

SEPTEMBER 15, 2025

PROPUNEREA TEHNICĂ ȘI METODOLOGIA DE IMPLEMENTARE  
CONFORM CERINȚELOR CAIETULUI DE SARCINI

ASCENTA CLOUD

## Table of Contents

|                                                            |    |
|------------------------------------------------------------|----|
| Înțelegerea obiectivelor proiectului .....                 | 2  |
| Înțelegerea sarcinilor specifice.....                      | 2  |
| Metodologie de implementare .....                          | 4  |
| Analiza și Planificarea .....                              | 5  |
| Definirea viziunii produsului și a roadmap-ului .....      | 6  |
| Definirea Backlog-ului de Produs .....                     | 6  |
| Definirea Planului de Lansare .....                        | 7  |
| Alte obiective de planificare.....                         | 8  |
| Planificarea Iterațiilor .....                             | 10 |
| Organizarea ședințelor zilnice Scrum .....                 | 12 |
| Execuția Sarcinilor .....                                  | 13 |
| Review și Demo.....                                        | 13 |
| Refinement-ul Backlog-ului de Produs .....                 | 14 |
| Servicii de Mentenanță și Garanție.....                    | 15 |
| Servicii de Mentenanță în Timpul Fazei de Dezvoltare ..... | 15 |
| Servicii de Mentenanță după Faza de Dezvoltare .....       | 16 |
| Servicii de Garanție .....                                 | 16 |
| Aspecte Generale privind Managementului Incidentelor ..... | 17 |
| Organizarea și Personalul implicat.....                    | 18 |
| Echipa de Proiect a Clientului .....                       | 18 |
| Echipa de Proiect a Consultantului .....                   | 18 |
| Echipa de Suport a Consultantului.....                     | 20 |

## Înțelegerea obiectivelor proiectului

SI „e-Integritate” este o suită de aplicații web orientate pe trei contururi distincte – Portalul public al declarațiilor, Portalul declarantului și Portalul Autorității – care operează într-o arhitectură client-server multinivel și se integrează cu platforme guvernamentale (MPass, MSign, MNotify, MConnect), registre de stat (RSP, RSUD, Registrul Fiscal, Registrul de Stat al Transportului, RBI) și alte sisteme conexe (ex. SI „e-Management”). Această arhitectură asigură autentificarea și semnarea, jurnalizarea evenimentelor critice, interacțiuni TLS/VPN și schimb de date cu sisteme externe, menținând în același timp o separare clară a rolurilor (Subiect, Inspector, Decident, Administrator).

Din punct de vedere tehnologic, soluția actuală este dezvoltată pe Yii Framework (Php) și PostgreSQL, într-un model pe straturi (UI / Logică / Acces date) – opțiuni care privilegiază reutilizarea codului, integritatea datelor și întreținerea facilă a componentelor. Cerințele de performanță prevăd timp mediu de răspuns  $\leq 3$  secunde la încărcare nominală și suport pentru >1000 utilizatori autorizați.

Funcțional, soluția acoperă: (i) Gestiunea subiecților declarațiilor, (ii) Depunerea declarațiilor (autentificare, completare, semnare/expediere), (iii) Controlul declarațiilor/sesizărilor (examinare prealabilă, inițiere control, acte de constatare, monitorizare în instanță), (iv) Generarea rapoartelor și (v) administrarea sistemului. Pe conturul public, sunt disponibile căutarea/filtrarea datelor și publicarea actelor privind rezultatele controlului.

Interoperabilitatea și securitatea sunt preocupări transversale: integrarea cu servicii guvernamentale, jurnalizarea, notificările și conexiunile securizate TLS sunt explicite, iar Caietul de sarcini solicită suplimentar întărirea securității aplicațiilor (colector.ani.md; declarant.ani.md; portal-declaratii.ani.md) conform celor mai bune practici.

## Înțelegerea sarcinilor specifice

Conform înțelegerii noastre în urma analizei Caietului de sarcini este nevoie de operat modificari la o serie de functionalități descrise in Cap.5.

1. Gestiunea sesizărilor (Portalul Autorității).
  - Implementarea unui algoritm de repartizare aleatorie către minim 2 inspectori cu încărcare minimă pe regimul de control vizat; extinderea jurnalizării cu evenimentele de creare/editare sesizare, creare dosar, listă inspectori, decizia de repartizare (inclusiv numere/ID-uri relevante).
  - Ajustarea meniului pentru rolul Operator Decident, cu acces dedicat la „Declarații” și „Registrul de Stat al Interdicțiilor”.
2. Depunerea declarațiilor (Portalul declarantului).
  - Adaptarea structurii de date și a funcționalităților la Legea nr.130/2021; menținerea fluxului de autentificare/completare/semnare.

- Validări client+server după tip/domeniu (CF 02.08), localizarea formularului în rusă (CF 02.09) și remedierea erorilor istorice în procesul de semnare/previzualizare PDF (CF 02.10).
3. Controlul sesizărilor / Verificarea declarațiilor.
    - Separarea fluxurilor: (a) controlul sesizărilor (încălcări 01–04) vs. (b) verificarea DAIP (05), cu stări/decizii și documente aferente (procese-verbale, demersuri, sesizări către alte autorități).
    - Optimizări UX/ergonomie pentru calendarul din dashboard (CF 03.01) și ordonarea/etichetarea registrelor de dosare (CF 03.02) cu reguli privind termenele.
  4. Lista actelor spre publicare.
    - Flux operator „Publicare”: vizualizare/descărcare act, atașare PDF prelucrat/hașurat și trimitere în listă; sincronizare cu compartimentul „Acte privind rezultatele controlului” din Portalul public; jurnalizarea fiecărei salvări/trimiteri.
  5. Registrul de Stat al Interdicțiilor (RSI).
    - Automatizări pentru verificarea statutului actelor de constatare după expirarea termenului de contestare; notificarea Operatorului Inspector; activare/anulare interdicție și sincronizare cu pagina web ANI.
    - Flux de rectificare cu aprobarea Operatorului Decident; trasabilitate completă; dashboard cu categorii de evenimente; interdicțiile nu se șterg, ci se anulează pe bază de documente aprobate.
  6. Portalul public al declarațiilor.
    - Ajustarea mecanismelor de raportare la noile structuri de date și operaționalizarea „Acte privind rezultatele controlului”.
  7. Interpelare MConnect.
    - Ajustări ale web-serviciilor pentru SI „Protecția socială” (CNAS) cu prezentarea datelor în tabele exportabile (Word/Excel/PDF) și actualizarea mesajelor privind temeiul legal/scopul pentru interpelarea RSP în profilul subiectului.
  8. Rol Președinte/Vicepreședinte ANI.
    - Creare rol cu nivel de acces superior: vizualizare sesizări repartizate aleatoriu, dosare/acte și rapoarte statistice.
  9. Generarea rapoartelor.
    - Revizuire logică de calcul la noile structuri de date și ajustarea formatelor de extragere.

10. Soluții suplimentare de securitate informațională.

- Consolidarea securității pentru toate aplicațiile e-Integritate, conform recomandărilor și celor mai bune practici de securitate cibernetică.

