

Programy wieloplatformowe

Często zdarza się, że chcemy napisać aplikację wieloplatformową tak, aby cały jej kod był przenośny. Co więcej, niektóre aplikacje dostają informację o urządzeniu dopiero podczas uruchomienia. Efekt ten można zauważyć nawet na różnych wersjach tej samej platformy - aplikacje na Androida wyglądają inaczej na różnych wersjach.

Twoim zadaniem będzie napisanie systemu który w prosty sposób umożliwia kompletną podmianę całego GUI.

Projekt zawiera:

1. Program.cs który zawiera funkcję main.
2. Interfejsy IDialogWindow, ISlider, IButton

Założenia:

1. Zawsze na raz będzie używana tylko jedna platforma.
2. Nie wolno modyfikować dostarczonych interfejsów
3. Program użytkownika (udawany przez stworzoną przez ciebie funkcję- szczegóły poniżej) powinien być nieświadomy wybranej platformy. Powinien mieć jednak możliwość tworzenia elementów interfejsu użytkownika.
4. Dodanie kolejnej platformy nie powinno mieć żadnego wpływu na program użytkownika.

Twoje zadanie:

1. Implementacja interfejsów dla trzech różnych platform wg wytycznych podanych poniżej.
2. Implementacja funkcji udającej program użytkownika wg wytycznych podanych poniżej.
3. Stworzenie funkcji main, która stanowić będzie tester naszego rozwiązania. Ma ona uruchomić program użytkownika (funkcja z pkt. 2) z wykorzystaniem wszystkich z dostępnych platform.

Implementacja interfejsów:

Konstruktor każdego z obiektów powinien wypisywać na ekran tekst zgodny ze schematem: "NazwaPlatformy NazwaObiektu created". Parametry potrzebne do działania danego elementu (parametry slidera: MinValue, MaxValue, IncrementStep, current value jest inicjalizowane wartością MinValue, parametry dialog window: lista sliderów, string z nazwą i lista buttonów, button: string content) powinny zostać przekazane przez konstruktor. DrawContent wszystkich sliderów wypisuje: "PlatformName slider from {MinValue} to {MaxValue} with step: {step}, current: {CurrentValue}".

Platforma nr 1: WP8

- ISlider - ChangeValue dodaje podaną liczbę do CurrentValue przemnożoną przez IncrementStep. Nowa liczba zostaje obcięta do możliwego zakresu (min i max value). Wypisuje nową wartość na ekran.
- IButton - zapamiętuje przekazaną zawartość. DrawContent wypisuje napis na ekran, ButtonPressed wypisuje "WP8 Button pressed, content: "+zawartość napisu.
- IDialogWindow: DrawDialog wypisuje na ekran "Drawing WP8 dialog:" + nazwa okienka, następnie wywołuje DrawContent na wszystkich sliderach i buttonach.

Platforma nr 2: XFCE

- ISlider - ChangeValue dodaje podaną liczbę do CurrentValue przemnożoną przez IncrementStep. Zakres wartości liczby powinien być zapętlony (np.: dla zakresu (0,100), jeśli liczba wyjdzie poza zakres, to powinna pojawić się z drugiej strony: $80+35 = 15$, $25-50=75$ - operacja). Wypisuje nową wartość na ekran.
- IButton - zapamiętuje napis, ale zamieniając wszystkie litery na małe. DrawContent wypisuje napis na ekran, ButtonPressed wypisuje "XFCE button pressed X times", gdzie X to numer kliknięcia danego buttona.
- IDialogWindow: DrawDialog wypisuje na ekran "XFCE Dialog:" + name, następnie wywołuje DrawContent na wszystkich sliderach, oraz informację "To accept press ^O+^X, to cancel press ESC".

Platforma nr 3: Solaris

- ISlider - ChangeValue dodaje podaną liczbę do CurrentValue przemnożoną przez IncrementStep, ale tylko jeśli jest ona dodatnia. Jeśli liczba jest ujemna to $CurrentValue = MinValue$. Wypisuje nową wartość na ekran.
- IButton - zapamiętuje napis, ale zawsze w środek napisu wkleja "Oracle". DrawContent wypisuje napis na ekran, ButtonPressed nic nie robi.
- IDialogWindow: buttony w konstruktorze powinny być ignorowane i zamiast nich powinien zostać stworzony button z napisem "Abandon all settings". DrawDialog wypisuje na ekran "Solaris Dialog" + name, następnie wywołuje DrawContent na wszystkich sliderach, oraz "You can change these settings but Solaris OS does not support applying them". Na końcu należy wywołać DrawContent na stworzonym buttonie.

Implementacja programu użytkownika:

Zaimplementuj funkcję BuildDialogWindow, jednakową dla wszystkich platform, która skonstruuje opisany poniżej interfejs a następnie wywoła metody na jego składowych.

Funkcja ta powinna po kolei:

- Stworzyć 3 slidery z różnymi wartościami min i max.
- Stworzyć 2 buttony: jeden powinien zawierać napis "Cancel", a drugi "Accept"

- Stworzyć dialogWindow z wcześniej utworzonymi elementami, oraz nazwą "UserSettings".
- Na każdym ze sliderów przeprowadzić kilka operacji zmiany wartości tak, aby jego wartość na pewno przekroczyła obie granice (maksymalną, minimalną).
- Wywołać ButtonPressed na każdym z buttonów dwa razy.
- Wywołać DrawDialog.