

# Raport științific și tehnic 2023

**MUSHNOMICS - UNLOCKING DATA-DRIVEN INNOVATION FOR IMPROVING  
PRODUCTIVITY AND DATA SHARING IN MUSHROOM VALUE CHAIN**

Referință 6436/15.11.2023

Manager de proiect: ***Rudolf Erdei***

## Istoricul versiunilor

Versiune	Autor	Modificări
0.1	Rudolf Erdei	Versiunea inițială
0.5	Rudolf Erdei	Versiunea intermediară
1.0	Rudolf Erdei	Versiunea finală

## Cuprins

<b>1</b>	<b>Introducere</b>	<b>5</b>
<b>2</b>	<b>Obiective planificate în perioada raportată</b>	<b>5</b>
2.1	Activități în perioada raportată . . . . .	6
2.1.1	Activități planificate . . . . .	6
2.1.2	Activități efectuate . . . . .	6
2.1.3	Devieri de la planificare . . . . .	6
<b>3</b>	<b>Metodologii</b>	<b>7</b>
3.1	Asigurarea calității . . . . .	7
3.2	Metodologie de testare . . . . .	9
3.2.1	Scop și obiective . . . . .	9
3.2.2	Domeniu de aplicare . . . . .	9
3.2.3	Responsabilitate . . . . .	9
3.2.4	Monitorizarea procesului . . . . .	12
3.3	Integrarea Continuă . . . . .	13
<b>4</b>	<b>Realizări tehnice</b>	<b>14</b>
4.1	Interfața cu utilizatorul . . . . .	14
4.2	Canalul de comunicație . . . . .	21
4.2.1	Criptarea cu chei publice SSL . . . . .	21
4.2.2	Formatul Mesajelor transferate în cadrul platformei . . . . .	22
4.2.3	Componentele principale ale Canalului de Comunicație și rolul acestora . . . . .	23
4.3	Aspecte legate de securitate . . . . .	23
4.4	Baza de cunoștințe . . . . .	24
<b>5</b>	<b>Artefacte</b>	<b>26</b>
5.1	UI - Interfața cu utilizatorul . . . . .	26
5.2	Knowledge base - Baza de cunoștințe . . . . .	26
5.3	Algoritm dinamic de generare a compoziției substratului . . . . .	26
5.4	Comparator de setup și rețete . . . . .	27
<b>6</b>	<b>Diseminare și exploatare</b>	<b>28</b>
6.1	Activități de diseminare . . . . .	28
6.1.1	Alte activități de diseminare . . . . .	28
6.2	Exploatare . . . . .	29
<b>7</b>	<b>Impact</b>	<b>29</b>
7.1	Indicatori de rezultat . . . . .	29
7.2	Impactul social . . . . .	29

7.3	Impactul asupra mediului . . . . .	30
7.4	Impactul economic . . . . .	30
<b>8</b>	<b>Concluzii</b>	<b>31</b>
8.1	Activități viitoare . . . . .	31

## Parteneri



Figura 1: Logo-ul proiectului MUSHNOMICS



(a) Holisun SRL (Holisun, Romania) Coordinator



(b) Pilze-Nagy Ltd (PILZE, Hungary)



(c) Department of Plant and Environmental Sciences, University of Copenhagen (UCPH, Denmark)



(d) University College Dublin (UCD, Ireland)

Figura 2: Partenerii proiectului *MUSHNOMICS*

## Sumar

Acest raport prezintă activitatea Holisun în anul al treilea din cadrul proiectului *MUSHNOMICS*. Activitatea este împărțită pe diferite capitole, discutându-se metodologiile de lucru utilizate precum și inovațiile ce au fost introduse în componentele dezvoltate. Se prezintă rolul Holisun în cadrul proiectului precum și rezultatele obținute. Structura și conținutul capitolelor este precum urmează:

Capitolul 1 conține informații despre proiect și despre conținutul prezentului document. Capitolul 2 vorbește despre obiectivele ce au stat la baza activităților din perioada de referință. În Capitolul 2.1.1 se arată pe scurt care sunt activitățile planificate pentru perioada de raportare, iar Capitolul 2.1.2 conține activitățile efectuate în perioada de referință.

În Capitolul 3 vorbim pe scurt despre metodologiile ce stau la baza proceselor de design, atât din punct de vedere grafic și experiența utilizatorului (UX), precum și a dezvoltării componentei software ce va controla această interfață cu utilizatorul. Capitolul 3.2 prezintă metodologia de testare a componentei soft rezultate, pentru asigurarea calității acesteia, precum și asigurarea validității din punctul de vedere al utilizatorului. Capitolul 3.3 prezintă metodologia de lucru ce are ca scop actualizarea și punerea în producție a noutăților din implementare, prin implementarea integrării continue.

Capitolul 4 prezintă realizările tehnice, implementarea ce a avut loc în cadrul proiectului. Capitolul 5 prezintă artefactele ce au fost rezultate în cadrul proiectului. În Capitolul 6 se prezintă activitățile de diseminare și exploatare în cadrul cărora rezultatele din cadrul proiectului au fost prezentate. În final, în Capitolul 8 se discută concluziile documentului privind desfășurarea proiectului.

## 1 Introducere

*MUSHNOMICS* își propune să dezvolte o platformă integrată pentru eficientizarea producției de ciuperci, precum și eficientizarea întregului lanț de producție și aprovizionare, eficientă din punct de vedere al costurilor, capabilă să monitorizeze sănătatea culturilor. În general, proiectul aspiră să culmineze cu producerea unei platforme radicale care va fi o paradigmă schimbătoare a modului în care inovația tehnologică bazată pe inteligența artificială poate deveni un instrument accesibil, accesibil tuturor și ușor de utilizat, aplicabil tuturor formelor de agricultură în medii controlate.

Prezentul raport oferă o imagine de ansamblu asupra cadrului operațional și a designului aplicației *MUSHNOMICS*, responsabilă cu interacțiunea care are loc între platformă și utilizatorul final. Aplicația va extinde și completa funcționalitățile Platformei digitale *MUSHNOMICS*, făcând-o o soluție completă pentru utilizarea acesteia.

Raportul descrie pe scurt proiectarea și implementarea platformei de date *MUSHNOMICS* cu cele două componente ale ei: sistemul de ingestie și prelucrare a datelor, precum și interfața cu utilizatorul. În cele din urmă, raportul clarifică amplasarea arhitecturală a componentelor digitale care deservește soluția generală *MUSHNOMICS*.

## 2 Obiective planificate în perioada raportată

În perioada de raportare, Holisun a efectuat activități de cercetare, inovare și implementare, necesare pentru buna desfășurare a proiectului *MUSHNOMICS*. Având în vedere rolul tehnic al Holisun, implementarea a ocupat un rol central în activitatea companiei în cadrul proiectului, pe toată durata desfășurării acestuia.

Scopul Holisun în cadrul proiectului a fost, printre altele, și realizarea interfeței grafice cu utilizatorul, interfață ce va fi folosită de către toți utilizatorii sistemului, indiferent de gradul lor de înțelegere. Astfel, obiectivele planificate pentru perioada de raportare, sunt:

- O1 - Testarea și validarea platformei cu ajutorul uneltelor disponibile, dar și cu ajutorul utilizatorilor;
- O2 - Evaluarea performanței platformei
- O3 - Analiza și îmbunătățirea securității platformei

- O4 - Îmbunătățirea experienței utilizatorilor prin feedback și optimizare
- O5 - Dezvoltarea în comun a modelelor de afaceri inovatoare bazate pe modulul MUSHNOMICS adaptat conținuturilor pentru luarea deciziilor informate de către cultivatorii/antreprenorii de ciuperci;
- O6 - Monitorizarea progresului și coordonarea echipei de proiect
- O7 - Exploatarea și diseminarea rezultatelor obținute din cadrul activităților aferente proiectului.

## 2.1 Activități în perioada raportată

### 2.1.1 Activități planificate

În perioada 06.12.2022 - 30.11.2023 au fost planificate următoarele activități:

- Testarea și validarea tuturor funcționalităților platformei MUSHNOMICS pentru a asigura conformitatea cu cerințele și nevoile stabilite anterior.
- Evaluarea performanței platformei sub diferite sarcini și încărcături pentru a identifica și corecta eventualele probleme de performanță.
- Analiza și testarea nivelului de securitate al platformei pentru a identifica și soluționa eventuale vulnerabilități.
- Colectarea feedback-ului de la utilizatori și aplicația schimbărilor sau îmbunătățirilor sugerate în platformă, pentru a îmbunătăți experiența utilizatorului.
- Identificarea punctelor problematice și optimizarea codului și a funcționalităților pentru a îmbunătăți performanța și eficiența platformei.
- Testele finale de validare pentru a asigura că platforma MUSHNOMICS este pregătită pentru lansarea publică.
- Întâlniri periodice online de progres;
- Întâlniri fizice de progres cu partenerii din consorțiu.

### 2.1.2 Activități efectuate

În perioada 06.12.2022 - 30.11.2023 au fost efectuate următoarele activități:

- Testarea și validarea tuturor funcționalităților platformei MUSHNOMICS pentru a asigura conformitatea cu cerințele și nevoile stabilite anterior.
- Evaluarea performanței platformei sub diferite sarcini și încărcături pentru a identifica și corecta eventualele probleme de performanță.
- Analiza și testarea nivelului de securitate al platformei pentru a identifica și soluționa eventuale vulnerabilități.
- Colectarea feedback-ului de la utilizatori și aplicația schimbărilor sau îmbunătățirilor sugerate în platformă, pentru a îmbunătăți experiența utilizatorului.
- Identificarea punctelor problematice și optimizarea codului și a funcționalităților pentru a îmbunătăți performanța și eficiența platformei.
- Testele finale de validare pentru a asigura că platforma MUSHNOMICS este pregătită pentru lansarea publică.
- 6 întâlniri periodice online de progres, în datele: 02.03.2023, 04.04.2023, 02.05.2023, 21.06.2023, 04.08.2023, 10.10.2023
- 3 întâlniri fizice de progres cu partenerii din consorțiu, în datele: 08.12.2022 - Dublin, Irlanda la partenerul UCD, 24-25.05.2023 - Budapesta, Ungaria la partenerul PILZE, 28.11.2023 - Budapesta, Ungaria la partenerul PILZE

### 2.1.3 Devieri de la planificare

În perioada raportată nu au fost devieri de la planificare, sub nici un aspect.

### 3 Metodologii

În acest proiect am utilizat o serie de metodologii standard, lucru care asigură o calitate crescută a livrabilelor și a rezultatelor proiectului.

În acest capitol discutăm diversele metodologii utilizate în cadrul proiectului:

- Asigurarea calității;
- Metodologia de testare;
- Integrare continuă.

#### 3.1 Asigurarea calității

Managementul calității *software* permite echipei care dezvoltă produse *software* să înțeleagă mai bine legătura dintre calitate internă și externă a produsului. Așa cum știm că calitatea externă este cea observată de utilizator, care este testată și care va sta la baza acceptanței produsului de către client. Calitate internă este acea componentă a calității produsului *software* care nu se poate observa, fiind reflectată de structura codului, de regulile de codificare, testare și mentenanță a produsului.

Scopul aplicării unor modele de calitate include evaluarea *software-ului* din diverse perspective legate de specificații, dezvoltare, utilizare, mentenanță, suport, asigurare calitate și audit. În cadrul HOLISUN utilizăm:

- Standardul ISO/IEC 25001:2014 - Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) este utilizat pentru planificare și management. [11]
- Standardul ISO/IEC 25010:2011 - Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) - System and software quality models - definește un model de calitate al produsului compus din opt caracteristici de bază (care sunt împărțite în subcaracteristici). Acest standard a fost revizuit și confirmat în 2017. [12]

Figura 3 prezintă modelul calității conform ISO 25010, așa cum este specificat în site-ul oficial al ISO și cum este implementat în cadrul HOLISUN. Sunt cele opt caracteristici de bază care sunt analizate și care au la rândul lor un set de sub-caracteristici.



Figura 3: Modelul Calității conform ISO 25010

Asigurarea calității *software-ului* este un concept care ar trebui să se întindă pe întregul ciclu de viață al dezvoltării *software-ului* și pe întregul proces de dezvoltare continuă. Indiferent de metodologia adoptată pentru dezvoltare sunt câteva etape comune.