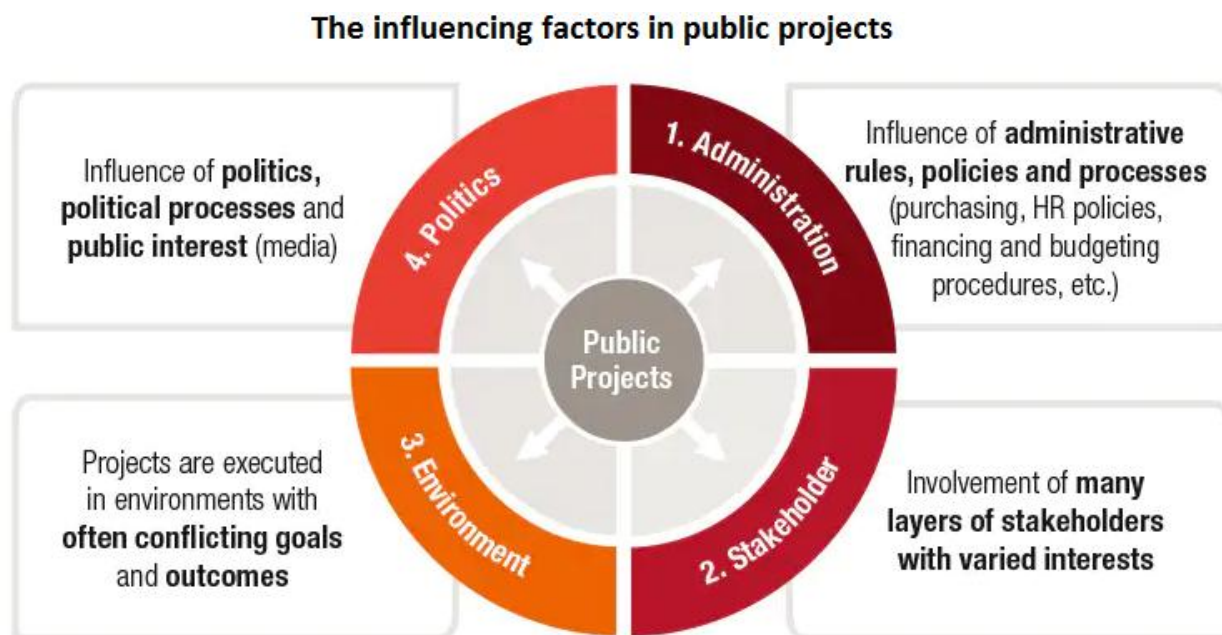
11. Migrarea conținutului bazei de date curente.

- Migrarea și actualizarea datelor (ex. nomenclatoare, structuri derivate), cu menținerea integrității și a compatibilității cu fluxurile ajustate.

## Metodologie de implementare

Pe baza experienței noastre cu proiecte de tip waterfall și agile, am concluzionat că pentru aplicațiile guvernamentale inovatoare și serviciile digitale, o abordare agilă sau hibridă este mai eficientă. Statisticile arată că proiectele de e-guvernare și digitalizare eșuează frecvent din cauza cerințelor neclare și a lipsei de comunicare. Aceste proiecte implică de obicei actori interdependenți multipli (ex. furnizori de servicii, furnizori de date și regulatori) care trebuie să coopereze pentru a stabili și menține standarde comune. Coordonarea între aceste părți este crucială pentru ca soluția să funcționeze eficient.

O strategie esențială pentru reconstruirea sistemelor tehnologice vechi este adoptarea mentalității „mai mic este mai bun”. Aceasta implică împărțirea sarcinilor complexe și interdependente în unități de lucru realizabile, ceea ce este fundamental pentru dezvoltarea modernă a software-ului în sectorul privat și public. Chiar și în proiectele cu cerințe tehnice predefinite, solicitările de modificare și provocările de acceptare sunt inevitabile din cauza complexității și influențelor externe.



source: <https://www.pwc.ch/en/insights/public-sector/are-public-projects-doomed-to-fail>

Prin urmare, preferăm să dezvoltăm în iterații, prezentând funcționalitățile dezvoltate părților interesate relevante în timpul fazei de dezvoltare pentru a obține feedback cât mai curând posibil, a ne adapta și a evita eforturile tardive de refacere. Abordarea agilă poate îmbunătăți semnificativ calitatea proiectelor, permițând construirea rapidă și adaptarea pe baza feedback-ului regulat. Cu toate acestea, aceasta necesită un angajament considerabil din partea tuturor celor implicați. Echipa de proiect, inclusiv părțile interesate, ar trebui să se concentreze pe aplicarea a ceea ce funcționează cel mai bine pentru sarcină, mai degrabă decât pe adoptarea strictă a metodologiilor.

### **Riscuri și Provocări ale abordării Agile în proiectele de digitalizare a guvernului:**

- Mentalitate de Tip Waterfall: Procesele tradiționale waterfall sunt adânc înrădăcinate în instituțiile guvernamentale.
- Buget și Termene: Echilibrarea receptivității la schimbări cu bugetele și termenele fixe poate fi o provocare.
- Pierderea Viziunii de Ansamblu: Concentrarea pe rezultatele sprintului poate duce la pierderea viziunii generale a proiectului.
- Probleme de Calitate: Accentul pe viteză poate compromite calitatea produsului din cauza atenției reduse asupra analizei și refactorizării.
- Implicarea Clientului: Nivelurile ridicate de participare a clientului cerute în proiectele agile pot fi dificile pentru unele instituții.

Deși agilul oferă multe beneficii, nu toate proiectele sunt potrivite pentru a fi implementate eficient cu practici pure agile. După coordonarea inițială a backlog-ului, activitățile suplimentare de analiză și dezvoltare pot continua în paralel pentru a scurta perioada de dezvoltare a proiectului. De asemenea, în faza mai avansată de dezvoltare pot fi adăugate activitățile de migrare, documentare și instruire.

### **Analiza și Planificarea**

O concepție greșită comună despre Agile este credința că nu implică planificare. Această neînțelegere provine din interpretarea greșită a valorilor Manifestului Agile. Manifestul Agile subliniază „răspunderea la schimbare în locul urmării unui plan”, nu „în loc de urmărirea unui plan”.

În practică, aceasta nu înseamnă că un proiect Agile nu are un plan. Înseamnă că planul este suficient de flexibil pentru a permite ajustări valoroase și importante, recunoscând că schimbarea este inevitabilă.

Echipa Agile trebuie să înceapă proiectul cu o viziune clară asupra a ceea ce clientul dorește să obțină ca rezultat. Având în vedere că nu toate detaliile sunt cunoscute de la început, cel mai mare risc este începerea dezvoltării având înțelegeri diferite ale scopului și ipotezelor proiectului. Proiectele complexe au numeroase interdependențe și dacă dezvoltatorii nu înțeleg pe deplin pentru ce codează, acest lucru duce de obicei la erori de integrare. Activitățile de analiză și proiectare a cerințelor asigură o bază solidă pentru dezvoltare. În proiectele Agile, arhitectura se schimbă în timp, dar o înțelegere clară a rezultatului dorit va minimiza riscurile de eșec. De aceea, considerăm important să alocăm timp pentru înțelegerea cerințelor, analiza sistemului și proiectare.

Livrarea valorii și calității maxime pentru oameni și societate în cel mai scurt timp posibil necesită o înțelegere fundamentală a sistemului construit. La începutul proiectului, se așteaptă ca Product Owner-ul desemnat să prezinte echipei ciclul de viață al proiectului și complexitatea acestuia, împărțind munca de sus în jos în faze și sub-faze. Echipa și clientul convin asupra livrabilelor într-un stadiu timpuriu al dezvoltării, oferind o mai bună înțelegere a rezultatelor așteptate.

