ADPQ2 Source Code Inventory

Com.Natoma.Adpq.Prototype.Api

Exposes business and database functionality to the web application

Root

Contains configuration and deployment details

Controllers

Public facing interface for handling JSON requests

Db

SQL to initiate the PostgreSQL database

Com.Natoma.Adpq.Prototype.Business

Contains business logic exposed by API project

Root

Contains configuration details

Data

Contains files needed for database interaction (POCOs that represent data tables and a context file that configures the data store)

Models

Contains files needed for representing data in services and controllers (POCOs and enums)

Auth

Contains models specific to the auth module

Notification

Contains models specific to the notification module

Request

Contains models specific to the request module

UserProfile

Contains models specific to the user profile module

Options

Provides strongly typed configuration

Services

Contains business services used by API controllers

Root

Contains the implementation details for each exposed service

Interfaces

Contains the definition for each exposed service

Utils

Contains utility classes

Com.Natoma.Adpq.Prototype.Web

Provides UI for the prototype web application

root

Contains configuration, deployment details, and readme

Wwwroot

Provides resources for public website

Css

Normal CSS

Html

Definition of each page comprising the website

Images

Images used in website

Js

Scripts to control website (folder generated by compiling Typescript classes. Front-end is Angular 2)

Root

Angular2 control scripts and scripts for pages that only need a single script

Admin

Scripts specific to admin module

Shared

Scripts shared across all modules

User

Scripts specific to user module

Lib

UI libraries used in browser

Dependencies

The various libraries downloaded from NPM necessary for the UI

App

TS scripts that generate the wwwroot.js scripts

Admin

TS scripts specific to the admin module

Shared

TS scripts shared between modules

User

TS scripts specific to the user module

Controllers

HomeController to bootstrap the application then Angular 2

Typings

Used by the Typescript compiler to enforce third party libraries' strong typing

Views

Two views used. One with the controller for the layout view to load JS/CSS for the non-angular third party libraries and the index view that bootstraps Angular 2