Vom prezenta câteva aspecte ale asigurării calității produsului pentru fiecare etapă de dezvoltare.

Aspecte care contribuie la asigurarea calității produsului și sunt utilizate de *Planning*:

- Asigurarea tuturor informațiilor necesare legate de client și de produs;
- Existența unor proceduri pentru transferul corect și concret de informație de la echipele de sales și presales spre echipele care vor asigura livrarea produsului;



- Realizare unui proiect plan adecvat și adaptat cerințelor;
- Stabilirea instrumentelor de lucru folosite pentru dezvoltare;
- Utilizarea unui software de managementul documentației și review-ul documentației;
- Elaborarea planului de asigurare a calității.

Aspecte care contribuie la asigurarea calității produsului în etapa de Analiza.

Un element foarte important în asigurarea calității software este modul în care sunt definite și structurate specificațiile (requirements) la orice nivel:

- Exista un proces foarte bine organizat pentru colectarea cerințelor de business (business requirements);
- Exista un proces pentru validarea înțelegerii corecte a specificațiilor;
- Un pas foarte important în domeniul calității este folosirea unui instrument (tool) pentru managementul requirementsurilor. Utilizam un tool intern care încorporează și test management și requirements management atunci se poate urmări mai ușor requirements-tests traceability.

Aspecte care contribuie la asigurarea calității produsului implementate în etapa de *Design*:

- Avem definite un proces corect pentru identificarea software și system design;
- Avem definite un proces corect pentru identificarea obiectivelor de testare;
- Utilizam template-uri corecte și a unor instrumente clare pentru etapa de design;
- Utilizam instrumente (tools) adecvate pentru etapa de design;
- Realizam documentații clare și concise ușor de întreținut bazată pe template-uri care să fie foarte utilă echipei cât și clientului final;
- Utilizarea unor tooluri specializate pentru managementul documentației pentru că documentele din acesta faza de dezvoltare vor fi utilizate în etapa de dezvoltare.

Asigurarea calității produsului, implementate în etapa de *Development*:

- Utilizarea unui tool folosit pentru gestiunea activităților de development – Redmine. Trebuie să existe reguli clare pentru implementarea cerințelor;
- Utilizam guideline-uri pentru scriere de cod, în funcție de tehnologia folosită, sau pentru customizare în cazul în care se folosesc alte pachete software pentru configurare;
- Exista un proces clar pentru cod review: mod de abordare, mod de stocare a informațiilor;
- Utilizarea SonarQube [6] pentru calitatea codului.
- Proces clar de elaborare a documentației codului
- Stabilirea unor proceduri clare pentru gestionarea activităților interne, cu ajutorul aplicației interne AIDA.

Aspecte de asigurarea calității produsului în etapa de *Testing*:

- Etapa de testare merge în paralel cu activitatea de dezvoltare;
- Realizarea unui sistem cât mai corect de testare internă;
- Am stabilit de la început reguli pentru bug și issues tracking și modalitățile de abordare și clasificare - Redmine [8];
- Guidelines-uri pentru crearea testelor automate (cu Selenium) și manuale;
- Acordarea unui segment important testării performanței (utilizând jMeter [9] )și securității sistemului (cu BurpSuite [13]).

Elemente care asigură calitatea produsului, vitale în etapa de *Deployment*:

- Elaborarea unor cerințe pentru deployment;

- Elaborarea procedurilor de deployment;
- Folosirea unor instrumente pentru deployment (să nu uităm de Continuous Integration și Continuous Delivery [16]);

Menționăm câteva aspecte care contribuie la asigurarea calității produsului și sunt necesare în etapa de *Maintenance*:

- Punerea la dispoziția echipei de mentenanță a documentațiilor;
- Procedurile concrete de Issue Tracking cu ajutorul aplicației arătate până acum;
- Utilizarea unui instrument pentru gestiunea tichetelor - idem.

## 3.2 Metodologie de testare

Metodologia de testare a fost făcută conform procedurii operaționale interne ale firmei: PRODUSE ȘI DEZVOLTARE SOFTWARE COD: PO 3.1.

### 3.2.1 Scop și obiective

Prezenta procedură documentează modalitatea în care se face planificarea, realizarea și controlul activității de testare.

Scop: Prezenta procedură documentează procesul de programare și testare a programelor/serviciilor software în condiții controlate, în scopul satisfacerii cerințelor clienților.

Obiectiv: Diminuarea până la eliminare a neconformităților cauzate de lipsa de planificare, control, indisponibilitate a informațiilor, neutilizarea unor metode adecvate de validare sau implementare sau chiar necunoașterea/nerespectarea procedurilor de lucru specifice.

### 3.2.2 Domeniu de aplicare

Procedura se aplică pentru toate procesele care concură la proiectarea și realizarea programelor, aplicațiilor și furnizarea / implementarea acestora.

### 3.2.3 Responsabilitate

a) Managementul de vârf are responsabilitatea și autoritatea pentru:

- Asigurarea resurselor financiare, umane și tehnice necesare desfășurării procesului;
- Analiza periodică a eficienței, eficacității și performanțelor procesului.

b) RMC are responsabilitatea și autoritatea pentru:

- Distribuirea prezentei proceduri;
- Asigurarea că întreg procesul de planificare și realizare a programelor se desfășoară în condiții controlate, adică are la bază proceduri specifice de lucru, specificații clare de date, proceduri specifice de testare, validare și documentare pentru client;
- Asigurarea că există proceduri unitare de lucru și că acestea se utilizează de către întreg personalul;
- Asigurarea că programele neconforme sunt evidențiate, separate și tratate conform procedurii de tratare a neconformităților;
- Oprirea imediată și retragerea pentru modificări a programelor neconforme și stabilirea de măsuri adecvate.

c) Responsabilul calitate are responsabilitatea și autoritatea pentru:

- Elaborarea prezentei proceduri;

- Disponerea abordării proiectării și programării numai în condițiile în care acestea au la bază specificații clare, reglementări și restricții legislative precizate, proceduri de lucru analizate riguros cu clientul și există posibilitatea realizării lor în condițiile controlate de asigurare a calității și eficienței;
- Coordonarea activităților în vederea atingerii obiectivelor propuse. Aplicarea procedurilor specifice privind programarea, testarea, validarea și implementarea. Dacă aceste proceduri nu există va urmări realizarea și aplicarea lor;
- Generarea și păstrarea înregistrărilor din proces și stabilirea măsurilor privind eliminarea oricărei neconformități de proces sau produs care poate compromite realizarea programelor;
- Disponerea măsurilor de retragere și modificare, actualizare, completare a programelor găsite necorespunzătoare sau incomplete, pentru a se reduce riscul utilizării lor în continuare;
- Informarea promptă a RMC și a responsabilului privind relația cu clientul asupra oricăror probleme, neconformități sau informații privind constatările și acțiunile întreprinse.
- Disponerea opririi utilizării și retragerea oricărei aplicații sau program care este de natură să aducă prejudicii firmei sau beneficiarilor, cu anunțarea prealabilă a responsabilului privind relația cu clienții și a RMC.

d) Programatorii au responsabilitatea și autoritatea pentru:

- Aplicarea procedurilor de lucru stabilite, iar acolo unde ele nu există – solicitarea de precizări privind modalitatea de abordare a programelor sau implementării;
- Informarea prealabilă completă privind ce se cere de făcut, de ce se cere, cum se dorește să se realizeze, care sunt restricțiile legislative, operaționale sau de altă natură și asupra procedurilor de lucru aplicabile;
- Realizarea activităților de proiectare, programare în condiții de eficacitate, eficiență și calitate – toate vizând satisfacerea cerințelor clientului, inclusiv cele de reglementare;
- Notificarea tuturor acțiunilor, modificărilor, îmbunătățirilor sau oricăror altor acțiuni de acest fel în vederea cunoașterii lor de către colegii de echipă sau implementare.
- Testarea, validarea rezultatelor și controlul fiecărui program în parte și a întregului din care fac parte în vederea asigurării controlului rezultatelor.

### **Descrierea activităților**

#### **Planificarea testării software:**

- Date de intrare: specificații și analiză de caz, cerințe beneficiar și reglementări, sesizări sau neconformități constatate, acțiuni preventive necesare, cerințe exprese ale clienților sau analiștilor de sistem.
- Testarea software se face în condiții planificate și controlate și se realizează de către managerul de calitate.
- Utilizarea echipamentului adecvat, salvările și actualizările de program intră în responsabilitatea programatorului și este urmărită de către Responsabilul producție.
- Personalul este instruit asupra procedurilor (instrucțiunilor) de lucru pe care le aplică. În cazul apariției unor precizări suplimentare acestea se regăsesc în documentația de execuție sau în altă parte.
- Întreg procesul este monitorizat în execuție în fazele prevăzute în documentație.
- Controlul, testarea și validarea se realizează de către personal cu experiență. Fiecare element este controlat, astfel ca nici un produs necorespunzător să nu fie admis la livrare.

Etapele planificării testării automate: Într-o abordare mai detaliată testarea automată înseamnă:

- Planificare
  - identificarea cerințelor și a funcționalităților.
  - gruparea acestora în condiții de test.

- crearea cazurilor de test pentru aceste condiții.
- Design
  - construcția scripturilor de test.
  - generarea testelor de rulare.
- Execuție
  - crearea scenariului de rulare a scripturilor.
  - rularea uneltelor monitor pentru înregistrarea datelor.
  - înregistrarea rezultatelor pentru fiecare rulare.
  - raportarea și gestionarea bug-urilor.
- Management
  - generarea rapoartelor și graficelor.
  - controlul dintr-un singur punct de comandă.
  - documentarea permanentă a stadiului curent al proiectului

### Descrierea procesului

**Testarea** este activitatea de concepție a cazurilor de test, de execuție a testelor și de evaluare a rezultatelor testelor, în diferite etape ale ciclului de viață al programelor.

**Un test** constă în execuția programului pentru un set de date de intrare convenabil ales, pentru a verifica dacă rezultatul obținut este corect.

**Un caz de test** este un set de date de intrare împreună cu datele de ieșire pe care programul ar trebui să le producă.

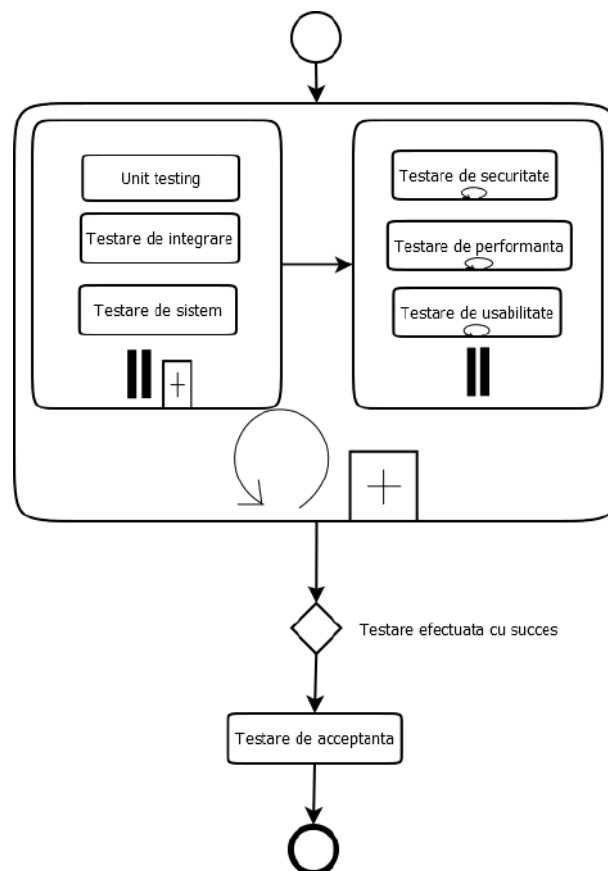


Figura 4: Arhitectura de testare

Procesul de testare se realizeaza in 2 etape:

- **Intern:** pe baza planului de testare, dezvoltat pornind de la cerintele clientului.

