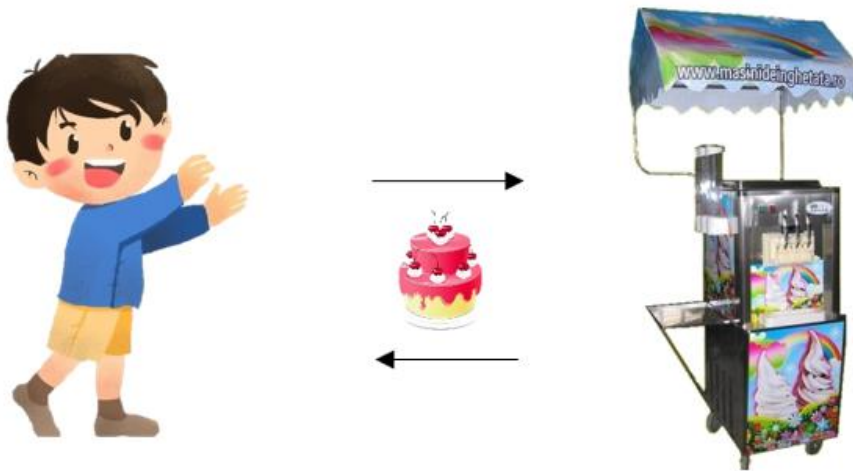


## PROIECT 1 – Cake Maker

**Cake Maker** este un proiect care necesită dezvoltarea unei **aplicații C++ tip consolă** care **simulează** un automat de făcut prăjituri.

Rolul lui principal este să servească clientul cu prăjitura dorită. În altă ordine de idei automatul va dispune de un meniu principal care îi va oferi clientului anumite posibilități de selecție.

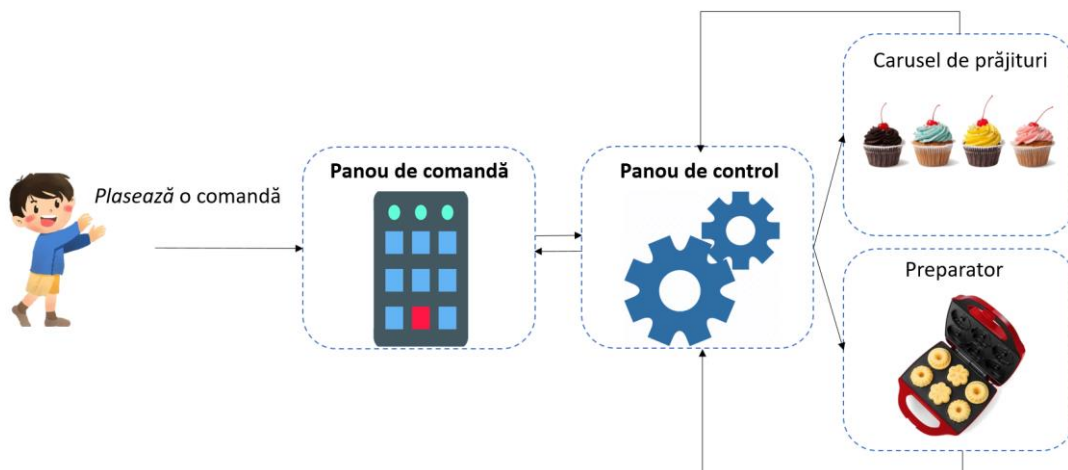
- Afișarea meniului de prăjituri
- Afișarea prăjiturilor existente
- Comandarea unui produs



### 1. Componentele aplicației:

Componentele aplicației sunt prezentate în figura 1 și acestea sunt:

- a) Panoul de comandă
- b) Panoul de control
- c) Depozit de prăjituri
- d) Preparatorul de prăjituri



## 2. Descrierea aplicației:

Pentru simularea acestui automat, programul C++ va dispune de un meniu care prezintă comenzile posibile pentru un utilizator. Aceste comenzi sunt:

- 1) Afișare prăjituri posibile
- 2) Comandă prăjitură
- 3) Afișare prăjituri existente

Pentru o prăjitură se cunoaște numele acesteia și rețeta. Iar o rețetă este caracterizată prin nume și timp de preparare exprimat în minute. Există două tipuri de prăjituri – prăjituri obișnuite și prăjituri pentru diabetici. Pentru prăjiturile diabetice se mai cunoaște și cantitatea de zahăr.

Depozitul de prăjituri al automatului este caracterizat printr-o listă de prăjituri, capacitate maximă.

