

Сессия 2

Модуль 4: Разработка desktop-приложений

Реализация отчетов точки проката горнолыжного курорта «Игора»

Реализуйте интерфейс для построения отчетов.

Реализуйте отчет по оказанным услугам:

- количество оказанных услуг по дням за период времени,
- количество заказов по дням за период времени по каждой услуге,
- количество заказов по дням за период времени.

Пользователь должен выбирать период с точностью до дня.

Все отчеты могут быть представлены в виде графика и таблицы, кроме того, пользователь может выбрать форму представления отчета: графиком или таблицей. Пользователь должен иметь возможность выгрузить данные отчеты в .pdf файл, выбрав один из трех вариантов выгрузки: только график, только таблица, график и таблица.

Разработка библиотеки

Разработайте библиотеку (отдельным проектом), которая позволит вернуть список свободных временных интервалов (заданного размера) в графике сотрудника точки проката горнолыжного курорта «Игора». Это необходимо для понимания того, сколько сотрудников должно быть на точке проката для уменьшения очередей в пиковые периоды, например, выходные или каникулярные дни.

Необходимо обязательно следовать правилам именования библиотек, классов и методов в них. В случае ошибок в рамках именования ваша работа не может быть проверена и ваш результат не будет зачтен. Классы и методы должны содержать модификатор public (если это реализуемо в рамках платформы), чтобы внешние приложения могли получить к ним доступ.

Вход:

- список занятых промежутков времени (в двух массивах: startTimes - начало, durations - длительность),
- минимальное необходимое время для консультации/оформления заказа (consultationTime),
- рабочий день сотрудника (начало - beginWorkingTime и завершение - endWorkingTime)

Выход:

- список подходящих свободных временных промежутков (в массив строк формата HH:mm-HH:mm)

Требования к именованиям и форматам:

	C#	Java	Python
Библиотека классов	SF2022User{NN}Lib.dll	SF2022User{NN}Lib.jar	SF2022User{NN}Lib
Название класса	Calculations	Calculations	Calculations
Название метода	AvailablePeriods()	availablePeriods()	available periods()
Входящие обязательные параметры	TimeSpan[] startTimes, int[] durations, TimeSpan beginWorkingTime, TimeSpan endWorkingTime, int consultationTime	LocalTime[] startTimes, int[] durations, LocalTime beginWorkingTime, LocalTime endWorkingTime, int consultationTime	time[] start times, int[] durations, time begin working time, time end working time, int consultation time
Возвращаемые параметры	string[]	string[]	string[]

Пример:

Вход	Выход
startTime duration	08:00-08:30
10:00 60	08:30-09:00
11:00 30	09:00-09:30
15:00 10	09:30-10:00
15:30 10	11:30-12:00
16:50 40	12:00-12:30
Working Times	12:30-13:00
08:00-18:00	13:00-13:30
Consultation Time	13:30-14:00
30	14:00-14:30
	14:30-15:00
	15:40-16:10
	16:10-16:40
	17:30-18:00

Модуль 7: Тестирование программных решений

Модульные тесты

Реализуйте 10 unit-тестов на основе технологии TDD для библиотеки, функционал которой описан ранее. Учитывайте требования к именованиям и входным данным, а так же формату выходных данных.

Важно, чтобы тестовые данные в unit-тестах предусматривали различные ситуации. Например, недостаточное время в промежутках между ранее созданными интервалами, либо в начале рабочего дня, либо в конце рабочего дня; различная длительность и т.д.

Тестовая документация

Для выполнения процедуры тестирования прецедента «Формирование заказа» Вам нужно описать пять сценариев. Добавление заказа может быть выполнено успешно, а может быть отклонено согласно требованиям предметной области. Необходимо, чтобы варианты тестирования демонстрировали различные исходы работы алгоритма. Для описания тестовых сценариев в ресурсах предоставлен шаблон testing-template.docx.

Модуль 6: Создание инсталляторов

Разработка инсталлятора

Разработайте установочный файл для desktop-приложения под ОС Windows.

Модуль 1: Анализ и проектирование требований, бизнес-процессов

Диаграмма деятельности

Разработайте диаграмму деятельности для прецедента «Формирование заказа». Сохраните файл с диаграммой в форматах .vsdx и .pdf.

Проектирование user story

Разработайте user story на основе технологии BDD для прецедента «Формирование отчета». User story предоставьте на проверку отдельным файлом согласно представленного шаблона в ресурсах.