

## UNITATE DE CONTROL TELEMATICĂ PENTRU SISTEME INTELIGENTE DE CONDUCERE

Narcis-Mihai HARITON<sup>1</sup>, Gabriel BECIU<sup>2</sup>, Cosmin-Ionuț MIHAI<sup>3</sup>, Ioan-Dragoș DEACONU<sup>4</sup>, Aurel-Ionuț CHIRILĂ<sup>5</sup> și Valentin NĂVRĂPESCU<sup>6</sup>  
<sup>1, 2, 3, 4, 5, 6</sup>Universitatea POLITEHNICA din București – Facultatea de Inginerie Electrică  
mihai.narcis@yahoo.com<sup>1</sup>, gabriel.beciu@gmail.com<sup>2</sup>, mihai.cosmin955@gmail.com<sup>3</sup>,  
dragos.deaconu@gmail.com<sup>4</sup>, aurel.chirila@gmail.com<sup>5</sup>, valentin.navrapescu@gmail.com<sup>6</sup>

**Rezumat.** Sistemul de unitate telematică de control utilizează funcționalitățile sistemelor A.D.A.S, E.C.M și A.B.S deja existente pe autovehicule implicând componentele hardware, software și mecanice oferite de către acestea. Sistemul este un serviciu care îi oferă utilizatorului siguranța în trafic și informare prealabilă asupra impedimentelor pentru eficientizarea traseului. În prezenta lucrare este prezentat modul în care se propune ca acest sistem să proceseze datele și să le transmită către un server care ulterior va transmite către o hartă interactivă aceste informații.

### 1. INTRODUCERE

Unitatea telematică de control este un sistem utilizat în prezent pentru transmiterea informațiilor recepționate de la traductoarele autovehiculului în scopul semnalării unei urgențe sau pentru facilități oferite conducătorului auto.

Toate automobilele produse după anul 2015 conțin cel puțin un sistem care să confere posibilitatea conectării la o rețea celulară sau pentru folosirea datelor mobile, de exemplu OnStar de la General Motors. Aceste funcții fac parte din telematică, însă o astfel de unitate poate prelua date de la senzorii vehiculului, poate prelua date prin satelit cu privire la trafic sau la condițiile meteorologice.

Unitatea telematică de control este un ansamblu care ar putea furniza într-adevar o varietate atrăgătoare de servicii pentru vehicule, astfel ar putea să revoluționeze experiența de a conduce.

Telematica poate ajuta autoritățile să facă progrese majore în ceea ce privește realizarea unui transport inteligent, reducerea timpului de transport și cel mai important reducerea numărului de decese datorate accidentelor rutiere.

Unitatea telematică de control este amplasată cât mai aproape de bordul vehiculului pentru a putea realiza o conexiune bluetooth cât mai bună cu dispozitivele din interiorul autovehiculului, dar această amplasare a fost realizată și datorită faptului că unitățile de bord trebuie să ocupe cât mai puțin spațiu în interiorul autovehiculului pentru a putea oferi loc părții mecanice. Cu toate acestea, mulți producători de autovehicule preferă ca unitatea telematică de control să fie amplasată în spatele autovehiculului împreună cu unitatea specifică funcțiilor media, însă acest lucru elimină din atribuțiile telematice ale unității.

### 2. AȘTEPTĂRI FUNCȚIONALE ALE MODELULUI IMPLEMENTAT

Sistemul inteligent de detecție a impedimentelor este un serviciu care îi oferă utilizatorului siguranța în trafic și informare prealabilă asupra impedimentelor pentru eficientizarea traseului. Acest sistem comunică automat cu serverul pentru a primi și a transmite informații despre zona în care a fost detectat evenimentul.

Sistemul inteligent de conducere crează o hartă care va conține impedimentele întâlnite pe partea carosabilă. Pentru realizarea unei astfel de hărți trebuie ca vehiculele să poată comunica cu același server de sortare și interpretare a datelor primite, deoarece trebuie ca un eveniment să fie confirmat cât mai des posibil pentru a nu interveni înlăturarea prealabilă a acestuia fără ca conducătorul auto să fie anunțat la apropierea de acel segment de drum.

