

Nr. înreg./.....

**MINISTERUL EDUCAȚIEI NAȚIONALE
COLEGIUL NAȚIONAL „MIHAI EMINESCU”**

Clasa a XII-a B

Specializarea *Matematică-informatică, intensiv informatică*

**LUCRARE PRACTICĂ DE ATESTARE
A COMPETENȚELOR PROFESIONALE ÎN SPECIALITATEA INFORMATICĂ**

Titlul lucrării

Capsule

**Coordonator,
Prof. Șandor Nicoleta**

**Candidat,
Elev Mureșan Paul**

Satu Mare

2016

TEMA LUCRĂRII

Program interactiv de luat notițe

CUPRINS

MEDIUL DE PREZENTARE	2
2.1 VISUAL STUDIO 2015	2
2.2 ADOBE PHOTOSHOP CC	2
PREZENTAREA LUCRĂRII.....	3
3.1 STRUCTURA APLICATIEI.....	3
3.2 DESIGN.....	6
3.3 MANIPULAREA FIȘIERELOR	9
3.4 FUNCTIA SEARCH SI LISTA CU NOTITE.....	10
3.5 TEXT-TO-SPEECH.....	10
3.6 API-UL PENTRU VREME	11
CONCLUZII	12
SURSE WEBGRAFICE SELECTIVE	13

ARGUMENT

“Nimic nu este prea dificil dacă împarți în pași mici ceea ce ai de făcut.” Henry Ford

Trăim într-o perioadă dificilă a omenirii. Tehnologia devine din ce în ce mai avansată, accesul la informație este din ce în ce mai mare, iar mulți dintre noi suntem suprasolicitați.

Acum un secol, în absența tehnologiei, ritmul de viață era mult mai lent. Consumerismul și noul secol ne-a schimbat ritmul de viață. Trăim mai alert, facem mai multe lucruri zilnic și ne este prezentată mult mai multă informație.

Zi de zi, ne lovim cu aceste informații la lucru, la școală, acasă sau pe internet, și din această cauză, mulți dintre noi uităm de lucrurile importante pe care le avem de făcut. De multe ori dentistul nostru are cabinetul gol, cutia de lapte nu este de găsit sau tema rămâne acasă pe masă.

Astfel, mi-am propus să creez un program care să rezolve aceste probleme, să ne scutească de suprasolicitrarea memoriei, care deja este încărcată oricum cu diferitele informații pe care le folosim zilnic.

Am realizat un program interactiv de luat notițe. Acestea pot fi liste de cumpărături, memento-uri, diferite lecții de la școală sau orice crezi că este important pentru tine.

Aplicații/resurse folosite:

1. Visual Studio 2015
Librării: *MahApps.Metro* și *MaterialDesignInXamlToolkit*
2. Adobe Photoshop CC

Configurație optimă:

CPU: Intel Core i5 5200U, 2.2 GHz Dual core

GPU: ATi Radeon HD 4850; 625 MHz, 512 MB GDDR3

RAM: 1GB DDR3

OS: Windows 8/10

Mediul de prezentare

2.1 Visual Studio 2015

Visual Studio include un set complet de instrumente de dezvoltare pentru generarea de aplicații ASP.NET, Servicii Web XML, aplicații desktop și aplicații mobile. Visual Basic, Visual C++, Visual C# și Visual J# toate folosesc același mediu de dezvoltare integrat (IDE) care le permite partajarea instrumentelor și facilitează crearea de soluții folosind mai multe limbaje de programare.



Pentru realizarea aplicației, am folosit **Windows Presentation Foundation** (pe scurt **WPF**), care este un subsistem grafic folosit pentru realizarea interfețelor pentru versiuni ale sistemului de operare Windows. WPF, numit anterior “Avalon”, a fost lansat inițial ca o parte a .Net Framework 3.0, și în loc să se bazeze pe subsistemul GDI, WPF folosește DirectX. WPF își propune să fie un model consistent de programare pentru realizarea interfețelor și aplicațiilor pentru business.

