

Metodologia de Implementare și Asigurare a Serviciilor de Garanție, Mentenanță și Suport

Proiect: Servicii de dezvoltare a soluției complexe care include componentă tehnică și de program destinate creării cheilor publice și private, și creării, verificării, validării a serviciilor de încredere calificate

Esempla Systems

Contents

Organizare.....	3
Obiectivele proiectului.....	3
Componentele cheie	3
Domeniul de Aplicare al Proiectului	3
Faze de Implementare	3
Riscuri și provocări în proiectele de transformare digitală	4
Avantajele abordării hibride de implementare:.....	5
Metodologia folosită pentru realizarea serviciilor	5
Metodologia de realizare a proiectului	5
Etapa de planificare și analiză.....	6
Etapa de dezvoltare	7
Procesul de soluționare a incidentelor.	9
Mentenanță corectivă	10
Managementul cererilor de schimbare	11
Managementul calității.....	12
Abordarea tehnică de implementare	14
Personalul utilizat pentru realizarea serviciilor.....	16
Organizare și Personal	16
Responsabilitățile echipei de proiect a beneficiarului.....	16
Structura echipei Prestatorului.....	16
Disponibilitatea echipei	17
Structura echipei de suport.....	17
Planul de lucru pentru realizarea serviciilor	22
Sub-plan analiză și design	22
Sub-plan dezvoltare	23
Sub-plan instalare, configurare și integrare	24
Sub-plan de testare și asigurare a calității	25
Sub-plan tranziție și punere în funcțiune.....	27
Sub-plan de dezvoltare a capacității umane	28
Sub-plan de activități de mentenanță și suport tehnic.....	29
Sub-plan de instruire utilizatori.....	31
Sub-plan de comunicare	31
Sub-plan de management al problemelor	33
Sub-planul de management al riscurilor	34
Sub-plan de management al schimbărilor.....	35

Organizare

Obiectivele proiectului

Obiectivul principal al proiectului este dezvoltarea și implementarea unei infrastructuri complete de chei publice (PKI) care să permită crearea, gestionarea și verificarea semnăturilor electronice calificate pentru entități publice și private din Republica Moldova. Soluția va integra infrastructura existentă, respectând standardele internaționale precum ETSI, FIPS și eIDAS, asigurând cele mai înalte niveluri de securitate și conformitate legală.

Componentele cheie

- **Autoritatea de Certificare (CA):** Soluția include platforma Enterprise Java Beans Certificate Authority (EJBCA), care gestionează emiterea și ciclul de viață al certificatelor digitale. EJBCA oferă verificarea în timp real a revocării certificatelor utilizând OCSP și CRL și se integrează cu module de securitate hardware (HSM) pentru protejarea cheilor private.
- **Dispozitivul de Creare a Semnăturii Calificate (QSCD):** Cryptomatic Signer 6 va fi utilizat pentru generarea și stocarea securizată a cheilor private. Acesta este conform cu standardele EN 419 241-1 și EN 419 241-2 și suportă algoritmi criptografici RSA și ECDSA.
- **Portaluri pentru Utilizatori și Organizații:** Portaluri personalizate vor permite gestionarea securizată a documentelor, generarea și verificarea semnăturilor electronice, oferind autentificare multi-factor (MFA) pentru securitate sporită.
- **Integrare API:** Sistemul va oferi API-uri pentru a asigura interoperabilitatea cu platforme externe, utilizând standarde precum PKCS#11 și Cloud Signature Consortium (CSC).

Domeniul de Aplicare al Proiectului

Implementarea va acoperi configurarea întregii infrastructuri PKI, inclusiv componente hardware, software și de integrare. Soluția are scopul de a extinde infrastructura de servicii de încredere, oferind tranzacții digitale sigure pentru utilizări guvernamentale și comerciale.

Faze de Implementare

- **Proiectare și Planificare (Săptămânile 1-3):** Definirea cerințelor tehnice și a arhitecturii sistemului pe baza Caietului de Sarcini.
- **Configurarea Mediului (Săptămânile 4-5):** Configurarea mediului de producție și pre-producție, inclusiv configurarea HSM și CA.
- **Dezvoltare (Săptămânile 4-15):** Dezvoltarea componentelor personalizate și a portalurilor, configurarea QSCD și integrarea API-urilor.
- **Testare și Validare (Săptămânile 8-16):** Testarea aprofundată a funcționalităților sistemului, securității și conformității cu standardele solicitate.

- Implementarea în Producție (Săptămânile 14-16): Implementarea sistemului complet testat în mediul de producție.
- Instruire (Săptămânile 16-18): Furnizarea de instruire pentru administratori și utilizatori finali privind utilizarea, configurarea și mentenanța sistemului.

În abordarea noastră pentru atingerea obiectivelor ne vom baza pe:

- Expertiza semnificativă de consultanță și tehnică obținută în cei peste 15 ani de activitate în domeniul transformării digitale a sectorului public.
- Expertiza specifică de implementare a soluțiilor similare.
- Înțelegerea clară a viziunii guvernelor actuale asupra transformării digitale, ecosistemelor serviciilor publice și a stivei tehnologice, grație participării noastre în implementarea unor proiecte de nivel sistemic.

În baza pe experienței noastre cu implementări de tip Cascadă (Waterfall) și Agile, am concluzionat că pentru sistemele guvernamentale inovative și serviciile digitale, o abordare agilă sau hibridă este mai eficientă.

Statisticile arată că proiectele de transformare digitală din sectorul public eșuează adesea din cauza cerințelor neclare și a lipsei de comunicare. Chiar și în proiectele unde există cerințe tehnice predefinite, provocările legate de solicitările de schimbare și acceptanță în timpul procesului de implementare sunt inevitabile în cazul soluțiilor complexe. De aceea, preferăm să dezvoltăm în iterații, să prezentăm funcționalitățile dezvoltate părților interesate relevante în timpul fazei de dezvoltare, pentru a obține feedback cât mai curând posibil, să ne adaptăm și să evităm efortul de refacere tardivă. Abordarea agilă poate îmbunătăți calitatea proiectelor, dar nu este o baghetă magică. Echipa de proiect, inclusiv părțile interesate, nu ar trebui să fie mai preocupate de adoptarea metodologiilor decât de aplicarea a ceea ce funcționează cu adevărat pentru proiect.

Abordarea agilă poate îmbunătăți calitatea proiectelor, deoarece permite echipei să construiască rapid și să se adapteze pe baza feedback-ului regulat. Menită să fie un proces de împuternicire, în esență, Agile oferă o metodologie promițătoare pentru finalizarea mai rapidă a proiectelor și un proces de dezvoltare mai eficient. Dar aceasta cere un angajament considerabil din partea tuturor celor implicați în proiect, mai ales din partea Beneficiarului.

Riscuri și provocări în proiectele de transformare digitală

- Gândirea și procesele de tip Waterfall sunt adânc înrădăcinate în instituțiile guvernamentale.
- Dificultatea de a combina abordarea de răspuns la schimbări cu un buget și termene fixe.
- Focusul pe rezultatele sprint-urilor poate duce la pierderea viziunii de ansamblu.
- Calitatea produsului poate scădea din cauza concentrării pe livrarea rapidă (și mai puțin pe analiză și refactorizare).
- Clienți foarte implicați care participă în proiect aproape zilnic poate constitui o problemă pentru unele instituții.

Așadar, deși Agile oferă multe beneficii, nu toate proiectele sunt potrivite pentru a fi implementate eficient cu practici Agile. Această metodologie funcționează bine atunci când sunt dezvoltate produse software, dar există riscul ca Agile pur să nu funcționeze bine atunci în cazul unui proiect complex de transformare digitală. Acesta este frecvent cazul proiectelor în care există cerințe semnificative de integrare și sisteme terțe moștenite. Datorită diferitelor circumstanțe externe (de exemplu, disponibilitatea sistemelor terțe, schimbări în legislație, conformitatea cu deciziile și procesele organizațiilor/ sistemelor terțe, etc.) care nu pot fi controlate de echipa de dezvoltare și chiar de Beneficiar, poate fi nevoie să se analizeze și să se planifice mai mult în avans.

În același timp, ar putea fi dificil să se implice plenar angajații STISC ca experți sectoriali (pe domenii de interes), așa cum ar fi necesar pentru o implementare Agile pură. De aceea, utilizarea tehnicilor Agile pentru a îmbunătăți rezultatele dezvoltării, menținând în același timp controalele și structura organizațională văzute în instituțiile guvernamentale, poate fi o alternativă.