Activarea sistemului inteligent de detecție a impedimentelor poate fi automată la pornirea autovehiculului sau poate fi oprit/pornit de către utilizator în orice moment prin apăsarea unui buton din panoul de comandă al vehiculului. Comunicarea cu serverul se face cu ajutorul unui modul GSM prin care se trimite un mesaj text cu locația GPS și codul de indentificare al problemei întâlnite pe traseu. Serverul va primi informația și o va transmite către vehiculele aflate în perimetrul delimitat prin program în cazul recepționării a cel puțin 3 mesaje de confirmare a aceluiaș impediment.

Sistemului inteligent de detecție a impedimentelor utilizează sisteme existente pe autovehicule cum ar fi totalitatea sistemelor care formează A.D.A.S. (Sisteme avansate de asistență pentru conducători auto), sistemul A.B.S. (Sistem de antiblocare a roților) și T.C.U. (unitatea telematică de control). Pentru realizarea sistemului trebuie ca sistemele menționate anterior să suporte anumite modificări, iar în cazul sistemului A.D.A.S. și T.C.U. trebuie implementate noi funcționalități ale acestora.

### *2.1 Modele similare*

Mulți producători auto lucrează la implementarea unui astfel de sistem inteligent de conducere, însă singurii care au realizat o implementare pe autovehicul sunt cei de la BMW și de la Range Rover. Acești doi producători au realizat până în momentul actual o detecție a gropilor de pe partea carosabilă, însă fiecare dintre aceștia au abordat problematica detecției în moduri diferite, astfel BMW a preferat să realizeze o detecție video a acestora pe baza umbrelor create de gropi și Range Rover a preferat să detecteze groapa în momentul în care suspensia suferă o mișcare suficient de mare încât să poată fi considerată groapă.

Modelele similare ale producătorilor auto sunt încă într-un stadiu de dezvoltare, deoarece datele care trebuie transmise către un server pentru a le gestiona sunt mult prea mari în prezent și astfel necesitatea implementării unei rețele 5G crește pentru a putea transmite toate aceste date fără a avea probleme de supraîncărcare a datelor la intrarea în server.

### *2.2 Arhitectură destinată sistemelor avansate de conducere asistată (ADAS)*

Scopul principal al sistemelor avansate de conducere asistată este acela de a asista șoferul în ceea ce privește aspectele legate de siguranță în timpul șofatului, să sporească confortul de conducere al conducătorului autovehiculului și să îmbunătățească echilibrul economic și de mediu.

Aceste sisteme dezvoltate pentru asistență în timpul procesului de condus, au nevoie de o putere de calcul „high-end” a circuitelor integrate („SoC - sistem pe un cip / SOC - sistem pe cip”) [1] pentru a putea analiza aria de deplasare a automobilului și pentru prelucrarea computerizată cognitivă cu ajutorul inteligenței artificiale. În plus pe lângă necesitatea unei puteri mari de calcul mai sunt necesare și implementarea unor funcții de control pentru asigurarea unui consum redus de energie [2].

În figura 1 este prezentată o arhitectură simplă a sistemelor avansate de conducere asistată. După cum se poate observa din această figură această arhitectură este împărțită în trei blocuri.

Primul bloc este denumit ca fiind blocul de detectare și analiză, acest bloc va cuprinde toate echipamentele de percepție și detecție (senzori, camere video, radar, etc.), monitorizare și conectivitate. Este denumit în acest fel deoarece în această zonă se preiau informații despre mediul înconjurător al autovehiculului și sunt analizate. În urma acestei analize datele procesate vor fi transferate către următorul bloc arhitectural denumit blocul procesul de decizie.

În cadrul acestui bloc arhitectural sistemul avansat de conducere asistată va prelua datele informative de la blocul arhitectural menționat anterior le va analiza și va decide dacă va prelua controlul autovehiculului sau dacă doar va asista conducătorul autovehiculului în decizia luată. Deciziile sistemului avansat de conducere asistată, vor fi bine definite de către producătorii și dezvoltatori de sisteme avansate, în momentul programării funcționalităților acestora, însă decizia finală va rămâne conducătorului autovehiculului.

Ultimul bloc este denumit ca fiind blocul control deoarece în această zonă va fi asigurată siguranța autovehiculului, a șoferului cât și a tuturor participanților în trafic. În acest perimetru sunt dezvoltate sistemele de siguranța și control al autovehiculului.

