

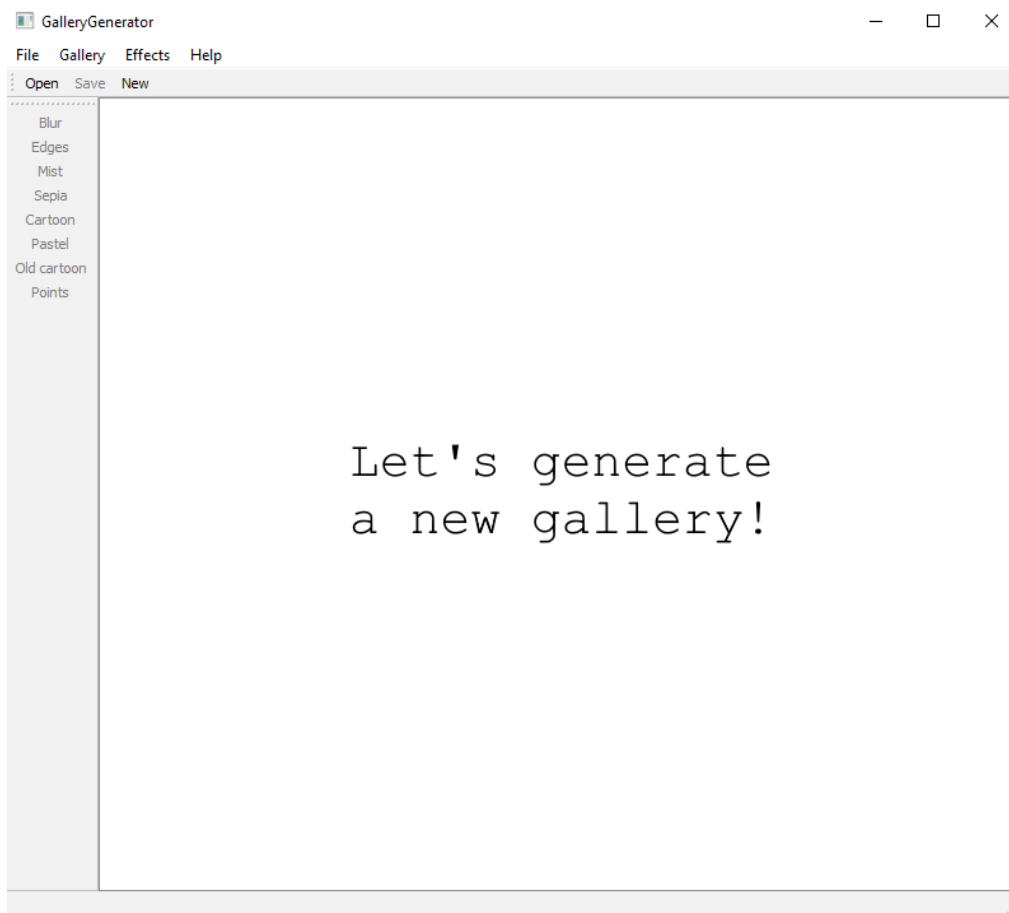
Generator fotogalerii

Łukasz Jaremek
Politechnika Warszawska
Wydział Elektroniki i Technik Informacyjnych
semestr zimowy 2020/2021

Projekt zrealizowany w ramach przedmiotu Podstawy Informatyki i Programowania

Spis treści:

1. Opis projektu i jego rozwiązania
2. Kluczowe klasy i funkcje
3. Instrukcja użytkowania aplikacji
4. Informacje szczegółowe
5. Testy projektu
6. Wnioski i obserwacje



1. Opis projektu i jego rozwiązania

Zadaniem moim było stworzenie aplikacji, która umożliwi wygenerowanie fotogalerii na wybrany przez użytkownika temat. Aplikacja miała zapewnić możliwość stworzenia kolażu, zapisania go do pliku i nałożenia efektów na zdjęcia.

Użyte biblioteki w programie:

- urllib – pobieranie zdjęć i informacji ze strony internetowej
- bs4 – przetwarzanie i oczyszczanie pobranych danych
- cv2 – operacje na zdjęciach (prycinanie, skalowanie, nakładanie efektów)
- numpy – operacje na zdjęciach
- PyQt5 – UI

Do projektu zdecydowałem się użyć strony internetowej unsplash.com, która udostępnia miliony darmowych zdjęć na różnorakie tematy.