Componentele backlog-ului trebuie definite pe baza cerințelor detaliate și a interacțiunilor fluxului de lucru. Prioritățile și dependențele sunt, de asemenea, atribuite în această etapă pentru a facilita planificarea muncii pentru sprinturile viitoare de dezvoltare.

## **Definirea viziunii produsului și a roadmap-ului**

Responsabilitatea Clientului este de a prezenta viziunea produsului, prioritățile și roadmap-ul inițial. Viziunea proiectului trebuie discutată în timpul întâlnirii de kick-off și actualizată sau confirmată pe parcursul ciclului de viață al proiectului. Acest lucru asigură că Clientul și întreaga echipă Agile au o înțelegere comună a scopului proiectului, priorităților, abordării și Produsului Minim Viabil (MVP). Când atât Clientul, cât și echipa de dezvoltare înțeleg obiectivele proiectului și motivele constrângerilor, vor căuta împreună soluții pentru a face munca potrivită la momentul potrivit.

MVP-ul ajută la evitarea irosirii timpului și banilor, începând cu o versiune simplă a produsului final care rezolvă cele mai importante probleme. Este vorba despre obținerea unui produs „suficient de bun” și livrarea acestuia „cât mai devreme posibil” utilizatorilor.

Ținând cont de specificul serviciilor publice, există câteva momente care nu trebuie ignorate atunci când se definește viziunea și prioritățile:

- Nevoile utilizatorilor finali
- Cerințele și/sau constrângerile legale
- Disponibilitatea integrărilor necesare

Rolul Product Owner-ului este crucial aici. Chiar și un proiect perfect din punct de vedere tehnic poate ajunge „pe un raft” dacă nu este conform cu legea sau nu este acceptat de utilizatorii finali. Ca și consultanți, putem face sugestii pe baza experiențelor anterioare, dar Clientul trebuie să se asigure că viziunea este consultată cu utilizatorii finali, ține cont de realitatea tehnică și legală și este definită în mod corespunzător.

## **Definirea Backlog-ului de Produs**

Definirea backlog-ului este unul dintre cele mai importante obiective ale Analizei și Planificării. Rolurile principale implicate în aceste activități sunt Product Owner-ul și experții în domeniu ai Clientului. Activitățile principale includ:

- Definirea Epics-urilor și Prioritizarea: Identificarea caracteristicilor mari și prioritizarea lor pe baza valorii și importanței pentru afacere.
- Descompunerea Epics-urilor în User Stories: Descompunerea epics-urilor în user stories mai mici și gestionabile care pot fi dezvoltate incremental.
- Descrierea User Stories: Furnizarea descrierilor detaliate ale fiecărei user story, conturând cerințele și criteriile de acceptare.

- Definirea și Descrierea Enablerilor Tehnici: Identificarea și descrierea enablerilor tehnici care susțin dezvoltarea user stories.
- Analiza Dependențelor între Elementele Backlog-ului de Produs: Examinarea și înțelegerea dependențelor între diferitele elemente din backlog pentru a asigura o secvențiere și prioritizare corespunzătoare.
- Definirea Priorităților Backlog-ului: Stabilirea priorităților pentru elementele din backlog pentru a ghida eforturile de dezvoltare în mod eficient.

Al 10-lea principiu al Manifestului Agile afirmă că "Simplitatea—arta de a maximiza cantitatea de muncă neefectuată—este esențială," încurajând echipa Agile să se concentreze pe dezvoltarea de soluții simple pentru livrarea timpurie, fără caracteristici suplimentare. Ca rezultat, Clientul primește un produs funcțional devreme, permițându-i să învețe rapid ce funcționează și ce nu, și ce doresc utilizatorii și ce nu doresc.

Obiectivul este să se abordeze cerințele principale și deciziile de proiectare în această fază inițială, lăsând detaliile și elementele cu prioritate mai mică să fie lucrate în paralel cu faza de dezvoltare iterativă.

## Definirea Planului de Lansare

Planul de lansare servește ca un obiectiv pe termen mediu în jurul căruia echipa se poate mobiliza. Pentru o analiză eficientă a sistemului și o planificare a efortului, este benefic să se discute un plan preliminar de lansare din etapele incipiente ale proiectului. În proiectele Agile, acest plan nu este strict obligatoriu, deoarece fiecare sprint livrează un produs funcțional, iar clientul are flexibilitatea de a decide când să îl implementeze în producție. Cu toate acestea, având un plan de lansare ajută echipa să înțeleagă nevoile de implementare și le permite să sugereze sarcini tehnice relevante pentru backlog, cum ar fi documentația, instruirea, migrațiile, ajustările și testările dedicate.

Un plan de lansare bine definit oferă mai multe avantaje:

- Aliniere: Asigură că echipa de dezvoltare și părțile interesate au o înțelegere comună a direcției și a cronologiei proiectului.
- Focalizare: Ajută la prioritizarea caracteristicilor și a sarcinilor critice pentru lansarea următoare, asigurându-se că echipa rămâne concentrată pe cele mai importante livrabile.
- Managementul Riscurilor: Permite o mai bună identificare și gestionare a riscurilor asociate cu implementarea, cum ar fi dependențele și potențialele blocaje.
- Pregătire: Asigură că toate activitățile necesare pre-implementare, cum ar fi instruirea și documentația, sunt planificate și executate în timp util.

Un alt aspect crucial al planului de lansare este efortul necesar pentru suportul post-implementare. Activitățile de dezvoltare trebuie coordonate cu activitățile de suport, asigurându-se că disponibilitatea personalului pentru ambele scopuri este luată în considerare atunci când se estimează capacitatea echipei. Această coordonare ajută la prevenirea conflictelor de resurse și asigură suport continuu pentru produsul implementat.

Product Owner-ul joacă un rol esențial în acest proces. Ca individ responsabil pentru gestionarea Backlog-ului de Produs și stabilirea priorităților conform planului de lansare, Product Owner-ul



trebuie să aibă autoritatea de a lua decizii privind prioritățile și cerințele de afaceri, mai ales atunci când apar contradicții și constrângeri. Această autoritate permite Product Owner-ului să:

- Rezolve Conflicte: Să abordeze rapid și să rezolve orice conflicte între diferite priorități sau cerințe.
- Adapteze Planurile: Să facă ajustările necesare planului de lansare pe baza feedback-ului și a circumstanțelor în schimbare.
- Asigure Alinierea: Să mențină alinierea între obiectivele proiectului și obiectivele de afaceri, asigurându-se că produsul livrat îndeplinește nevoile și așteptările clientului.

Prin integrarea unui plan de lansare bine gândit în fluxul de lucru al proiectului, echipa își poate gestiona mai bine eforturile, asigura implementări mai fluide și oferi suport eficient post-implementare, ducând în cele din urmă la un rezultat de proiect mai de succes.