Testarea functionala [10] face in vederea verificarii programului si foloseste date de test alese de participantii la procesul de dezvoltare a programului. Ea este efectuata la mai multe nivele: la nivel de unitate functionala, de modul, de subsistem, de sistem. Tipuri:

- Unit testing: Prima treapta a testării; se testează funcțiile sau module de cod. De obicei sunt făcute de programatori deoarece presupun cunoștințe avansate a design-ului intern al aplicației și codului.
- Testare de integrare: Testare combinată a diferitelor părți dintr-o aplicație pentru a vedea dacă funcționează corect. Aceste "parți" pot fi module de cod, aplicații individuale, aplicații client-server într-o rețea etc.
- Testare de sistem: Testare de tip Black-box bazata pe specificații care acoperă toate părțile sistemului

Testarea non-functionala [2]:

- Testare de securitate: Testare pentru a se evidenția cât de bine sistemul este protejat împotriva accesului intern sau extern neautorizat, deteriorarea intenționată, etc.; poate necesita tehnici de testare sofisticate
- Testare de performanta: Acest tip este adesea folosit în mod alternativ cu testarea de "stres" sau testarea de "încărcare".
- Test de stress: Folosit pentru a descrie testarea funcțională a sistemului în timp, sub încărcări neobișnuit de mari, repetarea în mai multe cicluri a unor anumite acțiuni sau intrări, inserarea de valori numerice mari, interogări complexe și mari la un sistem de baze de date, etc
- Testare de usabilitate: Testare prin care se determină dacă un software este "user-friendly" (prietenos pentru utilizatorul final. Evident aceasta testare este subiectivă și va depinde de utilizatorul sau clientul vizat. Interviuurile cu utilizatorii, sondajele, monitorizarea sesiunilor utilizatorilor sunt metode care pot fi folosite.

Categorie testare	Tip testare	Intrari	Sursa	Instrument
Functionala	Unit testing		Clase	jUnit
	Testarea de integrare	Plan de testare integrare	Module	Selenium
	Testarea de sistem	Plan de testare sistem	Sistem software	Selenium
Non-functionala	Testare de securitate	Standard securitate HOLISUN	Sistem software	Burpsuite
	Testare de performanta	Caiet de sarcini	Sistem software	jMeter
	Testare de usabilitate	Plan de usabilitate	Sistem software	Optimizely

Testarea interna este finalizata atunci cand nu mai exista erori de testare. Atunci cand aceasta etapa este incheiata, se trece la testarea externa, impreuna cu clientul.

- **Extern:** testarea de acceptanta, pe baza planului de acceptanta agreat cu clientul.

Testarea in vederea validarii programului, numita si testare de acceptare, are drept scop stabilirea faptului ca programul satisface cerințele viitorilor utilizatori. Ea se efectueaza în mediul in care urmeaza sa functioneze programul, deci folosindu-se date reale. Prin testarea de acceptare pot fi descoperite si erori, deci se efectueaza si o verificare a programului.

În cazul apariției neconformităților atunci acestea se soluționează conform procedurii de sistem "Controlul produselor/serviciilor neconforme" cod PS 8.3-01.

### 3.2.4 Monitorizarea procesului

Monitorizarea procesului de realizare sau modificare a programelor software se asigură de către Responsabilul producție care urmărește realizarea acestor programe conform cerințelor clientului și prezintă în analiza efectuată de management o sinteză a problemelor apărute la aceste programe.

### 3.3 Integrarea Continuă

*Integrarea Continuă* (CI) este practica de automatizare a integrării modificărilor de cod într-un singur proiect software, precum și de integrare a îmbunătățirilor în produsul final, pentru a se menține produsul într-o stare cât mai actuală. [7] Este o practică principală *DevOps* [4], care permite dezvoltatorilor să îmbine frecvent modificările de cod într-un depozit central unde apoi se execută *build*-urile și testele. Instrumente automate de scanare și testare sunt folosite pentru a afirma corectitudinea noului cod înainte de integrare.

Un sistem de control al versiunilor codului sursă este piesa centrală a procesului CI. Sistemul de control al versiunilor este, de asemenea, completat cu alte verificări, cum ar fi teste automate de calitate a codului, instrumente de revizuire a stilului de sintaxă și multe altele. CI este un instrument utilizat pe larg în cadrul metodologiei Agile [1] de dezvoltare a produselor software, metodologie care a fost adoptată și în cadrul proiectului *MUSHNOMICS* având în vedere că proiectul este unul cu o componentă de cercetare.

În cadrul companiei (și implicit în cadrul prezentului proiect), pentru implementarea metodologiei *Agile* și a CI, se utilizează următoarele componente software:

- **GitHUB**, pentru lucrul colaborativ cu codul, precum și o instanță internă de GIT;
- **Jenkins, Semaphore**, unelte pentru automatizarea procesului de CI. *Semaphore* oferă suport și pentru analiza statică a codului, fiind utilizat preponderent local, pentru stadiile inițiale de dezvoltare;
- **Leantime**, unealtă de management a proiectelor software, centrat pe metodologia Agile;
- **jUNIT, PhpUNIT**, unelte pentru testarea automată de tip *Unit Testing*;
- **PHP Mess Detector, PHP Code Sniffer**, unelte pentru îmbunătățirea codului, prin eliminarea de erori umane (datorate neatenției sau a lipsei de conformare la principiile interne de scriere a codului sursă);
- **SensioLabs Security Checker, Snyk**, unelte pentru verificarea dependențelor externe contra vulnerabilităților cunoscute, pentru a se putea elimina/mitiga/îmbunătăți securitatea platformei;

Dezvoltarea și punerea în producție a fiecărei componente (figura 5), precum și îmbunătățirile și actualizările ulterioare, se fac în trei etape, fiecare dintre ele ridicând calitatea codului scris precum și asigurând un nivel suplimentar de securitate și reziliență la probleme (*bug-uri*):

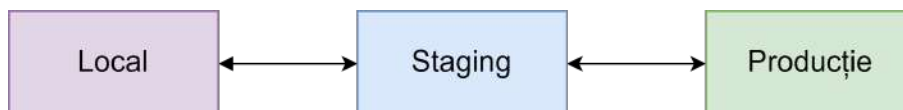


Figura 5: Etapele în proiectare software

- **Local** - etapa în care fiecare componentă este lucrată de către o echipă de dezvoltare. Codul este scris și analizat, testat la nivel de Unit Testing, se verifică aderența acestuia la nivelul de calitate necesar aplicației precum și la cerințele finale ale platformei. Orice problemă întâlnită se rezolvă pe loc, codul fiind promovat în etapa următoare doar atunci când nu mai există erori de funcționare. Această etapă asigură atât calitatea codului cât și rezolvarea a cel puțin 70% dintre problemele cele mai comune;
- **Staging** - etapa în care codul este introdus într-o simulare apropiată de mediul de producție. Se aplică testări automate, specifice sistemelor dezvoltate, precum și testare manuală, conform funcționalităților așteptate. O importantă componentă în această etapă este testarea din punct de vedere al securității și rezilienței la atacuri. Orice eroare găsită în această etapă întoarce codul în etapa locală, unde este rezolvată. Odată trecute toate testele, codul este gata de pus în producție. Această etapă asigură rezolvarea a cel puțin 99% din problemele posibile, de orice natură, făcând codul pregătit pentru serverul de producție;
- **Producție** - etapa de utilizare normală a produsului software. Deși există posibilitatea găsirii de probleme, acestea sunt rare și excepționale, caz în care codul se întoarce în *Staging* (pentru confirmarea problemei) și apoi în *Local* (pentru remedierea acesteia) și din nou în *Staging* pentru validarea rezolvării.

## 4 Realizări tehnice

În era digitală în continuă evoluție, dezvoltarea unei arhitecturi tehnologice robuste și eficiente este esențială pentru a răspunde nevoilor complexe ale utilizatorilor. Arhitectura (figura 6) propusă în această soluție reprezintă o sinteză armonioasă între dispozitive responsive pentru interfața utilizator (4.1), canale de comunicare securizate pentru transferul datelor(4.2), algoritmi avansați de substrat pentru prelucrarea informațiilor și o ontologie bine definită pentru a asigura coerența și semnificația datelor.

### 1. Dispozitive Responsive pentru UI:

În vârful acestei arhitecturi se află dispozitivele responsive pentru interfața utilizator (UI), concepute pentru a oferi o experiență fluidă și adaptabilă pe o gamă variată de dispozitive. Acestea asigură accesibilitatea și interacțiunea intuitivă, indiferent de platforma sau dimensiunea ecranului, facilitând astfel experiența utilizatorului.

### 2. Canal Securizat de Comunicare:

Pentru a proteja integritatea și confidențialitatea datelor, arhitectura include un canal securizat de comunicare. Acesta reprezintă un mediu sigur prin care informațiile sunt transferate între dispozitivele utilizatorului și sistemul central. Protocoalele avansate de securitate și criptare asigură că datele sunt protejate împotriva accesului neautorizat și a potențialelor amenințări cibernetice.

### 3. Algoritm dinamic de generare a compoziției substratului:

Odată ce datele sunt transferate prin canalul securizat, acestea sunt supuse prelucrării utilizând un algoritm dinamic de generare a compoziției substratului avansat. Acest algoritm complex efectuează calcule și analizează informațiile primite pentru a extrage multiple indici relevanți. Fiind optimizat pentru performanță și eficiență, algoritmul dinamic de generare a compoziției substratului joacă un rol crucial în transformarea datelor brute în informații valoroase și ușor de interpretat.

### 4. Ontologie referitoare la cultivarea ciupercilor:

La baza întregii arhitecturi stă o ontologie bine definită, furnizând un cadru semantic pentru datele procesate. Aceasta asigură coerența în interpretarea informațiilor și stabilește relații semantice între diferitele componente ale sistemului. Prin implementarea unei ontologii solide, se obține o înțelegere comună și coerentă a datelor, facilitând astfel luarea deciziilor și analiza informațiilor referitoare la cultivarea ciupercilor.

Prin combinarea acestor componente într-o arhitectură integrată, soluția propusă oferă o fundație solidă pentru o experiență utilizator remarcabilă, securizată și informațională. În secțiunile următoare, vom explora detaliat unele componente ale acestei arhitecturi, evidențiind rolurile și beneficiile aduse de fiecare în parte.

## 4.1 Interfața cu utilizatorul

Interfața cu utilizatorul, dezvoltată conform metodologiei UX, dispune de diferite etape, modelate în funcție de entitățile fizice, pentru ușurarea înțelegerii și a lucrului cu platforma. Acestea sunt aranjate într-o structură arborescentă, astfel:

- **Materiile prime (Feedstock)** reprezintă sursele pentru crearea substratului pentru ciuperci (vezi Figura 8);
- **Prepararea substratului** reprezintă calcularea mai multor indici cum ar fi: conținutul de azot și cenusa, volumul de apă necesar, numărul de galeti în care vor fi crescute ciupercile. (vezi Figura 9);
- **Cresterea ciupercilor** reprezintă estimarea ciupercilor obținute și a cantității de substrat uzat (vezi Figura 11);
- **Utilizarea substratului uzat** (vezi Figura 12);
- **Notitele** reprezintă secțiune în care utilizatorul își scrie notite despre rețete și despre creșterea ciupercilor (vezi Figura 13);

Utilizatorii principali ai platformei *MUSHNOMICS*, sunt împărțiți în 3 categorii (vezi Figura 7):

- **Utilizatori casnici / hobby** care au mai puțin de 10kg de biodeșeuri pe săptămână;

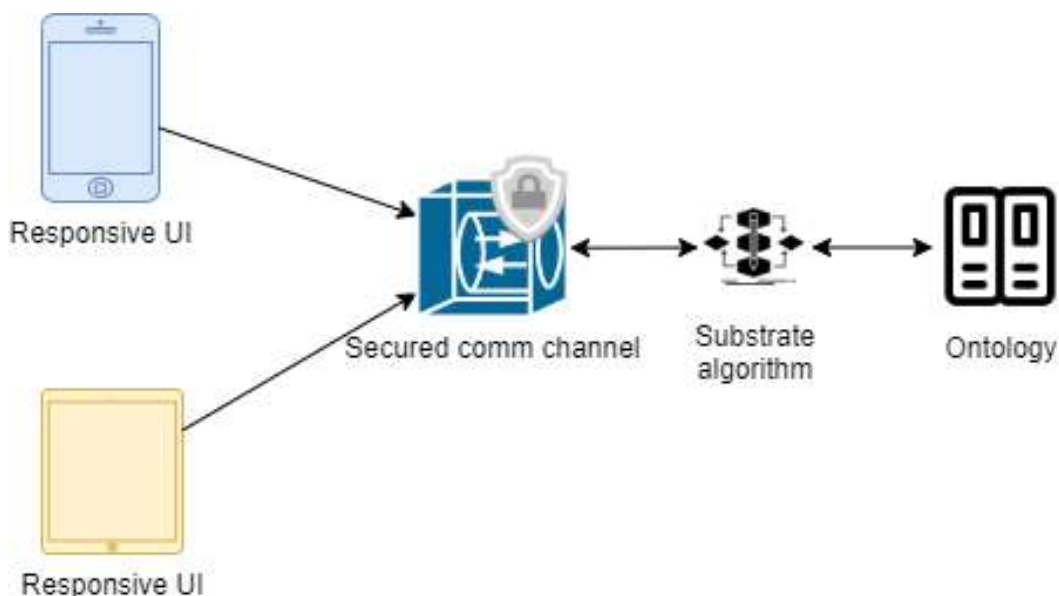


Figura 6: Arhitectura de nivel înalt a funcționalității platformei MUSHNOMICS

- **Utilizatori fermierii urbani** care au mai mult de 10kg de biodeșeuri pe saptamana;
- **Utilizatori pentru antrenament / training** care doresc să se familiarizeze cu platforma MUSHNOMICS.