Aplikacja składa się z dwóch głównych plików. Są to GalleryGenerator.py oraz UI.py. Pierwszy z nich ma za zadanie wygenerowanie fotogalerii o zadanym temacie, ilości zdjęć oraz żądanych rozmiarach.

Natomiast drugi plik tworzy interfejs dla użytkownika taki, aby korzystanie z aplikacji było łatwe i nie wymagało specjalnych umiejętności.

Dokładny opis klas i funkcji znajduje się w kolejnym rozdziale.

2. Kluczowe klasy i funkcje

klasa **GalleryGenerator**

Klasa zajmuje się generowaniem galerii, obrabianiem jej oraz zapisywaniem w wybranym miejscu. Przy jej inicjacji możemy podać szerokość i wysokość płótna, tj. obszaru, na którym program będzie rozmieszczać zdjęcia, rozmiar obszaru podawany w pikselach. Na płótnie o rozmiarze 1000x1000 zmieści się maksymalnie 9 zdjęć.

Funkcje:

- a) `generate_gallery(topic, number_of_photos, background)`
 - `topic` – temat kolażu
 - `background` – tło, na którym będą znajdować się zdjęcia. Może to być kolor lub przedmiot. Domyślnie ustawiony jest kolor czarny.
 - `number_of_photos` – maksymalna ilość zdjęć, które mają się znaleźć w kolażuFunkcja tworzy płótno na którym umieszcza zdjęcia,
- b) `save_gallery(file_name)`
 - `file_name` – nazwa pliku, pod którą zostanie zapisany kolażFunkcja zapisuje na dysku kolaż pod daną nazwą, domyślnie: `my_gallery.jpg`,
- c) `show_canvas()`
Funkcja wyświetla kolaż,
- d) `cut_canvas()`
Funkcja przycina płótno do minimalnie małego rozmiaru,
- e) `check_topic()`
 - `topic` – temat kolażuFunkcja sprawdza, jak dużo zdjęć o podanym temacie znajduje się na stronie `unsplash.com`. Funkcja zwraca krotkę (`bool`, `int`), która mówi, czy zdjęć jest więcej niż 10 oraz podaje ich przybliżoną ilość,
- f) `canvas()`
Funkcja zwraca kolaż w formacie `ndarray`,
- g) `background()`
Funkcja zwraca tło kolażu jako obiekt `Photo`,
- h) `photos()`
Funkcja zwraca listę zdjęć jako obiekty `Photo`, które znajdują się w kolażu.

klasa **Photo**

Klasa powstała aby usprawnić przekazywanie obrazów między klasami czy funkcjami.

W trakcie inicjacji podajemy:

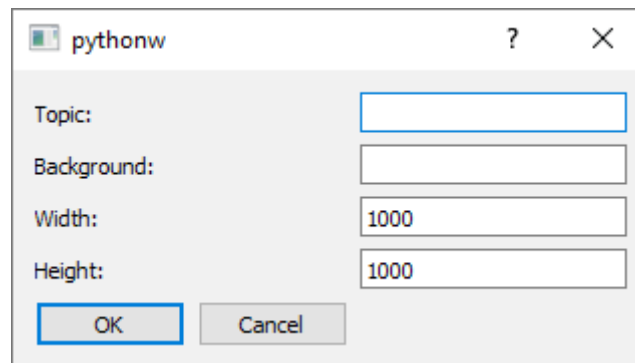
- image – obraz w dowolnym formacie
- x – koordynat x zdjęcia na płótnie
- y – koordynat y zdjęcia na płótnie
- width – szerokość zdjęcia
- height – wysokość zdjęcia
- name – nazwa zdjęcia

Funkcje:

