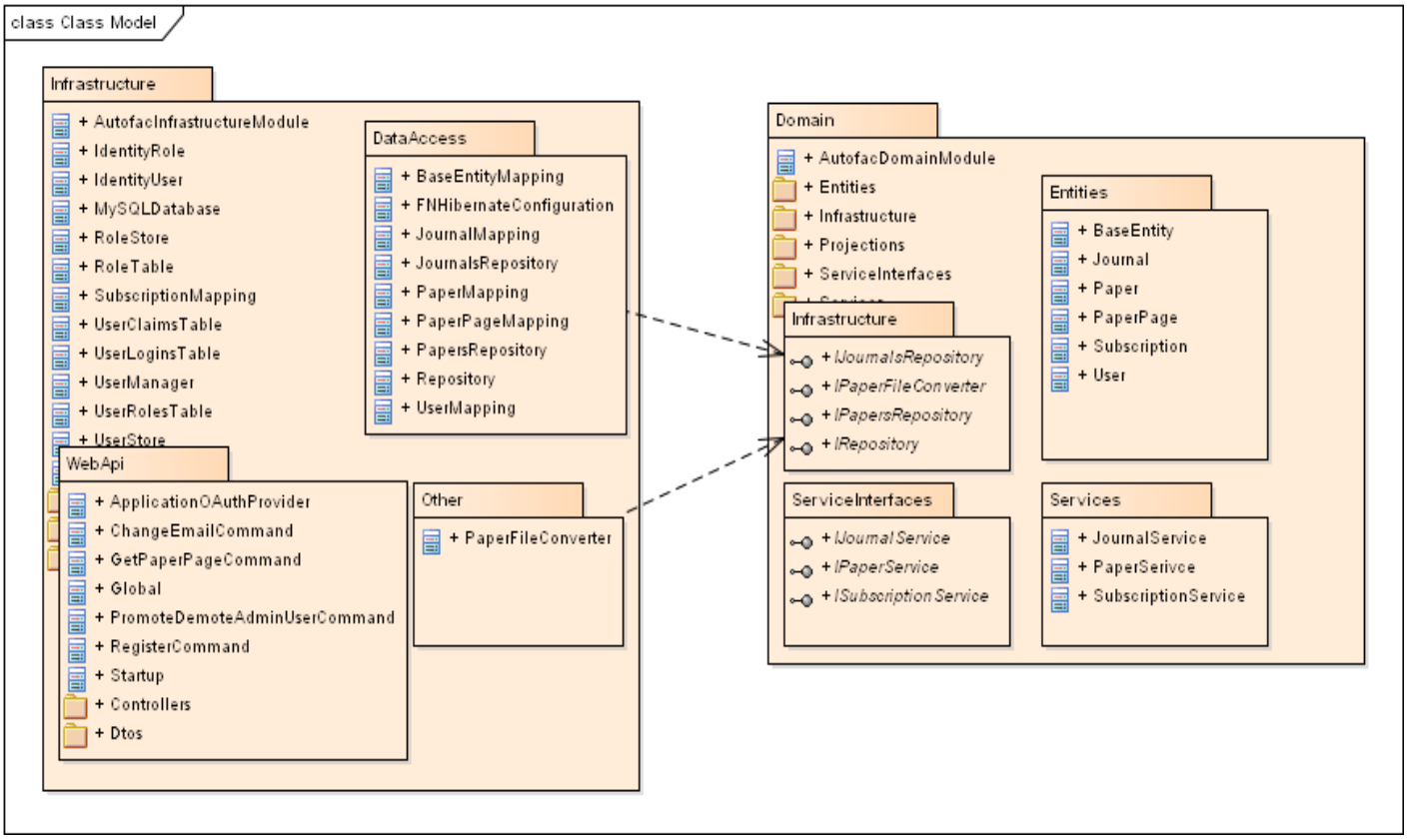
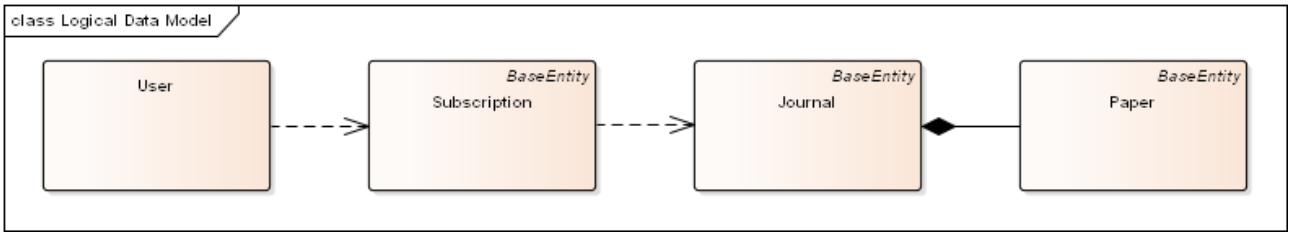
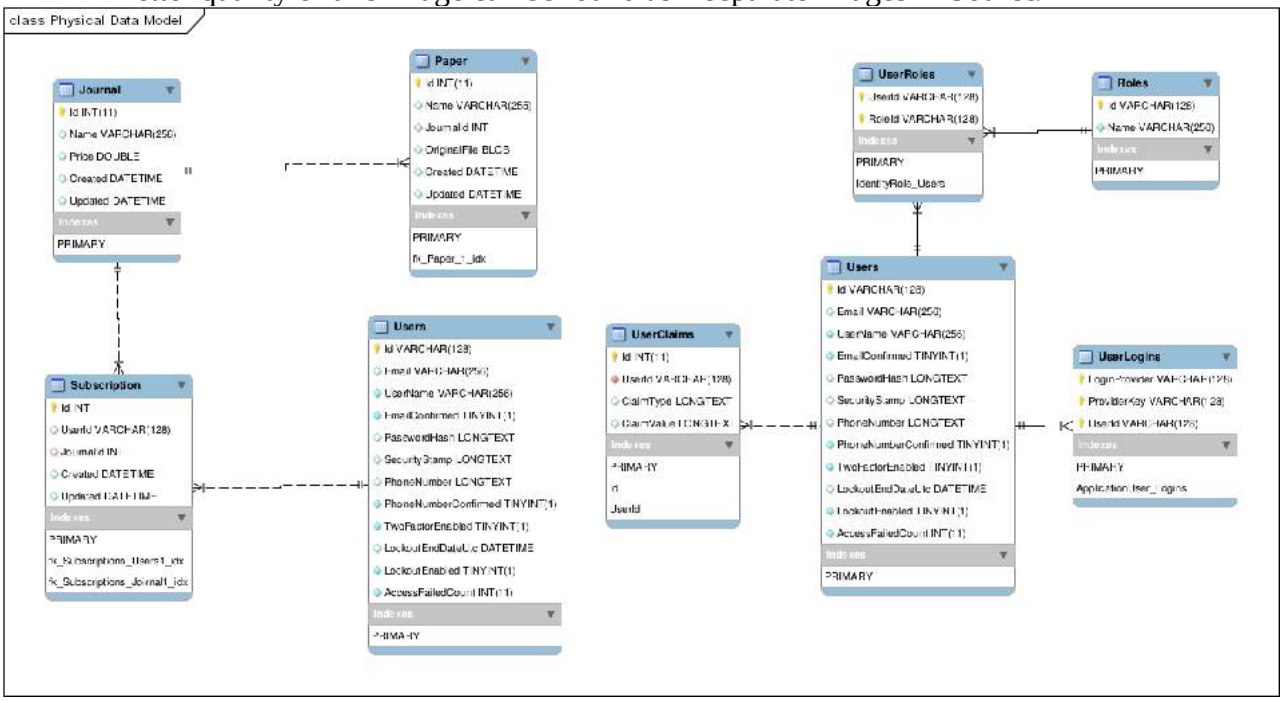