Având în vedere cele menționate mai sus, recomandăm aplicarea unei abordări hibride care să combine abordarea de dezvoltare Agile cu structura iterativă din Waterfall. Implicarea experților pe domenii de interes și coordonarea backlog-ului pot fi planificate într-un interval de timp cunoscut.

Avantajele abordării hibride de implementare:

- Fazele proiectului pot rula în secvență sau în paralel, fără a trebui să aștepte procedural finalizarea fazei anterioare; această abordare oferă un anumit nivel de flexibilitate pentru distribuirea resurselor și efortului de implementare.
- Permite reacția adecvată la schimbările de cerințe.
- Mai mult focus pe analiză și coordonarea proiectării proiectului.
- Livrarea iterativă facilitează dezvoltarea capacităților și transferul operaționalizării sistemului.

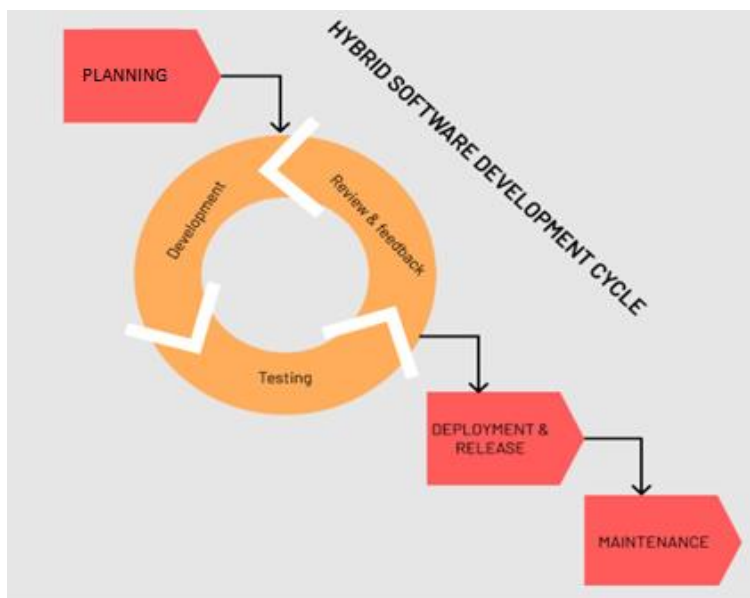
După o coordonare inițială a backlog-ului, activitățile ulterioare de analiză și dezvoltare ar putea fi realizate în paralel pentru a scurta durata de dezvoltare a proiectului. De asemenea, într-o fază de dezvoltare mai avansată, în paralel ar putea fi adăugate activitățile de migrare, documentare și instruire.

Metodologia folosită pentru realizarea serviciilor

Metodologia de realizare a proiectului

Metodologia hibridă de livrare a proiectului presupune o combinație de elemente a metodologiilor adaptive (agile) și predictive (cascadă).

Etapele de planificare și analiză, și tranziție se vor realiza predictiv, iar etapa de dezvoltare, testare și acceptanță adaptiv. Echipa de dezvoltare va dezvolta componentele software incremental, și va primi feedback continuu din partea beneficiarului în cadrul sedințelor de revizuire și sesiunilor de planificare.



Etapa de planificare și analiză

În cadrul acestei etape, adițional activității de analiză prezentată în caietul de sarcini, se vor efectua următoarele sub-activități:

- Activitatea de analiză va acoperi un coeficient de 70% din detalii, lasând restul de 30% să fie investigate și detaliate pe parcursul perioadei de dezvoltare, când anumite subiecte devin mai clare atât pentru prestator, cât și pentru beneficiar.
- Definirea backlog-ului în baza raportului de analiză. Backlog-ul va conține sarcini business (user story-uri) create de către analistul de business, cu suportul Beneficiarului (dacă va fi cazul). Aceste sarcini vor acoperi cerințele funcționale.

De asemenea, backlog-ul va conține și sarcini tehnice care vor acoperi cerințele nefuncționale, sarcini legate de infrastructură, migrare și testare care se vor crea de către arhitect, coordonatorul tehnic și coordonatorul de testare.

Pentru ambele componente ale sistemului informatic se va crea un singur backlog, care va permite gestiunea mai eficientă a dependențelor și o colaborare mai eficientă între echipe.

- Identificarea incrementelor (versiunilor) sistemului informatic în baza constrângerilor tehnice și de integrare, dependențelor existente și priorităților business, cât și planul de livrare a versiunilor sistemelor (release plan). Se vor organiza ședințe de planificare a incrementelor unde se va decide scopul fiecărui increment care urmează să fie livrat.

- Actualizarea planului de implementare (Gantt) care va conține activitățile la un nivel înalt ceea ce va permite urmărirea generală a progresului și dependențelor, fără a intra în detalii, care la un moment dat devin greu de gestionat și de urmărit, mai ales ca în etapa de dezvoltare se va utiliza un board Kanban care va conține sarcini detaliate.

Etapa de dezvoltare

Activitatea de dezvoltare se va executa în baza unui board de tip Kanban și prin aplicarea metodei adaptive de dezvoltare – SCRUM.

Astfel, vor fi aplicate principiile SCRUM, după cum urmează:

- echipa de dezvoltare va lucra în sprint-uri:
 - *Durata unui sprint*: 4 săptămâni, o astfel de durată fiind argumentată de existența unui backlog suficient de detaliat (a se vedea etapa de planificare și analiză).
 - *Cadența sprint-ului* va fi de miercuri (prima zi de sprint) până marți (ultima zi de sprint). O astfel de cadență s-a dovedit a fi cea mai potrivită unde activitățile finale de sprint și pregătirea pentru prezentare nu se va face cu un decalaj de 2 zile cum este în cazul cadenței luni-vineri.
- planificarea sprint-ului:
 - Scop: selectarea sarcinilor din backlog care urmează să fie executate în cadrul sprint-ului astfel încât în termenul definit - să fie livrat incrementul sistemelor informatice.
 - Durată: maxim 3 ore.
 - Cadența: în prima zi de sprint.
 - Participanți: echipa de dezvoltare.
 - Livrabile: backlog-ul sprint-ului.
- ședințele zilnice de progress (stand-up meetings):
 - Scop: identificare progresului, a problemelor și riscurilor
 - Durată: maxim 15 minute.
 - Cadența: în fiecare zi.
 - Participanți: echipa de dezvoltare.
 - Livrabile: actualizare registrului de riscuri și probleme.
- ședințele de clarificare:
 - Scop: prezentarea backlog-ului, sesiunea de întrebări și răspunsuri, ajustarea backlog-ului / sarcinii.
 - Durată: maxim 2 ore.
 - Cadența: o dată pe săptămână într-o zi fixată, sau de mai multe ori la necesitate.
 - Participanți: echipa de dezvoltare, opțional pot fi invitați și reprezentanții beneficiarului, în funcție de subiectele discutate.
 - Livrabil: ajustarea backlog-ului.

Deși ședințele de clarificare nu reprezintă un eveniment Scrum, considerăm important ca acestea să fie planificate și realizate pentru a livra cât mai multe cunoștințe echipei de dezvoltare, care se va putea concentra pe activitățile de programare și testare.

- ședințele de prezentare (review):
 - Scop: prezentarea funcționalităților dezvoltate în cadrul sprint-ului.
 - Durata: maxim 2 ore.
 - Cadența: în ultima zi a sprint-ului.
 - Participanți: echipa de dezvoltare, reprezentării beneficiarului.
 - Livrabil: solicitări de modificare.
- ședința de retrospectivă:
 - Scop: identificarea problemelor procedurale și soluții pentru a fi rezolvate.
 - Durata: maxim 2 ore.
 - Cadența: în ultima zi a sprint-ului.
 - Participanți: echipa de dezvoltare.
 - Livrabil: ajustarea procesului de lucru.

Atât în cadrul ședinței de planificare a sprintului, cât și ulterior în cadrul ședinței zilnice de progres, fiecare expert va avea asociat sarcini din backlog, în baza cărora se va analiza progresul și eventuale blocaje, riscuri, probleme.

La finalul dezvoltării unei versiuni de sistem care urmează să fie livrată conform planului de livrare, echipa de dezvoltare va prezenta beneficiarului un demo al sistemului informatic.

În ziua următoare, după prezentarea demo, beneficiarul și prestatorul vor avea o sesiune de planificare și feedback pentru a revizui planul pentru următorul increment și identifica modificările care sunt necesare.