WPF folosește XAML, un limbaj de programare bazat pe XML, pentru partea de definiție și legătură a elementelor din interfață. Aplicațiile WPF pot fi exportate ca aplicații pentru desktop și, de asemenea, obiecte în site-uri web. WPF dorește să unească un număr mare de elemente de interfață uzuale, cum ar fi randări 2D, documente fixe și dinamice, tipografie, grafică, animații și media pre-randată. Aceste elemente pot fi legate și manipulate în diferite evenimente și interacțiuni cu utilizatorul.

2.2 Adobe Photoshop CC

Adobe Photoshop este un software folosit pentru editarea imaginilor digitale pe calculator, program produs și distribuit de compania americană Adobe Systems și care se adresează în special profesioniștilor din domeniu.



Adobe Photoshop, așa cum este cunoscut astăzi, este vârful de lance al gamei de produse software pentru editare de imagini digitale, fotografii, grafică pentru tipar, video și Web de pe piață. Photoshop este un program cu o interfață intuitivă care permite o multitudine extraordinară de modificări: editări de luminozitate și contrast, culoare, focalizare, aplicare de efecte pe imagine sau pe zone (selecții) etc

Prezentarea lucrării



Fig. 1

Pentru lucrarea de atestat am decis să realizez o aplicație interactivă de luat notițe. Numele aplicației, “Capsule”(Capsulă) (Fig. 1), sugerează faptul că aceasta are rolul de a stoca, pe termen nelimitat, anumite informații sau idei ale utilizatorului pe care acesta s-ar putea să le uite sau pe care nu dorește să le păstreze încărcate în memorie.

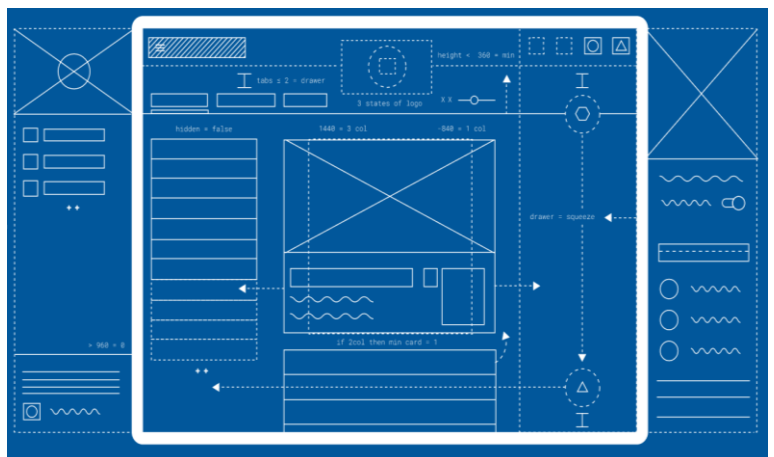
Aceste informații se pot stoca sub forma unor notițe, liste sau, pentru idei mai ample, sub forma unor eseuri, lecții etc. Programul permite, de asemenea, adăugarea de poze din fișierele locale ale calculatorului.

Funcțiile de bază prezente în orice editor de text, cum ar fi aldin(bold), subliniere(underline) sau cursiv(italic) sunt prezente și în acest program, însă, fiind gândit pentru notițe scurte și, de asemenea, pentru simplitate, am decis să nu adaug opțiuni pentru tipul fontului, mărimea sau culoarea acestuia. Opțiunile bold, underscore și italic se pot folosi cu prescurtările: ctrl+b, ctrl+u, ctrl+i.

3.1 Structura aplicației

Fiind proiectată într-un stil “material”, aplicația respectă regulile de structură impuse de Google pentru aplicațiile lor.

Principala resursă folosită pentru acest design este “material”-ul, fiind asemănător cu hârtia. Se pune mare accent pe realism, “material”-ul având unele principii preluate din viața reală dar și anumite principii care sfidează legile fizicii, folosite pentru îmbunătățirea experienței utilizatorului.



Ce este “material”-ul?

