Mini Project

Overview

Solution overview

Problem description

package diagram

Requirements

Class Diagram

<https://app.diagrams.net/#G1hUByVmsAylQ70ZGkWXAyAPqdIp4VS-np#%7B%22pageId%22%3A%22C5RBs43oDa-KdzZeNtuy%22%7D>

Testing Plan

Language, scope, intent

Class list

Screenshots

Pointlessness of console app

Header and footer

Code repo

Use of system libraries for authorisation and authentication, excluded from scope. Some classes to be throw away for facades for pre-built libraries to provide this functionality

Coding standards – brief mention

Error handling and logging

* Façade for logging / placeholder

Mention that the different paint types will be similar here, but in the final product will render differently, hence requiring different classes (to justify the multiple inherticed objects and the use of abstracts)

In external libs mention dot net core

Menus

Main menu

* ‘login’ or logout
* Find paint – by colour, keyword in title , manufacturer or range
  + List paints (paged?)

Add paint to my collection

Remove paint from my collection

Manage users - crud

Manage paints - crud

Admin level??

Suggestions for improvement

* Colour groups – multiple member ships

https://stackoverflow.com/questions/10423267/unit-testing-a-console-application-inside-visual-studio