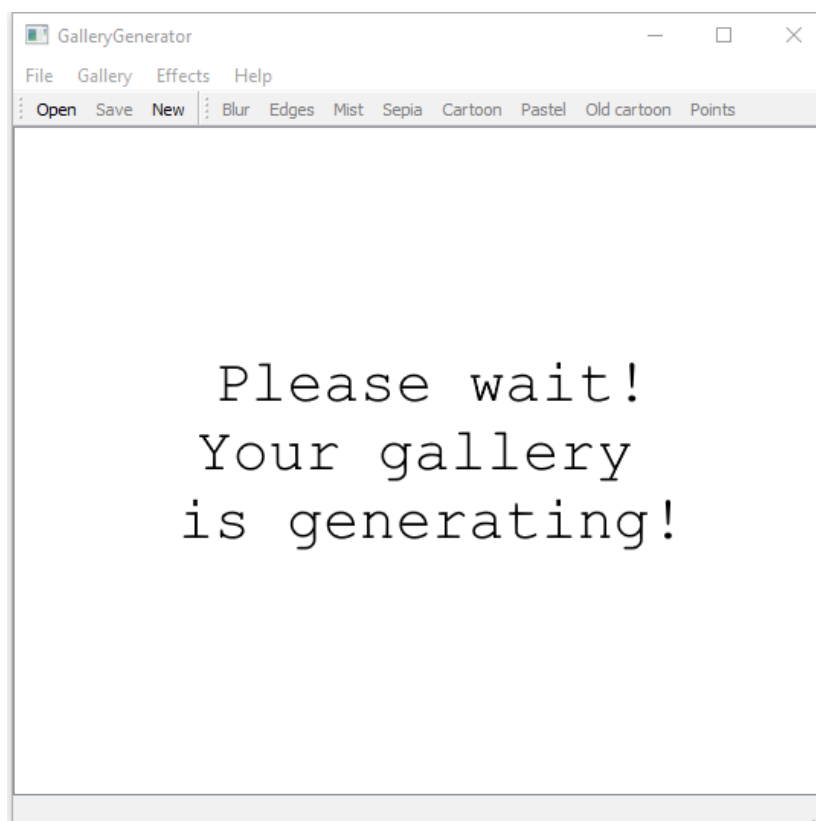
- a) image
Funkcja zwraca obraz,
- b) x
Funkcja zwraca x obrazu,
- c) y
Funkcja zwraca y obrazu,
- d) width
Funkcja zwraca szerokość obrazu,
- e) height
Funkcja zwraca wysokość obrazu,
- f) name
Funkcja zwraca nazwę obrazu,
- g) setx
Funkcja przyjmuje int oraz ustawia go jako x obrazu,
- h) sety
Funkcja przyjmuje int oraz ustawia go jako y obrazu.