Level of usage
1. Feedstock
2. Substrate preparation
3. Mushroom growing
4. SMS utilization

My Notes

Your level of usage:

**Home/Hobby**  
(<10kg biowaste/week)

**Urban Farmer**  
(>10kg biowaste/week)

**Training**  
(no biowaste, just learning)

*The usage level is estimated from the feedstock amount.*

Oyster mushroom cultivation is a popular and relatively straightforward method of growing edible mushrooms. Nevertheless, the biomass feedstock available will define the scale and technology options for the processing and growing process. In the platform, we differentiate two scales. If you have less than 10 kgs of biowaste available per week, then hobby growing is for you without any dedicated and expensive equipment just relying on everyday tools and own consumption. In this case, the platform will provide you guidance for the process. Many people start with small-scale home cultivation kits before moving on to larger setups. Above 10 kgs biomass per week, you may consider larger scale urban farming even purchasing specific set of equipment such as the ones from the MUSHNOMICS project.

Figura 7: Tipurile de nivele de utilizatori ai platformei MUSHNOMICS

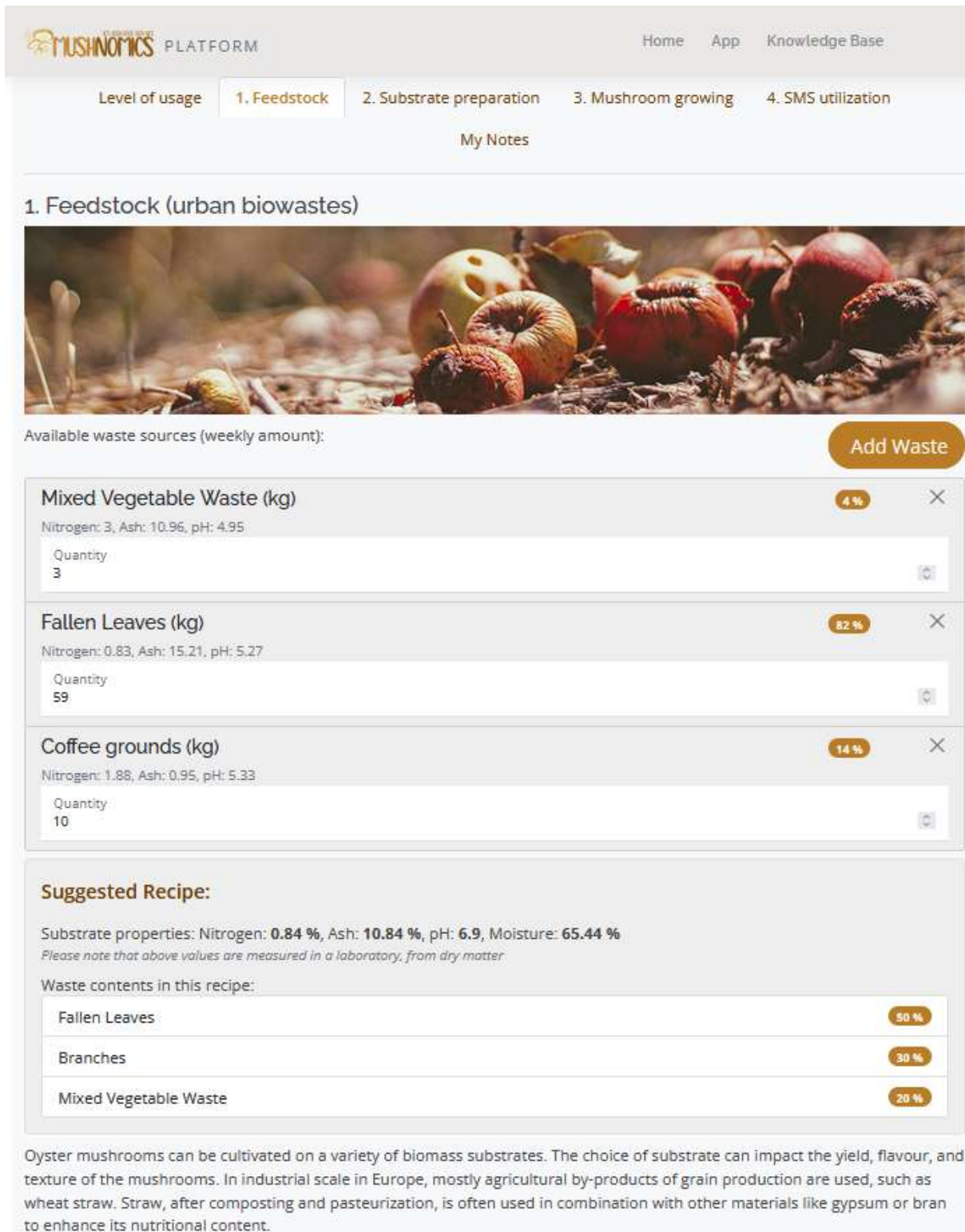
Inițial toți utilizatorii noi care se înregistrează în platforma MUSHNOMICS sunt de tipul *Training*, ulterior după ce în secțiunea materiilor prime de biodeșeuri utilizatorul își introduce tipurile de biodeșeuri pe care le generează și cantitatea acestora, se modifică nivelul la utilizatori casnici dacă are sub 10 kg de biodeșeuri sau la utilizator fermier dacă are peste 10 kg de biodeșeuri pe saptamana.

În secțiunea *Materii prime* din Figura 8 se poate vedea cum utilizatorul și-a selectat tipurile de biodeșeuri și cantitatea acestora pe care o generează, iar platforma i-a sugerat tipul de reteta pentru compostarea biodeșeurilor care include proportionalitatea materiilor prime.

În acest moment exista 4 tipuri de retete care sunt sugerate utilizatorilor in dependenta de ce biodeșeuri ei genereaza. Aceste retete si proportii sunt bine gandite din punct de vedere a cantitatii de azot, cenusa, pH și umiditatea compostului rezultat. Acești indicatori sunt de asemenea calculati automat in dependenta de reteta si cantitatea intro-



dusa de catre utilizator.



**MUSHNORICS PLATFORM** Home App Knowledge Base

Level of usage: **1. Feedstock** 2. Substrate preparation 3. Mushroom growing 4. SMS utilization

My Notes

### 1. Feedstock (urban biowastes)

Available waste sources (weekly amount): Add Waste

Waste Source	Quantity (kg)	Nitrogen (%)	Ash (%)	pH	Percentage
Mixed Vegetable Waste	3	3	10.96	4.95	4%
Fallen Leaves	59	0.83	15.21	5.27	82%
Coffee grounds	10	1.88	0.95	5.33	14%

**Suggested Recipe:**

Substrate properties: Nitrogen: **0.84 %**, Ash: **10.84 %**, pH: **6.9**, Moisture: **65.44 %**  
*Please note that above values are measured in a laboratory, from dry matter*

Waste contents in this recipe:

Fallen Leaves	50%
Branches	30%
Mixed Vegetable Waste	20%

Oyster mushrooms can be cultivated on a variety of biomass substrates. The choice of substrate can impact the yield, flavour, and texture of the mushrooms. In industrial scale in Europe, mostly agricultural by-products of grain production are used, such as wheat straw. Straw, after composting and pasteurization, is often used in combination with other materials like gypsum or bran to enhance its nutritional content.

Figura 8: Secțiunea de calcul a materiei prime pentru substrat

După ce utilizatorul își setează cantitățile și biodeșeurile, platforma calculează în secțiunea de *Prepararea substratului* (vezi Figura 9): *Cantitatea totală de deșeuri*, *Necesarul de apă estimat*, *Substratul total*, *Spawn-ul necesar*, *Numărul de găleți de 7 litri*.

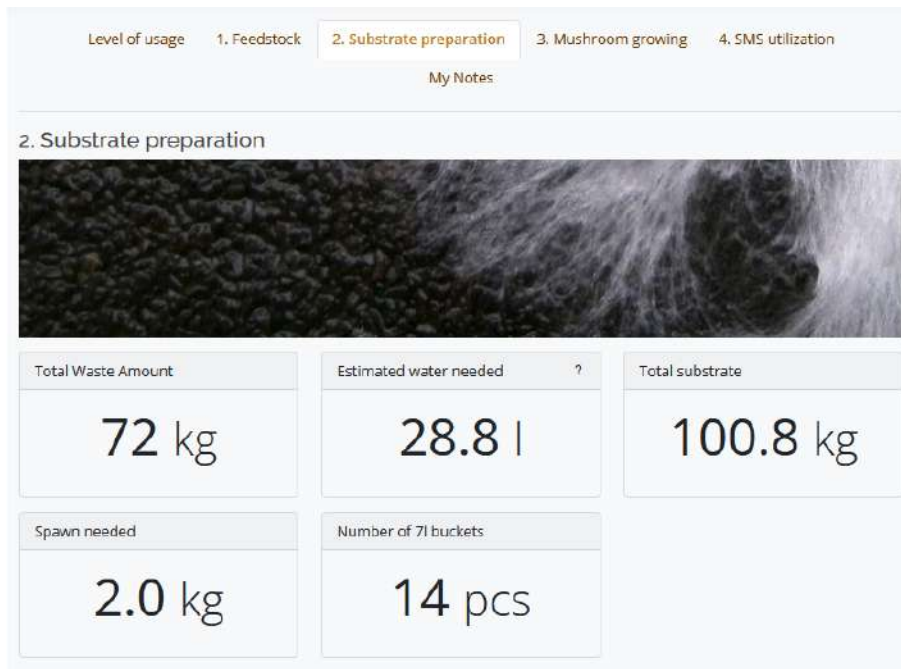


Figura 9: Secțiunea pentru prepararea substratului

În această secțiune sunt prezentate și informațiile de preparare a substratului pentru pregătirea substratului pentru cultivarea ciupercilor de pleurotus care implică crearea unui mediu potrivit pentru ca miceliul să colonizeze și să producă ciuperci.

Platforma MUSHNOMICS pe lângă faptul că efectuează diferite calcule ale indicilor de mai sus prezentate, ea și sugerează anumite corecții pentru nivelul de azot și cenușă din substrat pentru ca acesta să fie în condiții optime. În Figura 10 este prezentat cazul în care nivelul de azot e prea mare, nivelul optim fiind între 0,8 și 1,2. Iar pentru reducerea acestui nivel de azot sunt sugerate 2 opțiuni pentru atingerea pragului optim a substratului.

