

Opis wymagań

Łukasz Szewczyk

12 czerwca 2013

1 Wymagania

„Określenie wymagań, jakie musi spełniać oprogramowanie, jest tym miejscem projektu, w którym najbardziej i najwyraźniej stykają się interesy wszystkich jego udziałowców.” [1]

2 Wymagania funkcjonalne

2.1 Wykres liniowy

2.1.1 Tworzenie wykresu

Biblioteka musi umożliwiać utworzenie wykresu liniowego prezentującego dane z jednej lub kilku serii danych. Na wykres liniowy składają się:

- układ współrzędnych, zbudowany na dwóch osiach – jednej poziomej, odpowiadającej dziedzinie i jednej pionowej, odpowiadającej zbiorowi wartości,
- serie danych,
- łamane prezentujące dane z serii dostarczonych do wykresu, posiadające nazwę oraz unikalny w danym wykresie kolor,
- siatka, czyli zbiór prostopadłych linii dopasowanych do osi, ułatwiających szybkie, wzrokowe porównywanie danych,
- tytuły krzywych,
- tytuł wykresu,
- tytuły osi,
- legenda,
- tło.

Wyświetlanie każdego z elementów powinno być opcjonalne.

2.1.2 Modyfikacja stanu wykresu

Już po utworzeniu wykresu powinny być możliwe do skonfigurowania następujące elementy:

- zawartość serii danych,
- kolor krzywej,
- nazwa krzywej,
- dowolny z parametrów osi,
- dowolny z parametrów siatki,
- kolor/grafika tła.

2.1.3 Dostępne operacje

Operacje, które powinny być dostępne dla wykresu liniowego:

- skalowanie,
- zoom,
- zaznaczenie przedziałów dziedziny,
- zaznaczenie wierzchołków łamanej,
- zaznaczenie łamanych.

2.2 Wykres słupkowy

2.2.1 Tworzenie wykresu

Biblioteka musi umożliwiać utworzenie wykresu słupkowego prezentującego dane z jednej lub kilku serii danych. Wykres ten ma przyjmować jedną z dwóch orientacji: horyzontalną bądź wertykalną. Wertykalny wykres słupkowy jest czasem nazywany wykresem kolumnowym. Na wykres słupkowy składają się:

- układ współrzędnych, zbudowany na dwóch osiach – jednej poziomej i jednej pionowej, z których jedna odpowiada dyskretnej dziedzinie, a inna ciągłemu zbiorowi wartości,
- serie danych,
- prostokątne słupki o wysokości proporcjonalnej do reprezentowanej wartości. Kolor wszystkich słupków z danej serii danych jest jednaki i unikalny w skali wykresu,

- siatka,
- tytuł wykresu,
- tytuły osi,
- legenda,
- tło.

Wyświetlanie każdego z elementów powinno być opcjonalne.

2.2.2 Modyfikacja stanu wykresu

Już po utworzeniu wykresu powinny być możliwe do skonfigurowania następujące elementy:

- zawartość serii danych,
- kolor grupy słupków,
- nazwa grupy słupków,
- dowolny z parametrów osi,
- dowolny z parametrów siatki,
- kolor/grafika tła.

2.2.3 Dostępne operacje

Operacje, które powinny być dostępne dla wykresu słupkowego:

- skalowanie,
- zoom,
- zaznaczenie przedziałów dziedziny,
- zaznaczenie słupków,
- zaznaczenie grupy słupków.

2.3 Wykres kołowy

2.3.1 Tworzenie wykresu

Biblioteka musi umożliwiać utworzenie wykresu kołowego prezentującego dane z jednej serii danych. Na wykres kołowy składają się:

- seria danych

- wycinki kołowe o wspólnym środku, których kąt środkowy jest proporcjonalny do reprezentowanej wartości. Suma kątów środkowych wszystkich wycinków wynosi 360 stopni. Kolor danego wycinka jest unikalny w skali wykresu.
- tytuł wykresu,
- tytuły wycinków,
- legenda,
- tło.

Wyświetlanie każdego z elementów powinno być opcjonalne.

2.3.2 Modyfikacja stanu wykresu

Już po utworzeniu wykresu powinny być możliwe do skonfigurowania następujące parametry:

- zawartość serii danych,
- kolor wycinka,
- tytuł wycinka,
- kolor/grafika tła.

2.3.3 Dostępne operacje

Operacje, które powinny być dostępne dla wykresu kołowego:

- skalowanie,
- zaznaczenie wycinków,
- zaznaczenie całego wykresu,
- obracanie wykresu,
- przesuwanie wycinków.