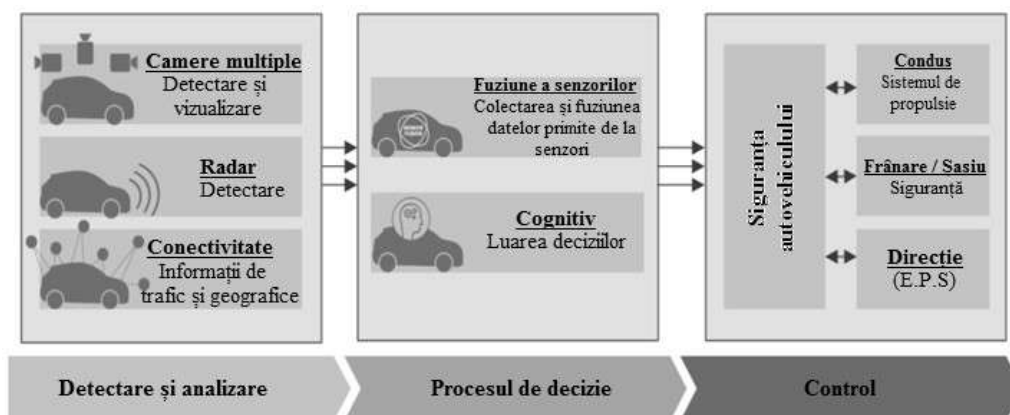


Figura 1. Arhitectura simplă a unui sistem avansat de conducere asistată [2]

### 2.3 Arhitectura sistemului conceput

În scopul realizării sistemului inteligent de conducere au fost create diagrame bloc ale elementelor componente care comunică pentru realizarea detecției și pentru transmiterea informațiilor.

A.D.A.S. preia informațiile de la senzorii existenți pe autovehicul și de la camera video, realizează procesarea datelor primite, verifică dacă în procesarea de verificare a detectat forme sau semnale care să indice existența unui eveniment. În cazul în care se găsesc astfel de informații se va face o procesare complexă pentru determinarea evenimentului. După ce este realizată această etapă, după caz dacă unitatea A.D.A.S. consideră că trebuie transmis un semnal către A.B.S. atunci se va transmite o comandă de gestionare a funcționalităților acestora în scopul prevenirii unor accidente sau deteriorări ulterioare ale vehiculului.

După detectarea evenimentului, A.D.A.S. va trimite informațiile către unitatea telematică de control care va alocă un cod evenimentului primit, va prelua locația GPS de la modulul intern și va transmite informațiile către navigație pentru a atenționa conducătorul auto în cazul unui impediment.

Unitatea telematică va verifica dacă există semnal sau dacă există o cartela SIM(Subscriber Identification Module) în interiorul sistemului, iar apoi dacă aceste condiții sunt îndeplinite se vor transmite informațiile către server.

Server-ul va transmite confirmarea primirii datelor și închiderea conexiunii cu vehiculul pentru ca mai apoi să determine dacă s-au primit 3 notificări despre acest impediment semnalat, iar dacă această condiție a fost îndeplinită se va transmite actualizarea de hartă către toate vehiculele apropiate de impedimentul semnalat.

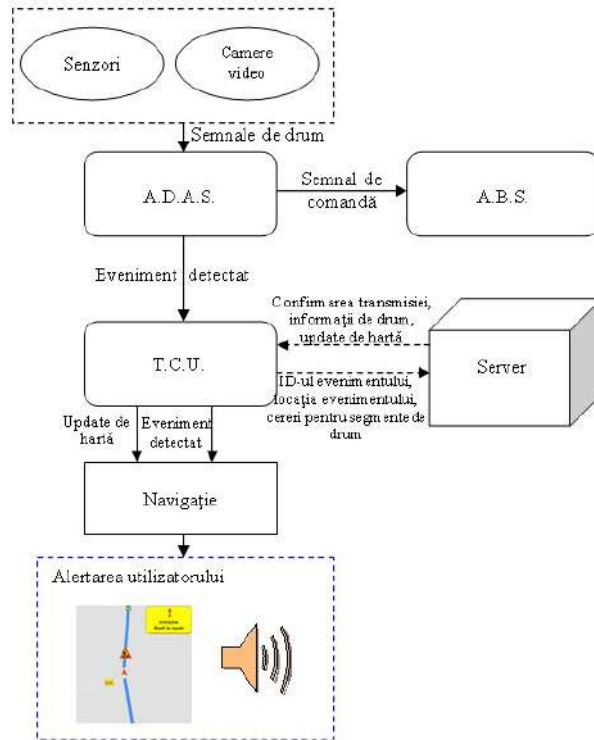


Figura 2. Arhitectura sistemului inteligent de conducere.