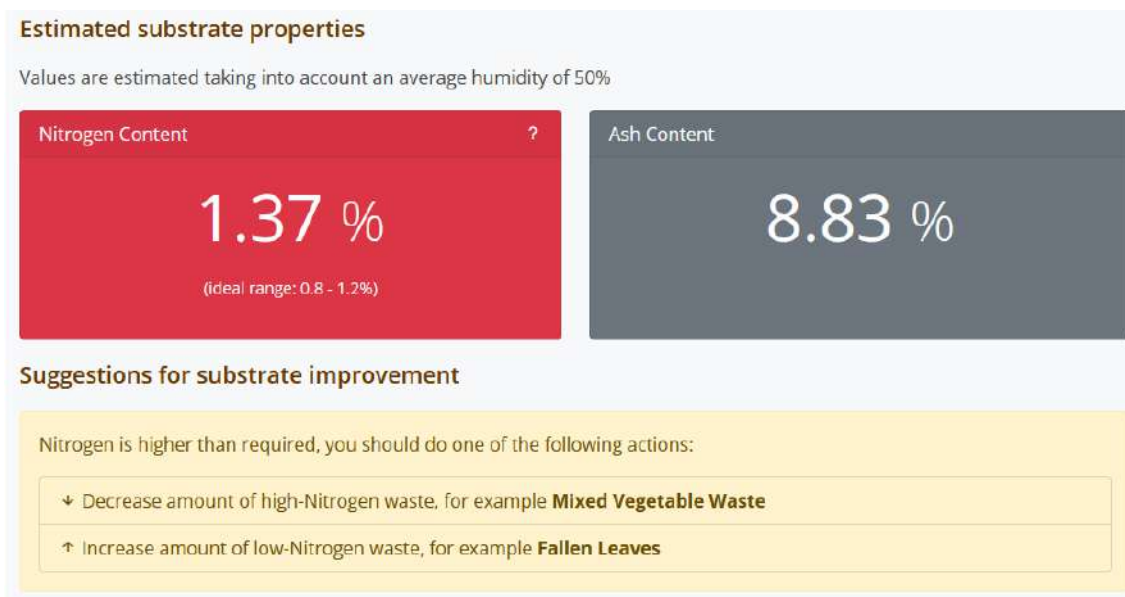


Figura 10: Sugestii pentru îmbunătățirea proprietăților substratului


După ce utilizatorii își prepara substratul împreună cu miceliuri, în secțiunea *Cresterea ciupercilor* (vezi Figura 11) se calculează automat cantitatea de ciuperci așteptată precum și cantitatea de substrat de ciuperci uzată așteptată.

În această secțiune sunt incluși și algoritmi de procesare a pozelor pentru detectarea ciupercilor. În Figura 11 se poate vizualiza cum algoritmul a detectat ciupercile din poza și le-a colorat cu verde.

Level of usage
1. Feedstock
2. Substrate preparation
3. Mushroom growing
4. SMS utilization

My Notes

### 3. Mushroom growing



Amount of mushroom expected

12 kg

(in a month, in two waves)

Amount of SMS expected

16 kg


(after ending the growth in a month)

**Process Image (Mushroom Detector)**

Please select an image file

No file selected.

Processed image



Oyster mushroom cultivation typically involves several distinct phases, each of which needs different environmental conditions. In order to monitor the environmental conditions, you will need a thermometer and a hygrometer, or one combined digital device.

Here are the key phases of oyster mushroom cultivation.

- 1. Incubation:** during this phase you will not see anything from outside of the bucket but inside there is hard work going on. The mycelia start to develop within the substrate and colonize the available space. As secreting enzymes and decomposing the substrate, this phase produces heat by the end of the first week which is a good indicator of the process. For successful incubation, you need to secure constant temperature (around 25-27°C) and humidity. Large changes in temperature need to be avoided as potentially can cause condensation. At home, ideally for this phase you can place your buckets into the kitchen or bathroom. Though you may be curious about what is happening, do not open the buckets as that would lead to contamination! This phase typically takes a few weeks.
- 2. Fruiting conditions:** after full colonization, the so-called pins start to push down the adhesive tapes from the holes which means now it is time to change the environmental conditions to initiate fruiting. Lower the temperature to 12-18°C

Figura 11: Secțiunea pentru creșterea ciupercilor

Următoarea secțiune este *Utilizarea substratului de ciuperci uzat*(vezi Figura 12). Utilizarea substratului de ciuperci uzat reprezintă o strategie inovatoare și durabilă în gestionarea deșeurilor organice. În cadrul acestei soluții, noi calculăm meticulos o serie de indicatori cheie, inclusiv Rata de Reducere a Deșeurilor, care măsoară procentul de diminuare a volumului total al deșeurilor organice prin procesul de compostare a substratului prin doua metode:

- Vermicompostarea;
- Compostarea cu muscă soldat negru.

În plus, evaluăm Indicele de Reducere a Deșeurilor, Rata de Conversie, Rata Relativă de Creștere și Rata de Conversie a Hranii pentru a cuantifica eficiența și sustenabilitatea procesului, contribuind astfel la o gestionare responsabilă a resurselor organice.

Figura 12: Secțiunea pentru utilizarea substratului uzat de la ciuperci

În ultima secțiune de *Notițe*(vezi Figura 13), fiecare utilizator își poate nota anumite observații sau informații despre prepararea substratului, procesului de creștere a ciupercilor precum și utilizarea substratului.

Figura 13: Secțiunea pentru tinerea de notițe despre o anumita rețetă, creștere de ciuperci

Pe lângă aceste secțiuni, se pot compara două instalații / rețete. (vezi Figura 14). Comparând cele două instalații, putem evalua eficiența lor în gestionarea deșeurilor utilizând indicatori precum Cantitatea Totală de Deșeuri, Sursele Deșeurilor, Conținutul de Azot, Conținutul de Cenușă, Cantitatea de Ciuperci Produse și Cantitatea de Substrat de Ciuperci Uzat.



Figura 14: Comparatie a doua rețete/instațatii

**Cantitatea Totală de Deșeuri:** Măsurând cantitatea totală de deșeuri eliminate, putem determina eficacitatea fiecărei instalații în reducerea volumului global de deșeuri generate.

**Sursele Deșeurilor:** Analizând sursele de deșeuri gestionate de instalații, putem identifica sursele predominante și dezvolta strategii specifice pentru minimizarea acestora.

Conținutul de Azot și Conținutul de Cenușă: Evaluând conținutul de azot și cenușă din deșeurile procesate, obținem informații cruciale despre calitatea și valoarea nutrițională a substratului rezultat.

Cantitatea de Ciuperci Produse și Cantitatea de Substrat de Ciuperci Uzat: Prin cuantificarea producției de ciuperci și a cantității de substrat de ciuperci uzat, putem evalua eficiența maximă în obținerea de produse utile din resursele disponibile.

Acești indicatori furnizează o perspectivă holistică asupra performanței instalațiilor în gestionarea deșeurilor și producția de ciuperci, contribuind astfel la optimizarea proceselor și luarea deciziilor informate pentru o gestionare durabilă a resurselor.

## 4.2 Canalul de comunicație

Platforma *MUSHNOMICS* este compusă din două componente majore ce pot funcționa fiecare în cel puțin două moduri distincte. Fiind componente Open Source, acestea se pot instala și la premisele utilizatorului, acestea rulând pe o mașină locală. În acest caz, utilizatorul este responsabil de securitatea aplicației, aceasta fiind asigurată printr-un firewall local, configurat de utilizator.

În caz contrar, cele două componente (Interfața cu Utilizatorul și Platforma de Achiziție și Prelucrare a Datelor) vor funcționa independent, fiind instalate în cloud-ul dezvoltatorilor acestora. Comunicarea între aceste componente se face prin intermediul rețelei Internet, făcându-se prin apeluri de tip REST [14]. Securizarea acestui tip de comunicație se face în straturi, astfel încât per total informația să fie cât mai dificil de extras, asigurându-se astfel și intimitatea informației.

Importanța securității și intimității datelor este dată de tipul de aplicație pentru care platforma a fost gândită, și anume o aplicație la scară mică, instalabilă în apartamente sau case cu suprafețe mici (deci utilizare domestică). Având în vedere legislația GDPR, este imperios necesară asigurarea unei securități cât mai ridicate. Pentru a se putea asigura aceste aspecte, s-a dezvoltat o arhitectură de canal de comunicație ce asigură securitatea și intimitatea datelor chiar și în momentul transferului acestora înspre/dinspre platformă, prin proiectarea diferitelor straturi care împreună compun infrastructura de comunicație și transfer de date.

Securitatea canalului de comunicație se împarte pe următoarele concepte, straturi și abstractizări:

- Se folosește criptarea cu chei publice ce este folosită implicit de sistemele de operare curente, sub forma certificatelor SSL;
- Formatul datelor transferate este standardizat, pentru a se putea asigura coeziunea și flexibilitatea necesare funcționării corecte și rapide a sistemului. S-a dezvoltat clasa `Message` ce înglobează toate câmpurile necesare atât funcțiilor ce pot fi executate (de exemplu `Start` sau `Stop`), dar și transferul clasic de date și/sau informații;
- Canalul de Comunicație dispune de un Registru unde se salvează toate informațiile referitoare la serviciile conectate precum și proprietățile acestora;

### 4.2.1 Criptarea cu chei publice SSL

Modul implicit de comunicare între serverele web și browsere permite trimiterea datelor în text simplu. Acest lucru îi face pe utilizatori vulnerabili în fața atacurilor care pot prelua și citi informațiile pe care le interceptează. SSL criptează comunicația în primă fază, astfel încât interceptarea datelor este aproape inutilă, necesitând un efort suplimentar semnificativ pentru a putea fi citite.

*Secure Sockets Layer* (SSL) este o tehnologie de securitate standard pentru stabilirea unei legături criptate între un server și un client. Securitatea SSL protejează datele sensibile împotriva captării sau modificării pe măsură ce două sisteme le trimit și le primesc, pentru a împiedica utilizatorii neautorizați să acceseze, să citească sau să modifice orice informații personale.

Cele două sisteme care transferă informațiile ar putea fi ambele servere, sau, de asemenea, un client și un server, cum este comunicarea dintre platforma de preluare a datelor și interfața utilizatorului, în cazul proiectului *MUSHNO-*

MICS . SSL este un protocol de securitate care determină modul de criptare a datelor folosind algoritmi specifici. Protocolul SSL securizat de socket evaluează atât datele de transmis, cât și legătura și determină variabilele de criptare pentru ambele.

#### 4.2.2 Formatul Mesajelor transferate în cadrul platformei

Standardizarea comunicațiilor prezintă multe avantaje pentru funcționarea într-un mod optim al platformei, din care putem aminti:

- verificarea structurii necesare este foarte simplă, putându-se astfel detecta dacă o conexiune este validă sau nu. În cazul în care un serviciu trimite un mesaj fără această structură, aceasta va fi automat blocată, considerându-se a fi malefică;
- viteza de execuție crește, sistemul fiind capabil să proceseze o cantitate mult mai mare de date într-un timp mai scurt și folosind o cantitate mult mai mică de energie electrică;
- securitatea sistemului va fi îmbunătățită, deoarece orice serviciu ce încearcă să trimită date în sistem și nu va utiliza acest format de comunicație va fi automat blocat.

Pentru buna funcționare, într-un mod flexibil, a platformei, am stabilit un format anume pentru mesajele ce vor circula în cadrul acesteia. [3, 5, 15] Aceste mesaje pot fi de două tipuri:

1. **Mesaje de execuție**, care pot comunica stări și comenzi între componentele sistemului, și
2. **Transfer de date**, care sunt folosite pentru transferarea bucăților de date între entități.

Aceste mesaje sunt împărțite pe mai multe clase, astfel:

- **Control** - Mesaje emise pentru a semnaliza comanda unei operațiuni ce va trebui executată de un nod final. Aceste mesaje pot fi emise doar de către componentele interfeței cu utilizatorul și vor fi transmise mai departe de platformă către instalații. Aceasta este una dintre cele mai importante clase de mesaje, întrucât permite sistemului să circule mesaje de control, ce vor comunica intenții către instalația fizică;
- **Status** - Mesaje emise ca răspuns la mesajele de Control. Acestea conțin rezultatul execuției și/sau actualizări referitoare la starea nodului;
- **Data** - Mesaje ce conțin date în cadrul sistemului. Acestea au dimensiuni mult mai mari și pot sosi într-un număr mai mare. În cazul datelor ce depășesc 1MB/mesaj, fiecare set de date se va partiționa în mesaje diferite, ce nu vor depăși 1MB;
- **Log** - Mesaje ce vor fi salvate în istoricul disponibil în cadrul platformei. Aceste mesaje indică diferite potențiale probleme ce pot fi întâlnite în cadrul platformei și care au fost semnalate de către noduri.