Pentru fiecare versiune de sistem prestatorul va pregăti livrabilele specificate pentru activitatea 2 și 3 în caietul de sarcini. Se vor livra adițional și notificări de instalare, unde se vor specifica versiunea sistemului, mediul unde este instalat, lista funcționalităților, lista defectelor rezolvate (opțional).

Testarea de acceptanță

Testarea de acceptanță se va realiza pentru fiecare increment (versiune) livrată conform planului de livrare definit în cadrul etapei de analiză și planificare. Această abordare va eficientiza procesul de validare a sistemului de către Beneficiar care va avea loc etapizat, pe un volum de funcționalități mai mic, ceea ce permite focusarea pe detalii, o analiză mai detaliată a defectelor și solicitărilor de modificare. De asemenea, permite integrarea mai rapidă a modificărilor, și ulterior punerea în producție a sistemelor în timpul specificat în caietul de sarcini.

Totuși, o testare de acceptanță finală va fi necesară pentru a confirma integritatea proceselor, care se va realiza în baza unor fluxuri de lucru de bază. De asemenea, în cadrul etapei de planificare și analiză se vor stabili nivelul ratei de trecere în baza cărora se vor lua deciziile de punere în producție sau remediere. Rata de trecere se va

calcula în baza cazurilor de testare care au fost executate cu succes. O propunere inițială ar fi următoare:

Grad de prioritate	Descriere	Rata de trecere
Major	Reprezintă cazuri de testare asociate fluxurilor de baza ale sistemului	100%
Mediu	Reprezintă cazuri de testare asociate unor funcționalități adiacente și fluxuri alternative	60%
Minor	Reprezintă niste cazuri de testare asociate funcționalităților care nu afectează în nici un mod buna funcționare a sistemului.	20%

De asemenea, se va lua în considerație și numărul de defecte care nu au fost rezolvate:

Grad de severitate	Număr permis
Major	5
Mediu	10
Minor	15

Procesul de soluționare a incidentelor.

Un incident / defect reprezintă o variație dintre rezultatul aștept și cel obținut în utilizarea sistemului. În perioada de dezvoltare, incidentele și problemele apărute în funcționarea sistemului se vor raporta după cum urmează:

- în cadrul testării interne se vor deschide sarcini de tip incident în backlog.
- persoanele autorizate ale beneficiarului vor raporta incidentele / defectele în cadrul unei aplicații de tip help-desk prin înregistrarea tichetelor. În cazuri critice, incidentele / defectele se vor semnala prin email, telefon, ulterior fiind create tichete de către Prestator. Modelul de raportare a incidentului / problemei, datele de contact vor fi furnizate în cadrul etapei de planificare și analiză.

Un tichet va conține un singur incident / defect, cu detalierea necesară pentru a fi reprodus de către Prestator. Aceste defecte vor fi incluse în ședințele de triaj.

Prestatorul va numi o persoană din echipa de testate cu rol de manager de defecte care va fi responsabilă de gestiunea incidentelor / defectelor raportate și raportarea aferentă. Totodată, managerul de defecte va asigura organizarea ședințelor de triaj a incidentelor și problemelor raportate pentru a identifica dacă incidentul / problema este un defect sau o solicitare de modificare, și a clarifica detalii, a seta prioritatea și severitatea incidentului/problemei.

Ședințele de triaj se vor organiza cel puțin o dată pe săptămână, cu o durată de cel mult 1 oră.

Incidentele / defectele raportate vor conține cel puțin:

- Sumar - o scurtă descriere a incidentului / problemei: ce s-a întâmplat, unde și când.
- Descrierea – descrierea pașilor și o detaliere suficientă pentru ca echipa de dezvoltare să înțeleagă problema, să fie capabilă să reproducă incidentul / defectului. Pot fi atașat și imagini pentru o mai bună înțelegere.
- Componentă afectată – se va indica care componentă este afectată de producerea incidentului.
- Nivelul incidentului / defectului - ex. la nivelul aplicației sau a bazei de date.

- Mediul – se va indica mediul unde s-a manifestat problema / incidentul.
- Prioritate – prioritatea incidentului / defectului se va stabili conform matricei:
 - 1 – sistemul nu poate fi livrat fără o soluție confirmată aferentă defectului / problemei, necesită o reacție imediată.
 - 2 - sistemul nu poate fi livrat fără o soluție confirmată aferentă defectului / problemei, dar nu necesită o reacție imediată.
 - 3 – se va soluționa în funcție de disponibilitatea resurselor, timpului, riscurilor.
- Severitatea unui incident se va identifica în baza matricei de mai jos:
 - Major – o funcționalitate vitală / critică nu este implementată, erori operaționale unde rezultatul obținut este diferit de cel așteptat, manifestându-se continuu sau repetat și care duce la blocarea operațiunilor și/sau pierderi de date.
 - Mediu – erori care nu sunt critice sau vitale pentru operabilitatea sistemului, acestea pot fi ușor simulate (manifestându-se continuu).
 - Minor – erori care țin de experiența de utilizare a sistemelor.
- Mod de rezolvare – se va indica cauza și soluția propusă.
- Alți parametri conform Caietului de sarcini se vor înregistra automat de către aplicație.

Ciclul de viață al incidentului / defectului:

- Înregistrat – este un incident / defect nou care va fi analizat în ședințele de triaj.
- Aprobat – incidentul / defectul este confirmat în cadrul ședinței de triaj, detaliile sunt suficiente pentru rezolvare.
- În curs de rezolvare – incidentul / defectul sunt în curs de soluționare de către echipa de dezvoltare.
- Rezolvat – soluția pentru remedierea incidentului / defectului este furnizată
- Închis – la închiderea unui incident/problemă se va indica motivul:
 - Soluționat – dacă soluția furnizată este confirmată de către raportor
 - Duplicat – dacă există un tichet/sarcină care descrie același incident
 - Refuzat – incidentul / defectul raportat nu reprezintă un defect.

Dacă soluția este livrată, dar nu este confirmată de către raportor în decursul a 5 zile lucrătoare, incidentul / defectul se consideră automat soluționat.

Mentenanță corectivă

Conform caietului de sarcini, sistemul informatic va beneficia de o perioadă de mentenanță corectivă de cca. 3 luni și o garanție de 2 ani de la acceptanța finală a sistemului.

Prestatorul va numi un manager de suport care va avea responsabilitatea să gestioneze incidentele/defectele și care va fi persoana de contact pentru Beneficiar.

Incidentele/defectele se vor raporta prin:

- Aplicație de tip help-desk, la care vor avea acces persoanele autorizate ale Beneficiarului. Disponibilitate sistem – 24/7.
- Hot line – un număr de telefon la care se pot raporta incidentele critice. Disponibilitate 5 zile din 7, 9:00 – 18:00.

- Email – o adresă dedicată de suport.

Incidentele / defectele raportate prin hot-line / email vor fi înregistrate obligatoriu în aplicația de tip help-desk.

Incidentele / defectele se vor raporta conform structurii prezentate în perioade de dezvoltare și vor avea același ciclu de viață.

Timpul de reacție și de rezolvare vor corespunde schemei SLA prezentate în cadrul Caietului de sarcini.

La instalarea noii versiuni a sistemului pe mediul de testare și ulterior producție se va crea și o notă de instalare care va include numărul versiunii, lista defectelor rezolvate, noi funcționalități (dacă este cazul).

Livrabilele vor corespunde listei prezentate în caietul de sarcini.

Managementul cererilor de schimbare

O schimbare reprezintă adăugarea, ștergerea sau îmbunătățirea scopului inițial planificat și aprobat, care poate avea impact asupra bugetului, calendarului de lucru, calității.

Necesitatea de a realiza o schimbare a scopului poate apărea atât în cadrul etapei de dezvoltare, cât și ulterior, la lansarea în producție a sistemului în cadrul etapei de mentenanță evolutivă. Indiferent de etapa la care va apărea necesitatea de a realiza o schimbare, se va urma același proces.

O schimbare va parcurge următoarele etape:

1. Depunerea cererii de schimbare. Oricine poate iniția o schimbare, care se va transmite părților conform canalelor de comunicare ale proiectului, ceea ce va asigura coordonare internă a schimbării. Canalele și regulile de comunicare se vor agreea în etapa de planificare și analiză. Schimbare propusă trebuie să fie suficient de detaliată astfel încât să fie clară. Cererea de schimbare se va completa în baza unui model agreat în cadrul etapei de analiză și planificare.
2. Cerere de schimbare se va înregistra în Registru, unde i se va acorda și un număr unic de referință.
3. Prestatorul va analiza schimbarea solicitată, dacă cererea nu este clară se vor solicita detalii adiționale.
4. Prioritizarea cererii de schimbare se va realiza de către beneficiar și Prestator. De asemenea, se va lua decizia dacă este necesară întocmirea unui Raport de analiză suplimentar sau nu.
5. Prestatorul va evalua impactul schimbării asupra scopului, bugetului, calității și termenului de implementare, cât și riscurile asociate schimbării. În rezultat va crea o ofertă de implementare care va fi transmisă Beneficiarului prin canalele de comunicare ale proiectului și conform regulilor de comunicare agreeate. De asemenea, se va decide și termenul de livrare a schimbării.