În figura următoare este prezentat modul de transmitere al informațiilor atât în interiorul vehiculului cât și datele care sunt transmise exterior către server.

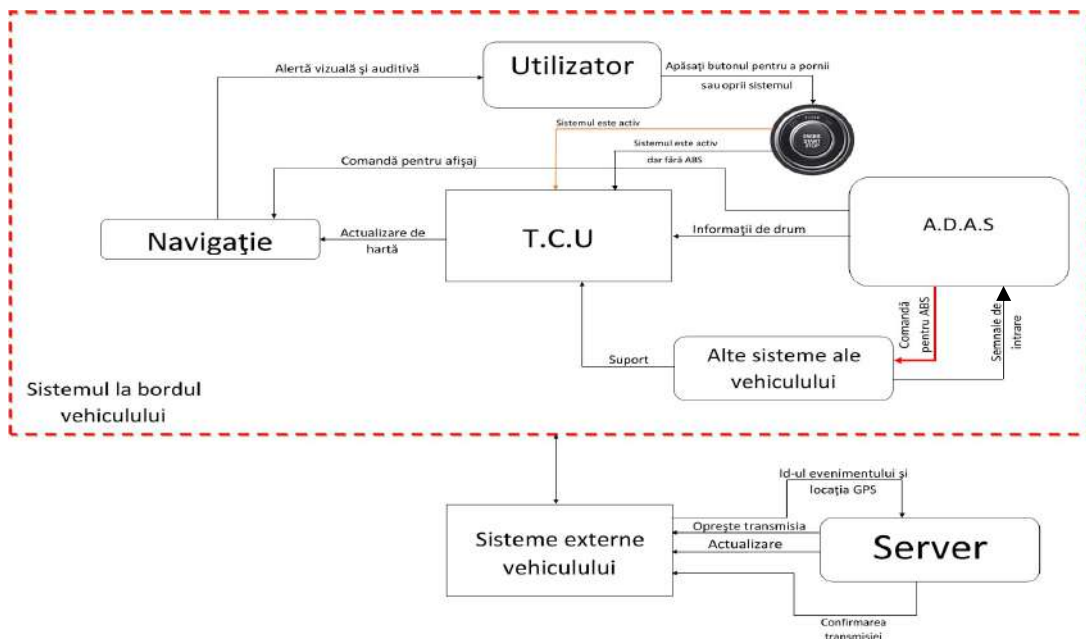


Figura 3. Modul de transmitere a datelor.

Utilizatorul sistemului are rolul de a porni vehiculul și de a selecta modul de funcționare al sistemului.

În urma alegeri făcute de către utilizator se va verifica modul de funcționare al sistemului și se vor face testele interne de verificare a funcționalităților sistemului. Sistemul în cazul în care este pornit avem prezente două posibilități de acționare ale acestuia: cu sistem de frânare activat și fara sistem de frânare.

Sistemul în cazul detectării unui impediment o să determine ID-ul acestuia și îl transmite către unitatea telematică de control (T.C.U.), dacă este cazul transmite comandă de acționare a sistemului de frânare către sistemele inteligente de pe vehicul destinate frânării automate. Sistemul transmite o comandă și către sistemul de navigație pentru a putea atenționa conducătorul auto în cazul detectării unui eveniment. Sistemul pentru a realiza toate aceste acțiuni trebuie să preia de la sistemele deja existente pe vehicul informațiile de drum necesare realizării detecției impedimentelor.

Unitatea telematică de control preia locația de la modulul GPS existent în interiorul acesteia și transmite ID-ul evenimentului și locația impedimentului către server.

Serverul transmite către vehicul, mai exact către T.C.U. confirmarea transmisiei, închiderea transmisiei și actualizarea de hartă dacă este necesar. Unitatea telematică de control, în momentul în care va recepționa actualizarea de hartă va transmite către navigație aceste informații.

### 2.3. Cazuri uzuale de utilizare.

În figura următoare este prezentat modul în care unitatea telematică trebuie să transmită datele preluate de la sistemul A.D.A.S și modul în care utilizatorul poate opri anumite funcții ale sistemului sau sistemul în sine.