După cum a fost menționat, Mesajele sunt unități atomice de informații și date care circulă în cadrul sistemului. Serviciile fac schimb de rezultate, solicită actualizări, trimit starea și multe altele, prin utilizarea mesajelor unice. Pentru ca acest lucru să fie posibil, structura unui Mesaj trebuie să găzduiască date precum și informații despre context (*tipul* de Mesaj care este trimis). Mesajul în sine este încorporat într-un plic, iar câmpurile sale obligatorii, împreună cu descrierile lor, sunt:

- **Job ID**: identificarea jobului curent, astfel încât toate componentele sistemului vor avea o înțelegere corectă a schimbului de mesaje;
- **Timestamp**: o coordonată temporală pentru generarea evenimentului;
- **Source-ID**: ID-ul înregistrat al producătorului evenimentului;
- **Event**: numele evenimentului care poate fi citit de om, folosit în scopuri de înregistrare și pentru informații care vor ajunge în interfața de utilizare și prezentate utilizatorului;

- **Message-type:** conține un șir standardizat care este tipul de mesaj. Tipul poate fi din mai multe clase și fiecare componentă poate emite sau primi doar anumite tipuri de Mesaje;
- **Destination:** este numele Serviciului care va consuma Mesajul și este folosit doar pentru cazuri speciale când este cunoscut de producătorul Mesajului;
- **Level of Access (Privacy):** *privat* - destinat unor destinații specifice; *protejat* – destinat oricărui consumator care deține cheia de decriptare (parola), respectiv *public*, care poate fi consumat de oricine primește Mesajul;
- **Priority:** *Scăzută (implicit)* / *Normală* / *Ridicată*;
- **CRC:** mecanism de asigurare a unei livrări valide a Mesajului;
- **Body-format:** descrie formatul Corpului. Poate avea orice format care poate fi serializat într-un String și va anunța receptorul cum să extragă datele din Body;
- **Body-compression:** anunță receptorul dacă corpul a fost comprimat pentru a reduce cantitatea de date transferate, precum și formatul de compresie (de ex.: zip);
- **Body:** corpul efectiv al Mesajului, care în cazul mesajelor de Control și Status poate să nu existe.

#### 4.2.3 Componentele principale ale Canalului de Comunicație și rolul acestora

- **Security Management:** oferă un nivel de securitate pentru prevenirea accesului neautorizat și a altor amenințări externe. Acest modul va bloca, de asemenea, atacurile precum DDoS de la blocarea sistemului;
- **Data Packet Validation:** oferă un mecanism de validare a mesajelor pentru a se asigura că toate mesajele redirecționate nu sunt corupte;
- **Data Safety Management:** oferă un nivel de siguranță pentru a se asigura că numai mesajele intenționate trec și sunt transmise numai dacă este permis; de asemenea, nu conțin informații dăunătoare și/sau cod rău intenționat;
- **BlockChain Based Traceability:** oferă un strat de persistență care permite înregistrarea sistemului și identificarea ulterioară a potențialelor probleme;
- **Delivery Management:** oferă un mecanism care asigură livrarea în siguranță a Mesajelor către toți consumatorii înregistrați, înainte de a le scoate din coadă.

#### 4.3 Aspecte legate de securitate

Securitatea aplicațiilor WEB este un subiect actual, numărul atacurilor fiind în continuă creștere. Deși multe atacuri sunt deja vechi, unele de decenii, toate tind să fie folosite în mod curent, unele datorită simplității lor, altele datorită faptului că există și azi aplicații ce nu sunt protejate împotriva lor. Așadar, rezolvarea tuturor găurilor de securitate dintr-o aplicație este absolut necesară, pentru a preîntâmpina orice breșă de securitate.

În practică, însă, acest aspect este dificil de implementat în mod complet, datorită numărului foarte mare (de ordinul miilor) a tipurilor de atacuri active. Pentru a se îmbunătăți totuși securitatea aplicației, s-a implementat o strategie de securitate în mai multe straturi, ce va filtra accesul în cadrul aplicației pentru diferitele tipuri de acces ce sunt disponibile în cadrul platformei *MUSHNOMICS*.

Pentru accesul WEB, strategia de securitate include parte de logare a utilizatorilor printr-un formular HTML. Posibilitățile de logare includ folosirea protocoalelor OpenID, prin rețeaua de socializare Facebook dar și prin identitatea oferită de profilul Google.

Pentru accesul prin API, strategia de securitate include controlul accesului la nivel de IP, la nivel de aplicație precum și utilizarea de chei de autentificare. Formatele mesajelor ce circulă prin platformă sunt complet standardizate, orice acces ce nu utilizează aceste formate va fi eliminat din cadrul sistemului, prin blocarea atât a cheii de autentificare cât și a IP-ului.

Pentru asigurarea calității codului și eliminarea practicilor necorespunzătoare, s-a folosit suita *BurpSuite*. După rezolvarea problemelor, sistemul va fi mult mai pregătit pentru a face față atacurilor.



Securitatea bazei de date cuprinde următoarele aspecte și mitigări ce au fost prevăzute, pentru a asigura reziliența sistemului:

- **Pericole interne**, cum ar fi ingineria socială, erorile umane sau alte persoane rău intenționate. Pentru mitigare, conținutul bazei de date (mai ales partea de parole de acces) s-a criptat. De asemenea, sistemul de drepturi este unul bine pus la punct, astfel încât o singură persoană are drept de acces pe întreaga bază de date;
- **Exploatarea vulnerabilităților SGBD-ului**. Accesul exterior la SGBD a fost restricționat, astfel încât doar prin intermediul aplicației tampon se poate face acest lucru. De asemenea, fiecare librărie și aplicație instalate, împreună cu sistemul de operare în sine, va fi menținut cu actualizările de securitate la zi, pentru a se preîntâmpina orice posibilitate de atacuri cunoscute;
- **Injecția de cod SQL**. Fiecare câmp destinat a fi inserat în baza de date este verificat în amănunt pentru a se elimina orice urmă de cod executabil SQL;
- **Atacuri Denial of Service (DoS/DDoS)**. Accesul la SGBD nu se poate face din exterior, orice acces neautorizat fiind blocat inclusiv la nivel de firewall;
- **Ștergerea/Pierderea tuturor informațiilor**, ca urmare a unui atac sau ca urmare a unei probleme hardware. Aceste tipuri de pierderi se pot mitiga în două feluri: prin menținerea unei copii de siguranță a tuturor datelor și prin duplicarea sistemelor active. În sistemul instalat, ambele moduri au fost configurate. Pentru SGBD există o imagine în oglindă ce este sincronizată la fiecare cinci minute cu noile modificări. De asemenea, strategia de backup include backup-uri parțiale, făcute la fiecare oră, dar și backup-uri complete ale bazei de date, făcute în fiecare zi;
- **Escaladarea privilegiilor**. Doar administratorul va putea face aceste operațiuni, și doar în urma unor discuții concrete și a hotărârilor luate împreună cu echipa de management a proiectului. Fiecare operațiune a fiecărui utilizator din cadrul sistemului este introdusă într-un log, astfel încât vor exista urme pentru orice acțiune executată de aceștia. În cazul improbabil în care va exista o scurgere de informații datorită unuia dintre utilizatori, acesta va fi tras la răspundere în fața legii.

#### 4.4 Baza de cunoștințe

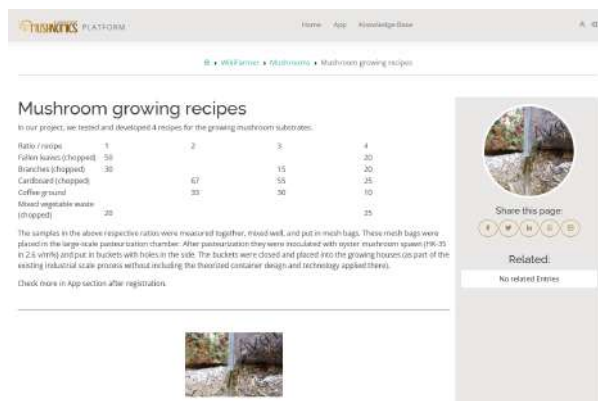
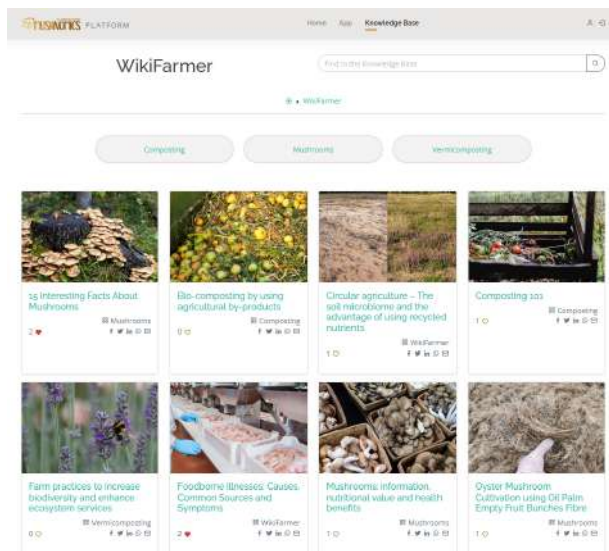
Proiectul *MUSHNOMICS* se adresează atât amatorilor cât și profesioniștilor în domeniul agriculturii verticale. Așadar, nivelul de cunoștințe pe care fiecare categorie de persoane îl are poate varia mult, acest aspect întărind necesitatea unei baze de cunoștințe accesibile tuturor.

În cadrul componentei de interfață cu utilizatorul, s-a dezvoltat un modul pentru baza de cunoștințe accesibilă utilizatorilor, la care s-a ținut cont de următoarele cerințe:

- Interfața cu utilizatorul trebuie să fie simplă și intuitivă, ținându-se cont de aspecte UX (experiența utilizatorului);
- Informațiile vor fi compilate în articole, ce vor putea fi însoțite de imagini (galerie de imagini) și video;
- Articolele vor fi grupate în categorii, fiecare articol putând aparține unei singure categorii;
- Fiecare articol va putea fi legat de un alt articol, care tratează un subiect similar;
- Utilizatorii vor putea marca articolele favorite (buton de *Love*), astfel ridicând-se importanța acestor articole.
- Utilizatorii vor putea distribui pe cel puțin trei căi articolele (atât rețele sociale cât și servicii de mesagerie instant);
- Baza de cunoștințe trebuie să poată fi interogată, astfel încât informația să fie ușor de găsit.

Analizând aceste cerințe, s-au modelat entitățile necesare și s-a făcut o schiță pentru interfața efectivă, cu ajutorul echipei specializate pe UX. Așa cum se poate observa în Figura 15a, interfața bazei de cunoștințe conține următoarele elemente:

- În partea de dreapta-sus găsim **câmpul de filtrare**, cu ajutorul căruia se pot căuta elemente în baza de cunoștințe. Căutarea se face atât în titlu articolului cât și în corpul acestuia, folosindu-se parametrul GET al cererii spre server, pentru a se putea salva căutarea, reveni la ea, sau partaja cu alte persoane;



(a) Interfața cu utilizatorul pentru baza de cunoștințe din cadrul proiectului *MUSHNOMICS*

(b) Design-ul pentru un articol din cadrul bazei de cunoștințe

Figura 15: Interfața pentru baza de cunoștințe și un articol din cadrul acesteia

- Între câmpul de căutare și lista de categorii se găsește secțiunea paginii unde s-a afișat locația curentă în cadrul bazei de cunoștințe (eng.: *Breadcrumb*). Această zonă va permite utilizatorului să cunoască în orice moment locația exactă în care se află. Utilizatorul poate alege să revină la categoria părinte, pentru a vedea și deschide și alte articole cu aceeași tematică;
- Lista de categorii afișează inițial toate categoriile existente în baza de cunoștințe. Dacă utilizatorul alege o categorie, în această zonă se vor încărca toate subcategoriile, permițând astfel o navigare rapidă și intuitivă prin structura arborescentă a categoriilor;
- După lista de categorii se găsește lista de articole disponibile în categoria curentă selectată și toate subcategoriile acesteia. Articolele disponibile au mai multe operațiuni posibile asupra lor, cum ar fi salvarea acestuia într-o listă de articole favorite (operațiunea de *Like*), deschiderea articolului (făcând click pe imagine sau pe titlu), sau partajarea acestuia pe o rețea de socializare, folosind butoanele aferente fiecărui articol în parte.

