

# ROOTLEAF

## Primul raport – Aplicația mobilă propusă

Stoian Andreea-Ionela  
Departamentul de Informatică  
Facultatea de Matematică și Informatică  
Universitatea de Vest, Timișoara  
Supervizor: Mafteiu-Scai Liviu Octavian

Mai 2021

### **Rezumat**

Acest raport conține o descriere succintă a proiectului aferent cursului Programare pe dispozitive mobile. Obiectivul este realizarea unei imagini de ansamblu asupra funcționalității și a evidențierii notelor distinctive oferite față de aplicațiile deja existente pe piață.

## Cuprins

1	Scop și utilizatori	3
2	Introducere	3
3	Aplicații similare	4
4	Contribuția autorului	5
5	Plan de dezvoltare	5

## 1 Scop și utilizatori

Grupul țintă al acestei aplicații este unul destul de specific. Reprezentând un spațiu de lucru aferent anumitor tipuri de arbori, utilizatorul va găsi o utilitate în folosirea acesteia doar în cazul în care este înrolat în cadrul unui curs ce utilizează structurile de date propuse sau în cazul în care dorește să aprofundeze caracteristicile sau operațiile aferente acestora.

Acest aspect nu generează neapărat concluzia unui grup țintă de dimensiuni reduse, întrucât structurile de date reprezintă o parte destul de însemnată în studiul informaticii, astfel încât atât universitățile, cât și ciclurile inferioare de studiu pot folosi aplicația ca și platformă de lucru.

Aici intervine și scopul principal al acestei aplicații, și anume oferirea unei platforme destinate integral utilizării drept spațiu de lucru în studiul și aprofundarea arborilor.

## 2 Introducere

Aplicația oferă oportunitatea alegerii arborelui asupra căruia vor fi efectuate operațiile. Punctul de pornire este reprezentat de un catalog ce conține arborii studiați în cursurile ce parcurg structurile de date abordate în cadrul *Universității de Vest din Timișoara: Algoritmi și structuri de date*, respectiv *Structuri de date avansate*. Astfel încât, pentru început, se poate opta pentru unul dintre următorii arbori: arbore binar de căutare, arbore Splay, arbore AVL, arbore B sau arbore Red Black.

Odată ales tipul de arbore asupra căruia se efectuează operațiile, este necesară optarea către una dintre operațiile disponibile.

Pe lângă operațiile comune tuturor tipurilor de arbori (inserare nod, ștergere nod, afișare arbore, succesorul/predecesorul unui nod, nodul minim/maxim, înălțime arbore, parcurgerea în inordine/preordine/postordine, etc), sunt prezente și operații specifice tipului de arbore ales (*în cazul arborelui Red Black, de exemplu, se poate selecta înălțimea black*).

Anumite operații necesită introducerea unor informații suplimentare (*introducerea valorii nodului pentru care se va căuta succesorul, de exemplu*), în timp ce altele returnează rezultatul în momentul selectării (*afișarea valorii nodului minim nu necesită decât selecția operației*).

La fiecare pas există opțiunea de a reveni la meniul principal pentru selecția unui nou tip de arbore.

Întrucât aplicația folosește ca și structuri de date doar arbori, a fost intitulată sugestiv *RootLeaf*, conținând două elemente comune tuturor tipurilor de arbori: rădăcina, respectiv frunzele.



Figura 1: App icon

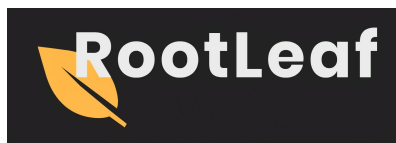


Figura 2: Logo

### 3 Aplicații similare

Deoarece vorbim strict de aplicații destinate utilizării mobile, piața nu cuprinde o gamă variată de opțiuni.

