Sistem de irigare

PROIECT RC

Autor: **Bandics Daniel, Demeter Reka, David Dana Dorina**

Conducător științific: **Dr. ing. Ioan Valentin Sita**

Cuprins

[1 Introducere 2](#_Toc477457095)

[1.1 Context general 2](#_Toc477457096)

[1.2 Obiective 2](#_Toc477457097)

[1.3 Specificații 2](#_Toc477457098)

[2 Studiu bibliografic 3](#_Toc477457099)

[3 Analiză, proiectare, implementare 4](#_Toc477457100)

[4 Concluzii 5](#_Toc477457101)

[4.1 Rezultate obținute 5](#_Toc477457102)

[4.2 Direcții de dezvoltare 5](#_Toc477457103)

[5 Reguli de formatare 6](#_Toc477457104)

[5.1 Formatarea paginii 6](#_Toc477457105)

[5.2 Titluri și stiluri 6](#_Toc477457106)

[5.3 Figuri, tabele și ecuații 7](#_Toc477457107)

[5.3.1 Figuri 7](#_Toc477457108)

[5.4 Tabele 7](#_Toc477457109)

[5.5 Ecuații 7](#_Toc477457110)

[5.6 Referințe bibliografice 8](#_Toc477457111)

[6 Bibliografie 9](#_Toc477457112)

# Introducere

## Context general

Îngrijirea planetlor de apartament devine tot mai dificilă în programul nostru zilnic aglomerat. Sistemele de udare automate simplifică sarcina de a asigura hidratarea plantelor. Cunoașterea momentului și a cantității potrivite de apă sunt elemente esențiale ale procesului de udare. Pentru a le oferi grădinarilor o experiență mai ușoară, am creat acest prototip.

## Obiective

Oamenii apreciază plantele pentru beneficiile lor și pentru plăcerea de a le îngriji. Totuși, mulți întâmpină dificultăți în a le menține sănătoase și viguroase. Pentru a răspunde acestei provocări, a fost creat un prototip care permite plantei să devină mai automată, udându-se singură dintr-un rezervor de apă și notificând utilizatorul când rezervorul trebuie reumplut. Automatizarea acestui sistem este gândită să ofere un sprijin real utilizatorului.

## Specificații

Acest proiect utilizează microcontrolerele Arduino Uno și NodeMCU ESP8266, programate pentru a detecta nivelul de umiditate și a furniza apă în funcție de necesitățile plantei. Sistemul integrează aplicația Blynk pentru a adapta udarea plantei în mod cord

# Studiu bibliografic

**2.1 Ce este IOT?**

**2.1.1 Definiția și importanța IoT**

Internet of Things este definit ca o rețea de dispozitive interconectate, echipate cu senzori, care colectează și transmite date, contribuind la o gamă largă de aplicații inteligente, cum ar fi case inteligente, orașe inteligente, monitorizare medicală și industrială. [1] Internet of Things a marcat începutul unei noi revoluții industriale, denumită „Industria 4.0”. Alături de Big Data și cloud computing, IoT este considerată una dintre descoperirile esențiale ale erei moderne în tehnologie. [2]

**2.1.2 Evoluția de la prima revoluție industrială la IoT**

Industria tehnologică a trecut prin patru revoluții majore: mecanizarea prin abur începând din 1760, electrificarea și producția de masă la finalul secolului XIX, automatizarea prin informatică în anii 1970 și, recent, digitalizarea avansată prin „Industria 4.0”. În această ultimă etapă, IoT joacă un rol central, conectând dispozitivele inteligente și facilitând schimbul rapid de date pentru o automatizare mai complexă.

**2.1.3 Cum poate IoT să îmbunătățească activitățile zilnice**

IoT sprijină utilizatorii în a lucra și trăi „mai inteligent”, simplificând activitățile zilnice și contribuind la bunăstare. Un exemplu important este automatizarea locuinței, unde IoT permite controlul de la distanță al electrocasnicelor prin sisteme conectate la internet, oferind rezidenților mai mult confort și eficiență.