### Alte obiective de planificare

Pe lângă activitățile principale de planificare, există și alte obiective critice care trebuie abordate pentru a asigura succesul proiectului. Aceste obiective includ:

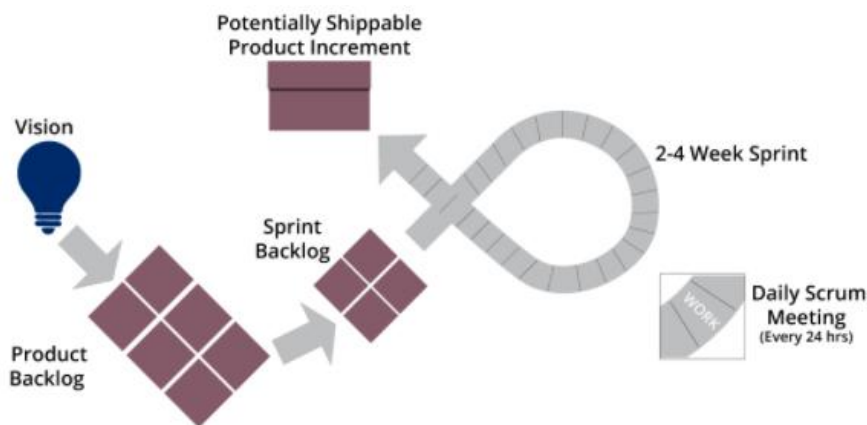
- Configurările mediului de proiect: Stabilirea și configurarea mediilor necesare de dezvoltare, testare și producție. Aceasta include configurarea serverelor, bazelor de date, pipeline-urilor de integrare continuă/livrare continuă (CI/CD) și a altor componente de infrastructură. Setările corespunzătoare ale mediului asigură că echipa poate lucra eficient și că tranziția de la dezvoltare la producție este fără probleme.
- Identificarea riscurilor proiectului: Efectuarea unei evaluări detaliate a riscurilor pentru a identifica potențialele riscuri care ar putea impacta proiectul. Aceasta implică evaluarea riscurilor tehnice, operaționale și de afaceri, precum și factorii externi. Odată identificate, aceste riscuri trebuie documentate, prioritizate și trebuie dezvoltate strategii de mitigare. Evaluările periodice ale riscurilor trebuie realizate pe tot parcursul ciclului de viață al proiectului pentru a asigura gestionarea proactivă a oricăror riscuri emergente.
- Dezvoltarea arhitecturii la nivel înalt: Proiectarea arhitecturii la nivel înalt a sistemului pentru a se asigura că îndeplinește cerințele proiectului și sprijină scalabilitatea, securitatea și mentenabilitatea. Aceasta implică definirea componentelor cheie, interacțiunile lor și structura generală a sistemului. Dezvoltarea arhitecturii la nivel înalt include și selectarea stivei tehnologice potrivite și asigurarea că aceasta se aliniază cu obiectivele și constrângerile proiectului.
- Abordarea și regulile de muncă: Stabilirea abordării și regulilor de muncă pentru a ghida execuția proiectului. Aceasta include definirea:
- Cadența și durata iterațiilor: Determinarea lungimii sprinturilor sau iterațiilor pentru a asigura un echilibru între livrarea rapidă și timpul suficient pentru dezvoltarea de calitate.
- Gestionarea Comunicării: Stabilirea protocolelor clare de comunicare, inclusiv întâlniri regulate (ex. stand-up-uri zilnice, planificarea sprinturilor, retrospective) și canale de comunicare (ex. email, chat, unelte de gestionare a proiectelor).

- Responsabilități: Definirea clară a rolurilor și responsabilităților fiecărui membru al echipei și al părților interesate pentru a asigura responsabilitatea și un flux de lucru eficient.
- Cerințe de Raportare: Stabilirea tipurilor și frecvenței rapoartelor (ex. rapoarte de progres, rapoarte de risc, actualizări de stare) pentru a menține transparența și a ține toate părțile interesate informate despre progresul proiectului.

Abordând aceste obiective suplimentare de planificare, echipa de proiect poate crea o fundație solidă pentru procesul de dezvoltare, asigurându-se că toate aspectele critice sunt luate în considerare și gestionate eficient. Această abordare cuprinzătoare a planificării ajută la atenuarea riscurilor, îmbunătățirea comunicării și creșterea performanței generale a proiectului.

## Dezvoltare Software

În general, urmăm abordarea Scrum în proiectele noastre Agile. Scrum este un cadru care promovează dezvoltarea iterativă și incrementală a produselor. Această concentrare pe iterație oferă numeroase beneficii asociate cu Scrum și Agile, inclusiv colaborarea cu clientul, inovația, identificarea și mitigarea timpurie a riscurilor și livrarea rapidă a software-ului.



Abordarea iterativă permite livrarea sistemului în etape, modulele fiind expediate imediat ce ating stadiul de Produs Minim Viabil (MVP). Acest lucru permite echipei să continue lucrul la îmbunătățirile viitoare, în timp ce livrează componente funcționale devreme. Testarea și revizuirea în timpul dezvoltării oferă vizibilitate timpurie a soluției pentru client, permițând acțiuni corective timpurii în cazul în care există probleme de înțelegere a cerințelor sau a scopului.

O iterație are o durată și o cadență fixă, convenite de Product Owner și echipă, în funcție de planul de lansare, distribuția optimă a efortului și roadmap.

Pe baza experienței noastre anterioare, am constatat că sprinturile mai scurte oferă avantajul unui feedback mai rapid, ceea ce este deosebit de util la începutul proiectului. Cu toate acestea, sprinturile mai scurte pot fi mai puțin eficiente în ceea ce privește proporția timpului petrecut în ședințe față de timpul de dezvoltare. Într-un sprint de două săptămâni, timpul alocat pentru planificare și revizuiră rămâne similar, lăsând puțin timp pentru dezvoltare și restrângând capacitatea echipei de a realiza sarcini mai complexe. Prin urmare, atunci când este planificată o

muncă complexă și cerințele sunt clare, recomandăm sprinturi de trei săptămâni. Această durată extinsă permite mai mult timp de dezvoltare concentrat, menținând în același timp bucle regulate de feedback.

Pentru a ne alinia și mai bine cu Caietul de Sarcini, procesul nostru de dezvoltare include următoarele îmbunătățiri:

- **Implicarea Continuă a Clientului:** Implicarea regulată a clientului asigură alinierea cu așteptările acestuia și încorporarea la timp a feedback-ului. Revizuirile și demonstrațiile sprintului permit clientului să vadă progresul și să ofere feedback regulat.
- **Documentație Cuprinzătoare:** Fiecare iterație include actualizarea documentației relevante pentru a se asigura că toate părțile interesate au acces la informații actuale despre progresul proiectului și funcționalitatea sistemului.
- **Asigurarea Calității:** Încorporăm teste riguroase în fiecare etapă a dezvoltării, inclusiv teste automate, teste de performanță și teste de acceptare a utilizatorilor. Această abordare ajută la identificarea și rezolvarea problemelor devreme, asigurând livrabile de înaltă calitate.
- **Managementul Riscurilor:** Prin împărțirea proiectului în iterații mai mici și gestionabile, putem identifica și atenua riscurile devreme. Evaluările periodice ale riscurilor și actualizările registrului de riscuri sunt părți integrale ale procesului nostru de dezvoltare.
- **Scalabilitate și Flexibilitate:** Abordarea iterativă permite scalabilitatea și flexibilitatea, acomodând schimbările în cerințe și priorități fără întreruperi semnificative. Acest lucru este deosebit de important pentru proiectele complexe de digitalizare guvernamentală care pot necesita adaptări la reglementările și nevoile părților interesate în evoluție.

Prin adoptarea acestor practici, ne asigurăm că procesul nostru de dezvoltare este robust, transparent și aliniat atât cu principiile Agile, cât și cu cerințele specifice din Caietul de Sarcini. Această abordare ne permite să livrăm soluții de înaltă calitate în mod eficient, menținând în același timp o colaborare puternică cu clientul.

## **Planificarea Iterațiilor**

Al 3-lea principiu al Manifestului Agile afirmă: "Software-ul funcțional este livrat frecvent, săptămâni mai degrabă decât luni."

