

## Proiect Procesarea de imagini

În acest proiect este vorba despre scanarea codului de identificare ISBN-10 sau ISBN-13 de pe diferite cărți/manuale/publicații, care este sub forma unui cod de bare, apoi se face identificarea informațiilor aferente codului scanat dintr-o bază de date și afișarea acestora utilizatorului. Apoi utilizatorul va selecta mai multe opțiuni privind cartea identificată și opțiunea respectivă va fi memorată în baza de date alături de codul de identificare a cărții.

Un cod ISBN-13 arată sub forma:  
Singura diferență dintre un cod ISBN-10 și ISBN-13 este aceea că ISBN-13 conține mai multe cifre din cauza apariției prea multor cărți și publicații, acestea fiind mult prea multe pentru a putea fi identificate individual printr-un cod unic ISBN-10.

ISBN-13 este o extindere a ISBN-10 cu scopul extinderii posibilităților de identificare a cărților în dimensiunea internațională.

Algoritmul implementat face singur transformarea codului de bare ISBN-10 într-unul ISBN-13 pentru a putea fi identificată cartea în baza de date [Google Books](#).

Baza de date este un fișier numit *db.dat* în care se memorează linii în felul următor:

COD\_BARE OPTIUNE INFORMAȚIE\_ADITIOINALĂ

Primul element: *COD\_BARE* fiind codul de identificare citit de pe carte.

Al doilea element: *OPTIUNE* reprezentând una din cele 3 opțiuni disponibile de marcare a cărții.

Al treilea element: *INFORMAȚIE\_ADITIOINALĂ* fiind informație necesară în cazul în care se selectează opțiunea cu numărul 2.

Opțiunile de marcare a unei cărți sunt următoarele:

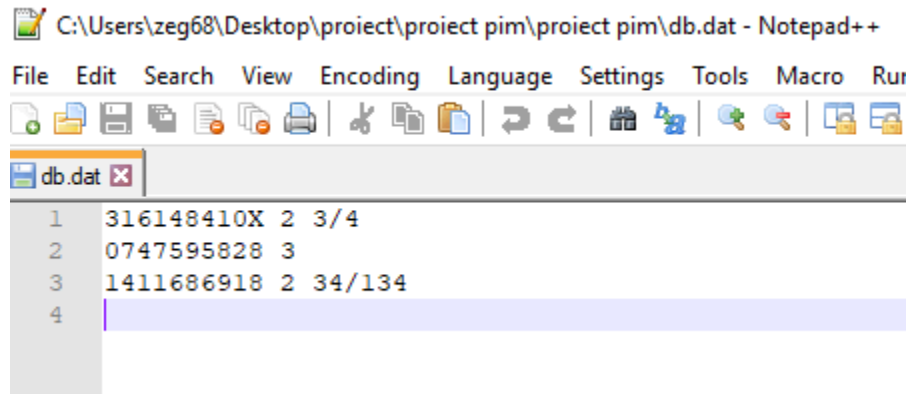
*Opțiunea 1.* Marcare carte a se dori sa se citească.

*Opțiunea 2.* Carte în curs de citire, adăugând numărul ultimei pagini și numărul total de pagini din carte.

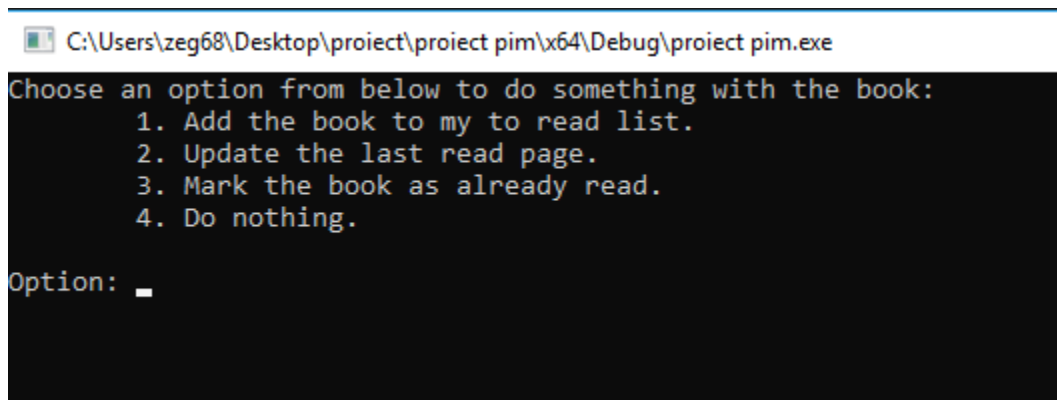
*Opțiunea 3.* Cartea fost terminată deja de citit.