Într-o lume din ce în ce mai rapidă, oamenii tind să își dorească automatizarea tuturor proceselor. Nevoie de control la distanță a devenit o cerință în viața modernă. [3]

De asemenea, IoT are un impact semnificativ și în agricultură. Tehnologii precum senzori wireless integrați în „agricultura inteligentă” pot transforma acest domeniu, oferindu-le fermierilor instrumente de decizie și automatizare pentru a spori eficiența. De exemplu, sistemele de grădină inteligentă permit îngrijirea optimă a plantelor, chiar și în absența proprietarului. Aceasta este o soluție utilă mai ales în contextul schimbărilor climatice, când o grădină bine îngrijită poate contribui la confortul casei. În absența unei astfel de tehnologii, lipsa de timp poate duce la ofilirea plantelor, impiedicând astfel beneficiile acestora. [4]

**2.1.4 Arhitecturile IoT**

Există două arhitecturi IoT principale descrise: arhitectura cu trei straturi și cea cu cinci straturi. Arhitectura în trei straturi include stratul de percepție, stratul de rețea și stratul de aplicație, fiecare având un rol specific în procesul de transmitere și analiză de datelor. Arhitectura cu cinci straturi adaugă straturile de transport și de afaceri, asigurând o prelucrare mai avansată a datelor și integrarea acestora în strategiile organizaționale. [5]

**2.1.5 Senzorii și actuatoarele în IoT**

Există diverse tipuri de senzori utilizate în IoT, cum ar fi cei pentru lumină ambientală, GPS, accelerometru, și senzori biometrici. Senzorii sunt cruciali pentru colectarea datelor din mediu, iar actuatoarele permit dispozitivelor să răspundă acestor dare prin acțiuni fizice, cum ar fi deschiderea unei vale sau pornirea unui motor. [5]

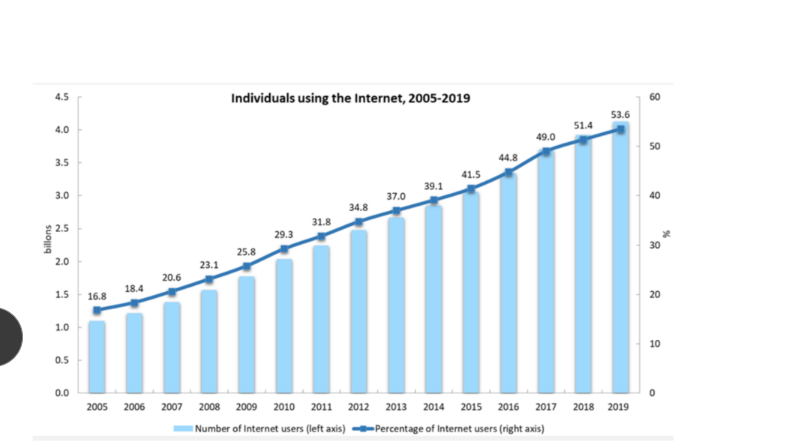
**2.1.6 Aplicații practice ale IoT**

IoT găsește aplicații într-o gamă largă de oameni, cum ar fi:

* Industria de producție: monitorizarea și automatizarea proceselor din fabrici pentru a optimiza performanța echipamentelor și a reduce timpii de nefuncționare
* Transport și mobilitate: dispozitivele IoT permit întreținerea predictivă a vehiculelor și îmbunătățesc siguranța prin monitorizarea continuă a condițiilor de drum și vehicul.
* Agricultură: senzorii monitorizează condițiile de sol și umiditate, ceea ce permite ajustări precise ale sistemelor de irigații pentru a îmbunătăți productivitatea

