

[Form, C++] (50 баллов) Планировщик расходов. На основной форме располагаются следующие элементы:

- поле для ввода текущего бюджета (*QLineEdit*) (значение не может быть отрицательным);
- список предполагаемых расходов (*QListWidget*); элементы списка должны отображать название расхода, дату и сумму, а также содержать кнопку, удаляющую элемент из списка;
- кнопка для создания нового расхода (*QPushButton*);
- текстовое поле (*QLabel*), в котором отображается остаток бюджета после вычета расходов (если остаток становится отрицательным, то цвет текста должен поменяться (при неотрицательном остатке цвет текста - зелёный, при отрицательном - красный)).

При нажатии на кнопку добавления в список расходов появляется модальный диалог, в котором пользователь должен выбрать дату (*QDateEdit*), ввести сумму и название расхода. Должна осуществляться валидация полей, введённых пользователем (дата не должна быть раньше текущей, сумма отрицательной, название может состоять только из букв латинского алфавита, цифр и символа пробела, не допускается использование пробелов в начале или конце названия, название должно содержать хотя бы один символ, отличный от символа пробела, названия расходов не могут совпадать; при непройденной валидации кнопка добавления должна стать неактивной). После ввода корректных данных и нажатии кнопки ОК добавляется запись о новом расходе в список расходов на основной форме.

При изменении списка расходов или поля текущего бюджета остаток должен пересчитываться. Реализовать возможности сохранения данных (списка расходов и бюджета) в файл и загрузки данных из файла (с валидацией).

Приложение должно содержать меню с пунктами About и Exit. Приложение должно быть реализовано с применением архитектурного паттерна проектирования (MVC, MVP, MVVM).

Срок сдачи: 23:00 12.12.2018