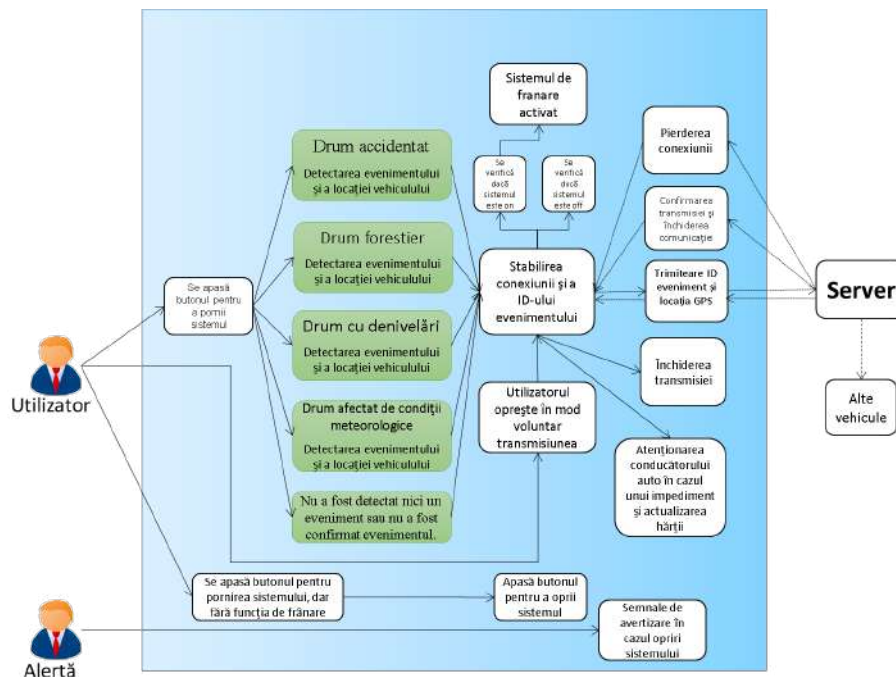


Figura 4. Modul de tratare al cazurilor uzuale de utilizare.

Diagrama cazurilor uzuale de utilizare exemplifică modul de decizie al sistemului în cazul în care a fost activat, dar și cum reacționează acesta în momentul în care este schimbată starea sistemului.

Utilizatorul pornește vehiculul cu ajutorul butonului specific pornind astfel și sistemului inteligent de detecție a impedimentelor. În momentul în care sistemul este pornit informațiile vor fi preluate de către sistemului inteligent de detecție a impedimentelor de la camera video și de la senzori, va analiza informațiile primite și în cazul în care se va detecta un impediment se vor urma pașii corespunzători cazului uzual respectiv. Se stabilește tipul evenimentului detectat și comunicarea cu unitatea telematică de control, se verifică dacă funcția de frânare este activă și se transmite comanda de frânare dacă este necesar. Se transmite id-ul evenimentului împreună cu locația către server, iar serverul dacă va primi mai multe notificări legate de acest eveniment va transmite mai departe către toate vehiculele o actualizare de hartă. Serverul transmite confirmarea comunicației, închiderea transmisiei și actualizarea de hartă către sistem.

Dacă utilizatorul oprește voluntar sistemul, atunci se vor închide toate conexiunile cu serverul până la repornirea sistemului.

Utilizatorul poate opri voluntar doar sistemul de frânare, rămânând însă activ sistemului inteligent de detecție a impedimentelor. Utilizatorul este alertat vizual și sonor de orice modificare adusă stării butonului pentru sistemului inteligent de detecție a impedimentelor, dar și în cazul fiecărui impediment detectat.

În cazul în care sistemul nu detectează un impediment semnalat de către server, atunci se va transmite această informație către server pentru a înlătura impedimentul la următoarea actualizare de hartă.

### 3. Realizarea modelului experimental.

În prima etapă am realizat o interfață capabilă să poată crea atât mesaje care vor fi transmise către server pentru a putea aplica impedimentele pe hartă și pentru a putea genera codul json specific pentru aceasta. Utilizatorul are posibilitatea de a menționa coordonatele impedimentului dorit, nivelul impedimentului și tipul acestuia cu ajutorul interfeței.