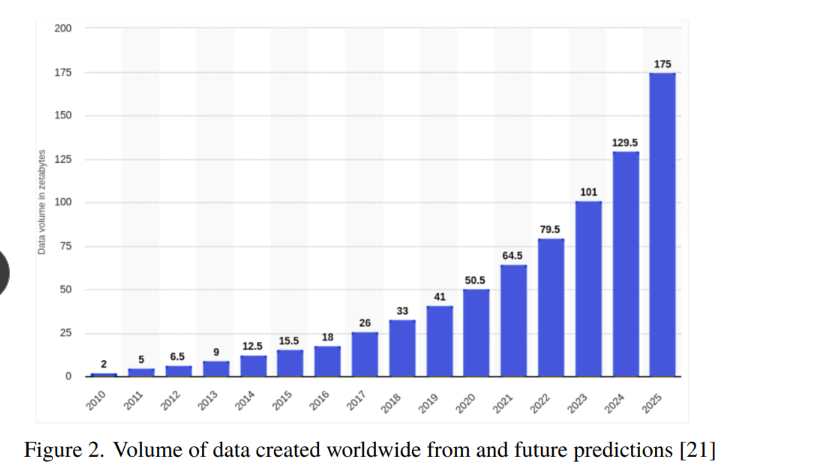
**2.1.7 Provocările actuale pentru IoT**

IoT se confruntă în prezent cu două limitări majore: numărul de dispozitive conectate și accesibilitatea datelor. Deși, după cum ilustrează Figura 1, numărul utilizatorilor conectați la inernet este în continuă creștere, ami există încă un drum lung până la o acoperire globală a internetuluo, având în vedere că peste jumătate din populația lumii nu are încă acces la internet.



*Figura 1*

Pe de altă parte, IoT se bazează considerabil pe date provenite din mediul înconjurător. În fiecare zi sunt generați peste 2,5 quintilione de octeți (1 QB= octeți) de date. Majoritatea acestor date provin de la dispozitive electronice interconectate, cum ar fi laptopuri, telefoane mobile, televizoare inteligente și sisteme inteligente de detecție a vehiculelor, echipate cu numeroși senzori. [3]

