**Тестовое задание на должность “Программист”**  
  
Задача: визуализация графика работ с возможностью его редакции.  
  
К письму приложен xml файл (test\_task\_model) следующего содержания:

class Model

{

List<Graph> graph //график работ

List<Visual> visual //5 уровней визуализации

}

class Graph

{

int id\_sublayer1 //идентификатор этажа

int id\_technology\_card //идентификатор технологической карты

DateTime start\_date //дата начала

DateTime end\_date //дата окончания

}

class Visual

{

int id //идентификатор уровня

int? parent\_id //идентификатор родителя

string name //название этапа, или технологической карты (если уровень 5)

int? id\_technology\_card //идентификатор технологической карты, если это уровень 5

DateTime start\_date //дата начала (минимальная дата начала его детей, если это не уровень 5)

DateTime end\_date //дата окончания (максимальная дата окончания его детей, если это не уровень 5)

}

Необходимо реализовать визуализацию данных из файла в виде, примерно отображённом на скриншоте.



Состав xml файла:  
  
1) **Visual** – данные для визуализации различных этапов графика, с их датами.  
Visual состоит из элементов пяти уровней, связи между которыми заданы с помощью parent\_id.  
Где первый уровень parent\_id == null и последний пятый уровень id\_technology\_card != null.   
И дат начала и окончания каждого элемента.  
Каждый уровень, кроме пятого имеет своих детей, и его даты зависят от сроков его детей.

2)**Graph** – график работ, на основе которого построен Visual.  
Включает в себя идентификатор технологической карты (которая является уровнем 5 из Visual).  
И идентификатор этажа, на котором производятся работы.  
Между датами работ по этажам, могут быть пропуски.  
  
Необходимо:

**А) Визуализировать информацию.**1) Квадратики - Каждый месяц делится на 3 части: c 1 по 10, с 11 по 20, с 21 по конец месяца. Если этап входит в данный период квадратика, квадратик закрашивается.  
2) Кнопки плюс и минус. Возможность сворачивать и разворачивать каждый этап с его под-этапами.  
3) Каждый квадратик. Закрашенный или не закрашенный должен быть интерактивным.   
При нажатии на него должно открыться окно с указанием:  
-периода за который он отвечает  
-названием уровня, к которому он относится  
4) Даты начала и окончания. Элемент datetimepicker. Даты по данному этапу.

**Б) Реализовать возможность изменения графика.**

**Важно. Все сдвиги, сжатия и разжатия необходимо производить не в Visual, а меняя сам график Graph. А после его изменения, уже на его основе изменять Visual.**

Предусмотреть 2 способа изменения графика:  
  
**1) Изменение уровней 1 2 3 4**

Пользователь может изменить у данных уровней дату начала или окончания.  
Данное изменение должно делать сдвиг всех технологических карт, которые входят в данный уровень.

Пример:   
Уровень 3 включает в себя 3 уровня 4. Каждый из уровней 4, включает в себя по 5 технологических. Итого необходимо сделать сдвиг, 20 технологических карт.  
  
Увеличив дату окончания с 1 на 10 число, производится сдвиг на 9 дней данного элемента.  
Пример:  
Изменили дату начала с 1 на 10 число.  
Было:  
Начало уровня 3, 1.10.2010  
Конец уровня 3, 20.12.2019  
Стало  
Начало уровня 3, 10.10.2010  
Конец уровня 3, 29.12.2019

**2) Изменение уровня 5**В отличии от изменения уровней 1 2 3 и 4. Происходит НЕ сдвиг, а именно смена даты. (сжатие или разжатие).  
Дата начала не может быть больше даты окончания.  
После того как сжатие и разжатие произошло, могут изменится даты 1 2 3 и 4 уровней.  
  
Graph, помимо технологической карты включает в себя еще и id\_sublayer1. Это идентификатор конкретного этажа, где производится работа. Если продолжительность работ увеличивают, то увеличивается промежуток производства работ у ПОСЛЕДЕГО этажа. Если работу “сжимают”, то уменьшается время производства работ у последнего этажа, пока она не станет ровна одному дню.   
  
НЕЛЬЗЯ сделать сжатие таким, чтобы продолжительность работ была меньше чем число этажей.

**Тех требования:**Необходимо реализовать данное задание в виде отдельного windows приложения.  
Реализовать его на C# с использованием WPF.  
  
Время открытия приложения: не более 5 секунд.  
Время на любой сдвиг с перестройкой визуализации: не более 2 секунд.  
  
**Дополнительные вопросы можно задавать на email: soft@ibpspb.ru**