Panoul de control dispune de următoarele funcționalități:

- Preluare comandă, în cazul în care prăjitura comandată există în depozit ea va fi servită imediat utilizatorului. În cazul în care prăjitura nu există în depozit, se va crea o comandă pentru preparatorul de prăjituri folosind rețeta aferentă prăjiturii comandate. După așteptarea timpului conform rețetei, prăjitura va fi servită utilizatorului
- Returnarea numelor prăjiturilor din depozit
- Umplerea automată a depozitului cu ultima prăjitură comandată, dacă depozitul ajunge la 0, în urma unei comenzi

Preparatorul de prăjituri dispune de următoarea funcționalitate: preia o comandă de preparare (o rețetă) și simulează un timp de așteptare aferent rețetei primite. După scurgerea timpului, acesta returnează prăjitura creată.

## 3. Inițializare automat:

Înainte de punerea în funcțiune a automatului el trebuie inițializat cu lista de prăjituri/rețete pe care le poate pune la dispoziția utilizatorului.

În programul C++, această inițializare se produce înainte de afișarea meniului prin citirea unei liste de prăjituri (fiecare cu rețeta aferentă) dintr-un fișier numit rețete.txt, prin intermediul panoului de control.

## 4. Afișare prăjituri posibile:

La primirea acestei comenzi, automatul va afișa toate prăjiturile pe care le va putea procesa. În programul C++, pe ecran va fi afișată lista de prăjituri cu care a fost inițializat automatul, prin intermediul panoului de control.

## 5. Afișare prăjituri existente:

La primirea acestei comenzi, automatul va afișa toate prăjiturile care se află în depozit. În programul C++, pe ecran vor fi afișate toate prăjiturile care se află în depozit, prin intermediul panoului de control.

## **6. Comandă prăjitură:**

La primirea acestei comenzi, automatul, prin intermediul panoului de control primește numele unei prăjituri și opțional numărul de bucăți. Apoi, el verifică dacă prăjitura comandată există în depozit, dacă da va returna prăjitura găsită, dacă nu, prin intermediul preparatorului se va crea o prăjitură folosind rețeta asociată. De asemenea, dacă în urma comenzii, depozitul nu este plin, acesta se va umplea cu ultima prăjitură comandată, până la capacitate maximă.

În programul C++, comanda introdusă va fi preluată de panoul de control care face verificarea existenței prăjiturii în depozit. Dacă aceasta există, prăjitura va fi afișată pe ecran, dacă nu există, se va afișa un mesaj de informare pentru utilizator că prăjitura nu există și trebuie să aștepte un timp – indicat de rețeta asociată prăjiturii comandate, apoi se va simula crearea unei prăjituri, folosind rețeta asociată, prin intermediul preparatorului. Preparatorul va simula așteptarea tipului necesar pentru gătire. După acest timp de așteptare, prăjitura va fi afișată pe ecran prin intermediul panoului de control. De asemenea se va verifica dacă depozitul este plin și se va acționa în consecință.

## **Observații:**

În cazul introducerii unei comenzi greșite se va arunca o excepție corespunzătoare, iar mesajul de eroare va fi afișat pe ecran și în același timp va fi stocat într-un fișier log.txt.

Simularea timpului de așteptare se poate face prin intermediul bibliotecii <Windows.h>, funcția „Sleep()”.