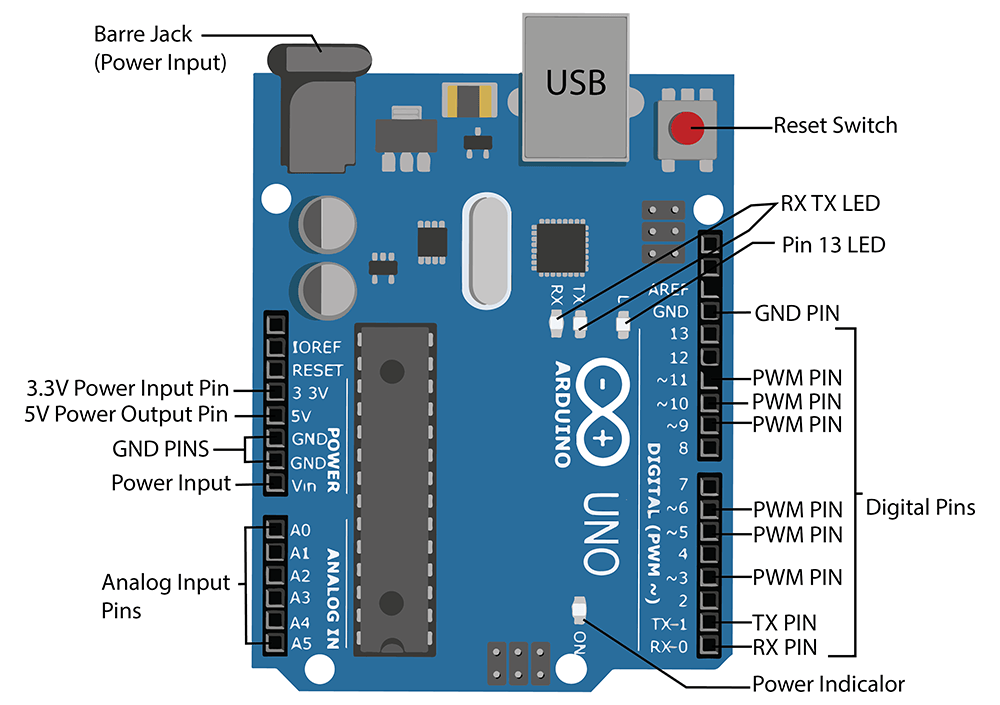


*Figura 2*

# Analiză, proiectare, implementare

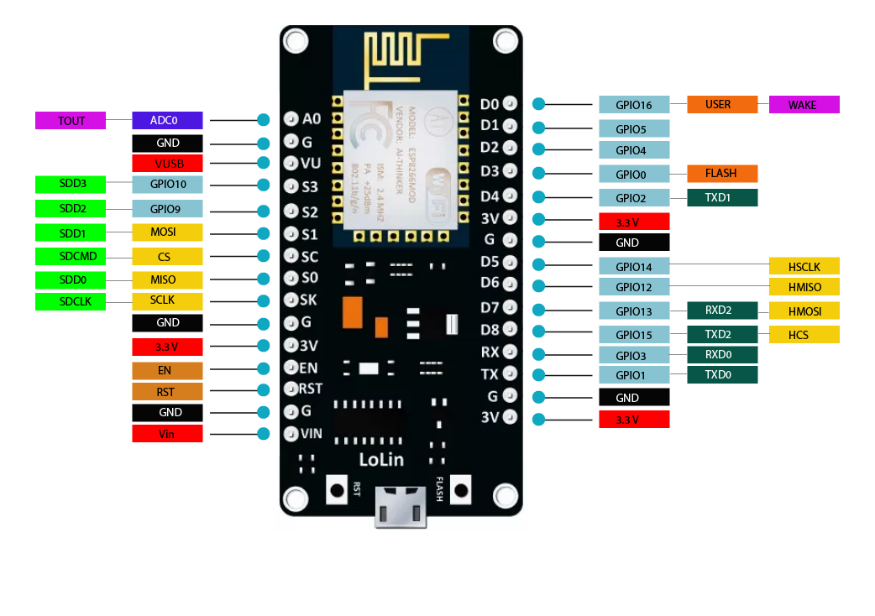
**3.1 Studiul componentelor**

**3.1.1 Arduino UNO placa de dezvoltare**

**

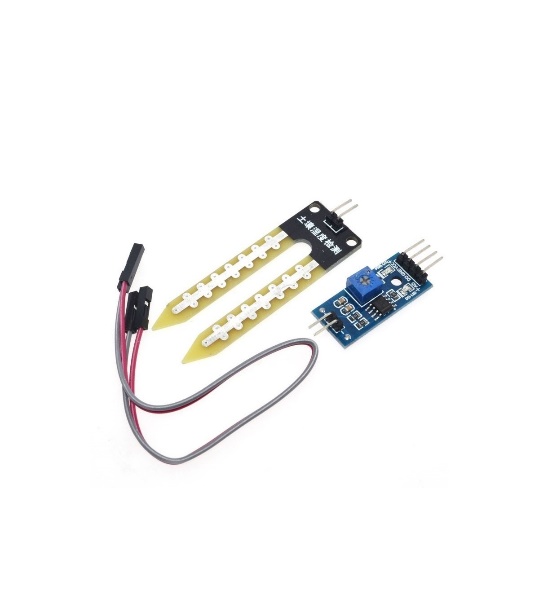
Arduino UNO este o placă de dezvolare bazată pe microcontrollerul Atmega328. Are 14 pini de intrare/ieșire (dintre care 6 pot fi folosiți ca ieșiri PWM), 6 intrări analog, un oscilator de 16MHz, o conexiune USB, mufă de alimentare, și un buton de reset.

