

# Documentație Busify

## Introducere:

Busify este o aplicație iOS care are ca scop ușurarea călătoririi cu transportul în comun din Cluj prin centralizarea și afișarea datelor despre vehicule și linii, atât sub o formă compactă tabelară, cât și pe o hartă live. Această documentație oferă un ghid detaliat și complet despre utilizarea și integrarea funcțiilor pe care aplicația le are de oferit.

Aplicația este disponibilă pe App Store cu un total de 330 descărcări în luna mai 2023, iar numărul continuă să crească.

## Instalare:

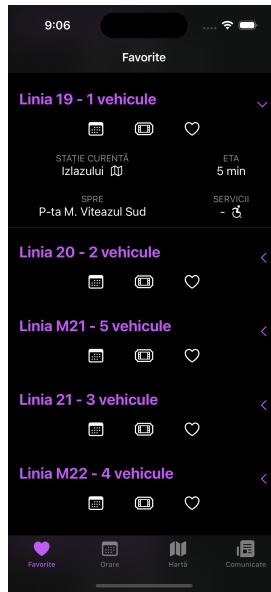
- Cererile sistemului: compatibil cu orice iPhone care rulează pe iOS 16.1 sau mai recent pentru versiunea de pe App Store, dar harta live este disponibilă și pe site-ul oficial. (<https://busify-cluj.web.app/>)
- Descărcare: Busify este disponibilă pe App Store. (<https://apps.apple.com/us/app/busify/id6447985147>)
- Permisuni: La prima deschidere a aplicației, aceasta va cere permisiuni optionale precum locația dispozitivului (pentru calcularea ETA-ului și centrarea corectă a hărții) și acces la notificări (pentru noile știri și pentru notificarea despre Vinerea Verde). Aceste permisiuni sunt optionale și pot fi ignorate, întrucât funcționarea corectă a aplicației nu depinde de acestea, dar au ca scop îmbunătățirea acesteia și sporirea convenienței utilizării ei.
- Alte cerințe: Conexiune obligatorie la internet pentru obținerea datelor din servere sau din dispozitivele din împrejurime.

## Interfață:

Utilizatorul beneficiază de o interfață prietenoasă, ușor de navigat, menită să facă utilizarea ei o plăcere și persoanelor mai în vîrstă prin modul de organizare a datelor, etichetelor atașate fiecărui tip de dată afișat (de exemplu eticheta "Stație curentă" și "ETA") și butoanelor mari, contrastante. Aplicația utilizează aceeași paletă de culori ca și vehiculele CTP (negru, mov, gri și indigo) pentru a da un sens de familiaritate și pentru a fi ușor de distins. Aceasta se împarte în 4 ecrane, dispuse în bara din partea inferioră a ecranului, anume: Toate Liniile, Orare, Hartă și Știri.

Ecranul Toate Liniile: dispune toate datele necesare utilizării de bază a aplicației sub forma tabelară, iar liniile sunt dispuse sub forma unei liste. La cerea utilizatorilor, pentru conveniență, vehiculele din cadrul unei linii sunt ascunse la început, iar utilizatorii pot extinde lista vehiculelor din butonul "<". Fiecare linie dispune de 3 butoane, fiecare având o pictogramă intuitivă, unul care duce utilizatorul la orarul linie, unul care duce utilizatorul pe aplicația de SMS-uri, cu toate câmpurile completate automat, având un mesaj gata de trimis pentru a cumpăra un bilet și încă unul care adaugă sau șterge linia de la favorite.