6. Aprobarea / respingerea cererii conform nivelelor de aprobare specificate în Caietul de sarcini. Se vor implementa doar cererile de schimbare care au fost aprobate.

Managerul de proiect din partea Prestatorului este responsabil de gestiunea Registrului de cererilor de schimbare. Dacă o cerere de schimbare a fost aprobată, Prestatorul va crea și o sarcină în backlog, în funcție de prioritate, echipa de dezvoltare va implementa schimbarea.

Managementul calității

Dezvoltând sisteme informatice de stat, atragem o atenție deosebită calității din perspectiva a 2 direcții:

- Calitatea proiectului (managementul proiectului)
- Calitate sistemului informatic livrat

Calitatea proiectului

Calitatea proiectului se va realiza prin intermediul proceselor de planificare, implementarea procedurilor (asigurarea calității) și controlul calității.

În cadrul etapei de planificare se vor descrie procesele proiectului care țin de:

- regulile de comunicare, canalele de comunicare, se va detalia diferența dintre comunicarea oficială și neoficială.
- organizarea, liniile de raportare și escaladare a problemelor, datele de contact ale persoanelor cheie.
- se va detalia modul de întocmire și aprobare a documentelor, ciclul lor de viață.
- se vor aproba modele documentelor, registrelor, rapoartelor.
- se vor indica persoanele responsabile.

Asigurarea calității este în responsabilitatea fiecărui participant al proiectului care se va asigura prin:

- urmărirea proceselor descrise în planul de implementare.
- stocarea documentelor astfel încât să fie accesibile participanților proiectului.
- elaborarea și revizuirea documentelor conform standardelor indicate în planul de implementare.
- realizarea rapoartelor.

Controlul calității se va realiza prin intermediul sesiunilor de audit. Auditul poate fi inițiat de către Prestator sau beneficiar la orice etapă a proiectului. Subiect de audit pot fi:

- Documentele aprobate.
- Registrele (de exemplu: de documente, riscuri, probleme, cereri de schimbare, etc.).
- Planul de implementare.

În urma desfășurării auditului se pot propune schimbări care vor duce la îmbunătățirea și eficientizarea proiectului. La fel, se pot propune sesiuni de instruire a membrilor proiectului privind procesele descrise în planul de implementare.

Calitatea sistemului informatic

Calitatea sistemului informatic care urmează să fie livrat se va asigura prin intermediul a 3 procese:

- Planificare calității
- Asigurarea calității
- Controlul calității

Planificarea calității presupune planificarea testării sistemului per fiecare nivel de testare – funcțională și non-funcțională. Astfel, pentru fiecare nivel de testare se vor realiza planul de testare care va include: scenariile de testare, datele de test, lista funcționalităților care nu pot fi testate și motivul, etc.

Asigurarea calității sistemului informatic se va asigura prin:

- Testarea funcțională și non-funcțională a sistemului informatic, cu livrarea de rapoarte per fiecare nivel de testare cu descrierea rezultatelor.
- Elaborarea și actualizarea continuă a documentației privind sistemul informatic (de ex. manuale de utilizare și administrare, proceduri aferente sistemului informatic, și alte documente).
- Asigurarea versionării și copiilor de rezervă.
- Instruirea utilizatorilor sistemului.
- Adicional, calitatea sistemului va fi asigurată de o echipa profesionistă cu experiență, de procesul de dezvoltare care prevede runde de feedback atât intern în echipă , cât și din partea beneficiarului.

Controlul calității

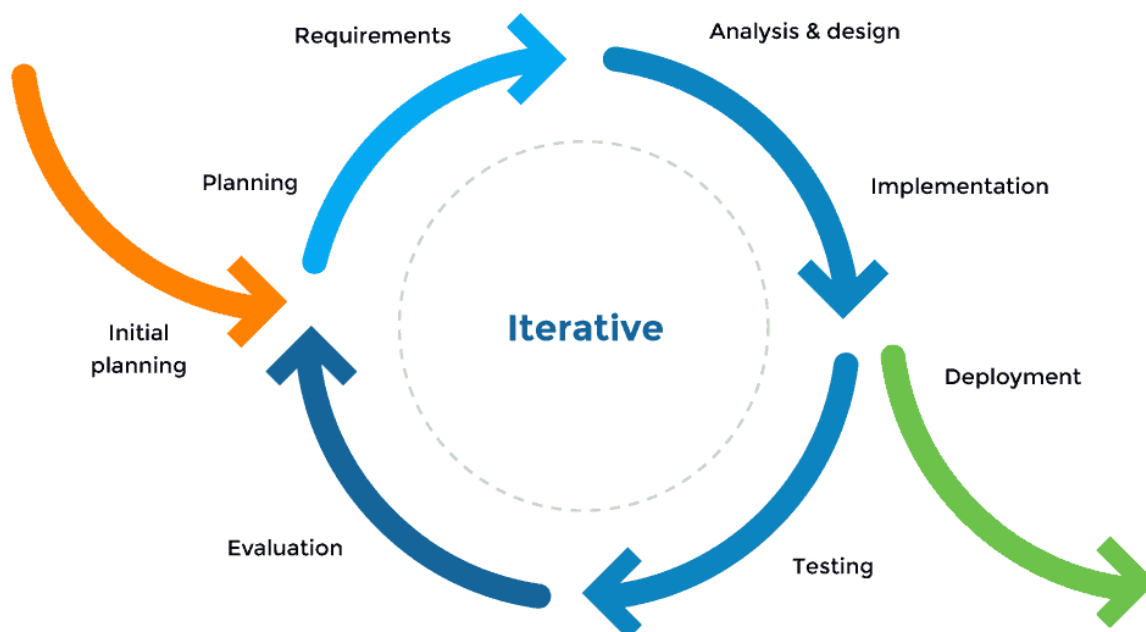
După punerea în producție a sistemului informatic calitatea tot rămâne a fi un aspect important. Astfel, vom asigura controlul calității prin:

- Inspectarea log-urilor, indicilor de performanță a sistemului și propunerea de soluții pentru îmbunătățirea performanței, proceselor.
- Corectarea defectelor apărute în baza SLA-ului definit și asigurând disponibilitatea sistemului de de tip help-desk 24/7.
- Actualizarea continuă a documentației pentru a reflecta ultimele modificări (mai ales în cadrul introducerii de noi funcționalități ca urmare a implementării schimbărilor).

Abordarea tehnică de implementare

Ciclul de viață al dezvoltării software (SDLC)

Întreg procesul de implementare a proiectului va fi unul iterativ. Procesul iterativ începe cu o implementare simplă a unui subset al cerințelor software și îmbunătățește iterativ versiunile evolutive până când întregul sistem este implementat. La fiecare iterație, se fac modificări de design și se adaugă noi capacități funcționale. Ideea de bază din spatele acestei metode este dezvoltarea unui sistem prin cicluri repetate (iterative) și în porțiuni mai mici la un moment dat (incrementale).



În acest model incremental, întreaga cerință este împărțită în diverse construcții (build-uri). În timpul fiecărei iterații, modulul de dezvoltare parcurge fazele de cerințe, design, implementare și testare. Fiecare versiune ulterioară a modului adaugă funcționalitate la versiunea anterioară.

Cheia pentru utilizarea cu succes a unui ciclu de viață de dezvoltare software iterativ este validarea riguroasă a cerințelor și verificarea și testarea fiecărei versiuni a software-ului față de aceste cerințe în cadrul fiecărui ciclu al modelului. Pe măsură ce software-ul evoluează prin cicluri succesive, testele trebuie repetate și extinse pentru a verifica fiecare versiune a software-ului.

În cadrul fiecărui increment (iterație) procesul de dezvoltare software se va baza pe principiile Scrum și DevOps.