Product Owner-ul și echipa de dezvoltare planifică sprinturile pe baza Backlog-ului de Produs existent. În timpul ședinței de planificare, clientul și membrii echipei de dezvoltare revizuiesc backlog-ul pentru a determina elementele de lucru gata să fie completate în următoarea iterație.

Echipele Agile nu dezvoltă caracteristicile produsului aleatoriu. Product Owner-ul definește obiectivul care trebuie atins în cadrul intervalului de timp al iterației. Această planificare are loc la începutul fiecărui sprint, cu participarea obligatorie a Product Owner-ului, Scrum Master-ului și echipei de dezvoltare.

Odată ce obiectivul iterației este stabilit, echipa Agile se angajează să finalizeze elementele de lucru. Aceste elemente de lucru angajate sunt considerate convenite de client și trebuie dezvoltate de echipa Agile exact așa cum sunt descrise în backlog. Echipa poate refuza să includă elemente de lucru în iterația planificată dacă acestea nu sunt suficient de clare. În astfel de cazuri, clientul și echipa pot conveni să transfere elementele neclare într-o altă iterație sau poate fi organizată o

ședință dedicată de rafinare pentru a analiza și coordona, eventual implicând părți interesate și experți relevanți.

Definiția Agreată de Ready (DoR) specifică cum trebuie să arate un element din backlog sau o user story pentru a fi acceptată în iterația următoare. Criteriile INVEST formulate de expertul în Extreme Programming Bill Wake funcționează adesea bine ca Definiție de Ready:

- I (Independent): Elementul este autoconținut și poate fi progresat fără a depinde de un alt Element din Backlog-ul de Produs (PBI) sau resursă externă.  
Ex: User story pentru acceptarea aplicațiilor online depinde de crearea user story pentru aplicația online și nu ar fi planificată pentru aceeași iterație.
- N (Negotiable): Elementul lasă loc pentru discuții privind implementarea optimă.  
Ex: O user story care prescrie că validarea formatului de email trebuie setată la nivelul back-end nu este negociabilă. În schimb, o user story care descrie nevoia de a evita formatele incorecte de email permite dezvoltatorilor să găsească cea mai bună soluție pentru a satisface cerința (validare back-end sau front-end).
- V (Valuable): Elementul aduce o valoare clară pentru client.  
Ex: Dacă clientul respinge rar solicitantii, dezvoltarea unei opțiuni de respingere nu este valoroasă.
- E (Estimable): Elementul are o dimensiune relativă față de alte PBI-uri.  
Ex: O user story care implică integrarea cu un sistem extern care nu este disponibil nu este estimabilă și nu ar fi inclusă în iterația următoare până când ghidurile de integrare sunt analizate și accesul este oferit.
- S (Small): Elementul este suficient de mic pentru a fi estimat cu precizie și planificat într-o casetă de timp pentru iterație.  
Ex: O user story care include atât opțiunile de acceptare, cât și cele de respingere pentru funcțiile de procesare a aplicațiilor nu este suficient de mică dacă implică două fluxuri de lucru diferite.
- T (Testable): Elementul are criterii clare de acceptare, permițându-i să fie testat.  
Ex: O user story fără criterii de acceptare nu este testabilă. Criteriile de acceptare precum „opțiunea de respingere trebuie să fie ușor de utilizat” sau „datele solicitate sunt încărcate rapid” nu sunt testabile.

Indiferent de intervalul de timp, echipa Agile decide câte elemente de lucru să își asume pentru o iterație. Nimeni (clientul, Product Owner-ul sau Scrum Master-ul) nu ar trebui să pună presiune pe echipă pentru a-și asuma prea multe, deoarece acest lucru poate duce la resentimente, neîncredere și muncă de calitate scăzută. Abordările tradiționale de planificare necesită identificarea, secvențierea și estimarea fiecărei sarcini individuale pentru a calcula cronologia. Planificarea Agile estimează elementele de lucru la un nivel înalt pentru a determina cât poate fi finalizat într-o iterație.

Estimarea timpului de finalizare poate părea o ghicire, dar anumite principii pot face estimarea mai precisă. Ca bună practică, echipa se bazează pe succesele anterioare pentru a estima efortul. De exemplu, dacă echipa a livrat X număr de user stories în iterația anterioară, se angajează să livreze X număr de user stories de complexitate similară în iterația următoare.

Un factor cheie al succesului este ca clientul și membrii echipei să recunoască și să accepte că estimările se vor îmbunătăți pe parcursul proiectului. După completarea mai multor iterații, echipa

Agile poate măsura viteza lor, care este timpul mediu petrecut pentru implementarea unui element de lucru. Această măsurătoare ajută la rafinarea estimărilor viitoare și la îmbunătățirea acurateții planificării.

## Organizarea ședințelor zilnice Scrum

Al 12-lea principiu al Manifestului Agile afirmă: "Echipa reflectă în mod regulat asupra modului de a deveni mai eficientă și ajustează în consecință."

În fiecare sprint, echipa de dezvoltare organizează ședințe zilnice scurte (15-20 minute) cunoscute sub denumirea de scrums zilnice. Aceste ședințe servesc pentru a coordona prioritățile zilei, a aborda orice constrângeri și a discuta îmbunătățirile necesare pentru a atinge obiectivul sprintului.

Aspecte cheie ale ședințelor zilnice scrum includ:

- **Coordonarea Zilnică:** Fiecare membru al echipei împărtășește ceea ce a lucrat în ziua precedentă, ceea ce plănuiește să lucreze astăzi și orice obstacole pe care le întâmpină. Acest lucru asigură că toată lumea este pe aceeași pagină și poate ajuta la rezolvarea promptă a problemelor.
- **Transparență și Responsabilitate:** Scrums zilnice promovează transparența și responsabilitatea între membrii echipei. Prin împărtășirea progresului și a provocărilor, membrii echipei se responsabilizează pe ei înșiși și pe ceilalți pentru angajamentele lor.
- **Rezolvarea Imediată a Problemelor:** Orice obstacole sau impedimente identificate în timpul ședinței pot fi abordate imediat, fie de către echipă, fie escaladate la Scrum Master pentru rezolvare. Acest lucru ajută la menținerea proiectului pe drumul cel bun și previne întârzierile.
- **Îmbunătățire Continuă:** Scrumul zilnic este o oportunitate pentru echipă de a reflecta asupra proceselor lor și de a face mici ajustări incrementale pentru a îmbunătăți eficiența și eficacitatea. Acest lucru este în conformitate cu principiul Agile de îmbunătățire continuă.
- **Focalizare și Aliniere:** Scrumul zilnic asigură că echipa rămâne concentrată pe obiectivul sprintului și că eforturile lor sunt alinate. Ajută la prevenirea creșterii necontrolate a cerințelor și asigură că toată lumea lucrează către aceleași obiective.

Participarea la ședințele zilnice scrum include de obicei Product Owner-ul, Scrum Master-ul și echipa de dezvoltare. Părțile interesate și alte persoane interesate pot participa ca observatori dacă sunt invitați, dar ședința este destinată în principal echipei de dezvoltare pentru a-și coordona munca.

## Execuția Sarcinilor

Procesul de dezvoltare se bazează pe al 9-lea principiu al Manifestului Agile: "Atenția continuă la excelența tehnică și un design bun."

La începutul sprintului, user stories asumate din backlog-ul sprintului sunt atribuite membrilor echipei, analizate și împărțite în sarcini de executat (TO DO). Clientul este responsabil pentru reprezentarea nevoilor produsului și asigurarea că membrii echipei de proiect sunt conștienți de acestea și că munca lor se concentrează pe aceste nevoi. Cu toate acestea, echipa decide cum vor fi împărțite activitățile între membri și cum va fi organizată munca lor. Echipa are control asupra procesului de livrare, mai degrabă decât fiind direcționată de un manager.