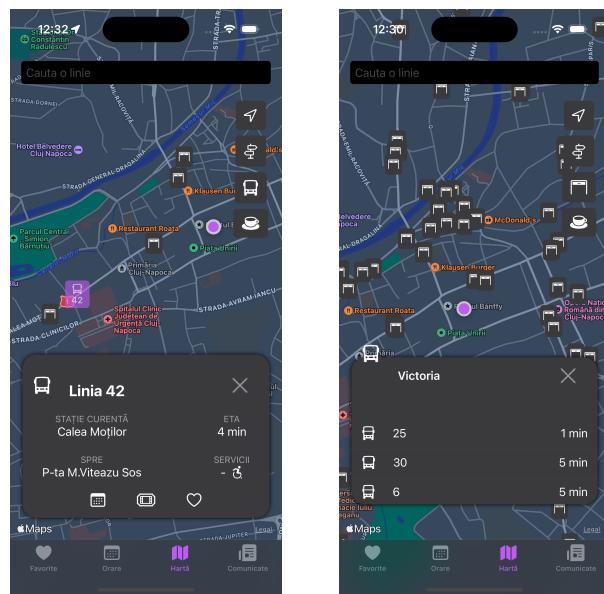
Aici sunt dispuse date precum stația curentă a unui vehicul (cu un buton pentru schimbarea ecranului pe "Hartă" și centrarea ei pe acel vehicul), încotro se îndreaptă, timpul aproximativ ca acel vehicul să ajungă la locația utilizatorului și serviciile de care dispune precum suport pentru biciclete sau rampă pentru persoanele cu dizabilități. De asemenea, acest ecran conține și un search bar pentru scrolarea rapidă la o anumită linie și un toggle pentru afișarea tuturor liniilor sau doar a celor favorite.



Ecranul Orare: dispune o listă cu toate liniile sub forma unui tabel. Deasupra lor, în același fel, sunt dispuse liniile favorite pentru acces rapid. La click-ul oricărei linii din aceste 2 liste se va deschide un ecran secundar cu orarul corespunzător. De asemenea, deasupra celor 2 liste, după caz, vor apărea alerte dacă programul vehiculelor este alterat din cauza unui eveniment și vor dispărea odată ce orarul special nu mai este valabil. Orarele sunt împărtășite pe 4 pagini (luni-vineri, sămătă, duminică și

stații), care se deschide automat pe pagina corespunzătoare și poate fi modificat ulterior de către utilizator.

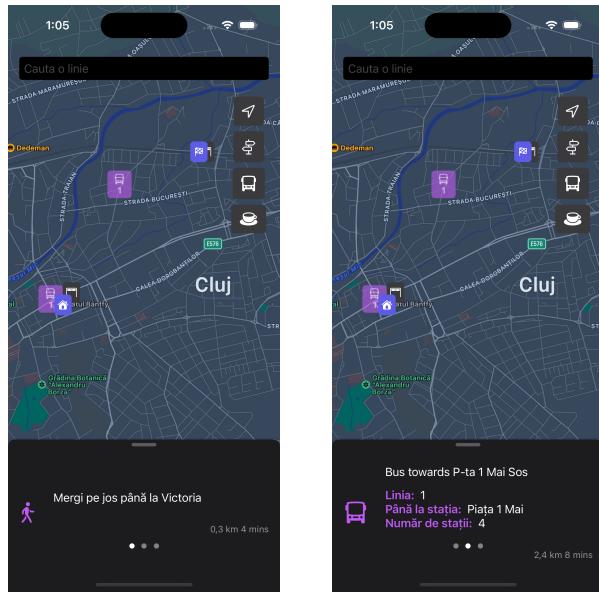
Ecranul Hartă: aici se află componența principală a aplicației, unde fiecare vehicul își are propriul marker, care conține pictogramă specifică tipului de vehicul și numărul liniei, plasat la coordonatele corespunzătoare din viața reală. La apăsarea fiecărui dintre aceste marker va apărea o fereastră, egală cu un sfert din înălțimea ecranului device-ului, asemănător cu cel din Toate Liniile, cu informații precum stația la care se află, ETA-ul, încotro se îndreaptă și serviciile pe care le oferă vehiculul. În același timp, toate celelalte vehicule, încăfară de cel selectat, vor dispărea și vor apărea markere pentru stațiile de pe ruta sa. La apăsarea unui marker corespunzător stațiilor va apărea o alta fereastră cu informații legate de traficul vehiculelor CTP din acea stație, asemănător cu ecranul din stațiile din viața reală.