3. Instrukcja użytkowania

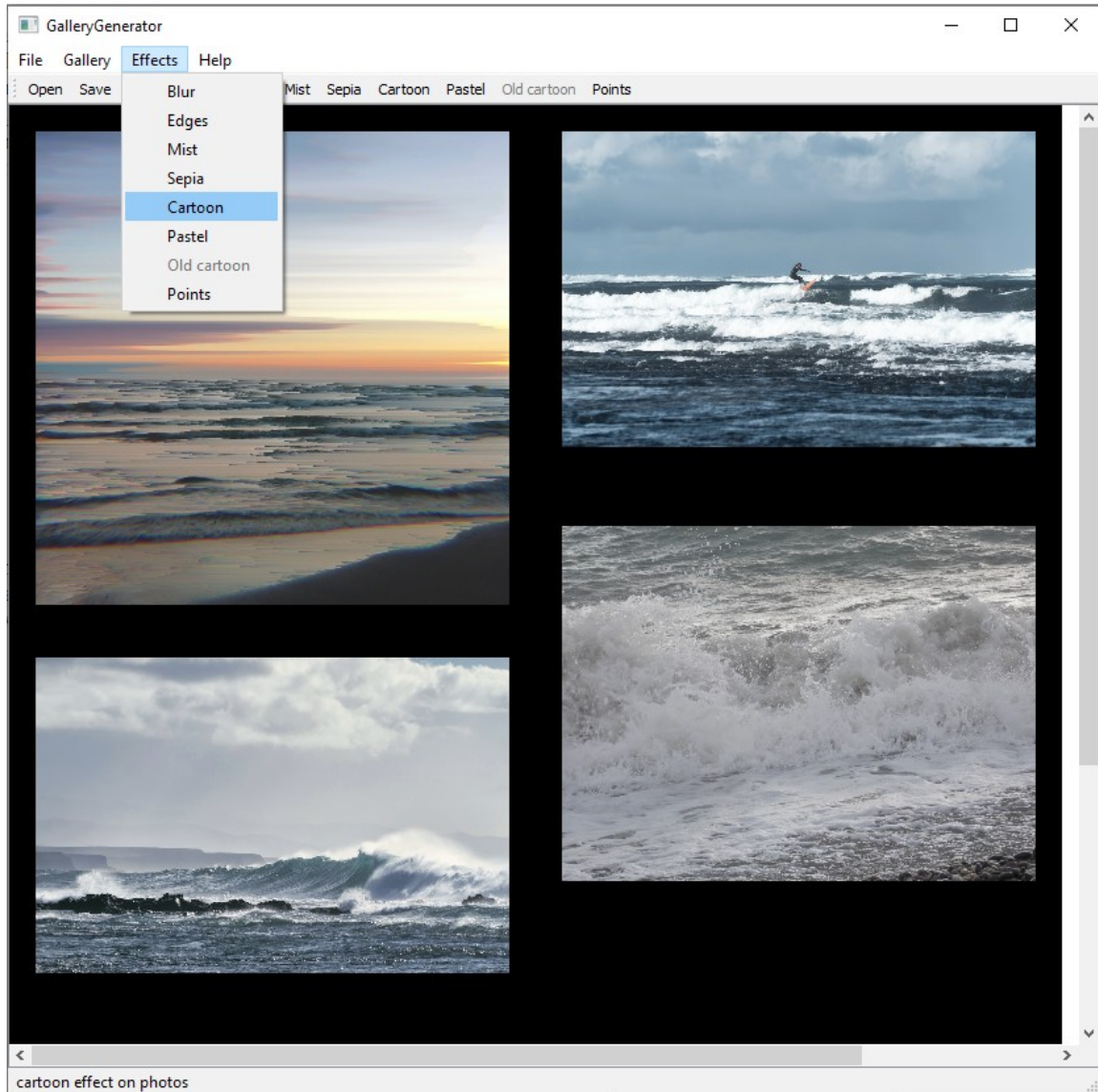
Aby uruchomić aplikację, należy wywołać plik **main.pyw**, po którego uruchomieniu ukaze nam się okno aplikacji. Żeby stworzyć nową galerię, należy wejść w: **Gallery** → **New** (lub nacisnąć **Ctrl+N**). Naszym oczom ukaze się okno dialogowe, gdzie podajemy parametry naszej galerii. Są to:
temat galerii, temat tła galerii (nieobowiązkowe), wysokość i szerokość podane w pixelach. Zalecane wymiary galerii są domyślnie wpisane (1000x1000).



Gdy uzupełnimy już wszystkie pola i naciśniemy Ok, program sprawdzi czy wpisane dane są poprawne a następnie przystąpi do generowania kolażu. Zostaniemy o tym powiadomieni nowym komunikatem.



I już! Po chwili naszym oczom ukaże się gotowa galeria. Jeśli na krawędziach kolażu jest za dużo tła, możemy je przyciąć skrótem **Ctrl+C**. Zapis dokonuje się za pomocą **Ctrl+S** natomiast w zakładce **Effects** możemy znaleźć dostępne efekty. Jeśli zdjęcia nam się nie podobają, możemy wygenerować nowe używając **Ctrl+N**. Program daje nam również możliwość wyczyszczenia tła bądź wczytania zdjęcia z dysku.



4. Informacje szczegółowe

SimpleCanvas.py

Pierwszym problemem, który napotkałem było rozmieszczanie zdjęć na płótnie tak, aby cała wolna przestrzeń została wypełniona po brzegi. Lecz jak to zrobić? Stawianie zdjęcia przy zdjęciu nie byłoby estetyczne, a generowanie losowych przerw również pozostawia wiele do życzenia. Dlatego powstał właśnie ten plik, ze specjalnym systemem przechowywania zdjęć. Plansza, na której rozmieszczamy fotografie jest podzielona na kwadraty o wymiarach 50x50 pixeli. W momencie gdy dodajemy zdjęcie o szerokości 210 pixeli i wysokości 180 pixeli, na płótnie zostaje zarezerwowany obszar 250x200 pixeli, dzięki czemu tworzy się szczelina 40 pixeli dla szerokości i 20 pixeli dla wysokości, której nie zajmie inne zdjęcie a w dodatku powstaje estetyczny efekt wizualny