“Material”-ul este un obiect solid, cu 3 mărimi: lungime, lățime și grosime. Acesta are umbră, fiind mereu situat la o anumită înălțime față de baza aplicației. Din această cauză, două coli de “material” nu pot ocupa același loc, iar acesta nu se poate îndoi. “Material”-ul poate apărea și se poate distruge spontan, se poate mișca pe ambele axe. Orice conținut aflat în “material” se poate deplasa oriunde în interiorul lui, dar nu poate ieși înafara lui. De asemenea, el poate să se “rupă” în mai multe bucăți egale, să se “vindece” și să devină iar o singură bucată și se poate și mări sau micșora.

Structura generală

Structura aplicației este și ea conform structurii “material” pentru desktop. Sus se afla o bară permanentă care conține butonul pentru meniul lateral și cel de setări. În centru se află pagina cu titlul și conținutul notiței. De asemenea, un element specific al design-ului material este “Float Button”-ul. Acesta este butonul care conține principalele acțiuni ale aplicației: creare notiță și adăugare imagine. Pentru a atrage atenția mai mult decât restul butoanelor, acesta este rotund. Pe lângă acest buton, aplicația mai conține butoane cu diferite opțiuni pentru notițe (ștergere, salvare și opțiunea Text-to-Speech) și butoanele pentru navigare între notițe (Fig. 2).

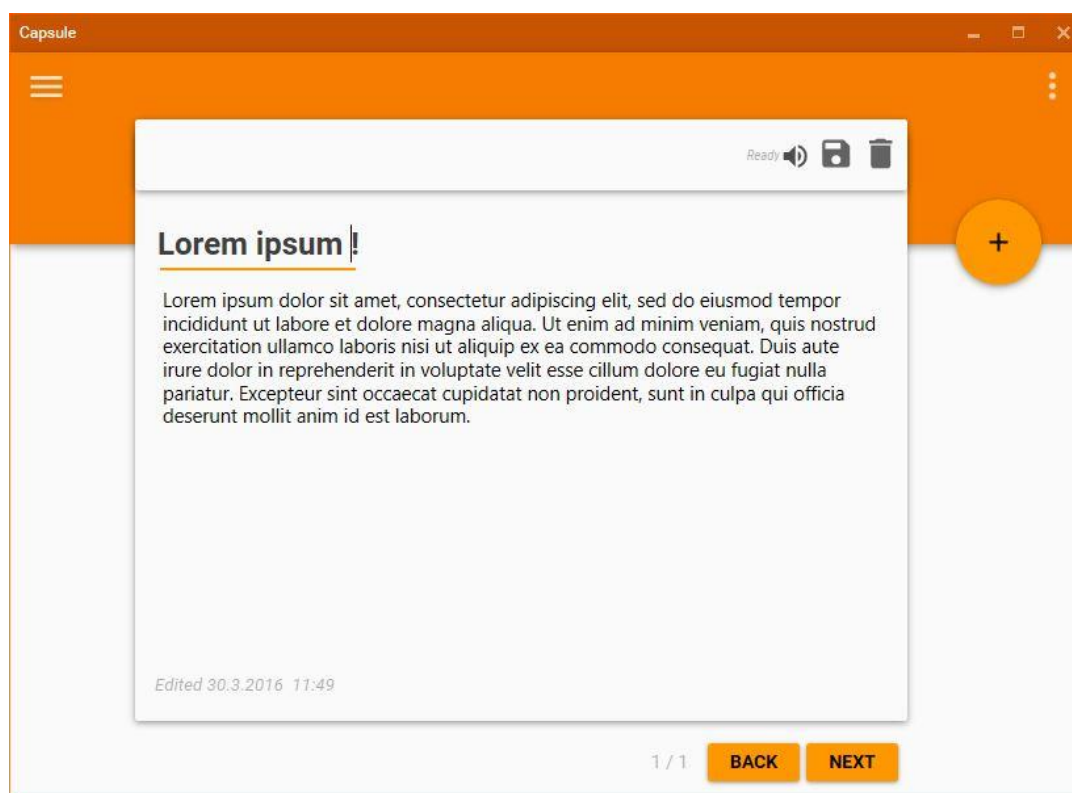


Fig. 2

În meniu, care glisează din partea stângă, se află o listă cu toate notițele (pentru o navigare mai rapidă când numărul acestora crește), un câmp pentru căutare, și diferite informații meteo locale (Fig. 3).