De asemenea, acest ecran conține un search bar care va recentra harta pe vehiculul căutat, plus o serie de 4 butoane. Primul buton are ca scop recentrarea hărții pe locația utilizatorului, al doilea deschide ecranul pentru direcții, al treilea schimbă modul de vizualizare a hărții de la toate vehiculele la toate stațiile, iar ultimul deschide ecranul pentru donații.

Ecranul de direcții este o fereastră egală cu jumătate din înălțimea ecranului și permite utilizatorilor abonați să introducă 2 adrese (sau să folosească și locația curentă) pentru a primi direcții pas cu pas până în acea locație. Odată introduse aceste date, fereastra se micșorează până la 25% din înălțimea ecranului device-ului și vor dispune direcțiile pas cu pas, împărțite pe pagini asemănătoare celor din ecranul Orare. Aceste direcții

pot fi de exemplu “mergi până la stația Memorandumului Nord” sau “Linia 25 până la Grădini Mănăștur” cu alte detalii precum durata fiecărui pas, distanța parcursă sau numărul de linii. Marker-ele, de asemenea, se vor schimba și vor arăta toate vehiculele din liniile pe care le poate lua utilizatorul, stațiile relevante și încă 2 markere semnificând de unde a plecat și unde va ajunge pentru a verifica dacă datele au fost procesate corect.



Ecranul Știri: prezintă un istoric conținând ultimele 10 știri publicate pe site-ul oficial CTP. Se află sub formă de listă, unde fiecare știre este descrisă de titlu și un preview, iar la click-ul oricăreia, utilizatorul va fi dus pe site-ul oficial CTP pentru întreg articolul.



## Funcționalități și algoritmi:

Gestionarea datelor: datele furnizate sunt gestionate în mod intelligent printr-un algoritm asemănător cu cel al Spotify, care la momentul actual este 90% implementat. Când utilizatorul își deschide aplicația, aceasta scanează device-urile din jur pe o arie de 50m printr-un algoritm de peer connectivity, se conectează la cel care are de asemenea Busify activ și deschis și furnizează datele din interiorul acelui dispozitiv, care ar face încărcarea datelor să dureze în jur de 1.3s. Algoritmul, la momentul actual, nu este public, doarece dispozitivul se poate conecta la un alt singur dispozitiv, dar dacă s-ar putea conecta la mai multe dispozitive deodată, informațiile ar fi furnizate din mai multe surse în același timp și ar putea tranșa drastic timpul de încărcare.

Dacă un astfel de dispozitiv nu este disponibil, iar algoritmul de peer connectivity nu poate fi aplicat, atunci, odată la 20s, se va face un apel la server-ul propriu care, la rândul lui, odată la 15s face apel la server-ele oficiale (ale primăriei) pentru obținerea datelor. Am ales să folosesc propriul server doarece este mai sigur ca utilizatorii să fie conectați într-un mediu separat, sigur și controlat, și pentru a reduce numărul de apeluri la server-ele oficiale (care la momentul actual sunt 50.000 pe zi). Acest tip de încărcare a datelor variază între 1.7s și 4s, depinzând de traficul aflat pe server-ul oficial la acel moment.

Datele sunt preluate sub formă de date JSON și decodate în aplicație, apoi atribuite în structurile specifice fiecăreia.

```
1 func getVehicles() async throws -> [Vehicle]{
2     let randomKey = Constants.keys.randomElement()
3     let _headers : HttpHeaders = ["Content-Type" : "application/json",
4                                   "X-Agency-Id" : "2",
5                                   "X-API-KEY" : randomKey ?? "aSVGUmECaW8VbAPYnzJFM21KeFiz1suP2z55ZzhL"]
6     let getRequest = AF.request("https://api.tranz.y.dev/v1/opendata/vehicles", method: .get, parameters: [:], encoding: URLEncoding.default, headers: _headers)
7     var responseJson : String!
8     do {
9         responseJson = try await getRequest.serializingString().value
10    } catch let err{
11        print(err)
12    }
13    let responseData = try? DecodingManager().decodeVehicles(jsonString: responseJson ?? "")
14    return responseData ?? [Vehicle]()
15 }
```

Apelul către server pentru obținerea datelor despre vehicule

Ecranul Orare: după selecarea unei linii, orarul acesteia se va deschide pe pagina corespunzătoare dintre cele 3, în funcție de ziua curentă sau de orarul special aflat în efect. De asemenea, pagina se deschide automat la poziția potrivită a scroll-ului, adică a urmatoarei plecărei, modificând culoarea și tăind cu o linie cursele deja începute și îngroșându-le pe cele care vor urma să înceapă în mai puțin de 15 minute.