În timpul dezvoltării, sarcinile vor fi marcate ca IN PROGRESS. Echipa scrum lucrează independent pe parcursul întregului sprint. Dezvoltatorii deschid și estimează sarcinile individual sau în timpul întâlnirilor zilnice. Sarcinile simple pot fi estimate pe măsură ce sunt deschise, dar estimarea timpului pentru sarcinile care necesită integrarea cu eforturile altor dezvoltatori poate fi dificilă. Dacă există o nepotrivire între modul în care două componente interacționează, reproiectarea acelor piese poate lua timp și complexitate semnificative, care sunt greu de măsurat înainte ca munca să fie finalizată.

Echipa de dezvoltare trebuie să decidă între ei cum să realizeze cel mai bine livrabilele. Scrum Master-ul și Product Owner-ul nu trebuie să microgestioneze sarcinile dezvoltatorilor, deoarece acest lucru poate fi demotivant și neproductiv.

Odată ce o caracteristică este considerată DONE, aceasta este livrată Product Owner-ului pentru acceptare. Bugs descoperite în timpul testării sunt remediate în fiecare sprint. În managementul tradițional și predictiv al proiectelor, managerul de proiect este responsabil pentru calitatea proiectului. În Agile, echipa este responsabilă pentru livrarea unui produs de înaltă calitate.

Echipa Agile folosește Definiția de Gata (Definition of Done) pentru a măsura completitudinea unui element de lucru livrat. Criteriile care trebuie îndeplinite pentru ca un element de lucru să fie considerat "gata" sunt convenite de echipă și de client, stabilind un prag clar de calitate și asigurând o înțelegere comună a ceea ce înseamnă "gata". Cu toate acestea, fiecare problemă trebuie să îndeplinească criteriile individuale de acceptare. În timp ce atât criteriile de acceptare, cât și Definiția de Gata determină când un element de lucru este finalizat, există diferențe în modul și locul în care sunt utilizate. Criteriile de acceptare sunt unice pentru user story, în timp ce Definiția de Gata este universală și se referă la fiecare element de lucru livrat.

Echipa ar trebui să ia în considerare și că pregătirea demo-ului necesită timp, inclusiv curățarea datelor, configurarea unui mediu de demo și/sau a fluxurilor de lucru. Aceste activități fac parte din sarcinile din backlog-ul sprintului.

## Review și Demo

Al 7-lea principiu al Manifestului Agile afirmă: "Software-ul funcțional este principalul indicator de progres."

La sfârșitul fiecărui sprint, Product Owner-ul și echipa efectuează un review al sprintului pentru a examina pachetul de lucru pregătit în timpul sprintului și pentru a evalua disponibilitatea acestuia pentru o posibilă implementare în producție.

În timpul ședinței de review al sprintului, Product Owner-ul explică Backlog-ul de Produs, iar echipa de dezvoltare prezintă munca efectuată, răspunzând la orice întrebări care pot apărea. Product Owner-ul verifică munca în raport cu "Definiția de Gata," oferă feedback pentru a se asigura că incrementul livrat îndeplinește nevoia de afaceri și fie acceptă munca, fie identifică modificările necesare.

Prezența altor părți interesate ale proiectului poate fi benefică, deoarece oferă o oportunitate de a primi feedback asupra progresului și rezultatelor proiectului. Recomandăm participarea părților interesate (de exemplu, utilizatori finali, reprezentanți ai autorităților care vor interacționa cu sistemul) la review-ul sprintului. Cu toate acestea, responsabilitatea de a decide și de a invita participanții cvați revine Product Owner-ului.

Product Owner-ul urmărește feedback-ul primit în timpul review-ului sprintului și îl încorporează în Backlog-ul de Produs pentru planificarea viitoare, după cum este necesar. Această buclă continuă de feedback asigură că proiectul rămâne aliniat cu nevoile și așteptările părților interesate.

Pentru review-urile sprinturilor finale, recomandăm participarea personalului tehnic care va fi responsabil pentru mentenanța sistemului. Implicarea lor asigură că echipa de mentenanță este familiarizată cu funcționalitatea sistemului și poate oferi suport fără probleme după implementare.

### **Refinement-ul Backlog-ului de Produs**

Rafinarea backlog-ului de produs este o activitate continuă care pregătește calea pentru sprinturile viitoare. Recomandăm cu tărie organizarea de ședințe periodice de rafinare, de obicei cu o durată de până la 2 ore.

În timpul acestor sesiuni, Elementele Backlog-ului de Produs (PBI-uri) sunt prezentate și explicate echipei de dezvoltare de către Product Owner sau un expert în domeniu. Echipa de dezvoltare poate pune întrebări, primi răspunsuri imediate și ajuta la descrierea mai detaliată a elementului. De asemenea, pot ajuta la împărțirea elementelor mai mari în sarcini mai mici și mai gestionabile, facilitând dezvoltarea și livrarea unei valori mai mari.

Când un PBI este suficient de clar, echipa de dezvoltare procedează la estimare. În prezent, există multe instrumente de estimare online disponibile, iar toți membrii echipei de dezvoltare sunt implicați în procesul de estimare. PBI-urile estimate de echipa de dezvoltare sunt atribuite puncte de poveste (story points). Elementele estimate la mai mult de 13 puncte de poveste nu pot fi incluse într-un sprint și trebuie împărțite în sarcini mai mici.

Aspecte cheie ale rafinării eficiente a backlog-ului de produs includ:

- Ședințe Regulate: Organizarea periodică a sesiunilor de rafinare asigură că backlog-ul este actualizat și prioritarizat continuu, menținându-l aliniat cu obiectivele proiectului și nevoile părților interesate.

- **Proces Colaborativ:** Implicarea atât a Product Owner-ului, cât și a echipei de dezvoltare în procesul de rafinare promovează colaborarea și asigură că toată lumea are o înțelegere clară a sarcinilor.
- **Feedback Imediat:** Permite echipei de dezvoltare să pună întrebări și să primească feedback imediat ajută la clarificarea cerințelor și reduce neînțelegerile.
- **Detaliere Incrementală:** Adăugarea treptată a detaliilor în PBI-uri pe măsură ce se apropie de implementare asigură că echipa are informațiile necesare pentru a continua fără a fi supraîncărcată cu detalii prea devreme.
- **Estimare și Prioritizare:** Estimarea PBI-urilor și prioritizarea acestora pe baza valorii de afaceri și a efortului necesar asigură că echipa se concentrează pe livrarea celor mai critice caracteristici mai întâi.

Prin menținerea unui backlog de produs bine rafinat, echipa poate asigura sesiuni de planificare a sprinturilor eficiente și o execuție eficientă a sarcinilor, ducând în cele din urmă la livrarea cu succes a proiectului.

### Servicii de Mentenanță și Garanție

Serviciile de mentenanță în timpul fazei de dezvoltare includ remedierea defectelor identificate de client. Această abordare proactivă asigură că orice problemă este abordată prompt, menținând stabilitatea și fiabilitatea sistemului pe parcursul procesului de dezvoltare.