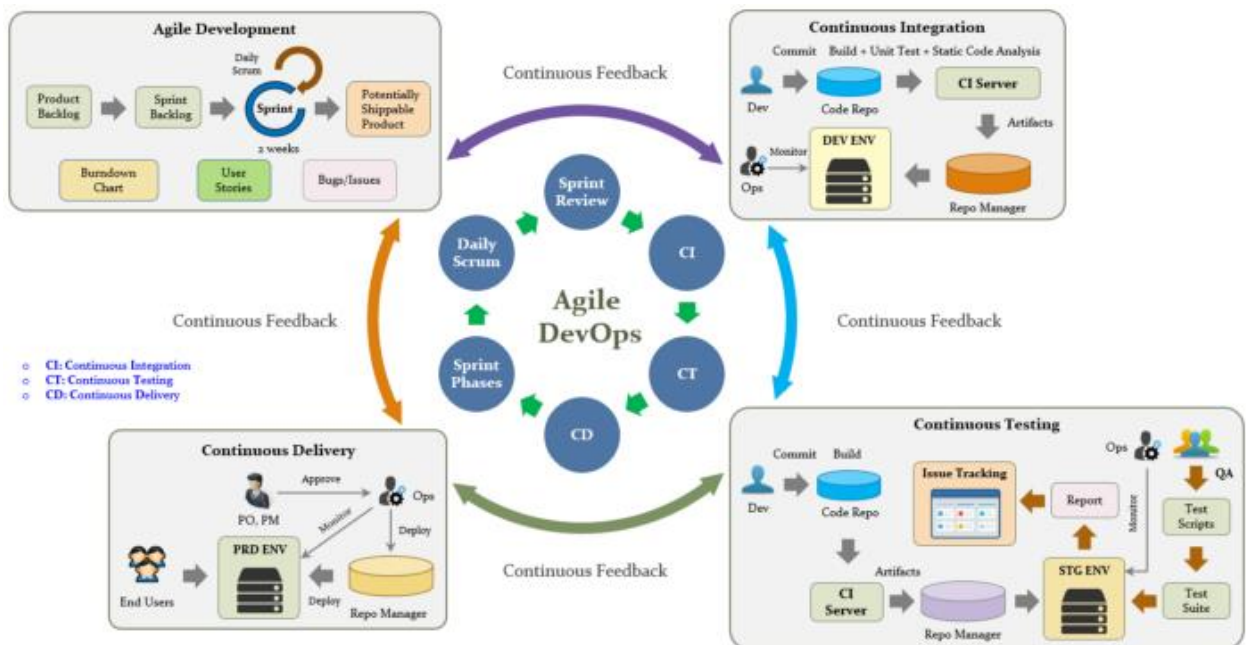
Scrum este cel mai popular și adoptat cadru Agile. Membrii echipei de dezvoltare, Product Owner-ul și Scrum Master-ul (cele 3 roluri Scrum) colaborează pentru a lucra în

iterații scurte de 30 de zile sau mai puțin, numite sprint-uri. Ei pornesc de la o listă de elemente de lucru, numită product backlog, și selectează un subset de elemente de lucru pentru a fi implementate într-un sprint. Echipa de dezvoltare își desfășoară sincronizările zilnice numite întâlniri zilnice Scrum și continuă să implementeze elementele de lucru. La sfârșitul sprintului, este produs un increment de produs sub forma unui software funcțional.

Construirea unui pipeline eficient de livrare a software-ului necesită cooperarea între echipele Dev și Ops. DevOps oferă integrare continuă / livrare continuă / implementare continuă și utilizează instrumente de automatizare și principii Lean. Echipa Dev, compusă din dezvoltatori și testeri, este responsabilă de scrierea și validarea software-ului. Partea Ops a DevOps se referă la lansarea software-ului și implică cunoștințe despre servere, middleware, configurații de rețea, stocare și tehnici de monitorizare, cu care majoritatea dezvoltatorilor nu sunt familiarizați.

Etapele standard de dezvoltare sunt:

- Colectarea și Analiza Cerințelor (Planificare)
- Design
- Implementare (Codare / Dezvoltare)
- Testare
- Implementare
- Mentenanță



Personalul utilizat pentru realizarea serviciilor

Organizare și Personal

Având în vedere propunerea de a utiliza o metodologie hibridă de implementare a proiectului, vrem să subliniem importanța punerii în valoare a indivizilor și a interacțiunilor în detrimentul proceselor rigide, pledând pentru împuternicirea și încrederea în membrii echipelor. Această perspectivă privește dezvoltarea și implementarea proiectului ca pe o activitate colaborativă între Prestator și Beneficiar, care lucrează împreună pentru a aduce valoare adăugată.

Responsabilitățile echipei de proiect a beneficiarului

De obicei, Beneficiarul desemnează un Manager de produs / proiect pentru a gestiona backlog-ul produsului, a asigura livrarea pachetelor de lucru, a răspunde la întrebările echipei, a aproba foaia de parcurs a proiectului și documentele aferente și a coordona cu părțile interesate externe, inclusiv potențiali parteneri tehnici.

Vrem să menționăm că implicarea specialiștilor cu expertiză sectorială este crucială în special în faza de planificare timpurie, pentru a oferi consultanță privind cerințele de business în cadrul proiectului. În plus, implicarea personalului tehnic din partea beneficiarului este benefică pentru gestionarea sistemului post-implementare, în special pentru funcționalitățile administrative.

Structura echipei Prestatorului

Conform abordării propuse, fiecare membru al echipei poartă responsabilitate individuală, cultivând astfel un puternic sentiment de proprietate și responsabilitate care îmbunătățește în mod natural productivitatea. Această abordare recunoaște și încorporează deciziile luate de membrii echipei, oferindu-le autonomie asupra procesului de dezvoltare și generării soluțiilor.

Pentru o livrare rapidă a valorii adăugate, abordarea noastră promovează luarea deciziilor într-un mod descentralizat, simplificând fluxul de dezvoltare și minimizând întârzierile. Cu toate acestea, anumite decizii cu implicații strategice mai ample pot necesita o abordare centralizată. Stabilirea unei structuri robuste de luare a deciziilor este esențială pentru cultivarea unei echipe eficiente. În consecință, Managerul de Proiect este responsabil pentru supravegherea logisticii proiectului, cum ar fi alocarea resurselor, raportarea progresului și îndatoririle administrative, fără a influența direct activitățile de dezvoltare, care sunt conduse de echipa de dezvoltare în conformitate cu principiile agile. Managerul de Proiect facilitează, de asemenea, comunicarea principală cu Managerul de produs / proiect a beneficiarului pentru a preveni întreruperile și a asigura un dialog clar și neîntrerupt.

Succesul proiectului este adesea atribuit prezenței analiștilor de business dedicați, arhitecților de sistem experimentați și membrilor competenți ai echipei care încorporează principiile agile nu doar ca metodologie, ci ca filosofie de bază. Cadrul Scrum conferă echipei de dezvoltare responsabilitatea de a naviga în călătoria de dezvoltare și de a inova soluții.

Un analist de business sau dezvoltator senior își va asuma rolul de Scrum Master, jucând un rol crucial în cultivarea unui mediu agil eficient. Aceasta implică promovarea muncii în echipă, eliminarea barierelor și protejarea dezvoltatorilor de posibile distrageri. Responsabilitățile Scrum Master-ului se extind la monitorizarea progresului echipei, oferirea de suport, îmbunătățirea comunicării și abordarea provocărilor pentru a spori eficiența echipei și a rezolva obstacolele.

Echipa de proiect este compusă din specialiști cu competențe extinse în analiză de business și de sistem, design UI/UX, dezvoltare software, baze de date, administrare de sisteme, DevOps și testare.

Menținerea unei culturi agile necesită o comunicare transparentă. Interacțiunile directe, nesupravegheate între Managerul de produs / proiect al beneficiarului și dezvoltatori sunt descurajate pentru a păstra transparența în toate comunicările. Discuțiile despre funcționalități și actualizări sunt documentate sistematic ca comentarii la funcționalități relevante, asigurând vizibilitatea pentru întreaga echipă.

Managerul de Proiect și Directorul companiei sunt responsabili pentru gestionarea dimensiunilor organizaționale și administrative ale contractului, garantând că toate activitățile proiectului sunt aliniate și clar articulate.

Disponibilitatea echipei

Pentru a asigura implementarea proiectului în conformitate cu cerințele caietului de sarcini, propunerea noastră este ca procesul de dezvoltare a ambelor sisteme să se desfășoare în paralel. Pentru asta suntem gata să implicăm experți non-cheie suplimentari, chiar de la etapa de analiză.