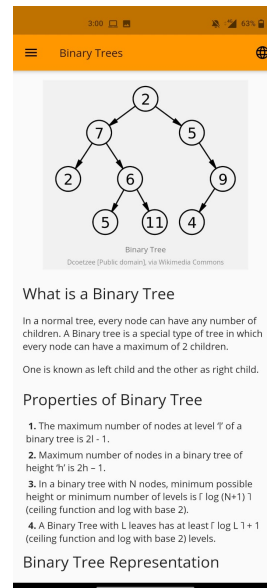
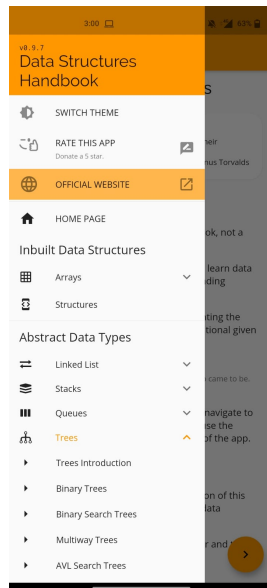
Aplicațiile deja existente se focusează asupra părții teoretice al acestui subiect, cuprinzând teoria generală a anumitor tipuri de arbori, respectiv exemplificarea acestora pe un exemplu fixat.

Prima aplicație de acest gen este *Algorithms and data structures in dynamics*, care odată accesată oferă posibilitatea alegerii a una dintre cele 4 opțiuni prezente: graf, vector, arbore binar de căutare sau hash table, fiecare dintre ele având un număr disponibil de 2, până la 6 operații. Odată aleasă operația, aceasta va fi explicată pe pași pe un exemplu fixat.



O altă aplicație asemănătoare o reprezintă *Data Structures Handbook*, care de data aceasta oferă o gamă mai variată de structuri de date, însoțite de teoria aferentă fiecăreia. Suplimentar, aceasta oferă și resurse adiționale de materiale teoretice și deține și o listă de vizualizatoare pentru fiecare dintre structurile de

date, care însă funcționează prin intermediul unui site afiliat și nu direct prin aplicație.



## 4 Contribuția autorului

Având în vedere axarea a majorității aplicațiilor deja existente asupra părții teoretice a structurilor de date, principala adăugare o reprezintă prezentarea părții practice a acestora. Utilizatorului i se oferă mai degrabă un spațiu de lucru în locul unui mediu de învățare.

Un alt adaos îl reprezintă și numărul operațiilor, întrucât în majoritatea cazurilor unui vizualizator se insistă doar pe operațiile de adăugare, ștergere și vizualizare.

De altfel, de regulă este tratat un singur tip de arbore, ca și în cazul aplicației *Algorithms and data structures*, diferit în cazul aplicației propuse fiind tratarea a mai multor tipuri de arbori simultan.

## 5 Plan de dezvoltare

Ca și mediu de dezvoltare, am folosit IDE-ul *Android Studio*, iar codul ce stă la baza implementării arborilor este realizat în limbajul de programare *Java*.

Interfața a fost realizată sub forma unui meniu principal din care se poate selecta tipul de arbore dorit. Cuprinde 5 butoane, fiecare aferent unui tip de

arbore care redirecționează utilizatorul către un meniu secundar, ce cuprinde operațiile aferente fiecăruia.