GalleryGenerator.py

Mamy już sposób na rozmieszczanie zdjęć, teraz skąd je pobrać? Na pomoc przychodzi strona internetowa unsplash.com, które oferuje darmowe zdjęcia na różnorakie tematy, które można legalnie wykorzystać. Wystarczy wpisać hasło a strona pokazuje zdjęcia z nim związane oraz ich ilość. Lecz jak efektywnie pobierać losowe zdjęcia? Na szczęście strona posiada podstronę, która zwraca pojedyncze zdjęcia na zadany temat, dzięki czemu w łatwy i szybki sposób można było ją wykorzystać do projektu. Co prawda, na ilość zdjęć odnoszących się do danego tematu nie ma żadnego wspomagania, lecz oczyszczenie strony w celu pozyskania takiej informacji było bardzo łatwe i sprawne, dzięki czemu przed stworzeniem kolażu można sprawdzić, czy na witrynie istnieje w ogóle wystarczająca ilość zdjęć, aby taki kolaż ułożyć.

Photo.py

Plik zawiera jedną, krótką, lecz bardzo pomocną klasę, która usprawniła działanie programu. Zdjęcia na początku były przesyłane jako lista bitów, co do rozmieszczenia ich na płótnie całkowicie wystarczało, lecz problem pojawił się przy rozwoju programu gdy powstała potrzeba przekazywać zdjęcia wraz z ich koordynatami, rozmiarami czy nazwą, aby móc zwykłe zdjęcie odróżnić od zdjęcia, które stanie się tłem galerii, tak prosty ruch pozwolił na poważny rozwój i krok na przód.

Generators.py

Plik zawiera generator tematów, które mają bardzo dużo zdjęć. Wyszukane i zatwierdzone tematy, są zapisywane do pliku *topics_base.txt*, z którego możemy korzystać podczas użytkowania aplikacji, gdy nie mamy pomysłu jaki utworzyć temat. W zakładce **Gallery → New...** jeśli naciśniemy przycisk *random* znajdujący się przy *topic* algorytm uzupełni pole *topic* losowym tematem z pliku tekstowego. Aby poszerzyć bazę tematów, wystarczy wywołać plik *Generators.py*.

5. Testy

W projekcie wykorzystałem biblioteki i technologie, z którymi nigdy przedtem nie miałem do czynienia, wskutek czego napisanie aplikacji było dla mnie sporym wyzwaniem, lecz i wspaniałą lekcją i doświadczeniem. Aplikację testowałem w dużej części sam, poprzez dziesiątki prób i setki błędów. Do tego kod udostępniałem znajomym oraz pisałem testy jednostkowe, których ze względu na strukturę programu nie ma dużo.

6. Wnioski i obserwacje

Jak już napisałem stronę temu, pisanie aplikacji było dla mnie olbrzymim zastrzykiem doświadczenia i radości. Pomimo wielu problemów i ciężkich chwil, zawsze znajdowałem siłę i chęć aby kontynuować pisanie kodu, co jest dla mnie dużym wyczynem ponieważ do tej pory do samego końca doprowadziłem tylko jeden projekt. Nieopisane wsparcie otrzymywałem również zawsze od prowadzących przedmiot. Niezależnie od pory dnia i nocy mogłem liczyć na ich pomoc. Choć zdarzało się, że na początku nie zgadzałem się i nie rozumiałem ich porad, prędzej czy później wprowadziłem je w życie, ponieważ owe rozwiązania okazywały się najlepsze.

Chciałbym tylko jeszcze napisać, że mam wiele pomysłów jak bym program rozwinął gdyby nie ograniczenie czasowe i zapewne wprowadzę te zmiany w wolnym czasie:

- historia operacji na kolażu – aby móc cofnąć nałożenie efektu, który nam się nie spodobał
- możliwość wygenerowania tylko jednego, nowego zdjęcia, aby nie musieć zmieniać całego kolażu
- możliwość zmiany tła kolażu
- możliwość rysowania na kolażu
- dodanie kolejnych przekształceń na kolażu/konkretnych zdjęciach – np. obracanie