Structura echipei de suport

Localizat în Chișinău, Moldova, Centrul nostru de Suport cu două niveluri abordează nevoile clienților prin Suport de Primă Linie pentru analiza inițială și Suport de Linie a Doua pentru depanare completă. O linie telefonică de urgență în timpul orelor de lucru și un helpdesk online disponibil 24/7 asigură disponibilitatea suportului, cu Managerul de Proiect ca principal contact pentru probleme urgente.

Pașii pregătitori pentru perioada de garanție, cum ar fi permiterea accesului la sistemul de suport, sunt întreprinși cu o săptămână înainte de începerea acesteia. Un Manager de Suport dedicat este desemnat pentru comunicarea concentrată pe contract.

#	Nume Expert	Cod Expert	Rol propus	Responsabilități în proiect	Experiență
1.	Daniela Motpan	Expert cheie [01]	Project Manager	- Managementul Echipei de Proiect: - o Managementul, organizarea, alocarea și planificarea echipei de experți pentru întregul proiect. - o Conducerea echipei și organizarea coordonării implementării întregului proiect. - Planificare și Monitorizare: - o Planificarea inițială a proiectului, actualizarea planurilor, stabilirea punctelor de decizie și identificarea posibilităților de suprapunere a unor activități. - o Raportarea regulată a progresului proiectului și monitorizarea continuă - o Urmărirea realizării alocărilor în proiect și respectarea tuturor termenelor limită. - Plan de Comunicare: - o Elaborarea și actualizarea planului de comunicare și gestionarea relațiilor cu toate părțile implicate în proiect. - o Facilitarea comunicării în cadrul echipei și cu stakeholderii externi. - Identificarea și Rezolvarea Problemelor: - o Identificarea problemelor ce pot apărea și luarea de măsuri corespunzătoare pentru soluționarea acestora. - o Verificarea documentelor întocmite de echipa de proiect. - o Propunerea de soluții pentru evitarea și diminuarea riscurilor aferente serviciilor prestate. - o Prestarea serviciilor conform termenelor stabilite. - Documentație Tehnică: - o Asigurarea documentației tehnice de specialitate (livrabile) în cadrul proiectului. - o Lucrul direct cu toți experții implicați în proiect, cu personalul din cadrul STISC și echipa de implementare. - o Asigurarea suportului necesar pentru a răspunde incidentelor apărute ca urmare a implementărilor. - Reprezentarea și Monitorizarea: - o Reprezentarea prestatorului în relațiile cu toate părțile implicate și monitorizarea modului în care sunt desfășurate activitățile. - o Monitorizarea modului în care se desfășoară activitățile de proiect pentru realizarea obiectivelor acestuia. - o Monitorizarea activităților de mentenanță, rezolvarea eventualelor probleme legate de managementul acestora, inclusiv activitățile de instruire. - Managementul Schimbărilor: - o Asigurarea aplicării corespunzătoare a managementului schimbărilor, resurselor umane, calității, riscurilor și comunicării pe tot parcursul proiectului. - o Gestionarea domeniului de aplicare al proiectului, cronologiei, bugetului, riscurilor, problemelor și livrabilelor proiectului.	15 ani
2.	Dumitru Virtosu	Expert cheie [02]	Arhitect de sistem	- Propunerea Soluțiilor de Arhitectură: - o Proiectarea soluțiilor de arhitectură pentru toate componentele sistemului informatic - o Coordonarea integrării componentelor sistemului, având în vedere interconectările și influențele directe și indirecte ale modificărilor efectuate asupra tuturor componentelor.	10 ani

				- Dezvoltarea Soluțiilor Tehnice: - o Propunerea soluțiilor tehnice pentru implementarea funcționalităților sistemului informatic conform cerințelor caietului de sarcini, legislative și de business. - o Asigurarea că soluțiile propuse sunt în concordanță cu arhitectura existentă. - Documentarea și Actualizarea Informațiilor: - o Documentarea și păstrarea informației și/sau modelelor arhitecturii de sistem și ale structurilor de date. Actualizarea și modificarea acestor informații pe parcursul derulării activităților. - o Documentarea și menținerea informației și modelelor arhitecturale, actualizându-le pe măsură ce activitățile progresează. - Evaluarea și Asigurarea Conformității: - o Efectuarea evaluării tehnice din punctul de vedere al aderenței la principiile arhitecturale agreeate și formularea recomandărilor pentru modificarea acestora sau solicitarea definirii de soluții alternative dacă e cazul. - o Evaluarea conformității tehnice cu principiile arhitecturale agreeate și recomandarea modificărilor sau soluțiilor alternative, când este necesar. - Colaborarea cu Experții Cheie: - o Participarea alături de ceilalți experți cheie implicați în cadrul activității de implementare a cerințelor de securitate și confidențialitate pentru sistemul informatic. - o Colaborarea cu experții pentru implementarea cerințelor de securitate și confidențialitate, asigurând conformitatea cu legislația și politicile de securitate. - Monitorizarea Performanței Arhitecturale: - o Monitorizarea și evaluarea performanței arhitecturale a sistemului, identificând și propunând îmbunătățiri. - o Asigurarea că arhitectura susține scalabilitatea, fiabilitatea și eficiența.	
3.	Ștefan Condrea	Expert - cheie [03]	Business / System Analyst	- Analiza și specificarea cerințelor legislative și de bune practici: - o Analizarea și specificarea cerințelor business si de sistem. - o Identificarea și documentarea cerințelor detaliate. - Analiza sistemului existent și noilor cerințe: - o Implicarea în analiza sistemului existent și a noilor cerințe, proiectarea și validarea specificațiilor aferente cerințelor business si legislative. - o Crearea de modele vizuale, cum ar fi diagramele de flux sau hărțile de proces pentru a documenta procesele și fluxurile de lucru ale proiectului. - Proiectarea și validarea soluțiilor: - o Colaborarea cu ceilalți experți pentru a propune soluții pentru realizarea cerințelor menționate în Caietul de sarcini specifice fiecărei componente a sistemului informatic. - Asigurarea conformității și integrarea funcționalităților: - o Asigurarea că funcționalitățile implementate sunt conform tuturor specificațiilor funcționale de la toate nivelurile și că acestea sunt integrate corect în cadrul sistemului.	15 ani

				○ Asigurarea standardelor de documentare și conformitate: Menținerea documentației de înaltă calitate pentru cerințele de design, testare și fazele de implementare, asigurând conformitatea cu standardele organizaționale. - Pregătirea și execuția testelor funcționale: ○ Implicarea în pregătirea și execuția testelor funcționale alături de Coordonatorul de testare/Tester. - Analiza și gestionarea cerințelor de schimbare: ○ Analizarea solicitărilor de schimbare conform cerințelor legislative și de business pe perioada mentenanței evolutive și adaptive, indiferent de componenta sistemului informatic. - Monitorizarea și gestionarea riscurilor și schimbărilor de proiect, asigurându-se că orice ajustări sunt comunicate și documentate corespunzător.	
4.	Anders Nielsen	Expert - cheie [04]	Consultant tehnic	- Oferirea de suport pentru clienții activi, inclusiv cunoștințe detaliate despre configurațiile acestora și mediul local de instalare. - Pregătirea și documentarea nevoilor specifice ale clienților pentru o instalare folosind produsele Cryptomathic. - Pregătirea materialelor de instruire și desfășurarea sesiunilor de training pentru Soluțiile de Semnătură Electronică. - Pregătirea materialelor de instruire și desfășurarea sesiunilor de training pentru Utimaco HSM.	6 ani
5.	Peer Madsen	Expert - cheie [05]	Consultant tehnic	- Oferirea de suport pentru clienții activi, inclusiv cunoștințe detaliate despre configurațiile acestora și mediul local de instalare. - Pregătirea și documentarea nevoilor specifice ale clientului pentru o instalare folosind produsele Cryptomathic. - Pregătirea materialelor de instruire și desfășurarea de sesiuni de training pentru soluțiile de semnătură electronică. - Pregătirea materialelor de instruire și desfășurarea de sesiuni de training pentru Utimaco HSM.	10 ani
6.	Sergiu Chernev	Expert - cheie [06]	Coordonator tehnic / Dezvoltator software	- Ghidarea echipei, conducerea eforturilor de proiectare și dezvoltare, și asigurarea respectării celor mai bune practici și standarde de codare. - Gestionarea alocării sarcinilor, urmărirea progresului și colaborarea cu echipele interfuncționale pentru a asigura livrarea la timp a etapelor proiectului. - Stabilirea liniilor directe pentru arhitectura software, revizuirea documentelor de design pentru scalabilitate și performanță, și efectuarea de revizui ale codului. - Proiectare și dezvoltarea aplicațiilor .Net / Java, performante și scalabile, asigurându-se că calitatea codului respectă cele mai bune practici și standarde. - Traducerea cerințelor business în specificații tehnice, furnizarea de actualizări regulate ale proiectului. - Supravegherea testării automate, coordonarea cu echipele de QA și rezolvarea defectelor și problemelor de performanță. - Alocarea resurselor echipei, identificarea lacunelor de competențe, organizarea de instruire și asigurarea că echipa dispune de instrumentele și mediile necesare.	10 ani
7.	Adrian Ursachi	Expert - cheie [07]	Dezvoltator software, Full stack	- Dezvoltarea aplicațiilor .Net / Java, performante și scalabile, asigurându-se că calitatea codului respectă cele mai bune practici și standarde. - Efectuarea de revizui ale codului pentru a menține calitatea acestuia și oferirea de mentorat pentru dezvoltatorii juniori, ajutându-i să își îmbunătățească abilitățile tehnice și înțelegerea codului. - Crearea și menținerea documentației tehnice pentru proiectarea sistemului, arhitectură și detalii de implementare pentru a asigura claritatea și consistența în echipa de dezvoltare.	10 ani