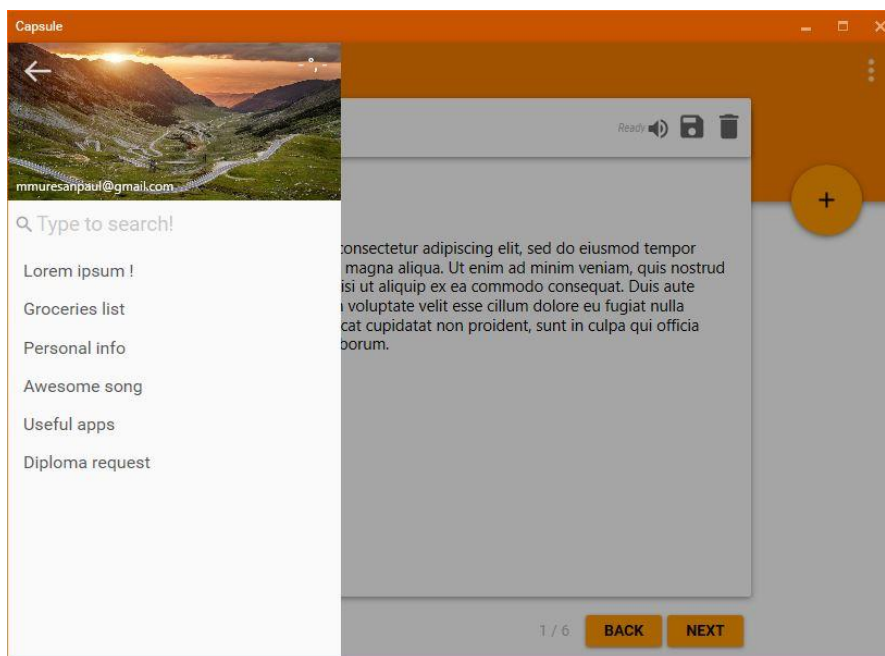


Fig. 3

În setări se află diferitele opțiuni pentru personalizare(culoarea principală a aplicației și modul zi/noapte) și opțiunile pentru Text-to-Speech (viteza voci și tipul de voce) (Fig.4).