În cadrul articolului ce se găsește în baza de cunoștințe, avem acces la diferite funcționalități, așa cum se poate observa în Figura 15b. În partea de sus, antetul articolului, se poate observa locația curentă din cadrul bazei de cunoștințe (*Breadcrumb*), similar cu afișarea categoriei. Această locație va fi utilă vizitatorilor pentru a putea interpreta corect locația și a se putea orienta în cadrul bazei de cunoștințe.

Articolul efectiv este format dintr-un titlu, conținut, precum și o galerie foto. Conținutul poate fi introdus și editat printr-un editor WYSIWYG (*Ce se vede este și ce se obține*). Astfel, se pot introduce imagini, video-uri de pe diverse site-uri ce oferă această funcționalitate (YouTube, Vimeo, etc), tabele, liste și text formatat. Toate aceste funcționalități vor ajuta editorii să conceapă articole ce vor explica într-un mod complex și complet anumite concepte, idei, tutoriale.

În partea dreaptă a paginii găsim bara de acțiuni (*sidebar*), unde se poate vedea imaginea de bază a articolului, butoanele de partajare pe rețelele de mesagerie instant și rețelele sociale, precum și o listă cu articole legate (cu legătură între ele). Astfel se pot defini termeni asociați articolului curent precum și serii de articole cu o temă comună, ce trebuie citite într-o ordine prestabilită (cum ar fi cazul tutorialilor).

Galeria de imagini se află în partea jos a paginii articolului. Aici se vor putea vedea toate imaginile ce au fost încărcate pentru articolul deschis, existând posibilitatea de a deschide oricare dintre ele. Odată deschisă o imagine, navigarea între ele se face cu ajutorul săgeților de pe tastatură sau făcând click pe imagine, pe controalele aferente.

## 5 Artefacte

Rolul HOLISUN în cadrul proiectului *MUSHNOMICS* a fost unul tehnic, astfel încât activitatea noastră a fost concentrată mai degrabă pe rezultate palpabile, de nivel tehnologic ridicat, ușor de adus la TRL 9, pregătite pentru a fi marketate.

Tabela 1: Artefacte generate de HOLISUN în cadrul proiectului *MUSHNOMICS*

Denumire produs	Tip	Descriere	Data
<i>MUSHNOMICS</i> UI	Produs informatic	Platforma inteligenta pentru monitorizarea și creșterea de ciuperci.	30.10.2023
Baza de cunostinte	Produs informatic	Baza de cunostinte extensibila si colaborativa pentru intretinerea informatiilor publice.	15.09.2023
Algoritm dinamic de generare a compoziției substratului	Produs informatic	Algoritm dinamic de generare a compoziției substratului	15.09.2023
Comparator de setup	Produs informatic	Comparator de setup	20.10.2023

### 5.1 UI - Interfața cu utilizatorul

Interfața cu utilizatorul este una dintre cele mai importante componente ale unui sistem IT, deoarece este chipul aplicației, modul de contact al acesteia cu utilizatorii. O interfață bine definită și cu un design plăcut, va aduce experiența utilizatorilor la un nivel înalt, făcându-i pe aceștia să prefere aplicația și să revină la ea cât de des este nevoie.

În cadrul proiectului *MUSHNOMICS* am dezvoltat o interfață ce aderă la toate principiile designului precum și la principiile experienței cu utilizatorul (UX). Acestea sunt foarte importante în acest context, deoarece oferă dezvoltatorilor uneltele necesare pentru a înțelege și a crea componente software calitative.

Ca rezultat al dezvoltării, interfața dezvoltată de HOLISUN a fost testată și validată cu ajutorul unor potențiali clienți și entuziaști, ce își doresc achiziționarea sau construcția unei astfel de platforme. Întrucât interfața și aplicația ce o controlează au fost construite într-un mod flexibil, acestea se pot integra și în alte sisteme, pe tematici diferite față de proiectul de față, rezultând astfel o componentă flexibilă și marketabilă facil.

### 5.2 Knowledge base - Baza de cunoștințe

Baza de cunoștințe (KB) este una dintre cele mai importante componente în orice aplicație destinată entuziaștilor. KB-ul permite schimbul de informații între utilizatori, fără a obliga discuțiile directe, prin realizarea de materiale și articole deschise, instructive, la care și restul utilizatorilor pot contribui prin marcarea acestora ca favorite.

Marcarea ca favorit a unui articol are două rezultate, unul fiind ridicarea importanței acestuia, al doilea fiind salvarea în lista de favorite a articolului respectiv. Astfel, utilizatorii vor fi motivați să salveze doar articolele care le-au fost cu adevărat utile.

Făcând parte din componenta UI, KB-ul este o unealtă colaborativă de indexare și reținere a cunoștințelor. Deși aici implementată într-o aplicație specifică, ea poate fi implementată rapid în orice aplicație web similară, ridicându-i astfel gradul de importanță. La fel ca și UI-ul, KB-ul este o componentă flexibilă, gata de a fi marketată facil atât ca și o componentă de sine stătătoare, cât și ca parte a unei platforme extinse de management al cunoștințelor.

### 5.3 Algoritm dinamic de generare a compoziției substratului

În cadrul algoritmului dezvoltat pentru generarea compoziției substratului și estimarea proprietăților substratului, mai multe parametri sunt luați în considerare pentru a asigura o evaluare precisă și detaliată. Algoritmul are ca scop furnizarea unor informații utile privind compoziția și calitatea substratului rezultat din procesul de fermentație al ciupercilor.

Iată câteva aspecte cheie ale acestui algoritm, adaptate conform cerințelor:

1. *Cantitatea Totală de Deșeuri:*

Algoritmul evaluează cantitatea totală de deșeuri procesate, oferind astfel o măsură a eficacității în reducerea volumului global de deșeuri generate în procesul de producție.

2. *Necesarul de Apă Estimat:*

Parametrul acesta cuantifică estimativ cantitatea de apă necesară pentru procesul de fermentație al ciupercilor, având un impact direct asupra calității și a randamentului substratului.

3. *Substratul Total:*

Algoritmul calculează volumul total al substratului rezultat în urma procesului de fermentație, oferind informații despre disponibilitatea acestuia pentru utilizare ulterioară.

4. *Spawn-ul Necesar:*

Parametrul spawn reprezintă porțiunea de miceliu încorporată în substrat pentru a începe procesul de creștere a ciupercilor. Algoritmul estimează cantitatea optimă de spawn necesară pentru o producție eficientă.

5. *Numărul de Găleți de 7 Litri:*

Acest aspect cuantifică numărul de recipiente de 7 litri necesare pentru a conține substratul și spawn-ul, oferind o măsură a volumului de lucru și a eficienței logisticilor.

6. *Cantitatea de Azot și Cantitatea de Cenușă:*

Algoritmul analizează conținutul de azot și cenușă în substrat, furnizând informații cruciale despre valoarea nutrițională și compoziția minerală a substratului.

Prin integrarea acestor aspecte, algoritmul dinamic de generare a compoziției substratului aduce o abordare științifică și eficientă în evaluarea calității substratului obținut din procesul de compostare a biodeșeurilor. Acesta reprezintă un instrument esențial pentru gestionarea și optimizarea proceselor, facilitând utilizarea durabilă a resurselor și producția eficientă de substrat de calitate.

## 5.4 Comparator de setup și rețete

În procesul de comparare a două instalații, detaliile legate de *Cantitatea Totală de Deșeuri* oferă o privire asupra eficienței sistemelor în reducerea impactului asupra mediului. Prin analiza surselor de deșeuri gestionate de fiecare instalație, se pot identifica potențialele surse predominante și se pot dezvolta strategii personalizate pentru o abordare mai sustenabilă a eliminării deșeurilor.

*Conținutul de Azot și Conținutul de Cenușă* sunt indicatori cheie în determinarea calității substratului rezultat din procesul de fermentație al ciupercilor. Un conținut echilibrat de azot este esențial pentru o nutriție adecvată a ciupercilor, în timp ce conținutul de cenușă reflectă nivelul de minerale și substanțe minerale persistente în substratul final, oferind astfel informații despre calitatea nutrițională și reciclarea substanțelor.

*Cantitatea de Ciuperci Produse și Cantitatea de Substrat de Ciuperci Uzat* sunt două aspecte cruciale care indică eficacitatea în utilizarea resurselor disponibile. Cu cât o instalație produce o cantitate mai mare de ciuperci utilizând o cantitate mai mică de substrat uzat, cu atât procesul este considerat mai eficient și sustenabil. Aceste două măsurători contribuie la evaluarea randamentului global al instalației în transformarea deșeurilor în produse utile.

În concluzie, compararea instalațiilor în funcție de acești indicatori oferă o imagine comprehensivă asupra performanței lor în gestionarea deșeurilor organice. Această analiză detaliată facilitează identificarea celor mai eficiente practici și optimizarea proceselor pentru a atinge obiectivele de sustenabilitate și reducere a impactului asupra mediului.

## 6 Diseminare și exploatare

### 6.1 Activități de diseminare

Proiectul a fost diseminat în următoarele moduri:

- pe pagina web: <https://research.holisun.com/ro/proiecte/agriculture-4-0/mushnomics-ro>, având un număr de 180 de vizitatori lunari
- pe contul de LinkedIn: <https://www.linkedin.com/company/holisun>, cu 433 de adepți
- pe contul de LinkedIn al proiectului <https://www.linkedin.com/company/mushnomics-project/>, cu 79 de urmăritori
- pe pagina de Facebook: <https://www.facebook.com/Holisun.IT/>, având 2.2k de urmăritori
- pe contul de Twitter al proiectului: <https://twitter.com/mushnomics>, având 26 de urmăritori.

Au fost desfășurate o serie de activități de diseminare în cadrul unor evenimente de afaceri, expoziții și evenimente de brokeraj sau networking, listate în Tabelul 2.

Tabela 2: Lista activității de diseminare

Nume	Data	Link	Participanti	Rezultate
Barcelona Cybersecurity Congress	31 Ianuarie - 02 Februarie 2023	<a href="https://www.barcelonacybersecuritycongress.com/">https://www.barcelonacybersecuritycongress.com/</a>	Rudolf Erdei, Daniela Delinschi	Diseminare rezultate <i>MUSHNOMICS</i>
International B2B Software Days	08-09 Mai 2023	<a href="https://2023.b2bsoftwaredays.com/">https://2023.b2bsoftwaredays.com/</a>	Rudolf Erdei, Daniela Delinschi, Oliviu Matei	Prezentare <i>MUSHNOMICS</i>
Intalnire cu stakeholderi din Budapesta, Ungaria	25 Mai 2023		Rudolf Erdei, Daniela Delinschi	Prezentare <i>MUSHNOMICS</i>
Cybersecurity and the Future Europe	20 Iulie 2023	<a href="https://research.holisun.com/ro/noutati/atelierul-bieco-de-securitate-cibernetica">https://research.holisun.com/ro/noutati/atelierul-bieco-de-securitate-cibernetica</a>	Rudolf Erdei, Daniela Delinschi, Oliviu Matei	Prezentare <i>MUSHNOMICS</i>
Intalniri cu stakeholderi din Baia Mare, Romania	15 Septembrie 2023		Oliviu Matei	Prezentare <i>MUSHNOMICS</i>
Technology & Innovation Summit-Networking Event din Cluj-Napoca, Romania	26 Octombrie 2023	<a href="https://bit.ly/40TjJ8O">https://bit.ly/40TjJ8O</a>	Oliviu Matei	Prezentare <i>MUSHNOMICS</i>
Digital Twin Workshop la USV	10 Noiembrie 2023		Rudolf Erdei, Daniela Delinschi	Diseminare rezultate <i>MUSHNOMICS</i>

#### 6.1.1 Alte activități de diseminare

Proiectul a mai fost diseminat prin următoarele canale:

Canal	Adresa	Indicatori
Pagina web	<a href="https://mushnomics.org/">https://mushnomics.org/</a>	Vizitatori: 1.1K
LinkedIn	<a href="https://www.linkedin.com/company/mushnomics-project/">https://www.linkedin.com/company/mushnomics-project/</a>	Postări: 47
Twitter	<a href="https://twitter.com/mushnomics">https://twitter.com/mushnomics</a>	Tweet-uri: 56

## 6.2 Exploatare

Exploatarea rezultatelor implică înțelegerea pieței, pentru a se putea proiecta un plan de afaceri potrivit acesteia. Pentru aceasta, s-a luat în considerare abstractizarea (sau generalizarea) conceptelor ce există în cadrul platformei, pentru a putea fi adaptată ușor și altor piețe ce utilizează sisteme similare, cum ar fi piața auto, logistica, microclimatele, sistemele smart-home, și multe altele, piețe în care aplicațiile IoT sunt în centrul atenției.