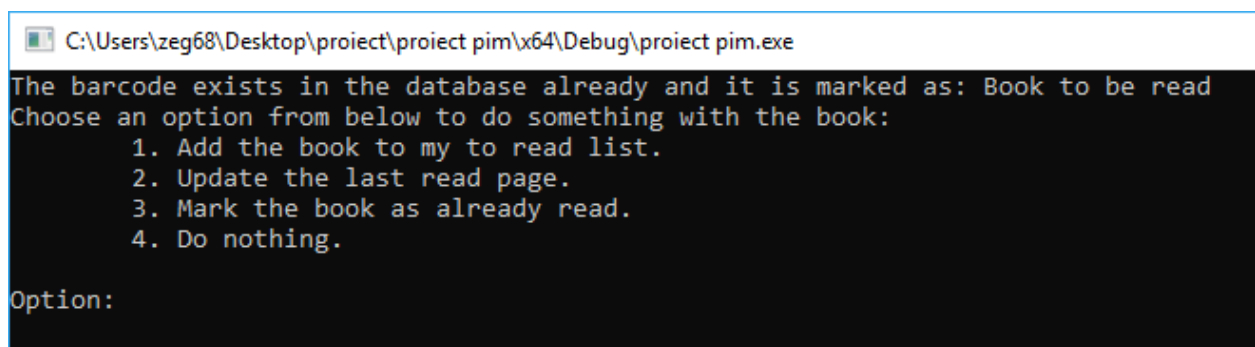
Baza de date arată sub următoarea formă:



Pentru a selecta ce opțiune dorim să alegem, programul va afișa următorul mesaj:



În cazul în care cartea există deja în baza de date, se va afișa următorul mesaj în locul celui de mai sus:

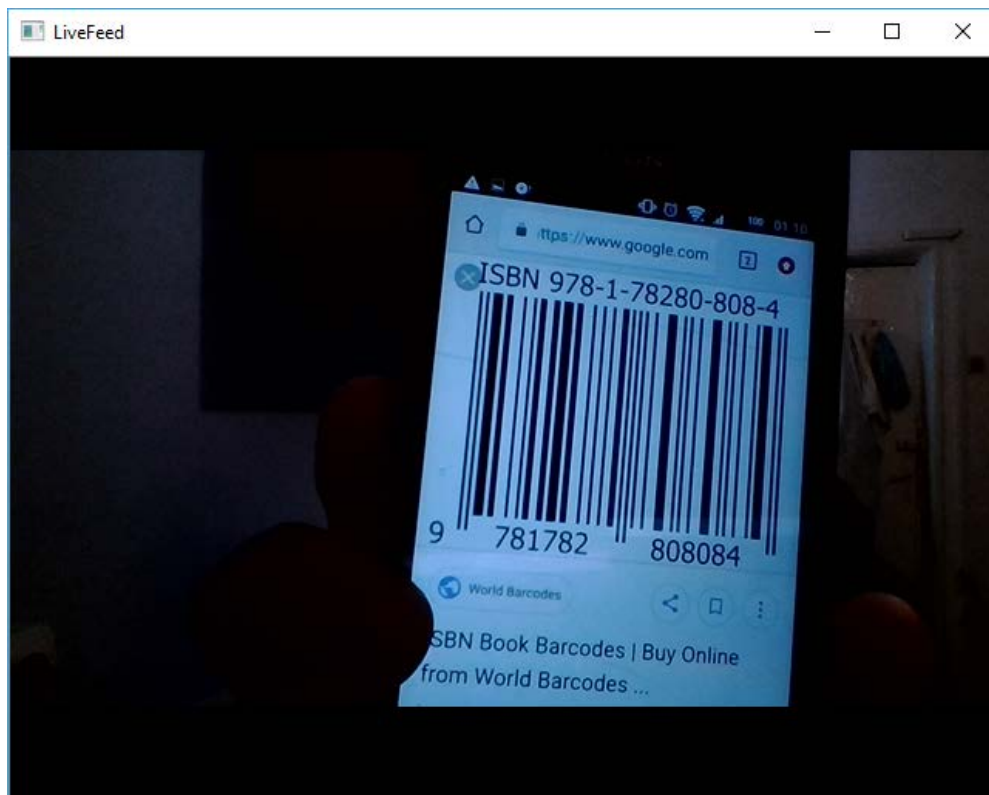


Scanarea se face în maxim 10 secunde de la apăsarea butonului ESC. Dacă nu a fost scanat nimic în acel interval de timp, programul va returna un mesaj de eroare după care se va închide.

În timpul celor 10 secunde, în fereastra LiveFeed este redat în timp real fluxul video înregistrat de pe camera web conectată la dispozitivul ce rulează programul. Programul scanează câte trei frame-uri consecutive, pentru a se focaliza imaginea în cazul unei mișcări bruște a cărții. Dacă nu s-a găsit nimic în cele 3 frame-uri consecutive, programul reia din nou scanarea altor 3 frame-uri după un interval de timp egal cu durata de procesare a restul instrucțiunilor până la revenirea la instrucțiunea de scanare.

Când s-a detectat un cod, frame-ul din LiveFeed devine blocat și aplicația începe să procese informația de pe acesta.

Fereastra arată în felul următor la blocare:



Mai departe se va afișa pe ecran codul de bare identificat alături de ora la care a fost identificat și tipul codului:

```
C:\Users\zeg68\Desktop\proiect\proiect pim\x64\Debug\proiect pim.exe
[1:10:46] 0 decoded ISBN-10 symbol "1782808086"
Scanned barcode: 1782808086
[ WARN:0] terminating async callback
Success!
```

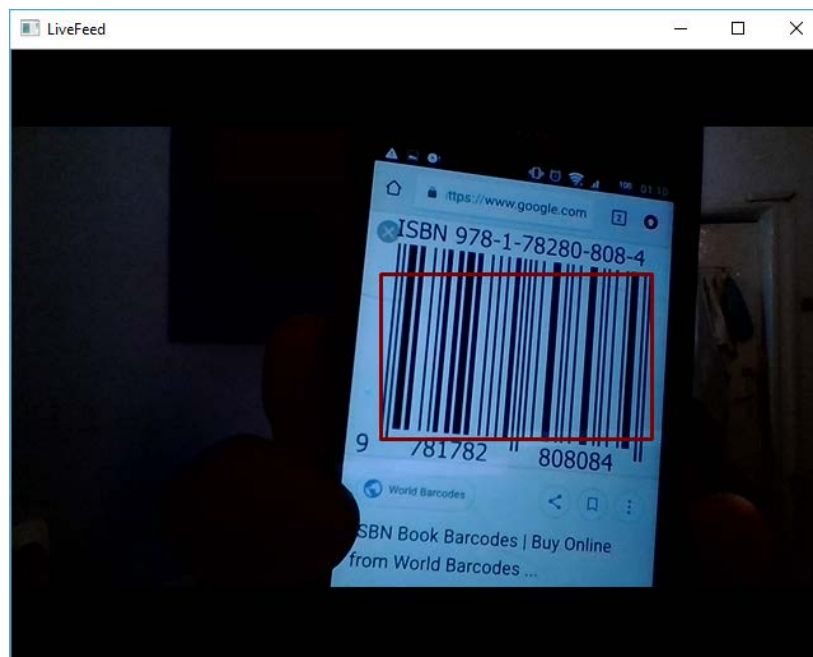
Informațiile identificare de pe Google Books vor fi afișate în continuare sub următorul format:

```
C:\Users\zeg68\Desktop\proiect\proiect pim\x64\Debug\proiect pim
ISBN-13: 9781782808084,
Title: From Adam To Christ,
Authors: [Fino Z. Dlamini],
Publisher: ,
Year: 2017,
Language: en
```

Aici se poate observa că s-a făcut conversia automat de la ISBN-10 la ISBN-13.

După ce se introduce una din opțiunile afișate mai sus, fereastra LiveFeed va fi actualizată cu poziția codului de bare scanat, acesta fiind înconjurat de un chenar roșu.