2.4 Elementy składowe wykresu

2.4.1 Serie danych

Biblioteka musi obsługiwać próbki oraz serie danych. Każda z próbek powinna posiadać następujące parametry:

- wartość liczbowa z dziedziny,

- co najmniej jedną wartość liczbową, która będzie prezentowana, np. w postaci słupka

Seria danych powinna zawierać zbiór próbek. Niektóre z wykresów mogą wymagać, aby było to zbiór posortowany ze względu na wartość z dziedziny. Umożliwi to poprawne odwzorowanie próbek na wykresach.

Wymagane są operacje dodawania, modyfikowania i usuwania danych z serii. Każda z tych operacji powinna skutkować aktualizacją wykresu powiązanego z daną serią.

2.4.2 Legenda

Dla wykresów obsługujących wiele serii danych, powinna prezentować kolory oraz tytuły tych serii. Natomiast dla wykresów jednoseryjnych prezentowana powinna być informacja o kolorze i tytule każdej z próbek. Legenda powinna reagować na zmianę stanu wykresu, w szczególności na zmianę danych. Legenda powinna przyjmować jedną z dwóch orientacji: horyzontalną lub wertykalną.

2.4.3 Siatka

Element składający się ze zbioru prostopadłych linii, ułatwiający porównywanie danych różnych próbek. Powinna istnieć możliwość określenia grubości i koloru linii oraz ziarnistości samej siatki.

2.4.4 Osie

Oś ma być reprezentowana przez odcinek o jednym z końców zaznaczonym strzałką. Odcinek ten ma być przecinany krótkimi, prostopadłymi odcinkami oznaczającymi kolejne wartości z przypisanej do osi serii. Takie znaki na osi zwane są tick-ami. Każdy z tick-ów powinien posiadać podpis. Powinna być możliwość zmiany następujących parametrów osi:

- kolor,
- grubość,
- tytuł,
- widoczność tytułu,
- widoczność podpisów tick-ów.

Opcjonalnie dla dwóch osi tworzących układ współrzędnych może istnieć możliwość ustalenia punktu przecięcia innego niż $(0,0)$.

2.4.5 Tło

Tłem wykresu może być kolor, gradient bądź obrazek.

2.5 Operacje na wykresach

2.5.1 Skalowanie

Powinno być możliwe skalowanie wykresu. Wykres powinien dostosowywać swój rozmiar do przekazanego mu obszaru przeznaczonego na jego odryśowanie.

2.5.2 Zoom

Musi istnieć możliwość przybliżenia konkretnego fragmentu wykresu. Przybliżanie powinno być możliwe dla jednej z osi bądź dla obu jednocześnie. Oddalanie jako operacja symetryczna również powinna być dostępna.

2.5.3 Zaznaczanie przedziałów dziedziny

Powinno być możliwe zaznaczenie elementów z zadanego przedziału dziedziny. Zaznaczenie powinno być widoczne przez narysowanie przezroczystego prostokąta pokrywającego wszystkie żądane elementy i tylko te elementy. Kolor prostokąta i poziom jego przezroczystości powinny być konfigurowalne.

2.5.4 Zaznaczenie elementów reprezentujących próbki

Dla wykresu liniowego zaznaczenie będzie sygnalizowane poprzez narysowanie okręgu wokół wybranych wierzchołków łamanej. Z kolei dla wykresów słupkowego i kołowego będzie to narysowanie przylegającej ramki wokół odpowiednio słupków lub wycinków. Grubość i kolor obramowań powinny być konfigurowalne.

2.5.5 Zaznaczenie elementów reprezentujących serie danych

Powinna istnieć możliwość zaznaczenia wszystkich elementów reprezentujących dane z wybranej serii. Dla wykresu liniowego będzie to łamana, dla słupkowego grupa słupków, a dla kołowego cały wykres. Zaznaczenie wybranych elementów powinno skutkować obrysowaniem ich przylegającą ramką. Grubość i kolor obramowań powinny być konfigurowalne.

2.5.6 Obracanie wykresu

Powinna być możliwość obracania wykresu kołowego względem jego środka o zadany kąt. Proces ten powinien być możliwy do animacji z zastosowaniem standardowych rozwiązań Qt.

2.5.7 Przesuwanie wycinków

Powinna istnieć możliwość przesuwania wycinków wzdłuż prostej wyznaczonej przez promień dzielący wycinek na połowy. Powinna być przyjmowana nieujemna liczba pixeli, o którą wycinek powinien zostać przesunięty względem środka. Proces przesuwania wycinków powinien być możliwy do animacji z zastosowaniem standardowych rozwiązań Qt.

2.6 Interfejsy

Głównym celem mojej pracy inżynierskiej jest stworzenie uniwersalnego silnika do tworzenia wykresów. Tym niemniej możliwe i porządane jest stworzenie dwóch przykładowych interfejsów wykorzystujących nowopowstały silnik. Potencjalne interfejsy powinny powstać zarówno dla klasycznego Qt jak i Qt Quick 2.