Numărul tot mai mare de aplicații IoT crește nevoia de securitate, ceea ce conduce piața. IoT va consta din miliarde de dispozitive digitale, servicii și alte obiecte fizice care au potențialul de a se conecta, interacționa și face schimb de informații fără probleme. Piața auto devine, de asemenea, un element important al IoT, conectat la lumea exterioară printr-un număr tot mai mare de tehnologii wireless. Deși beneficiile conexiunii îmbunătățite sunt benefice, de asemenea, deschide o mare de oportunități atacatori. Conform informațiilor privind securitatea cibernetică a anului 2020 de la Upstream Security, din 2018 până în 2019, a existat o creștere cu 99% a incidentelor de securitate cibernetică.

Astfel, activitățile desfășurate în cadrul prezentului proiect, și anume de proiectare și implementare a unui sistem rezilient la atacuri, distribuit, rapid și modern, asigură posibilitatea Holisun de a proiecta sisteme de ultimă generație.

## 7 Impact

Prezentarea impactului proiectului *MUSHNOMICS* este prezentat în primul rând prin prisma Indicatorilor de rezultat, dar și ca impact social, economic și asupra mediului.

### 7.1 Indicatori de rezultat

Cu toate ca in propunerea de finantare nu au fost definiti indicatori de rezultat, dorim sa prezentam in acest raport sub forma de indicatori, rezultatele obtinute.

Indicatori		
<b>Indicatori referitori la Platforma MUSHNOMICS</b> Numar de utilizatori incepatori - 22 Numar de utilizatori avansati - 3 Numar de vizitatori unici - 32 Numar de instalatii create in platforma - 43	<b>Indicatori referitori la baza de cunostinte</b> Numar de articole - 15 Numar de accesari - 35 Numar de like-uri - 57 Numar mediu de like-uri - 1 like/zi	<b>Indicatori referitori la diseminare</b> Postari pe LinkedIn - 47 Postari pe Twitter - 56 Postari pe Facebook - 42 Articole pe mushnomics.org - 12 Numar de vizitatori unici - 1k Numar de vizite - 1.1K

Figura 16: Indicatori de rezultat

### 7.2 Impactul social

**Includerea Comunitară și Autonomie Alimentară:** Sistemul MUSHNOMICS, prin intermediul containerului pentru creșterea ciupercilor și platformei pentru crearea rețetelor de substrat, promovează includerea comunitară și autonomia alimentară. Locuitorii pot fi implicați direct în procesul de producție alimentară, contribuind la consolidarea comunității și reducând dependența de surse externe.

**Promovarea Educației Nutriționale:** Prin cultivarea ciupercilor pleurotus în comunitate și accesul la informații despre crearea substratului, sistemul încurajează educația nutrițională. O comunitate informată poate face alegeri alimentare mai sănătoase, contribuind la îmbunătățirea stării de sănătate generală.

**Stimularea Economiei Locale:** Implementarea sistemului MUSHNOMICS stimulează economia locală prin promovarea producției locale și a achiziționării de resurse și servicii la nivel comunitar. Acest lucru poate spori sustenabilitatea economică și reduce dependența de lanțurile de aprovizionare globale.

Aceste aspecte evidențiază impactul pozitiv al sistemului MUSHNOMICS asupra aspectelor sociale, încurajând comunitatea să devină mai autonomă, educată și implicată în procesele legate de producția alimentară.

### 7.3 Impactul asupra mediului

**Reducerea Deșeurilor Organice:** Sistemul MUSHNOMICS contribuie semnificativ la reducerea deșeurilor organice prin compostarea ulterioară a substratului uzat. Această practică ecologică transformă deșeurile organice în resurse valoroase, reducând impactul asupra depozitelor de deșeuri și prevenind poluarea mediului.

**Utilizarea Eficientă a Resurselor:** Platforma pentru crearea rețetelor de substrat permite optimizarea compoziției substratului, asigurând astfel utilizarea eficientă a resurselor. Aceasta contribuie la minimizarea consumului de materii prime și la o producție mai sustenabilă a ciupercilor.

**Energie și Emisii Reduse:** Containerul pentru creșterea ciupercilor poate fi proiectat pentru a utiliza energie eficientă și surse regenerabile. Prin adoptarea practicilor eco-friendly în procesele de creștere și gestionare, sistemul MUSHNOMICS reduce amprenta de carbon și emisiile asociate producției de alimente.

**Îmbunătățirea Calității Solului prin Compostare:** Compostarea substratului uzat aduce beneficii suplimentare mediului, contribuind la îmbunătățirea calității solului. Substratul compostat oferă nutrienți esențiali și îmbunătățește structura solului, susținând biodiversitatea și ciclul natural al nutrienților.

### 7.4 Impactul economic

**Creșterea Veniturilor Locale:** Implementarea sistemului MUSHNOMICS poate contribui la creșterea veniturilor locale prin vânzarea ciupercilor pleurotus produse. Comunitățile pot valorifica produsele locale, generând astfel o sursă suplimentară de venituri pentru fermieri și producători locali.

**Eficiența în Utilizarea Resurselor:** Prin utilizarea eficientă a resurselor și a tehnologiilor, sistemul MUSHNOMICS poate reduce costurile de producție. Acest aspect contribuie la creșterea eficienței economice, facilitând accesul la produse de calitate la prețuri mai competitive.

**Catalizarea Sectorului Agroalimentar Local:** Sistemul MUSHNOMICS poate funcționa ca un catalizator pentru dezvoltarea sectorului agroalimentar la nivel local. Generarea unei producții consistente de ciuperci și substraturi poate stimula dezvoltarea altor ramuri ale industriei agroalimentare, cum ar fi prelucrarea alimentelor și distribuția.

**Reducerea Dependentei de Surse Externe:** Prin promovarea producției locale, sistemul MUSHNOMICS poate contribui la reducerea dependenței de importurile de alimente. Aceasta crește reziliența economică a comunității față de fluctuațiile pieței internaționale și schimbările din lanțurile de aprovizionare globale.

**Generarea de Oportunități Antreprenoriale:** Odată cu implementarea sistemului, se deschid oportunități pentru antreprenori locali să dezvolte afaceri în domeniul producției de ciuperci sau serviciilor conexe, cum ar fi consultanța pentru substraturi sau prelucrarea produselor rezultate.

**Inovație și Transfer de Tehnologie:** Dezvoltarea și implementarea sistemului MUSHNOMICS pot contribui la inovație și transfer de tehnologie în comunitate. Aceasta poate crea un mediu propice pentru dezvoltarea unor inițiative noi și pentru adoptarea unor tehnologii inovatoare în sectorul agroalimentar.

Prin aceste impacte economice, sistemul MUSHNOMICS devine nu doar o soluție durabilă pentru producția de alimente, ci și un catalizator pentru dezvoltarea economică locală, creând o serie de oportunități pentru antreprenori și contribuind la consolidarea stabilității economice în comunitate.

## 8 Concluzii

Sistemul MUSHNOMICS se afirmă ca o soluție holistică și inovatoare în domeniul culturii de ciuperci pleurotus. Integrarea containerului pentru creșterea ciupercilor și a platformei de creare a rețetelor de substrat aduce beneficii semnificative în diverse domenii.

Din perspectivă socială, MUSHNOMICS contribuie la stimularea comunităților locale prin promovarea educației nutriționale și implicarea directă a rezidenților în producția alimentară. Aceasta nu doar sprijină autonomia alimentară, ci și favorizează o înțelegere mai profundă a proceselor agroalimentare.

Din punct de vedere economic, sistemul MUSHNOMICS reprezintă un motor al dezvoltării locale prin generarea de venituri și crearea de oportunități antreprenoriale. Eficiența în utilizarea resurselor și inovațiile aduse în procesele de producție pot contribui la stabilirea unor practici sustenabile în sectorul agroalimentar.

La nivel ecologic, MUSHNOMICS se dovedește a fi o alternativă prietenoasă cu mediul prin reducerea deșeurilor organice și promovarea unui ciclu închis de producție și compostare. Această abordare susține conceptele de sustenabilitate și economie circulară.

Prin urmare, MUSHNOMICS nu este doar o soluție de cultivare a ciupercilor, ci un sistem comprehensiv ce aduce beneficii semnificative atât comunităților locale, cât și mediului înconjurător.

### 8.1 Activități viitoare

Activitățile viitoare se axează pe îmbunătățirea sistemului și exploatarea rezultatelor. Mai exact, preconizăm activități referitoare la:

1. **Aducerea platformei la maturitate tehnologica TRL 7-8.** Pentru aceasta, toate funcționalitățile vor fi testate în condiții relevante de funcționare, atât în cadrul PILZE Kft, cât și împreună cu alți potențiali clienți ce vor fi coopțați;
2. **Aducerea Interfeței cu Utilizatorul la maturitate tehnologica TRL 6.** Datorită flexibilității, această interfață va putea fi implementată și altor tipuri de aplicații;
3. **Dezvoltarea unei noi linii de business** cu ajutorul platformei, ce va conține atât direcția curentă, în producția de ciuperci, cât și direcții alternative, cu ajutorul altor clienți din domeniul agricol.



## Referințe

- [1] Abrahamsson, P., Salo, O., Ronkainen, J., Warsta, J.: Agile software development methods: Review and analysis. arXiv preprint arXiv:1709.08439 (2017)
- [2] Dave, M.S., Patel, M.H., Prajapati, M.A.: Non-functional testing in agile development (2018)
- [3] Delinschi, D., Erdei, R., Matei, O.: Ontology driven high performance messaging system for distributed software platforms. In: 2022 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR). pp. 1–6. IEEE (2022)
- [4] Ebert, C., Gallardo, G., Hernantes, J., Serrano, N.: Devops. *Ieee Software* **33**(3), 94–100 (2016)
- [5] Erdei, R., Delinschi, D., Pașca, E., Matei, O.: Orchestrator architecture and communication methodology for flexible event driven message based communication. In: Computational Intelligence in Security for Information Systems Conference, International Conference on European Transnational Education. pp. 127–137. Springer (2023)
- [6] Ferenc, R., Langó, L., Siket, I., Gyimóthy, T., Bakota, T.: Source meter sonar qube plug-in. In: 2014 IEEE 14th International Working Conference on Source Code Analysis and Manipulation. pp. 77–82. IEEE (2014)
- [7] Fowler, M., Foemmel, M.: Continuous integration (2006)
- [8] Gata, G., Wardhani, N.K., Gata, W.: The use of web-based application system redmine setting in project management. *ISIT 2010* **1**(1), 142–146 (2010)
- [9] Halli, E.H.: Apache JMeter. Packt Publishing Birmingham (2008)
- [10] Howden, W.E.: Functional program testing. *IEEE Transactions on Software Engineering* (2), 162–169 (1980)
- [11] Iso/iec 25001:2014 systems and software engineering - systems and software quality requirements and evaluation (square). Standard (Mar 2014)
- [12] Iso/iec 25010:2011 systems and software engineering - systems and software quality requirements and evaluation (square) - system and software quality model. Standard (Mar 2011)
- [13] Mahajan, A.: Burp Suite Essentials. Packt Publishing Ltd (2014)
- [14] Masse, M.: REST API design rulebook: designing consistent RESTful web service interfaces. " O'Reilly Media, Inc." (2011)
- [15] Matei, O., Erdei, R., Delinschi, D., Andreica, L.: Data based message validation as a security cornerstone in loose coupling software architecture. In: Computational Intelligence in Security for Information Systems Conference. pp. 214–223. Springer (2021)
- [16] Shahin, M., Babar, M.A., Zhu, L.: Continuous integration, delivery and deployment: a systematic review on approaches, tools, challenges and practices. *IEEE Access* **5**, 3909–3943 (2017)