Aspecte cheie ale serviciilor noastre de mentenanță includ:

- **Rezolvare Promptă a Defectelor:** Abordarea rapidă a defectelor raportate pentru a minimiza întreruperile și a asigura progresul continuu.
- **Suport Continu:** Oferirea de suport continuu pentru a rezolva orice probleme operaționale care pot apărea, asigurându-se că sistemul rămâne funcțional și eficient.
- **Colaborare cu Clientul:** Lucrul îndeaproape cu clientul pentru a identifica, raporta și remedia defectele, asigurând o comunicare transparentă și o rezolvare eficientă a problemelor.

### Servicii de Mentenanță în Timpul Fazei de Dezvoltare

Serviciile de mentenanță sunt furnizate în timpul dezvoltării sistemului prin remedierea defectelor identificate de client.

*Service level agreements:*

| Cod | Severitate | Descriere                                                                                                                                                       | Timp răspuns | Timp recuperare* |
|-----|------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------|------------------|
| 1   | Critic     | Sistemul este indisponibil pentru utilizarea publică pentru a recupera informații.                                                                              | 30 minute    | 2 ore            |
| 2   | Non-critic | Incidente Minore: Incidente care afectează funcționarea sistemului, dar care, dacă nu sunt rezolvate, vor afecta productivitatea și nu operațiunile de afaceri. | 1 oră        | 4 ore            |



|   |                    |   |                   |     |
|---|--------------------|---|-------------------|-----|
| 3 | Solicitare telefon | - | 10 minute         | N/A |
| 4 | Solicitare email   | - | 2 zile lucrătoare | N/A |

\*Timpul de Recuperare – se va opri din numărare atunci când incidentul raportat este blocat din cauza lipsei de informații, indisponibilității datelor pentru reproducere, lipsei accesului etc.

### Servicii de Mentenanță după Faza de Dezvoltare

De asemenea, serviciile de mentenanță se vor extinde pentru 3 luni după lansarea finală.

| Cod | Severitate       | Descriere                                                                                                                                                                     | Timp răspuns  | Timp recuperare* |
|-----|------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------|------------------|
| 1   | Critic           | Sistemul este indisponibil pentru utilizarea publică pentru a recupera informații.                                                                                            | 30 minute     | 2 ore            |
| 2   | Non-critic       | Incidente minore care afectează funcționarea sistemului, dar care, dacă nu sunt rezolvate, vor afecta productivitatea și nu operațiunile de afaceri.                          | 1 oră         | 4 ore            |
| 3   | Change request** | Funcționalitățile noi și modificările funcționalităților deja implementate identificate în perioada de garanție vor fi implementate în funcție de disponibilitatea bugetului. | La solicitare | La solicitare    |

\* Timpul de Recuperare – se va opri din numărare atunci când incidentul raportat este blocat din cauza lipsei de informații, indisponibilității datelor pentru reproducere, lipsei accesului, etc.

\*\* Change requests - vor fi analizate de Consultant și oferta va fi aprobată de Client, în funcție de disponibilitatea bugetului.

### Servicii de Garanție

Perioada de garanție a sistemului este de 6 (șase) luni calendaristice de la data acceptării sistemului de către client și implementării în mediul de producție.

#### Acorduri de Nivel de Serviciu (SLA)

Orice incidente raportate de client după lansare vor fi rezolvate de echipa noastră de suport conform Acordurilor de Nivel de Serviciu (SLAs) definite în Anexa 4, p.4.1.5 Garanție și Suport, și anume:

| Cod | Severitate | Descriere | Timp răspuns | Timp recuperare* |
|-----|------------|-----------|--------------|------------------|
|-----|------------|-----------|--------------|------------------|

|   |            |                                                                                                                                                      |           |       |
|---|------------|------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|-------|
| 1 | Critical   | Sistemul Informatic este indisponibil pentru utilizarea publică pentru a recupera informații.                                                        | 15 minute | 2 ore |
| 2 | Non-critic | Incidente minore care afectează funcționarea sistemului, dar care, dacă nu sunt rezolvate, vor afecta productivitatea și nu operațiunile de afaceri. | 1 oră     | 4 ore |

\* Timpul de Recuperare – se va opri din numărare atunci când incidentul raportat este blocat din cauza lipsei de informații, indisponibilității datelor pentru reproducere, lipsei accesului, etc.

### Aspecte Generale privind Managementul Incidentelor

În perioada de mentenanță și garanție, incidentele vor fi urmărite și gestionate utilizând Azure DevOps prin elemente de lucru de tip bug. La începutul perioadei de garanție, vom alocă o linie telefonică dedicată pentru a asigura o comunicare eficientă. Un raportor de incident poate fi clientul sau un specialist de suport din prima linie al consultantului.

Aspecte cheie ale managementului incidentelor includ:

- Sumar: O descriere concisă, dar detaliată a incidentului—ce s-a întâmplat, când și unde.
- Pași pentru Reproducerea Incidentului: Pași detaliați care capturează suficiente informații astfel încât ceilalți membri ai echipei să poată înțelege pe deplin defectul de cod. Acest lucru include acțiunile întreprinse pentru a găsi sau reproduce bug-ul și comportamentul așteptat. Pot fi prezentate capturi de ecran pentru a susține descrierea.
- Informații despre Sistem: Specificația mediului software și a build-ului unde a apărut bug-ul, inclusiv configurațiile relevante, cum ar fi versiunea sistemului de operare și a browserului.
- Prioritate: Grila de priorități definește urgența incidentului pe baza următoarelor metrici:
  - 1: Sistemul nu poate fi implementat fără rezolvarea cu succes. Necesită acțiune imediată.
  - 2: Sistemul nu poate fi implementat fără rezolvarea cu succes, dar nu trebuie abordată imediat.
  - 3: Rezolvarea elementului de lucru este opțională, în funcție de resurse, timp și risc.
- Severitate: Clasificarea incidentului în categorii precum critic, necritic sau cerere de schimbare.

Prin implementarea acestor practici structurate de management al incidentelor, ne asigurăm că toate problemele raportate sunt abordate sistematic și eficient, conducând la rezolvări mai rapide și o fiabilitate mai mare a sistemului. Această abordare asigură o comunicare clară, prioritizarea și rezolvarea incidentelor, îmbunătățind procesul general de mentenanță și suport.

## Organizarea și Personalul implicat

Manifestul Agile prioritizează forța de muncă umană în detrimentul proceselor și instrumentelor, încorporând al 5-lea principiu: "Proiectele sunt construite în jurul indivizilor motivați, care trebuie să fie de încredere."

Construirea sistemelor este, în mod inerent, o activitate socială. Furnizorii și clienții sunt parteneri în fluxul de valoare al dezvoltării. Într-un mediu colaborativ, construit pe o fundație pe termen lung de încredere, persoanele implicate interacționează pentru a construi sisteme mai bune.

### Echipa de Proiect a Clientului

Conform Caietului de Sarcini, Secțiunea 8.6, Clientul va desemna un Product Owner pentru proiect. Product Owner-ul va fi responsabil pentru:

- Analiza business și definirea backlog-ului de produs: Asigurarea că cerințele sunt clare și bine documentate.
- Furnizarea clarificărilor, accesului și documentației pentru echipa de dezvoltare: Facilitarea comunicării eficiente și accesului la resursele necesare.
- Definirea Obiectivelor Sprintului: Stabilirea obiectivelor clare pentru fiecare sprint.
- Aprobarea Roadmap-ului Produsului și Planului de Lansare: Supravegherea direcției strategice și a cronologiilor.
- Aprobarea documentelor legate de proiect: Asigurarea că toate documentele îndeplinesc standardele și cerințele.
- Facilitarea comunicării cu părțile interesate externe: Coordonarea cu personalul tehnic de la Agenția de Guvernare Electronică, Agenția pentru Servicii Publice, Serviciul Tehnologiei Informației și Securitate Cibernetică, etc.
- Implicarea personalului tehnic din serviciul/departamentul de suport al Clientului va fi un avantaj, deoarece aceștia vor prelua administrarea sistemului atunci când acesta va fi implementat în producție. Contribuția lor legată de modulul administrativ al sistemului va fi benefică.
- Including technical staff from the Client support service/department will be an advantage, as they will undertake system administration when the system is deployed into production. Their input related to the administrative module of the system will be beneficial.