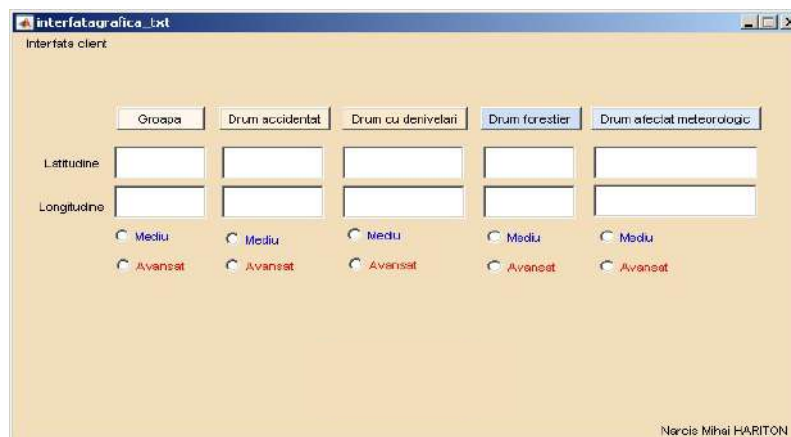


Figura 5. Interfața cu utilizatorul pentru transmiterea codului către server.

Interfața creată respectă funcționalitatea dorită și astfel prin apăsarea butonului respectiv impedimentului dorit este generat un fișier de tip text “impedimente.txt” unde salvează toate datele scrise pentru a putea contoriza și la client datele trimise de acesta. Odată cu apăsarea butonului se va transmite un mesaj de tip text către server și către propriul telefon mobil pentru a putea fi actualizată harta impedimentelor.

Prin implementarea acestei interfețe, harta cu utilizatorul nu va mai fi deschisă la client deoarece această interfață trebuie doar să genereze mesaje de tip text pentru a fi transmise cu ajutorul unui modul GSM conectat la unul dintre porturile usb ale unui laptop. Pentru a putea reprezenta harta și la utilizator ar trebui ca harta să fie postată online prin intermediul unei platforme de cloud.

### 3.1. Implementarea hărții impedimentelor.

Harta impedimentelor are rolul de a prelua codul Json generat de către server, de a analiza informațiile primite și de a genera punctele indicate pe hartă. Aceasta utilizează funcții specifice oferite de către google maps pentru generarea propriei hărți, dar și elemente ale limbajelor HTML, CSS și javascript.

În figura de mai jos este prezentată o hartă a impedimentelor puse cu ajutorul interfeței create și anume interfața grafică pentru generarea codului Json.

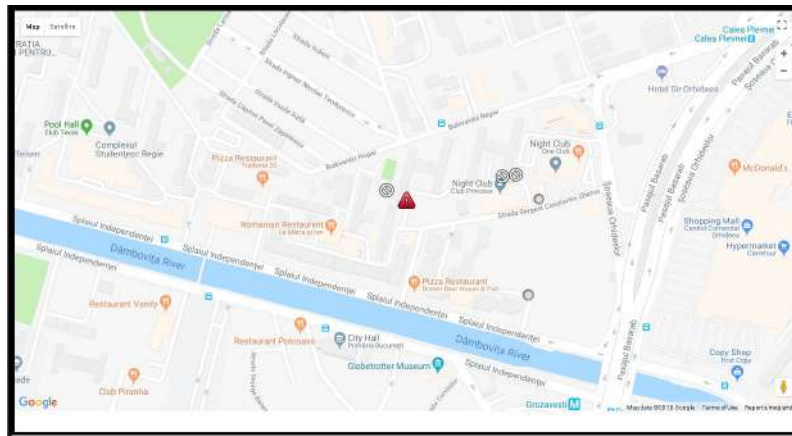








Figura 6. Harta impedimentelor.

Simbolurile pentru această hartă sunt următoarele:

- Pentru a simboliza o groapă avem: 
- Pentru a simboliza un drum accidentat avem: 
- Pentru a simboliza un drum forestier avem: 
- Pentru a simboliza un drum afectat de condițiile meteorologice avem: 
- Pentru a simboliza un drum închis avem: 
- Pentru a simboliza un drum cu denivelări avem:  [3]

Toate aceste simboluri au fost preluate de pe un site care este asociat comenzilor google pentru a putea realiza aplicarea acestora pe harta impedimentelor, în caz contrar era necesară crearea unei legături între funcția specifică și imaginile dorite care ar fi trebuit la rândul lor postate pe un server.

Programul pentru harta impedimentelor utilizează instrucțiuni de tip html pentru realizarea formatului pagini afișare în partea de antet a programului unde sunt definite cu ajutorul funcțiilor specifice din css dimensiunea hărții, dimensiunea pagini afișate și marginile acesteia. Tot în antet este definit și titlul pagini deschise în momentul rulării programului.

În corpul programului avem accesarea unor coduri specifice oferite open source de către google maps pentru generarea hărții.