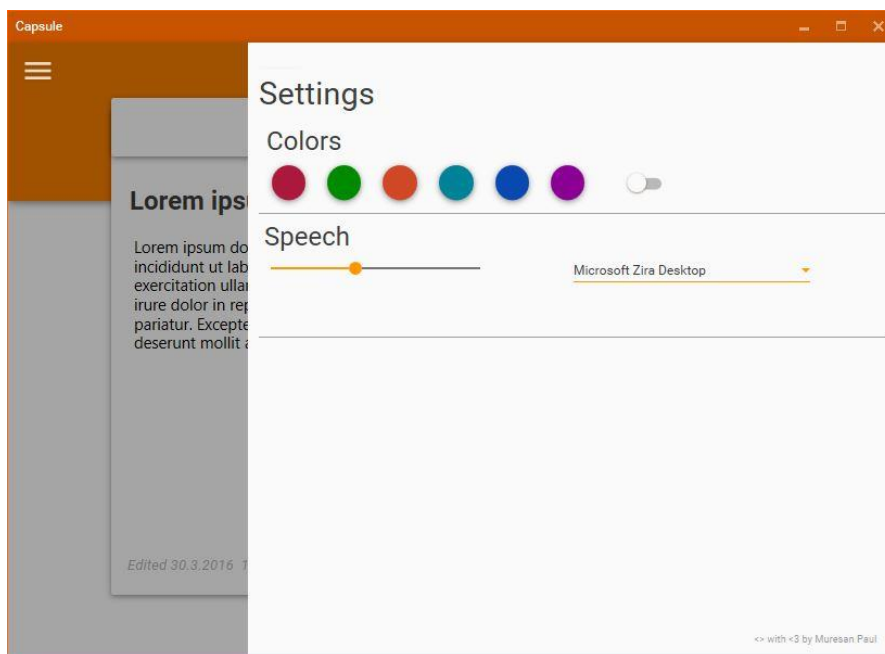


Fig. 4

3.2 Design

Culori

Culorile se bazează pe o paletă de culori (Fig. 7) care este alcătuită din 3 categorii: culoarea principală, culoarea secundară și culoarea de accent. Culoarea principală este cea folosită cel mai mult, fiind prezentă peste tot în aplicație, conferindu-i tema.

Culoarea secundară se folosește pentru a indica acțiuni sau informații, aceasta fiind o nuanță mai deschisă a culorii principale.



Culoarea de accent este folosită pentru “Action Buttons” și elemente interactive, cum ar fi butoanele, controale, slidere etc.

De asemenea, este acceptabil să se folosească aceeași culoare ca și culoare secundară și culoare de accent. Eu am ales această opțiune pentru ca mi s-a părut prea încărcat programul având culoarea de accent diferită de cea secundară.

Pentru a conferi un nivel mai ridicat de personalizare aplicației, utilizatorul poate alege o temă de culori dintr-o listă de teme valabile: roșu, portocaliu, indigo, albastru deschis, mov și verde. Programul conține și un mod zi/noapte, care schimbă culorile aplicației pentru a reduce tensiunea cauzată asupra ochiilor utilizatorului pe timp de noapte (Fig. 5 și Fig. 6).