2.6.1 Interfejs C++

Interfejs dla klasycznego Qt powinien możliwie dobrze wpisywać się w styl Qt opisany w artykule [2].

2.6.2 Interfejs QML

Interfejs ten powinien spełniać paradygmaty programowania deklaratywnego.

2.7 Przenośność

Biblioteka musi wpisywać się w politykę Qt brzmiącą: *pisz raz, kompiluj wielokrotnie*. Musi być przenośna na poziomie kodu źródłowego między najważniejszymi wspieranymi przez Qt platformami. Minimum to uruchomienie na systemach:

2.7.1 Windows NT

2.7.2 Linux (Ubuntu)

2.8 Efekty graficzne

Wszystkie opisane tu funkcjonalności są opcjonalne, a ich realizacja nie jest konieczna do zakończenia prac nad biblioteką.

2.8.1 Motywy

Dodatkiem, który podniósłby atrakcyjność wykresów jest wysokopoziomowy mechanizm motywów, podobny do *QStyle* [3]. Z jego pomocą, tworzenie zbioru

ru wykresów o jednolitej kolorystyce i czcionkach stałoby się łatwe. Zmiana motywu dla danego wykresu powinna sprowadzać się do prostej operacji.

2.8.2 Animacje

Powinno być możliwe animowanie procesu budowania wykresu. Mechanizm ten powinien korzystać z dostępnego API Qt[4] i działać w sposób spójny z już istniejącym framework-iem.

2.8.3 Generowanie grafik

Powinno być możliwe generowanie plików graficznych na podstawie istniejących wykresów w formatach .PNG i .SVG.

3 Wymagania pozafunkcjonalne

3.1 Struktura biblioteki

Biblioteka powinna mieć przejrzystą strukturę, która umożliwi programistom sprawne przeanalizowanie jej działania i szybkie przystąpienie do tworzenia wykresów.

3.2 Wymiennność biblioteki

Biblioteka powinna wykorzystywać mechanizmy pozwalające na tworzenie bibliotek dynamicznych wymiennych pomiędzy wersjami. Wprowadzenie nowej wersji biblioteki z niezmiennym interfejsem nie powinno wymagać przebudowania całej aplikacji.

3.3 Nowoczesność i uniwersalność

Biblioteka powinna wykorzystywać możliwie nowe technologie, m.in. Qt5. Jednak użycie standardu *C++11* nie jest wskazane ze względu na ograniczenie liczby potencjalnych odbiorców. Komponenty dostarczane do użytku programistom powinny być możliwie wysokopoziomowe i uniwersalne w użyciu.

3.4 Wydajność

Jako, że okoliczności wykorzystania wykresów biurowych są znacząco inne niż wykresów technicznych oraz natura ich danych jest dużo bardziej statyczna, optymalizacja nie jest tu kwestią najważniejszą.

3.4.1 Wydajność wykresów w uk. współrzędnych

Wstępnie zakłada się, że operacje dokonywane na wykresach osadzonych w układzie współrzędnych powinny działać względnie płynnie dla danych do tysiąca próbek. Nie jest wykluczona późniejsza optymalizacja pozwalająca komfortowo operować na większych ilościach danych.

3.4.2 Wydajność wykresu kołowego

Wykres kołowy, nie powinien powodować „zamrażania aplikacji” przy liczbie próbek nie przekraczającej 100.

3.5 Niezawodność

Jak już zostało wcześniej stwierdzone, natura oraz zastosowania wykresów biurowych różnią się od technicznych, a co za tym idzie, mają również inne wymagania dotyczące niezawodności. Przewiduje się, że biblioteka będzie przeznaczona do aplikacji finansowych i biurowych, a nie systemów czasu rzeczywistego. Jednakowoż w celu minimalizacji liczby błędów w kodzie, powinny zostać zastosowane testy jednostkowe. Zachowanie biblioteki w warunkach ekstremalnych, np. przepełnienia stosu, nie jest głównym celem projektu.

3.6 Skalowalność

Zarówno dodawanie nowych jak i usuwanie już istniejących elementów biblioteki powinno być łatwe i nie powinno mieć wpływu na stabilność pracy biblioteki. Dodawanie nowych elementów powinno być możliwe dzięki uniwersalnym interfejsom. Natomiast usuwanie istniejących elementów powinno sprowadzać się do wyłączenia ich z procesu kompilacji biblioteki.

Literatura

- [1] Inżynieria oprogramowania – rozdział 2, Krzysztof Sacha, Wydawnictwo Naukowe PWN, 2010, ISBN: 978-83-01-16179-8
- [2] API w stylu Qt <http://qt-project.org/wiki/API-Design-Principles>
- [3] QStyle <http://doc.qt.digia.com/4.7-snapshot/qstyle.html#details>
- [4] Animacje w Qt <http://doc.qt.digia.com/4.7/animation-overview.html>