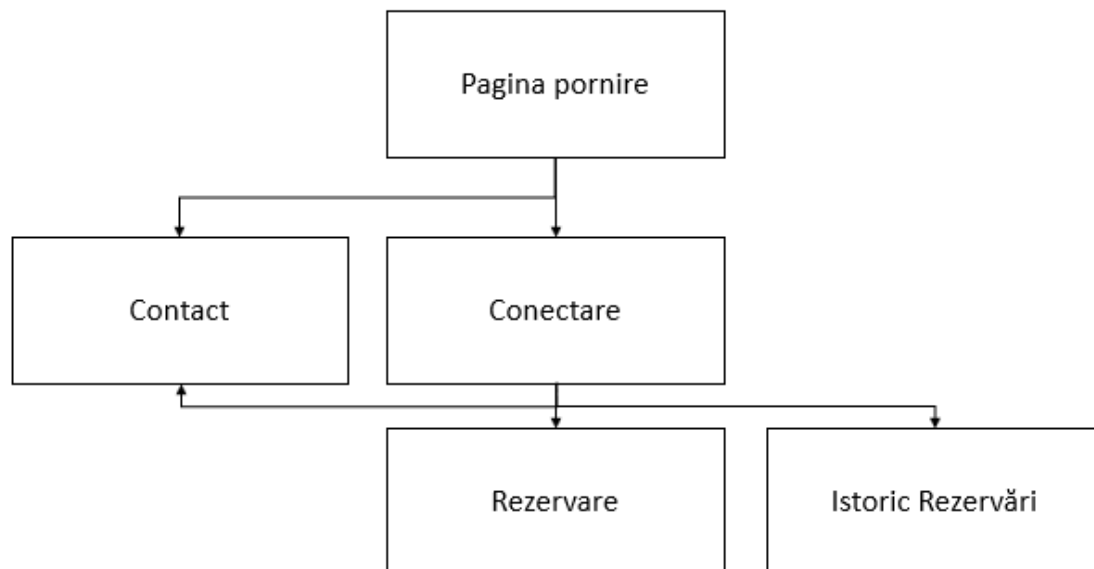
## PROIECT 2 - Booking App

**Booking App** este un proiect care necesită dezvoltarea unei **aplicații C++ tip consolă** care **simulează** o aplicație de tip web (<https://rezervari.upt.ro/>).

Programul permite realizarea anumitor rezervări de către studenți și profesori în cadrul bazei sportive a universității. **Pentru dezvoltarea acestui program, există 4 cerințe încadrate în chenarele de culoare albastră!** Fiecare cerință presupune unul sau mai multe meniuri cu diferite funcționalități.

### 1. Componentele aplicației:

Componentele aplicației sunt prezentate în schema 1.



Schema 1 – Componentele aplicației

### 2. Pagina de pornire (a unui utilizator neautentificat)

Prima pagină a website-ului este prezentată în figura 1, iar simularea c++ este prezentată în cerința 1. În această pagină utilizatorul va avea două opțiuni:

- 1) *Conectare*, care presupune deschiderea unei alte pagini care va prezenta alte trei opțiuni:
  - 1) *Înregistrare* (dacă este un utilizator nou)
  - 2) *Autentificare* (dacă este un utilizator existent)
  - 3) *Resetare parolă*
- 2) *Contact*, care permite utilizatorului să transmită un email către echipa de suport (funcționalitatea va fi descrisă în secțiunea de contact a unui utilizator autentificat)

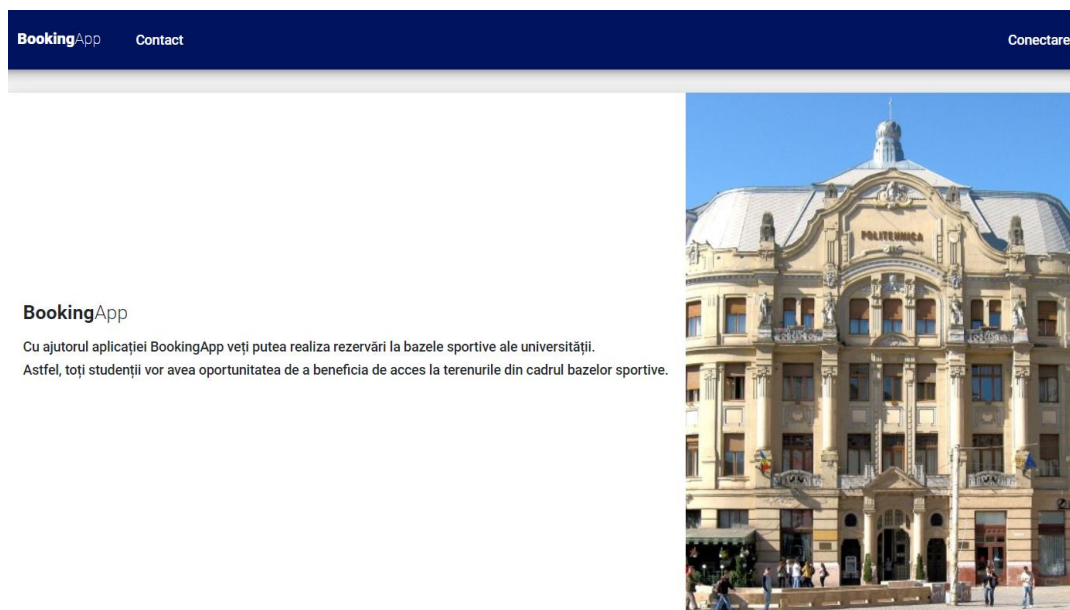


Figura 1 – Prima pagină a website-ului

**CERINȚA 1.** Se va crea un meniu pentru prima pagină care va afișa pe ecran un mesaj așa cum este prezentat în ceea ce urmează, urmat de cele două opțiuni **Conectare** și **Contact**:

BookingApp

Cu ajutorul aplicației BookingApp veți putea realiza rezervări la bazele sportive ale universității.