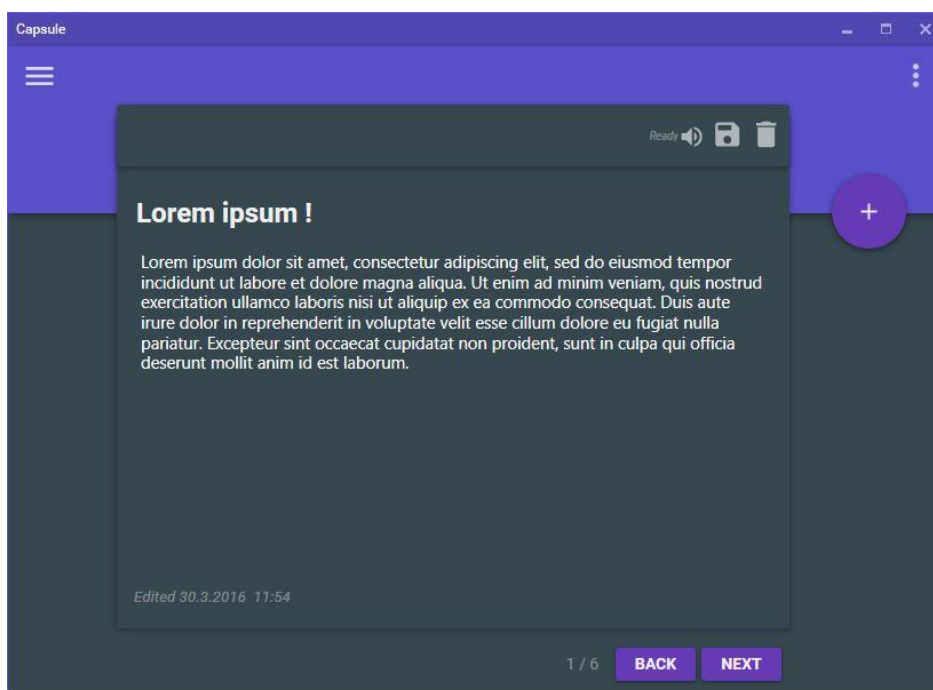


Fig. 5

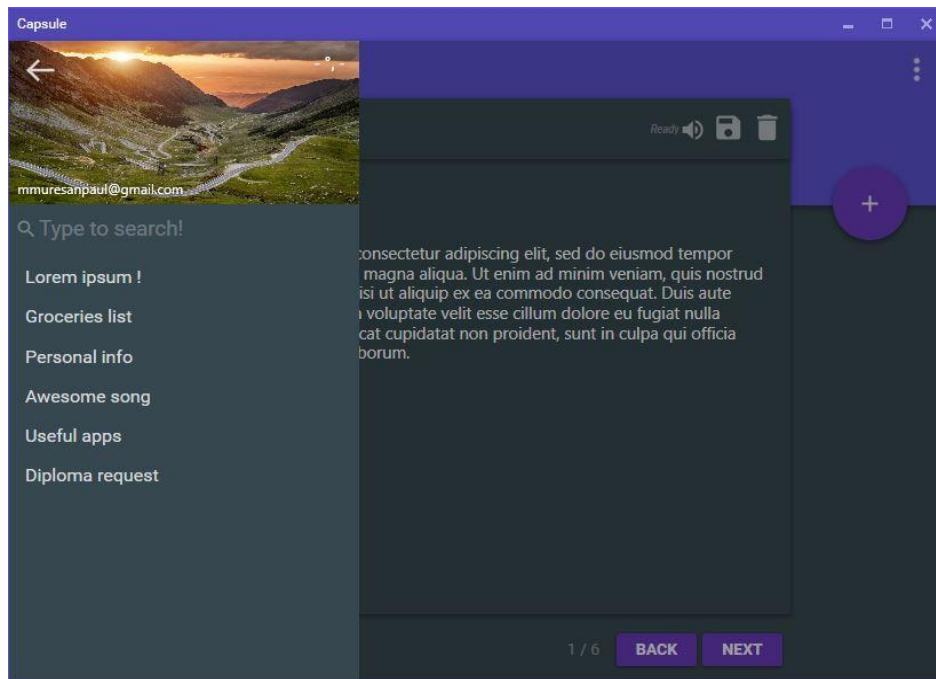


Fig. 6

Primary — Purple	
500	#9B26AF
700	#7A1EA1
800	#691A99
Accent — Green	
A200	#68EFAD

Fig. 7

Tipografie

Pentru text, aplicația folosește de asemenea fontul preferat de Google, *Roboto*.

Acesta are 6 variante: *Thin*, *Light*, *Regular*, *Medium*, *Bold* și *Black*, care, împreună cu diferite mărimi și culori, formează mai multe nivele de atracție.

De exemplu, pentru butoane se folosește Roboto Mediu cu mărimea 14, scris cu litere mari.

Display 4

Display 3

Display 2

Display 1

Headline

Title

Subheading

Body 2

Body 1

Caption

Button

Light 112sp

Regular 56sp

Regular 45sp

Regular 34sp

Regular 24sp

Medium 20sp

Regular 16sp (Device), Regular 15sp (Desktop)

Medium 14sp (Device), Medium 13sp (Desktop)

Regular 14sp (Device), Regular 13sp (Desktop)