### Echipa de Proiect a Consultantului

În Agile, fiecare membru al echipei este responsabil. Acest lucru îmbunătățește sentimentul de proprietate și responsabilitate, ceea ce, la rândul său, îmbunătățește productivitatea. Abordarea Agile oferă putere membrilor echipei să ia decizii și să gestioneze procesul de dezvoltare, încurajând inovația și eficiența.

Realizarea livrării rapide de valoare necesită luarea deciziilor descentralizate, reducând întârzierile și îmbunătățind fluxul dezvoltării produsului. Cu toate acestea, unele decizii strategice și globale justifică luarea deciziilor centralizate. Crearea unui cadru de luare a deciziilor fiabil este crucială pentru construirea eficientă a echipei. De obicei, Managerul de Proiect al Consultantului sau

Product Owner-ul este responsabil pentru coordonarea activităților proiectului, cum ar fi monitorizarea resurselor, depunerea rapoartelor și sarcinile organizaționale, fără a controla activitățile de dezvoltare, acestea fiind gestionate de echipa de dezvoltare conform principiilor Agile. Pentru a evita întreruperile echipei și comunicările greșite, Scrum Master-ul va gestiona comunicarea principală cu Product Owner-ul.

Rolul de Scrum Master va fi atribuit specialistului Asigurare a Calității (QA) care are expertiza și experiența necesare. Această persoană va asigura un mediu Agile eficient, ajutând echipa să colaboreze pentru a livra produse de calitate, eliminând obstacolele și protejând dezvoltatorii de întreruperi.

Pentru a atinge obiectivele sarcinii, echipa de proiect propusă include experți cu expertiză dovedită în analiza de afaceri și sistem, design UI, dezvoltare software, proiectarea și optimizarea bazelor de date, DevOps, testare și managementul proiectelor Agile.

Prin asigurarea unei organizații bine structurate și a unor roluri clar definite, echipa de proiect poate opera eficient, promovând un mediu de încredere, colaborare și îmbunătățire continuă.

| Expert                                 | Nume Expert | Responsabilități cheie                                                                                                                                                                                                                                                                                                                                                                                              |
|----------------------------------------|-------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Manager de proiect                     | Sergiu Robu | <ul style="list-style-type: none"> <li>- Planificarea și coordonarea activităților de proiect.</li> <li>- Gestionarea resurselor, bugetului și calendarului proiectului.</li> <li>- Asigurarea comunicării și raportării către părțile interesate.</li> <li>- Monitorizarea calității, riscurilor și respectării standardelor.</li> <li>- Coordonarea echipelor implicate (dezvoltare, testare, suport).</li> </ul> |
| Dezvoltator software / Lider de echipă | AU          | <ul style="list-style-type: none"> <li>- Analiza sistemului</li> <li>- Arhitectura sistemului</li> <li>- Proiectarea și dezvoltarea bazei de date</li> <li>- Integrarea sistemului</li> <li>- Dezvoltarea software (back end)</li> <li>- DevOps</li> </ul>                                                                                                                                                          |
| Dezvoltator Software                   | SC          | <ul style="list-style-type: none"> <li>- Analiza sistemului</li> <li>- Design și dezvoltare Web/UI</li> <li>- Dezvoltare software (back end, front end)</li> <li>- Integrarea sistemului</li> <li>- Testare unitară</li> <li>- DevOps</li> </ul>                                                                                                                                                                    |
| Software tester                        | OC          | <ul style="list-style-type: none"> <li>- Managementul defectelor</li> <li>- Dezvoltarea și execuția cazurilor de testare</li> <li>- Monitorizarea stării defectelor și confirmarea soluțiilor</li> </ul>                                                                                                                                                                                                            |

|  |  |                                          |
|--|--|------------------------------------------|
|  |  | - Facilitarea eliminării impedimentelor. |
|--|--|------------------------------------------|

Al 8-lea principiu al Manifestului Agile afirmă că: "Procesele Agile promovează dezvoltarea sustenabilă. Sponsorii, dezvoltatorii și utilizatorii ar trebui să fie capabili să mențină un ritm constant pe termen nelimitat.

Dezvoltarea sustenabilă se înțelege ca fiind necesitatea de a ritma munca astfel încât membrii echipei să nu devină supraobosiți sau să sufere de burnout.

O cultură agilă este susținută de transparența în comunicare. Prin urmare, comunicarea directă între Product Owner și dezvoltator ar trebui evitată (de exemplu, emailuri directe fără CC către Scrum Master). Recomandăm ca discuțiile privind caracteristicile și detaliile să fie documentate ca comentarii la user stories relevante în Azure, transparente și vizibile pentru toți membrii echipei.

Aspectele organizaționale și administrative ale contractului ar trebui adresate atât Scrum Master-ului, cât și/sau Directorului companiei.

### Echipa de Suport a Consultantului

Conform Caietului de Sarcini, înțelegem că este necesară o garanție a sistemului pentru o perioadă de 6 (șase) luni după acceptarea sistemului de către client.

Avem un Centru de Suport pentru Clienți al cărui scop principal este să asigure furnizarea serviciilor de mentenanță și suport post-implementare. Locația Centrului de Suport este Chișinău, Moldova. Centrul nostru de Suport este structurat pe 2 niveluri:

#### Nivel 1 - Suport de Primă Linie.

Specialistul de suport de primă linie este responsabil pentru analiza și categorisirea defectelor, oferirea unei soluții temporare (dacă este posibil). Dacă specialistul de primă linie nu poate oferi o soluție imediată pentru defectul raportat, acesta va escalada tichetul către suportul de a doua linie. Responsabil pentru suportul de primă linie va fi desemnat un tester din echipa de dezvoltare a proiectului.

#### Nivel 2 - Suport de a Doua Linie.

Inginerul de suport de a doua linie are toate instrumentele necesare pentru depanarea defectului și furnizarea soluției.

Suportul tehnic va fi furnizat prin 2 canale de comunicare:

- Linie telefonică, disponibilă între orele 9:00AM și 6:00PM (ora Chișinăului), de luni până vineri.
- Sistem de helpdesk online, disponibil 24/7.

Notă: în cazul incidentelor urgente (conform Nivelului de Severitate), persoana principală de contact va fi Managerul de Proiect.

|                                                                |            |                                                                    |
|----------------------------------------------------------------|------------|--------------------------------------------------------------------|
| Permiterea accesului la sistemul de helpdesk al Consultantului | Consultant | Cel târziu cu 1 săptămână înainte de data de începere a garanției. |
|----------------------------------------------------------------|------------|--------------------------------------------------------------------|

|                 |            |                                                                    |
|-----------------|------------|--------------------------------------------------------------------|
| Support hotline | Consultant | Cel târziu cu 1 săptămână înainte de data de începere a garanției. |
|-----------------|------------|--------------------------------------------------------------------|

De obicei, desemnăm o persoană de contact dedicată contractului (denumită Manager de Suport)