				- Supravegherea integrării diferitelor componente software și servicii terțe, asigurându-se de funcționalitatea și performanța fără probleme în cadrul arhitecturii generale a sistemului. - Identificarea și rezolvarea blocajelor de performanță în aplicație, optimizarea codului și a interogărilor în baza de date pentru a îmbunătăți eficiența și capacitatea de răspuns a sistemului.	
8.	Nicoleta Ionaș	Expert - cheie [08]	Tester	- Crearea și implementarea planurilor de testare, cazurilor de testare și scenariilor de testare. - Dezvoltarea și menținerea testelor automate. - Identificarea, documentarea și urmărirea defectelor găsite în timpul testării, utilizând sisteme de management al defectelor. - Analizarea cerințelor software. - Lucrul strâns cu dezvoltatorii, managerii de proiect și alte părți interesate.	7 ani
9.	Vadim Balan	Expert - cheie [09]	Expert securitate	- Efectuarea testelor de penetrare și evaluarea vulnerabilităților pentru aplicații web și infrastructura IT. - Definirea și implementarea cerințelor de securitate în cadrul ciclului de viață al dezvoltării software (SDLC). - Realizarea evaluărilor detaliate ale riscurilor de securitate pentru software și infrastructura IT. - Dezvoltarea și implementarea controalelor de securitate și a sistemelor de management al securității informației (ISMS) conform standardelor precum ISO 27001, NIST și PCI DSS. - Efectuarea auditurilor IT și asigurarea conformității cu standardele de securitate aplicabile, validând că software-ul respectă cerințele de securitate.	10 ani

Planul de lucru pentru realizarea serviciilor

Sub-plan analiză și design

În contextul dezvoltării sistemului informatic, scopul planului de analiză și design este de a oferi o structură detaliată și bine definită pentru întreaga fază de analiză și proiectare a sistemului. Acest plan va servi drept ghid strategic pentru echipa de proiect și include o serie de obiective și activități esențiale pentru a asigura dezvoltarea și implementarea cu succes a sistemului informatic.

Fiind o fază critică în ciclul de viață al proiectului, etapa de analiză și design va include o serie de activități esențiale menite să clarifice cerințele, să definească arhitectura și să planifice detaliile implementării sistemului. Etapa de analiză și design va conține următoarele activități de bază:

1. Identificarea și documentarea cerințelor:
 - Analiza nevoilor și cerințelor: În această fază se vor efectua interviuri și discuții cu beneficiarul, utilizatorii finali și alte părți interesate pentru a înțelege complet cerințele funcționale și non-funcționale ale sistemului. Scopul este să se obțină o înțelegere clară a așteptărilor și necesităților utilizatorilor și beneficiarului în ceea ce privește sistemul, componentele care urmează să fie dezvoltat precum și interacțiunea cu alte sisteme.
 - Documentarea cerințelor: Se vor înregistra și documenta cerințele într-un format structurat. Aceasta include specificații precise pentru funcționalități, performanță, securitate, fiabilitate, scalabilitate și alte caracteristici importante ale sistemului.
2. Proiectarea arhitecturii sistemului:
 - Concretizarea arhitecturii: Se vor clarifica toate aspectele legate de arhitectura sistemului și a componentelor lui.
 - Designul componentelor: Definirea și proiectarea modulelor, componentelor și interfețelor sistemului. Se vor stabili relațiile și interacțiunile între acestea pentru a asigura o integrare corectă și eficientă a funcționalităților.
3. Proiectarea detaliată:
 - Proiectarea bazelor de date: În cadrul acestei activități se vor stabili structurile de date, relațiile între tabele și strategiile de acces și manipulare a datelor în baza de date. Alegerea și proiectarea tipului de bază de date adecvată sunt parte integrantă a acestei activități.
 - Designul interfeței utilizator: Crearea wireframe-urilor, mockup-urilor și a designului grafic al interfeței utilizator (UI). Se planifică și se proiectează aspecte cum ar fi layout-ul paginilor, fluxurile de navigare și interacțiunile utilizatorului cu sistemul.
4. Planificarea testelor:

- Set de activități care au obiectivul de a actualiza planul de testare pentru a verifica și valida funcționalitatea, performanța și securitatea sistemului. Se vor defini scenariile de testare și se vor planifica în detaliu etapele de testare pentru a se asigura că sistemul va îndeplini toate cerințele și va funcționa în mod corespunzător în medii reale.

5. Documentarea sistemului

- Se compilează și se completează documentația descriptivă a sistemului, inclusiv toate specificațiile, diagramele arhitecturale. Această documentație este esențială pentru dezvoltarea ulterioară, întreținerea și suportul sistemului.

Sub-plan dezvoltare

Etapă de dezvoltare reprezintă faza în care specificațiile și design-ul sistemului urmează a fi transformate în cod executabil. În această fază, programatorii iau documentația detaliată și planurile create în etapele anterioare și încep să scrie efectiv codul care va constitui fundamentul software-ului. Aceasta implică mai multe activități și sub-etape critice, fiecare având un rol esențial în realizarea finală a sistemului. Iată o detaliere a acestei etape:

1. Pregătirea mediului de dezvoltare care va include:
 - Configurarea mediului de dezvoltare integrat (IDE) și a instrumentelor necesare.
 - Instalarea și configurarea serverelor de dezvoltare, locale sau virtuale, a bazelor de date și a altor infrastructuri necesare.
2. Scrierea codului sursă/Implementarea funcționalităților de bază:
 - Implementarea modulelor individuale:
 - Dezvoltarea componentelor software conform specificațiilor de design.
 - Scrierea codului pentru funcționalitățile principale și pentru modulele auxiliare.
 - Asigurarea unei arhitecturi modulare unde fiecare componentă are o responsabilitate clar definită.
 - Dezvoltarea și optimizarea algoritmilor specifici necesari pentru funcționarea corectă a sistemului.
 - Implementarea interfețelor de utilizator (UI):
 - Crearea și testarea interfețelor grafice pentru utilizatori.
 - Asigurarea unei experiențe de utilizare intuitive și prietenoase.
 - Implementarea interfețelor de programare a aplicațiilor (API):
 - Dezvoltarea API-urilor pentru comunicarea între diferite componente ale sistemului. Dezvoltarea interfețelor grafice pentru utilizatori, asigurând o experiență de utilizare intuitivă și prietenoasă.
 - Asigurarea compatibilității și a securității acestor interfețe. Verificarea funcționalității și a responsivității interfeței de utilizator pe diferite dispozitive și platforme.
3. Gestionarea codului sursă:

- Utilizarea sistemelor de control al versiunilor (de ex. Git) pentru a gestiona modificările codului.
 - Crearea și menținerea unor ramuri (branches) pentru diferite versiuni și pentru dezvoltări paralele.
4. Integrarea modulelor:
 - Integrarea modulelor individuale într-un sistem complet și coerent.
 - Asigurarea că toate componentele funcționează împreună conform specificațiilor.
 5. Testarea:
 - Testarea unitară:
 - Crearea și rularea testelor unitare pentru fiecare modul.
 - Verificarea funcționalității fiecărei părți individuale a sistemului.
 - Testarea de integrare:
 - Verificarea interacțiunilor dintre diferitele module.
 - Asigurarea că sistemul integrat funcționează corect.
 - Testarea de sistem:
 - Testarea întregului sistem pentru a verifica dacă îndeplinește cerințele inițiale.
 - Identificarea și remedierea defectelor și a incompatibilităților.
 6. Documentarea codului:
 - Scrierea comentariilor și a documentației necesare pentru înțelegerea codului.
 - Crearea documentației tehnice pentru utilizatori și pentru echipele de suport.
 7. Revizuirea codului:
 - Realizarea revizuirilor de cod (code reviews) de către membrii echipei pentru a asigura calitatea și conformitatea cu standardele.
 - Implementarea modificărilor sugerate în urma revizuirilor.
 8. Refactorizarea codului:
 - Îmbunătățirea structurii și a clarității codului fără a schimba comportamentul funcțional.
 - Optimizarea performanței și a eficienței codului.
 9. Managementul configurației:
 - Asigurarea că toate componentele și dependențele sunt corect configurate și gestionate.
 - Utilizarea unor instrumente de automatizare a configurației (de ex. Docker, Ansible) pentru consistență și eficiență.