**3.1.2 NodeMCU ESP8266 placa de dezvoltare**



NodeMCU este o firmware open-source bazat pe Lua și o placă de dezvoltare special concepută pentru aplicații IoT. Acesta include un firmware care rulează pe SoC-ul Wi-Fi ESP8266 de la Espressif Systems și hardware bazat pe modulul ESP-12E. Acest microprocesor suportă RTOS și operează între frecvențele 80Hz și 160Hz, frecvența fiind una ajustabilă. Are 128 KB RAM și o memorie flash de 4MB pentru stocarea datelor de program.

**3.1.3 Senzor de umiditate**



Poate fi folosit pentru a testa umidiatea solului. Acționează ca un rezervor variabil, cu cât mai multă apă este în sol cu atât conductivitatea dintre plăcuțe va fi mai mare și rezistența mai mică. Oferă posibilitatea reglării sensibilității din potențiometrul. Tensiunea de lucru este între 3,3V - 5V.

**3.1.4 Pompa de apă**



Este o pompă de apă mini submersibilă, funcționează la o tensiune de 3V sau 5V, fiind idealp pemtru alimentarea de la baterii sau surse de alimentare USB. Debitul maxim este aproximativ 80-120 litri pe oră (în funcție de tensiunea aplicată).

**3.1.5 Ecran LCD Display 1602 + adaptor i2c**



Caracteristici tehnice:

* Tensiune de alimentare: 5V
* Curent: 2mA
* Tensiune alimentare backlight: 4.2V
* Curent lumină de fundal: 250mA

**3.1.6 Aplicația Blynk**



Este o platformă IoT care permite utilizatorilor să controleze și să monitorizeze dispozitive inteligente de la distanță, folosind o aplicație mobilă. Are o interfață prietenoasă care oferă o interfață grafică ușor folosit, unde poți adăuga butoane, slider-e, grafice și alte widget-uri pentru a controla și a vizualiza datele de la dispozitivele conectate. Are conectivitate la cloud pentru a conecta dispozitivele cu aplicația mobilă, are un control în timp real și primirea de feedback instant de la senzori, se poate crea automatizări și scenaroi pentru dispozitivele, bazate pe condiții cum ar fi timp, locație sau date de la senzori.

## 3.2 Analiza sistemului

Dispozitivul mai are un senzor de umiditate care trimite valori analogice către controller, între 0 și 1024. Aici este invers, dacă valoarea senzorului ajunge aproape de 1024 atunci acesta indică faptul să solul este uscat, iar daca ajunge aproape de 0 atunci indică faptul că solul este umed.

Electrovalva se închide la valoarea de 1 logic și în mod normal este deschis. Pentru a obține o stare normal închisă sistemul folosește un releu conectat la o baterie de 9V, care este comandat de un semnal 1 logic pentru a deschide valva și 0 logic pentru a închide valva.

Butoanele produc valori de 1 logic când ele sunt apăsate, iar LED-ul se aprinde la valoarea de 1 logic și se oprește la valoarea de 0 logic.

LCD-ul de pe dispozitiv folosește biblioteca <LiquidCrystal\_I2C.h>. LCD-ul poate afișa pe 2 rânduri câte 16 caractere prin funcțiile bibliotecii (lcd.clear(); lcd.print(); lcd.setCursor ()).

## 3.3 Implementare

### 3.1.1Hardware

Dispozitivul este alcătuit din 2 compartimente:

Primul compartiment este rezervorul:

Apa din rezervor se golește printr-o electrovalvă. Electrovalva este închis în cazul în care primește alimentare, altfel este deschis. Deci are o bornă de VCC și o bornă de GND.

Al doilea compartiment:

Aici se afla cele 2 butoane fizice.

Deasemenea pentru un control mai ușor a fost implementat și butoane virtuale, folosind aplicația Blynk. Butonele fizice și cele virtuale din Cloud sunt legate de NodeMCU, iar NodeMCU trimite un semnal HIGH către Arduino, atunci când un buton este apăsat.