Inițial au fost definite “iconBase” și “icons” ca două variabile care precizează locul și numele pozei pe care să o atribuie fiecărui impediment, astfel având câte o declarație specială a variabilei “iconBase” unde se regăsește link-ul serverului care ne oferă imaginile la care se adaugă numele imaginii pentru a ști unde exact se află acea poză.

Funcția de inițializare a programului crează un obiect de tip map care accesează google.maps.Map. Pentru a stabili poziția inițială a hărții este necesar să definim latitudinea și longitudinea la care dorim să ne deschidă harta, zoom-ul necesar la care să se deschidă acesta fiind stabilit prin încercări repetate și poziția elementelor de control cum ar fi zoom in și zoom out.

Funcția findImpediment are rolul de a accesa programul “impedimente.json” și pentru a putea interpreta datele regăsite în interiorul acestuia. Astfel în cadrul acestei funcții se vor atribui funcții pentru preluarea coordonatelor și anume latitudine și longitudine, pentru preluarea tipului impedimentului și pentru afișarea acestora pe hartă. Pentru fiecare punct sa definit un mesaj de tip pop-up care să fie afișat la apăsarea cu mouse-ul pe unul dintre acestea. Acest mesaj preia parametrii tip și nivel.

### 3.2. Implementarea serverului.

Implementarea serverului a fost realizată cu ajutorul mediului de dezvoltare Matlab pentru a deservi la recepționarea mesajelor de tip text cu rolul de a realiza actualizarea hărții modelului experimental al sistemului. Această interfață a fost realizată sub forma unei structuri datorită ușurinței de realizare a funcțiilor suplimentare deoarece cu modulul “figure” implementarea funcțiilor pentru modulul GSM este destul de anevoioasă.

Pentru implementarea interfeței este necesar un modul de tip gsm prin usb pentru a putea realiza atât recepționarea mesajelor cât și transmiterea acestora, însă pentru a putea funcționa recepționarea informațiilor primite de la modelele experimentale ale clienților este necesar să deconectăm modulul din a se configura cu programul său intern, trebuie urmărit ce com ni se alocă pentru funcția de gsm deoarece după caz putem avea 2 sau 3 com-uri alocate pentru un modul și nu în ultimul rând trebuie să realizăm întreruperi de conexiune ale modului în momentul în care dorim să comunicăm cu acesta.

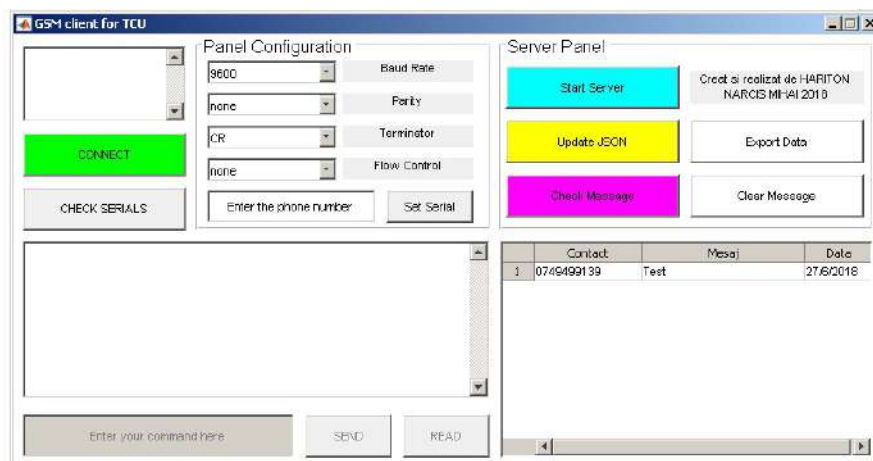




Figura 7. Implementarea modelului experimental al serverului.

Interfața realizată are rolul de a ne oferi posibilitatea de setare a parametrilor modulului pentru a putea fi compatibilă cu oricare dintre modulele existente, în caz contrar ar fi trebui să utilizăm un singur tip de modul care s-ar fi încadrat în parametrii prestabiliți în program. Astfel în partea de “Panel Configuration” regăsim toți parametrii necesari pentru a putea trimite și transmite un mesaj care trebuie să fie identici cu cei pe care îi regăsim în comul alocat pentru modul.

În vederea conectării cu modulul GSM trebuie să vedem dacă avem un astfel de modul conectat la server și de aceea avem create butonul “Check serials”. După apăsarea acestui buton vom avea afișate în partea din stânga sus a interfeței porturile disponibile, iar în momentul selectării portului cu care dorim să comunicăm vom putea da conectare pentru a putea primi și transmite informații către modulul GSM.