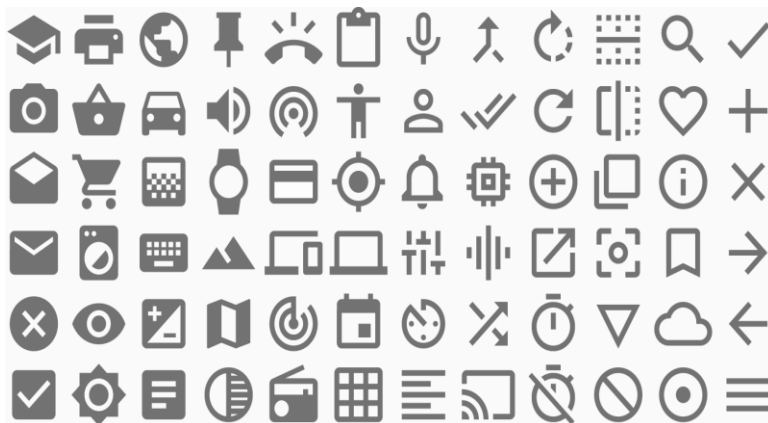
Regular 12sp

MEDIUM (ALL CAPS) 14sp

De asemenea, pentru lizibilitate, line height-ul se determină în funcție de mărimea fontului și tipul de font. Acesta se folosește doar pentru mărimile de font mai mici, sau unde sunt mai multe rânduri de text. În rest se recomandă a nu se depăși un rând de text.

Imagini de sistem

Imaginile de sistem, sau UI icons, simbolizează o comandă, un fișier, un dispozitiv sau un director. Ele sunt folosite de asemenea pentru acțiuni comune precum: ștergere, salvare, printare etc. Design-ul imaginilor de sistem este simplist, modern și prietenos. Fiecare imagine este redusă la forma ei minimală, cuprinzând, în esență, ideea ei.



Toate acestea asigură lizibilitate și claritate, chiar și la mărimi mai mici. Formele lor sunt îndrăznețe și geometrice, simetria și consistența lor oferindu-le o calitate unică, menținându-le simple.

3.3 Manipularea fișierelor

Fiecare notiță este memorată în folderul “My documents” în folderul creat prin cod, numit “My Notes”. Într-un vector de tip struct s-au memorat informațiile cuprinse în fiecare notiță, precum titlu, conținut, data ultimei modificări.

```
class Note
{
    public struct date
    {
        public Int32 Day;
        public Int32 Month;
        public Int32 Year;
        public Int32 Hour;
        public Int32 Minute;
    }
    public date[] Date = new date[100];
    public String[] Title = new String[100];
    public String[] Content = new String[100];
}
```

Conținutul se salvează într-un fișier de tip “.xml”, pentru ca să se păstreze formatul conținutului, cum ar fi mărimea fontului, tipul fontului, culoarea și, eventual, imaginile inserate în conținut.

Titlul notiței și data ultimei modificări se salvează separat într-un fișier de tip “.txt”.

Pentru a ușura procesul de ștergere, numele fișierelor sunt, defapt, indicii de poziție a notiței.

De exemplu: a 5-a notiță se va numi 5.txt respectiv 5.xml.

Procesul de manipulare a fișierelor

În momentul în care programul se deschide, acesta verifică dacă în folderul “My documents” există folderul “My Notes”. Dacă acesta nu există, îl va crea și de asemenea va adăuga și o primă notiță. Dacă deja există notițe create, acesta le va citi în ordine și le va introduce în program prin vectorul de tip struct, în care va avea loc orice modificare. La ieșirea din program, se vor actualiza notițele, modificate anterior în vector, în folderul în care sunt salvate.

Pentru ștergerea unei notițe, se folosește un algoritm asemănător cu cel de ștergere a unui element dintr-un vector.

Pentru salvare sau modificare, programul salvează în vector datele introduse în câmpurile de titlu, conținut și, printr-o funcție care returnează data și ora curentă, schimbă ultima dată a modificării.

3.4 Functia search si lista cu notițe

Când programul se deschide sau are loc o modificare, lista cu notițe se actualizează, afișând orice modificare.

În momentul în care se scrie ceva în căsuța de Search, conținutul listei se golește și se adaugă ulterior itemi care satisfac condițiile funcției de căutare.

Funcția search se bazează pe algoritmul folosit pentru căutarea binară unui element într-un vector. Acesta convertește conținutul introdus în căsuța de search, schimbând toate literele, ca acestea să fie mici și face aceleași lucru cu datele din vector, pentru că diferența dintre o literă mare și una mică să nu influențeze căutarea. După, compară conținutul introdus cu fiecare titlu din vector, căutând o apariție a conținutului introdus în titlu. Dacă conținutul introdus se regăsește în unul dintre titluri, acesta este introdus în listă.