Pentru afișarea interfeței avem un ecran LCD care are 4 borne. O bornă pentru VCC, o bornă pentru GND, o bornă pentru transmisia de date SDA, și o bornă pentru clock SCL. Ultimele două borne sunt legate la intrările analogice ale controllerului așa cum se vede pe imaginea următoare:

Pentru controlul electrovalvei am folosit un releu cu contact normal închis, care este legat de o baterie de 9V. Releul are o bornă de VCC, o bornă de GND și o bornă de comandă care este legat la ieșirea digitală a controllerului.

Mai avem un senzor de umiditate la sol cu modul LM393. Senzorul este legat de modul printr-un fir de VCC și print-un fir de GND. Iar modulul este legat de Arduino prin bornă de VCC, borna de GND, și prin borna Analog Output, care furnizează valori analogice către Arduino.

Componentele sunt legate de Arduino Uno în următorul mod:

Ca și intrări analogice se leagă la portul A0 senzorul de umiditate, iar la porturile A4 și A5 sunt legate SDA respectiv SCL de la ecranul LCD.

Ca și intrări digitale avem cele două butoane (MODEBtn, ON/OFFBtn), fizice și cele virtuale, care sunt procesate de microcontrollerul NodeMCU și sesizate în porturile 8 și 9 digitale si la porturile 11 si 12 digitale sunt conectate iesirile din placa nodeMCU.

Ca și ieșiri digitale avem Controlul releului care este introdus în portul 7 digital.

Semnalele VCC și GND sunt introduse într-un breadboard la care sunt legate alimentările componentelor.

Componentele sunt legate de NodeMCU Esp8266 în următorul mod:

Ca și ieșire avem cele două ieșire pentru OK si NEXT porocesate la porturile 2 și 3 digitale.

### 3.1.2Software

Mediul de implementare folosit la acest proiect este Arduino UNO.

Ca si functinalitate sistemul se foloseste de cele 2 butoane pentru a selecta modurile de utilizare. Systemul porneste pe modul manula cu pompa inchisa iar de acolo cu ajutorul butoanelor putem porni motorul sau sa schimbam pe modul automat.

Modul automat functioneaza in felul urmator: daca senzorul de umiditate scade sub o valoare prestabilita atunci acesta porneste iar daca senzorul in timp ce uda planta ajunge inapoi la unul dintre valorile prestabilite atunci acesta se opreste.

## 3.4 Testare

Testarea dispozitivului se face automat la fiecare ciclu prin software.

Testarea se realizează în funcțiile loop() și în moistureSenzor(). Verificarea valorii senzorului de umiditate se face în funcția moistureSenzor(), și dacă valoarea care se dorește returnată este 0 atunci se apelează funcția errorMessage(int errorCode) cu codul 1.

Funcția errorMessage(int errorCode) afișează diferite mesaje pe ecranul LCD în funcție de parametrul primit.

# Concluzii

## Rezultate obținute

Evidențiați toate rezultatele pe care le-ați obținut și trageți concluzii din ele. Puteți prezenta o analiză critică a ceea ce ați realizat comparativ cu alte lucrări/studii anterioare.

Includeți o listă a contribuțiilor pe care le-ați avut în domeniul temei abordate.

## Direcții de dezvoltare

Descrieți direcțiile posibile de dezvoltare.

# Reguli de formatare

## Formatarea paginii

* + Dimensiunea paginii: A4
  + Margini: 2.5 cm (sus, jos, stânga, dreapta)
  + Antet și subsol: 1.27 cm de la marginea paginii
  + În antetul paginii (header): titlul capitolului, centrat, stil: Header\_style
  + În subsolul paginii: numărul paginii, centrat

## Titluri și stiluri

Titlurile capitolelor și subcapitolelor se marchează cu stilurile Heading 1 – 4, conform documentului model anexat în format Word. Descrierea stilurilor utilizate în document este prezentată în Tabelul 5.1.