Astfel, studenții și profesorii vor avea oportunitatea de a beneficia de acces la terenurile din cadrul bazelor sportive.

Alegeți o opțiune:

1. Conectare
2. Contact

### 3. Fereastra de conectare

La accesarea opțiunii de conectare pe ecran va fi afișată o fereastră care permite utilizatorului să se conecteze la aplicație, să se înregistreze sau să își reseteze parola. Interfața este prezentată în figura 2 iar simularea în cerința 2.

**CERINȚA 2.** Programul C++ va pune la dispoziția utilizatorului un alt meniu cu cele 3 opțiuni **Autentificare**, **Înregistrare** și **Resetare parolă**.

Conectare

Alegeți o opțiune:

1. Autentificare
2. Înregistrare
3. Resetare parolă
4. Înapoi

Conectare

Adresă email

Trebuie să introduci o adresă de email.

Parolă

Trebuie să introduci o parolă

Autentificare

Nu ai încă un cont? [Înregistrare](#)

Ai uitat parola? [Resetează parola](#)

Figura 2 – Fereastra de conectare

La selectarea opțiunii **Înregistrare**, utilizatorul va trebui să introducă:

- a) Dacă este student: **numele, prenumele, CNP-ul** (unic), **anul, grupa**, o adresă validă de **e-mail** (unic) și o **parolă** (trebuie să conțină cel puțin 3 caractere).
- b) Dacă este profesor: **numele, prenumele, CNP-ul** (unic), **vechime**, o adresă validă de **e-mail** (unic) și o **parolă** (trebuie să conțină cel puțin 3 caractere).

Aceste date vor fi stocate într-un fișier text, după care va fi afișat un mesaj de succes.

La selectarea opțiunii *Autentificare*, utilizatorul va trebui să introducă o adresă de e-mail validă și o parolă. În cazul în care adresa specificată și parola corespund unui utilizator înregistrat, autentificarea are succes, iar pe ecran va apărea un nou meniu, un meniu al unui utilizator autentificat. În toate etapele, vor fi afișate mesaje corespunzătoare.

La selectarea opțiunii *Resetare parolă*, utilizatorul va trebui să introducă numele, prenumele și adresa de e-mail. Dacă toate acestea corespund unui utilizator înregistrat, atunci se va introduce o nouă parolă care va înlocui parola actuală pentru utilizatorul identificat. După care se va afișa un mesaj corespunzător de succes.

## 4. Pagina unui utilizator autentificat

După ce utilizatorul se autentifică, acesta dispune de o nouă pagină, prezentată în figura 3, respectiv simularea c++ din cerința 3, care oferă următoarele opțiuni:

- a) Rezervare
- b) Istoric rezervări
- c) Contact (același meniu de contact de care dispune un utilizator neautentificat)

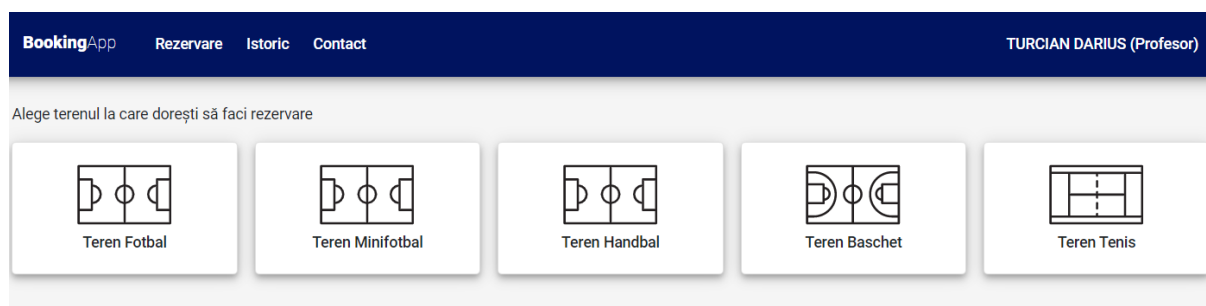


Figura 3 – Pagina unui utilizator autentificat

**CERINȚA 3.** Implementarea C++ presupune afișarea următorului meniu:

BookingApp (Nume, Prenume – Tip: student/profesor)