3.5 Text-to-Speech

Pentru funcția Text-to-Speech am folosit o librărie .NET 4.0, **System.Speech** introdusă și folosită de Windows în versiunile mai noi (Windows 7/8/10) pentru aplicația “Narrator” care ajută oamenii cu deficiențe de vedere. **Speech** produce o voce umană artificială care vorbește cu utilizatorii. Engine-ul din spatele librăriei convertește textul dintr-o anumită limbă în vorbire directă. Aplicația are acces la mai multe pachete de voci preinstalate în sistemul de operare și folosește și un pachet aditional de voci pentru limba română. De asemenea, librăria permite și modificarea ritmului vorbirii.

În aplicația mea, această opțiune se poate activa cu ajutorul butonului de Text-to-Speech care funcționează astfel:

- Click pentru pornirea Text-to-Speech, naratorul începe să vorbească notița curentă
- Click încă o dată pentru pauză, naratorul se oprește din vorbit iar următorul click îl va determina să continue de unde s-a oprit anterior
- Long Click pentru oprire, următorul click va porni funcția Text-to-Speech de la începutul notiței curente

3.6 API-ul pentru vreme

Ce este un API?

Application Programming Interface (API) reprezintă o interfață pentru programarea de aplicații. De obicei, este vorba despre interfața dintre programele de aplicație și sistemul de operare, care stabilește în amănunt modul în care programele de aplicație pot accesa (apela) serviciile sistemului de operare sub care rulează.

Pentru aplicația **Capsule**, am folosit un API prevăzut de openweathermap.org. Conferind-ui anumiți parametrii, cum ar fi locația în coordonate sau numele locației, acesta returnează într-un fișier XML anumite date despre vreme în locul specificat. Folosind aceste date, aplicația oferă utilizatorului informații despre vremea actuală în meniul cu notițe.

Concluzii

Folosind librăriile/funcțiile menționate mai sus, librăriile Mah.Apps și MaterialDesignInXamlToolkit și respectând ghidajele impuse de Google referitoare la design și user experience, am creat o aplicație interactivă, cu personalitate, fiind foarte intuitivă și simplă, dar în același timp practică și de încredere.

În concluzie, consider că realizarea acestui proiect m-a ajutat să îmi îmbogățesc cunoștințele de programare în C# WPF, software design și user experience și mi-a stârnit și mai tare dorința de a avea o carieră în domeniul informaticii.

Pentru realizarea unui astfel de proiect este necesară multă planificare și de asemenea, multe încercări, învățare din greșeli, mai ales din mențiunile colegilor mei care au încercat programul și m-au ajutat la categoria “user-friendly” . Până la sfârșitul proiectului, programul a trecut prin 20 de versiuni, fiecare adăugând o îmbunătățire în design și funcționalitate.

Pe viitor, doresc să dezvolt aplicația, adăugând versiuni pentru telefon, un website personal și funcții noi, cum ar fi conturi multiple și online syncing.

Surse webgrafice selective

[https://msdn.microsoft.com/en-us/library/gg145021\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/gg145021(v=vs.110).aspx)

<http://www.c-sharpcorner.com/uploadfile/nipuntomar/text-to-speech-in-wpf/>

(Accesat in data de 10.12.2015)

<http://mahapps.com/>

(Accesat in data de 23.01.2016)

<https://github.com/ButchersBoy/MaterialDesignInXamlToolkit>

(Accesat in data de 12.02.2016)

<https://design.google.com/>

(Accesat in data de 04.10.2015)

<http://static.thousandwonders.net/Transfagarasan.original.15709.jpg>

(Accesat in data de 10.03.2016)

Precizare

Descrierile bibliografice ale surselor de documentare din webgrafia prezentă respectă standardele normelor internaționale **ISO 690 / 1996** (*Documentare - Referințe bibliografice - conținut, forma și structură*) și **ISO 690-2 / 1997** (*Informatică și documentare - Referințe bibliografice - Partea 2: Documente electronice sau părți ale acestora - privind în special documentele electronice*).