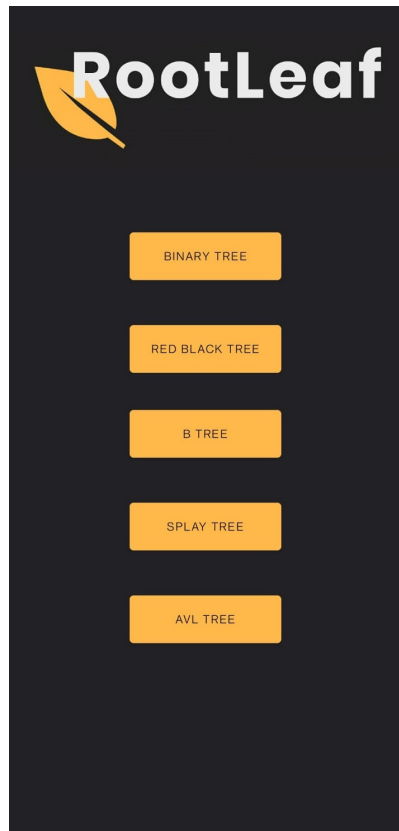


Figura 3: Meniu principal

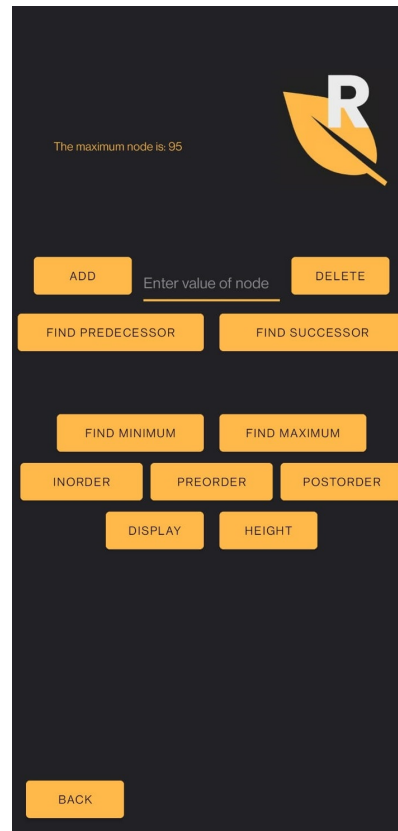


Figura 4: Meniu secundar

Meniul secundar cuprinde un *EditText* destinat inserării nodurilor pentru operațiile de adăugare/ștergere/găsire succesori/găsire predecesori. Dacă oricare dintre aceste butoane va fi apăsat fără inserarea unui nod, pe ecran va apărea un *Toast* sugestiv (Figura 5). Asemănător se va întâmpla și în cazul apăsării unui buton aferent unei operații care necesită existența unui arbore, însă arborele este încă gol (Figura 6). Pentru afișare este folosit un *TextView* care în funcție de butonul apăsat, va afișa un text sugestiv aferent rezultatului (Figura 4).

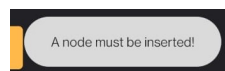


Figura 5: mesaj Toast

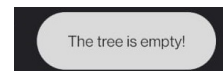


Figura 6: mesaj Toast

## Anexe

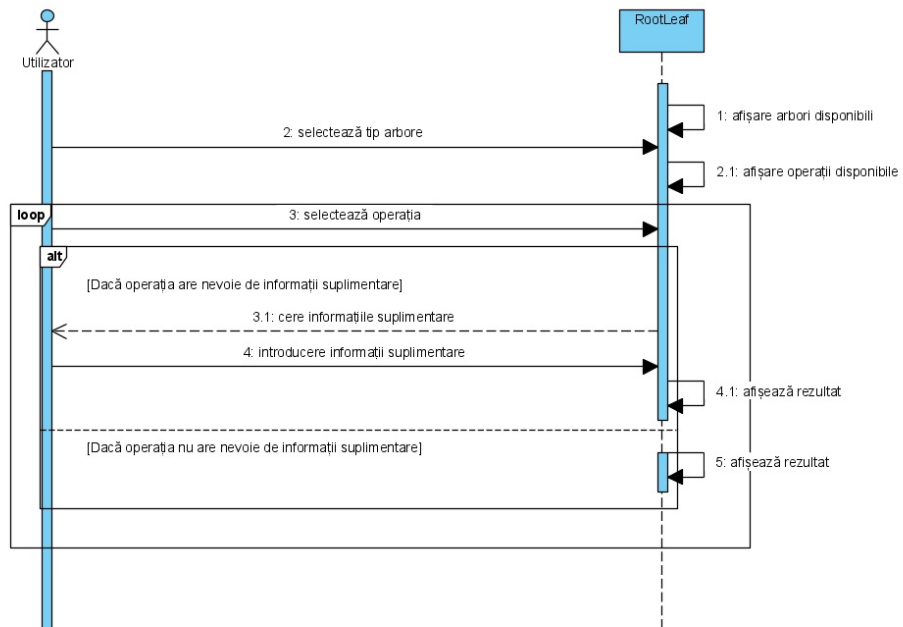


Figura 7: Diagrama de secvență