De asemenea, este posibilă descărcarea orarului unei linii în aplicație, care va putea fi accesat ușor și rapid în situațiile în care o conexiune la internet este necesară.

La apăsarea fermă pe una dintre plecări, se va deschide un context menu, conținând un buton pentru a primi o notificare în momentul în care acel vehicul începe cursa respectivă.

Ecranul Hartă: pentru a face distincția între vehiculele de pe sensuri de mers diferite, am introdus un algoritm care calculează cea mai apropiată stație față de utilizator și schimbă opacitatea vehiculelor care nu se află pe acea rută, astfel evidențiându-le numai pe cele relevante. Această funcționalitate a fost implementată tot la cerința utilizatorilor.

```

1  Grid {
2      ForEach(item.vehicles, id:\.self) {vehicle in
3          GridRow {
4              VStack{
5                  Text("STANȚIE CURENTĂ")
6                      .font(.footnote)
7                      .foregroundColor(.secondary)
8                  HStack {
9                      Text("\(vehicle.statie ?? "Necunoscută")")
10                     Image(systemName: "map")
11                     .font(.callout)
12                 }
13                 .padding(.bottom)
14                 .onTapGesture {
15                     let location = CLLocationCoordinate2D(latitude: vehicle.latitude ?? 0, longitude: vehicle.longitude ?? 0)
16                     let span = MKCoordinateSpan(latitudeDelta: 0.005, longitudeDelta: 0.005)
17                     locationManager.region = MKCoordinateRegion(center: location, span: span)
18                     selectedTab = 2
19                 }
20             }
21             VStack {
22                 Text("ETA")
23                     .font(.footnote)
24                     .foregroundColor(.secondary)
25                 Text("\(vehicle.eta ?? 0) min")
26                     .padding(.bottom)
27             }
28         }
29         .contextMenu {
30             Button {
31                 let location = CLLocationCoordinate2D(latitude: vehicle.latitude ?? 0, longitude: vehicle.longitude ?? 0)
32                 let span = MKCoordinateSpan(latitudeDelta: 0.005, longitudeDelta: 0.005)
33                 locationManager.region = MKCoordinateRegion(center: location, span: span)
34                 selectedTab = 2
35             } label: {
36                 Label("Vezi în hartă", systemImage: "map.fill")
37             }
38         }
39     }
40     GridRow {
41         VStack {
42             Text("SPRE")
43                 .font(.footnote)
44                 .foregroundColor(.secondary)
45             Text(vehicle.headsign ?? "Necunoscut")
46         }
47         VStack{
48             Text("SERVICII")
49                 .font(.footnote)
50                 .foregroundColor(.secondary)
51             HStack{
52                 if vehicle.bikeAccessible == "BIKE_ACCESSIBLE"{
53                     Image(systemName: "bicycle")
54                 } else {
55                     Text("-")
56                 }
57                 if vehicle.wheelchairAccessible == "WHEELCHAIR_ACCESSIBLE"{
58                     Image(systemName: "figure.roll")
59                 } else {
60                     Text("-")
61                 }
62             }
63         }
64     }
65     Divider()
66         .background(Color.purple)
67         .transition(.move(edge: .top))
68     }
69 }

```

Ecranul Știri: Informațiile despre știri nu sunt preluate direct pe site-ul CTP, ci am interceptat sursa lor printr-un serviciu proxy și am găsit server-ul public RSS la care se fac apelurile, la care ulterior am conectat restul algoritmului.

