

## **FOOD-DELIVERY**

- O companie de food delivery dorește să țină evidența asupra comenzilor mai multor restaurante locale prin implementarea unei aplicații. Se dorește ca programul dezvoltat să ofere o experiență eficientă și transparentă atât clienților cât și personalului angajat în ceea ce privește fiecare proces prin care trece o comandă.
- Astfel, se dorește implementarea următoarelor funcționalități destinate clienților:
  - **Plasarea Comenzii:** Utilizatorul trebuie să aibă abilitatea de plasa comenzi de la un anumit restaurant prin intermediul aplicației. Aceasta funcție trebuie să permită clientului adaugarea unor produse în comanda sa și plasarea unui punct unde aceasta va fi livrată.
  - **Urmărirea Comenzii:** Utilizatorul va putea vizualiza starea comenzii sale în timp real. (“Comandă Confirmată”, “În curs de preparare”, “Comandă Ridicată”, “În curs de livrare”, “Comandă livrată”)
- Restaurantele vor dispune de funcționalități precum:
  - **Management Comanda:** Personalul trebuie să poată accepta comenzi și să modifice starea acestora în timp real.
  - **Timp Estimat:** În funcție de comanda primită aplicația trebuie să calculeze un timp estimat privind procesarea și livrarea, bazat pe numărul de comenzi primite și timpul mediu de pregătire, stabilit de fiecare restaurant.
- **Interfața:**
  - Interfața aplicației trebuie să fie intuitivă și ușor de utilizat indiferent de statutul utilizatorului (client/personal restaurant), cu o modalitate de navigare fluidă între diferitele funcționalități și secțiuni ale acesteia.
  - Aplicația fiind destinată unei multitudini de clienți, trebuie să ofere suport lingvistic aferent preferințelor utilizatorului.

## **INDICAȚII:**

În vederea realizării aplicației se pot folosi următoarele resurse:

- Metoda Greedy (Destinat claselor a IX-a - a X-a):  
<https://www.geeksforgeeks.org/greedy-algorithms/>
- Algoritmul Bellman-Ford:  
[https://www.researchgate.net/figure/The-minimum-cost-maximal-2-matching-problem-with-euclidean-distances\\_fig1\\_362831396](https://www.researchgate.net/figure/The-minimum-cost-maximal-2-matching-problem-with-euclidean-distances_fig1_362831396)  
<https://www.geeksforgeeks.org/bellman-ford-algorithm-dp-23/>  
<https://www.infoarena.ro/problema/bellmanford>
- Algoritmul Edmonds-Karp:  
[https://www.w3schools.com/dsa/dsa\\_algo\\_graphs\\_edmondskarp.php](https://www.w3schools.com/dsa/dsa_algo_graphs_edmondskarp.php)
- OpenStreetMap (Pentru apelul API):  
<https://www.openstreetmap.org/#map=14/45.4350/28.0363>  
<https://technologyadvice.com/blog/information-technology/how-to-use-an-api/>
- Google Maps (Pentru testarea rutelor alese)
- Teorie Grafuri:

<https://www.pbinfo.ro/articole/810/grafuri-neorientate>

<https://www.pbinfo.ro/articole/509/grafuri-orientate>

- Realizarea interfeței:

<https://en.cppreference.com/w/cpp/links/libs>

[https://www.reddit.com/r/cpp/comments/babfl5/a\\_pretty\\_big\\_list\\_of\\_c\\_gui\\_libraries/](https://www.reddit.com/r/cpp/comments/babfl5/a_pretty_big_list_of_c_gui_libraries/)

- Se pot adresa întrebări pe serverul de Discord dedicat acestei secțiuni:

<https://discord.gg/xPC7a5nr>

## **CRITERII JURIZARE :**

Capitolul I. Arhitectura aplicației – 10 puncte

Secțiunea I.1. – Tehnologiile folosite:

- Alegerea tehnologiilor potrivite

Secțiunea I.2. – Proiectarea arhitecturală:

- Utilizarea corectă a paradigmatelor și tehnicilor de programare

Capitolul II. Implementarea aplicației – 20 puncte

Secțiunea II.1. – Eleganța implementării

- Clase/module/funcții ușor de extins, re folosibile și încapsulate
- Complexitatea tehnică

· Lizibilitatea codului și respectarea standardelor de programare (variabile definite cu nume semnificative, cod consecvent, documentat, etc.)

Secțiunea II.2. – Testarea aplicației

- Modalități de testare
- Absența erorilor (dacă există o versiune nefinalizată dar fără erori ce se va prezenta juriului, se va lua în calcul. Astfel se pot veni cu explicații asupra unei versiuni incomplete).

Capitolul III. Interfață – 10 puncte

Secțiunea III.1. – Impresia generală

- Posibilitatea de a se adapta la mai multe rezoluții (Responsiveness)
- Internaționalizare, posibilitatea modificării limbii pentru minorități
- Corectitudine gramaticală

Secțiunea III.2. – Ușurința în folosire

- Interfață intuitivă, ușor de parcurs

Capitolul IV. Conținut – 40 puncte

Secțiunea IV.1. – Funcționalitate, utilitate și interactivitate

- Utilitatea funcționalităților

Secțiunea IV.2. – Evaluare și feedback

- Îl ajută pe utilizator să înțeleagă ce a greșit

Secțiunea IV.3. – Posibilitatea de a actualiza și gestiona conținutul

- Simplitatea actualizării/gestionării conținutului (din aplicație/metode alternative)

Capitolul V. Originalitate și inovație – 10 puncte

Secțiunea V.1. · Originalitatea ideii sau inovații aduse față de soluții existente/adaptarea unor soluții existente pentru un rezultat unic.

Capitolul VI. Prezentarea proiectului – 10 puncte

Secțiunea VI.1. – Impresia generală a proiectului

- Impresia generală a proiectului
- Sunt aduse răspunsuri coerente altor participanți/juriului în timpul/după prezentarea proiectului

Secțiunea VI.2. – Documentația proiectului

- Sunt prezentate informații generice despre proiect

**Concurs regional de creativitate în IT „Severin Bumbaru” - Ediția 2024**

Secțiunea Programare - Clasele IX-XII

- Este descrisă arhitectura aplicației
- Sunt justificate tehnologiile alese