Alegeți o opțiune:

1. Rezervare
2. Istoric
3. Contact
4. Înapoi

1. La selectarea opțiunii **Rezervare**, pe ecran va fi afișat un nou meniu cu terenurile disponibile. Pentru un profesor, vor fi disponibile următoarele terenuri: fotbal, minifotbal, handbal, baschet și tenis. Pentru un student, vor fi disponibile numai terenurile: minifotbal, handbal, baschet și tenis. Meniul afișat va fi (ex. student – dacă se alege opțiunea pentru Teren Fotbal va fi afișat un mesaj de eroare):

BookingApp (Nume, Prenume – Student)

Alegeți terenul pentru care doriți să faceți o rezervare:

1. Teren Minifotbal
2. Teren Handbal
3. Teren Baschet
4. Teren Tennis
5. Înapoi

La selectarea oricărui teren, se va adăuga o nouă rezervare, care va conține numele, prenumele, e-mailul, tipul (profesor/student), terenul selectat și data și ora programării. Se va verifica dacă nu există o altă programare la aceeași dată, pe același teren și de asemenea se va verifica, ca data introdusă să fie în viitor. Datele se vor stoca într-un fișier text.

2. La selectarea opțiunii **Istoric**, pe ecran se vor afișa toate rezervările efectuate de utilizatorul autentificat.

3. La selectarea opțiunii **Contact**, la fel, ca și în meniul unui utilizator neautentificat, va fi afișată pe ecran pagina din figura 4, respectiv simularea c++ din cerința 4.

## 5. Pagina de contact

Figura 4 – Pagina de contact

### CERINȚA 4. Implementarea C++ va dispune de următorul meniu:

BookingApp (Nume, Prenume – Tip: student/profesor, dacă utilizatorul este autentificat)

Suport tehnic

În cazul în care întâlniți dificultăți echipa de suport vă va sta la dispoziție.

Tot ce trebuie să faceți este să completați următorul formular în care trebuie descrisă problema.

Alegeți o opțiune:

1. Trimitere sesizare
2. Înapoi

La selectarea opțiunii Trimitere sesizare, va trebui introdusă o problemă. Sesizarea va conține astfel: numele și prenumele, adresa de e-mail, CNP-ul și conținutul problemei. Se va transmite un e-mail către adresa [support@bookingapp.com](mailto:support@bookingapp.com) (pentru testare se va utiliza adresa unui prieten). Adresa care va transmite e-mailul de mai sus va fi cea a programatorului împreună cu datele de conectare ale acestuia (se va utiliza numai la testare, de asemenea credențialele se pot utiliza dintr-un fișier text local – datele private nu trebuie expuse în proiect).

Forma unui e-mail poate arăta după cum urmează:

Buna ziua,  
Subsemnatul Nume Prenume, avand CNP: 1250122202020  
Nu pot accesa website-ul, va rog sa ma ajutati in aceasta privinta.  
Multumesc,  
Nume Prenume

## Observații

În toate etapele, atunci când utilizatorul va introduce date greșite, mesajele de eroare se vor stoca în fișierul *log.txt*.

Se vor utiliza liste, supraîncărcarea operatorilor, scrierea în fișiere, tratarea excepțiilor.

### Transmitere e-mail

Pentru transmiterea unui email dintr-o aplicație c++ este necesară conectarea la o rețea de internet fără proxy (se pot utiliza date mobile în acest caz).

Pași necesari:

1. Descărcare cURL <https://curl.se/windows/>
2. Extragere arhivă și copiere **conținut** folder *bin* în același folder cu proiectul C++
3. Includere bibliotecă <Windows.h>, creare comandă *curl* și lansarea curl.exe prin comanda **WinExec("comanda trimitere email", SW\_HIDE);** sau prin utilizarea CreateProcess.
4. Informații creare comanda pentru trimitere e-mail utilizând cURL: <https://ec.haxx.se/usingcurl/usingcurl-smtp>

În ceea ce privește adresa de e-mail a programatorului există și posibilitatea înregistrării unei aplicații pentru a putea permite server-ului de e-mail (ex. gmail) să se autentifice și să transmită un e-mail, însă calea cea mai simplă este modificarea setării "*Allow less secure apps: ON*" - pentru un cont **gmail**. Mai multe detalii la: <https://myaccount.google.com/lesssecureapp>.