App Clip: este un cod asemănător cu cel de jos, sau poate fi chiar și un cod QR, care odată scanat cu un dispozitiv iOS va instala numai harta live a aplicației, în cazul în care utilizatorul nu are deja Busify instalat. Acest lucru este folosit în scenariul în care cineva așteaptă în stație și poate accesa cu ușurință locația vehiculului dorit. După 8 ore, harta live se va șterge automat, dar poate fi instalat din nou prin scanarea codului.



### **Integrare cu servicii externe:**

Datele care descriu un program special sunt scrise manual și postează într-un fișier pe site-ul aplicației, la care aplicația face apel pentru a verifica noutăți de fiecare dată când este deschisă.

ETA-ul este calculat dinamic, în funcție de viteza vehiculului, distanța între el și user și datele despre trafic pe ruta aceluia vehicul (provenit prin API-ul de la Google Maps explicitat mai jos).

Informațiile despre direcții provin de la API-ul de direcții de la Google Maps, fiind în esență un alt server la care se fac apeluri folosind cheia privată, coordonatele de început, coordonatele destinației și tipul de călătorie dorit (adică transport în comun).

Informațiile despre vehicule provin de la nou API Tranz, la care m-am alăturat încă de la lansare și la care am ajutat prin testare continuă și feedback constructiv pentru a rezolva potențialele erori. De menționat este faptul că harta site-ului oficial CTP folosește aceeași sursă precum Busify.

Doarece implementarea oficială a achizițiilor în aplicație este complexă, folosesc un serviciu extern numit RevenueCat, care înlocuiește clase întregi de cod în cîteva linii, iar restul se face pe site-ul lor oficial într-o manieră mai intuitivă și ușor de înțeles.

Notificările dinamice provin de la un serviciu de încredere pe care l-am folosit și în alte proiecte numit OneSignal. În concordanță cu acest serviciu folosesc Zapier pentru a “asculta” server-ul RSS pentru noi articole. Când se iveste o nouă știre, Zapier trimite o notificare către OneSignal care creează o notificare și o trimite tuturor utilizatorilor care au acceptat să primească astfel de notificări.

Server-ul propriu este oferit de către DigitalOcean și momentan se află în Varșovia, folosit la stocarea datelor din server-ele oficiale și la care se conectează fiecare dispozitiv în momentul deschiderii aplicației.

### **Troubleshooting și bug-uri cunoscute:**

- Serviciile de localizare: Dacă ETA-ul nu este calculat corespunzător, sau centrarea hărții nu se realizează corect verifică dacă aplicației îi este permis accesul la locația device-ului.
- Conexiunea la internet: În instanțele în care aplicația refuză să se încarce se va afișa un mesaj avertizând acest lucru. Verifică că dispozitivul are o conexiune activă la internet.
- Notificări: În cazul în care dispozitivul nu livrează notificări precum cele pentru Vinerea Verde sau noile știri, verifică că aplicația are acces să trimită notificări din setări.

Doarece atât utilizatorii, cât și testerii, mereu au avut căi să mă contacteze, bug-urile pe care le-au descoperit au fost adresate cât de repede a fost posibil. La momentul actual, luna mai 2023 (versiune App Store 1.2.2), următoarele bug-uri sunt cunoscute și urmează să fie rezolvate în următoarele versiuni:

- Uneori butonul de back din ecranul Orare nu funcționează.
- La deschiderea ecranului “Hartă”, centrarea nu se realizează corect în unele instanțe.

### **Securitatea datelor:**

Busify evită pe cât posibil utilizarea datelor cu caracter personal. Locația dispozitivului este singurul de tip de date colectat pentru statistică, dar nu poate folosi că utilizatorii să fie identificati. Datorită lipsei datelor cu caracter personal este posibil și realizarea algoritmului de peer connectivity, iar dispozitivele pot distribui datele aplicației între ele fără grija distribuirii și acestor tip de date.

Datele sunt ținute în siguranță prin utilizarea unui server propriu, aflat în Varșovia, care a fost configurat astfel încât să colecteze un singur “OK” de la dispozitive în cazul încărcării corecte a datelor pentru a putea fi inchis sesiunea dintre cele două.