Chenarul nu este afișat pe tot codul disproporțional, ci este un dreptunghi creat de colțurile din stânga jos și dreapta sus.



Inițial, pentru a scana codul de bare, este necesară o linie de grosime de minim câțiva pixeli, oriunde în dimensiunea codului de bare. Este posibilă prezentarea a unei părți foarte mici a codului de bare în fața camerei web pentru a fi scanat, atât timp cât se prezintă o linie întreagă.

Proiectul folosește 2 biblioteci importante și un API pentru a face posibilă funcționarea acestuia.

1. OpenCV 4.0.0  <https://opencv.org/opencv-4-0-0.html>

2. Zbar  <http://zbar.sourceforge.net/>  
<https://github.com/dani4/ZBarWin64>

3. isbnlib  
Python embedded into C++  
Google Books API <https://pypi.org/project/isbnlib/>  
<https://docs.python.org/2/extending/embedding.html>  
<https://developers.google.com/books/>

În final, programul are următoarele funcții implementate:

```
void CaptureWEB(Mat& img1, Mat &img2, Mat &img3)
```

Această funcție citește de la fluxul camerei web 3 frame-uri și le afișează unul după altul în fereastra LiveFeed într-o buclă infinită până la apăsarea butonului ESC, după care vor fi transmise variabilelor img1, img2 și img3. În această funcție s-au implementat apeluri ale unor funcții din biblioteca OpenCV.

```
int getBarcode(Mat img)
```

Funcția getBarcode va primi pe rând imagini, începând cu cele returnate de funcția CaptureWEB pentru a identifica codul de bare din acea imagine. Dacă nu se identifică un cod de bare în imaginea img atunci funcția returnează valoarea 0 care înseamnă trecerea la scanarea următoarei imagini, verificându-se dacă nu s-a depășit limita de timp impusă pentru scanare.

Această funcție încarcă în variabila globală *scanned\_barcode* și identifică apoi afișează dreptunghiul delimitator al codului de bare în imaginea blocată în fereastra LiveFeed. Acest

lucru însă se întâmplă cu întârziere, efectuându-se de abia după selectarea unei opțiuni din meniul bazei de date.

La succesul acestei funcții se va returna valoarea 1 și algoritmul va merge mai departe la partea de identificare a codului de bare în baza de date Google Books și afișarea informațiilor aferente acestuia, folosind funcția

```
int process_barcode(char* barcode)
```

Această funcție implementează în C++ operații din modulul isbnlib făcut pentru Python.

Rolul acestei funcții este de a căuta informații despre codul de bare scanat în baza de date Google Books. În cazul existenței codului în baza de date respectivă, se va apela funcția meta() a modulului isbnlib care returnează informații utile despre ce se află în spatele codului de bare scanat. Aceste informații vor fi procesate cu funcția void process\_string(char\* str) pentru a le afișa într-un format mai elegant.

Dacă codul de bare nu este de tip ISBN sau nu s-a găsit în Google Books, atunci funcția va returna valoarea -1 ceea ce va duce la terminarea imediată a programului.

Mai departe vom trece la prelucrarea datelor pentru baza de date.

```
void search_barcode_in_file_new(char* barcode)
```

După terminarea funcției precedente, este rândul de a selecta o opțiune din baza de date și de a actualiza baza de date în cazul că dorește utilizatorul aplicației acest lucru.

Această funcție va căuta mai întâi *barcode* în baza de date. În cazul în care există deja în fișierul db.dat, atunci se va apela funcția string edit\_existing\_line(string line) care va prelucra linia din baza de date și va primi înapoi noua linie cu noua opțiune a utilizatorului de introdus în baza de date.

Dacă nu există codul de bare în baza de date, atunci se va cere o opțiune apelând funcția int menu() care va ruga utilizatorul să selecteze ce dorește să facă cu cartea identificată, așa cum este arătat în primele pagini ale acestui document.

În final după ce s-au inserat toate liniile în documentul temp.dat, fișierul db.dat va fi șters și temp.dat va fi redenumit în db.dat. Acest lucru a fost efectuat pentru a putea actualiza baza de date într-un mod cât mai eficient. Cum dimensiunea fișierului db.dat este mica, citirea fiecărei linii în parte și apoi scrierea ei în alt fișier nu va ține un timp influențabil în performanța algoritmului.