Tabelul 5.1. Stiluri utilizate în acest document

| Nr. | Stil | Utilizat pentru | Format |
| --- | --- | --- | --- |
| 1 | Normal | Text normal | Font: (Default) Cambria, 12 pt, Justified, Line spacing: Multiple 1.1 li, Space After: 6 pt |
| 2 | Titlu | Titlul proiectului, prima pagină | Font: 24 pt, Small caps, Centered Line spacing: single, Space Before: 126pt, After: 0 pt, |
| 3 | Titlu2 | Titlul proiectului, pagina de prezentare | Font:14pt, Bold, Centered |
| 4 | Heading 1 | Titlurile capitolelor (nivel 1) | Font: 24 pt, Indent: Left: 0 cm Hanging: 0.76 cm, Space Before: 24pt, After: 12pt |
| 5 | Heading 2 | Titlurile subcapitolelor (nivel 2) | Font: 14 pt, Bold, Indent: Left: 0 cm  Hanging: 1.02 cm, Space Before: 18pt, After: 12pt |
| 6 | Heading 3 | Titlurile secțiunilor (nivel 3) | Font: Bold, Indent: Left: 0 cm Hanging: 1.27 cm, Space Before: 6 pt, After: 6pt |
| 7 | Heading 4 | Titlurile secțiunilor (nivel 4) | Font: Italic, Indent: Left: 0 cm Hanging: 1.52 cm, Space Before: 2 pt, After: 0 pt |
| 8 | Caption | Legenda figurilor și tabelelor | Font: Italic, Font color: Text 1, Line spacing: single, Space After: 10 pt, |
| 9 | Header\_style | Antetul paginii | Font: 10 pt, Italic, Centered, Border: Bottom: (Single solid line, Background 1, 0.5 pt Line width) |

## Figuri, tabele și ecuații

### Figuri

Figurile se inserează în text centrate, cu etichetă de numerotare și legendă (Caption) în partea de jos a figurii. Numărul figurii include și numărul capitolului, după exemplul prezentat în Figura 5.1.



Figura 5.1. Figură exemplu, stil: Caption

## Tabele

Tabelele se inserează în text centrate, cu etichetă și legendă (Caption) în partea de sus a tabelului, aliniată la stânga. Numărul tabelului include și numărul capitolului, după cum este prezentat, de exemplu, în Tabelul 5.1.

## Ecuații

Ecuațiile se inserează în text centrate, cu numerotare în partea dreaptă. Numărul ecuației include și numărul capitolului, conform exemplului din relația (5.1).

|  |  |
| --- | --- |
|  | (5.1) |

## Referințe bibliografice

Se recomandă ca citarea referințelor bibliografice să fie făcută în formatul IEEE.

În secțiunea Bibliografie sunt prezentate exemple pentru: o citare a unui capitol dintr-o carte [1], un articol publicat într-o revistă [2] și un articol publicat la o conferință [3].

Detalii cu privire la formatul citării diverselor tipuri de referințe pot fi găsite în [4] sau [5].

Referințele bibliografice se pot insera în text utilizând facilitățile Word de a adăuga surse și bibliografie unui document (References -> Citations & Bibliography). Dacă formatul IEEE pentru bibliografie nu este instalat implicit în Word, se poate descărca gratuit de la:

<https://bibword.codeplex.com/wikipage?title=Styles&referringTitle=Home>

Instrucțiunile de instalare pentru diferite versiuni de Word se pot obține de la aceeași adresă.

# Bibliografie

|  |  |  |
| --- | --- | --- |
| [1] |  | K. Mulchandani, "PLAWATOS-An Automatic Plant Watering IoT Based System Using ESP8266". |
| [2] |  | S. Suyoti, "Automatic Plants Watering System for Small Garden". |
| [3] |  | A. Gazis, "What is IoT? The Internet of Things explained," *Academia Letters, Article 1003,* 2021. |
| [4] |  | A. G. a. T. Gazi, "Big Data Applications in industry fields," *OUP ITNOW,* Vols. 63, no 2, pp. 50-51, 2021. |
| [5] |  | R. A. R. A. Mouha, "Internet of things (IoT)," *Journal of Data Analysis and Information Processing ,* vol. 9.02, p. 77, 2021. |