uc Primary Use Cases
System Boundary
Implemented
The System Boundary shows the logical interface between users and the system being described.
Access Content
Manage Users
Subscribe for content
Manage Content
Schedule recurring payment
Subscription Reports
User / Customer
Admin / Publisher
«include»
«Consumer»
«Consumer»
«Consumer»
«invokes»
«include»
«include»


cmp Components
Backend Repository Implementations
Repository Interfaces
Infrastructure
WebApi
Domain
Read Content
Publish Content
Manage Subscriptions
«Angular 2 app»
Client
«desktop»
Desktop Browsing client
«web»
Backoffice client
«web»
Frontoffice client


sd Content Lifecycle
(from Actors)
Admin / Publisher
(from Actors)
User / Customer
«web»
Backoffice client
(from Components)
«desktop»
Desktop Browsing client
(from Components)
«web»
Frontoffice client
(from Components)
WebApi
(from Components)
Create content()
Create Content()
Subscribe()
Access Content()


class Class Model
Infrastructure
+ AutofacInfrastructureModule
+ IdentityRole
+ IdentityUser
+ MySQLDatabase
+ RoleStore
+ RoleTable
+ SubscriptionMapping
+ UserClaimsTable
+ UserLoginsTable
+ UserManager
+ UserRolesTable
+ UserStore
DataAccess
+ BaseEntityMapping
+ FNHibernateConfiguration
+ JournalMapping
+ JournalsRepository
+ PaperMapping
+ PaperPageMapping
+ PapersRepository
+ Repository
+ UserMapping
WebApi
+ ApplicationOAuthProvider
+ ChangeEmailCommand
+ GetPaperPageCommand
+ Global
+ PromoteDemoteAdminUserCommand
+ RegisterCommand
+ Startup
+ Controllers
+ Dtos
Other
+ PaperFileConverter
Domain
+ AutofacDomainModule
+ Entities
+ Infrastructure
+ Projections
+ ServiceInterfaces
Infrastructure
+ IJournalsRepository
+ IPaperFileConverter
+ IPapersRepository
+ IRepository
Entities
+ BaseEntity
+ Journal
+ Paper
+ PaperPage
+ Subscription
+ User
ServiceInterfaces
+ IJournalService
+ IPaperService
+ ISubscriptionService
Services
+ JournalService
+ PaperSerivce
+ SubscriptionService



Better quality of this image can be found as 2 separate images in Source/