În partea din dreapta a interfeței grafice regăsim toate funcțiile pentru server unde avem posibilitatea de a porni și de a opri serverul, de a face actualizările fișierului de tip Json pentru a actualiza modelul experimental a hărții impedimentelor, avem posibilitatea citirii mesajelor recepționate, ștergerea acestora și exportarea datelor salvate. În caseta de dialog vor fi afișate toate mesajele primite de fiecare dată când apăsăm butonul “Check Message”.

În figura de mai jos este prezentat modul în care interfața grafică a modelului experimental pentru server reacționează în momentul conectării sau în momentul în care transmitem către modulul GSM comenzi de tip AT.

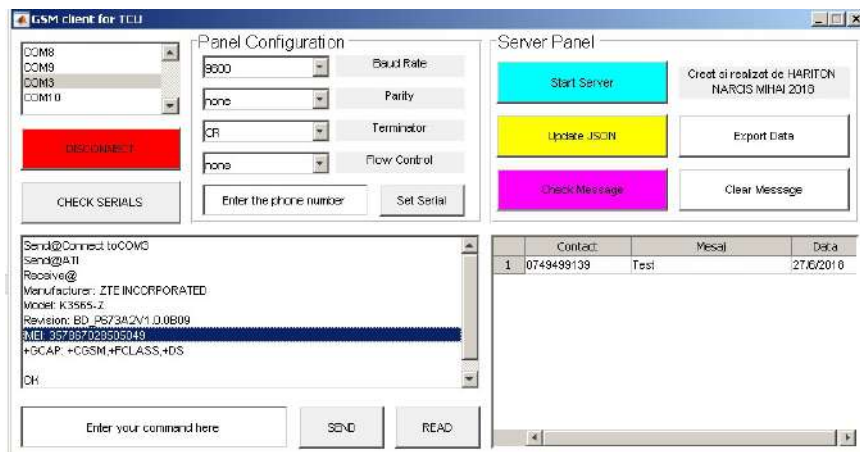


Figura 8. Testarea interfeței grafice a modelului experimental.

În figura 7. regăsim modul de afișare în momentul conectării cu modulul de tip GSM prin usb, comanda care ne oferă răspunsul asupra realizării conexiunii cu modulul sau nu, iar mai apoi a fost transmisă o comandă de verificare a faptului că modulul GSM este capabil în acest moment să comunice cu interfața. Prin intermediul comenzii “ATI” introdusă în locul în care avem afișat “Enter your command here” putem vedea toate datele modulului cum ar fi: versiunea de software, modelul produsului, firma care a creat acest modul și imei-ul acestuia.

#### 4. CONCLUZII

Soluția găsită pentru modulul GSM de a putea recepționa mesajele este Matlab-ul, deoarece acesta utilizează procesorul laptopului și astfel comenzile sunt executate mult mai rapid, însă pentru o bună conexiune la rețea trebuie utilizat un modul GSM prin usb care să se poată conecta singur la rețea. Făcând aceste lucruri am putut realiza implementarea serverului și crearea fișierului de tip json pentru harta impedimentelor care mai apoi să fie actualizată la fiecare reîmprospătare a pagini.

Conform studiului efectuat este de preferat ca unitatea telematică de control să conțină un modul pentru transmiterea terestră, o conexiune minimă la internet pentru 3G și un circuit de reglaj și gestionare a tensiunilor la intrare în cazul în care vor apărea tehnologi ulterioare să nu existe problema alimentării pentru modulele respective, acest lucru ajutând ca unitatea telematică să se poată dezvolta mult mai ușor în viitor.

## **BIBLIOGRAFIE**

- [1] Wikipedia. [10 11, 2020.] System on a chip. Retrieved from Wikipedia:  
[https://en.wikipedia.org/wiki/System\\_on\\_a\\_chip](https://en.wikipedia.org/wiki/System_on_a_chip)
- [2] Renesas. [10 11, 2020.] Advanced Driver Assistance System (ADAS). Retrieved from Renesas:  
<https://www.renesas.com/en-us/solutions/automotive/adas.html>
- [3] <http://kml4earth.appspot.com/icons.html>. Maps Icons. Google Earth. [Online] Google Inc. KML4Earth. [Cited: 10 11, 2020.] <http://kml4earth.appspot.com/icons.html>.