Sub-plan instalare, configurare și integrare

Scopul etapei de instalare, configurare și integrare este de a asigura că soluția dezvoltată poate fi implementată eficient și funcționează corect în mediile de operare prevăzute. Această etapă este crucială pentru a transforma componentele sistemului într-o soluție operațională, gata de utilizare. În acest sens, prestatorul va lua în considerare cerințele specifice privind mediile de instalare, va efectua teste riguroase implementa mecanisme de securitate adecvate pentru autentificarea și autorizarea utilizatorilor. Documentarea și validarea fiecărei activități specifice în cadrul acestei

etape sunt esențiale pentru a asigura o implementare fără probleme și pentru a demonstra conformitatea cu cerințele legale și tehnice.

Componentele etapei de instalare, configurare și integrare sunt:

1. Analiza cerințelor legale și tehnice
 - Analizarea prevederilor legale și a cerințelor tehnice pentru instalarea sistemului informatic pe medii virtuale și cloud. Evaluarea compatibilității infrastructurii existente cu cerințele proiectului.
2. Testarea înainte de livrare
 - 2.1. Planificarea și realizarea testelor
 - Crearea unui plan de testare detaliat care să acopere toate funcționalitățile sistemului informatic. Definirea criteriilor de acceptare și a metodologiilor de testare pentru medii virtuale și cloud.
 - 2.2. Realizarea testelor
 - Implementarea testelor pe medii virtuale și cloud pentru a verifica compatibilitatea și funcționarea corectă a sistemului.
 - Testarea instalării, configurării și operării sistemului în ambele medii, asigurându-se că funcționează în parametri normali.
 - 2.3. Documentarea rezultatelor testelor
 - Documentarea detaliată a rezultatelor testelor, inclusiv eventualele probleme identificate și soluțiile propuse.
 - Realizarea de rapoarte de testare care să demonstreze conformitatea cu cerințele specificate.
3. Raportul de instalare
 - 3.1. Documentarea activităților specifice
 - Crearea unui raport detaliat de instalare care să descrie fiecare activitate desfășurată în cadrul procesului de instalare, configurare și integrare.
 - Includerea în raport a detaliilor privind platformele utilizate, procedurile de instalare, problemele întâmpinate și soluțiile adoptate.
 - 3.2. Validarea și aprobarea raportului
 - Revizuirea raportului de instalare de către echipa de proiect și de către părțile interesate relevante.
 - Obținerea aprobărilor necesare pentru a confirma că toate activitățile au fost realizate conform planului și că sistemul este gata pentru livrare și utilizare.

Sub-plan de testare și asigurare a calității

Scopul acestui sub-plan este de a stabili strategia și procesul de asigurare a calității a livrabilelor din cadrul proiectului prin testare continuă și practice de asigurarea a calității.

Testare

Obiectivele testării sunt:

- Identificare defectelor și problemelor cât mai curând posibil
- Confirmarea că sistemul informatic corespunde cerințelor funcționale și standardelor
- Evaluarea cerințelor non-funcționale, și anume:
 - Performanța

- Securitate
- Comoditatea de operare

Principiile de bază aplicate în procesul de testare și asigurare a calității:

- Focusul va fi pe funcționalități critice, și după – pe funcționalitățile opționale
- Se vor executa cu prioritate scenariile pozitive și cele mai des utilizate scenarii alternative, înainte de testarea cazurilor excepționale.

Nivele de testare care se vor executa:

- Testarea unitară (unit testing): verificarea componentelor individuale
- Testarea integrată: verificarea interacțiunii dintre componente
- Testarea sistemului: verificarea sistemului ca un tot întreg
- Testarea de acceptanță: verificare că sistemul corespunde cerințelor business

Modalitatea de testare:

- Testarea de către Prestator se va realiza în cadrul fiecărui sprint
- Testarea de acceptanță se va derula pentru fiecare increment și la final de proiect.

Procesul de testare:

- Planificare. În cadrul procesului de planificare se vor realiza:
 - Crearea planului de testare care va include scopul testării, resursele necesare, durata testării per increment al sistemului
 - Revizuire bazei de testare (raportul de analiză, registrului de riscuri, probleme, etc.)
 - Identificarea subiectelor care nu pot fi testate
 - Identificare datelor de test
 - Elaborarea cazurilor de testare atât pentru cerințele funcționale, cât și non-funcționale
 - Prioritizarea cazurilor de testare
- Executarea.
 - Executarea cazurilor de testare în baza priorităților
 - Identificare și analiza defectelor
 - Confirmarea rezolvării defectelor
 - Înregistrarea rezultatelor testării
- Evaluarea rezultatelor testării:
 - Actualizarea registrului cazurilor de testate
 - Evaluarea dacă este necesare testări adiționale sau sunt necesare ajustări a documentației
 - Elaborarea raportului de testare

Modelele registrului de cazuri de testare, a cazurilor de testare și raportului de testare se vor agreea în cadrul etapei de planificare.

Asigurarea calității

Pentru asigurarea calității în cadrul proiectului se vor realiza următoarele activități:

- Revizuirea continuă a codului
- Auditul periodic
- Revizuirea documentației

Instrumente și resursele utilizate:

- Raportare defectelor de către testerii se va realiza prin Azure DevOps prin deschiderea unei sarcini de tip defect.
- Pentru echipa de acceptanță din partea Beneficiarului de va pune la dispoziție un sistem de tip help desk pentru raportare defectelor
- Pentru automatizarea testelor de regresie se vor utiliza instrumente dedicate (de ex. Cucumber, SpecFlow) și framework-uri specializate.

Roluri și responsabilități:

- Testerii sunt responsabili de executarea testării, identificarea, raportarea și monitorizare defectelor, respectarea planului de testare
- Programatorii sunt responsabili de colaborarea cu testerii pentru identificarea cauzei defectelor și rezolvarea lor
- Managerul de defecte este responsabil de gestiunea defectelor
- Coordonatorul de testare este responsabil de supravegherea respectării prezentului sub-plan, instruirea corespunzătoare a echipei, dacă este cazul. Comunicare rezultatelor testării.

Sub-plan tranziție și punere în funcțiune

Planul de tranziție și punere în funcțiune are ca scop de a asigura o tranziție lină și eficientă de la stadiul de dezvoltare sau implementare la operaționalizarea completă a sistemului informatic. Obiectivul principal urmărit este de a gestiona procesul de implementare și de a minimiza riscurile asociate cu schimbările și integrările noi în mediul de producție.

Activitățile de bază ale etapei de tranziție și punere în funcțiune vor consta din următoarele:

1. Planificarea și organizarea urmărește stabilirea unei strategii clare și a structurii de gestionare pentru implementarea planului. Este esențial ca la această etapă să fie elaborat un plan detaliat și bine structurat pentru tranziția de la vechiul sistem sau de la faza de dezvoltare la noul sistem operațional. Planul va include definirea obiectivelor și a sarcinilor echipei de implementare, identificarea resurselor necesare, stabilirea responsabilităților și crearea unui program detaliat al activităților.
2. Pregătirea infrastructurii are ca scop asigurarea infrastructurii necesare pentru implementarea și testarea sistemului. În cadrul acestei etape se va realiza instalarea și configurarea echipamentelor hardware și software, implementarea mediilor de dezvoltare și de testare, precum și asigurarea resurselor umane necesare pentru administrarea infrastructurii.
3. Testarea și validarea sistemului are ca scop verificarea funcționalității și performanței sistemului înainte de implementarea în producție. La această etapă, se va realiza efectuarea testelor de integrare, testelor de performanță, testelor de

- securitate și alte teste relevante pentru a asigura că sistemul îndeplinește cerințele și standardele stabilite.
4. Formarea și documentarea are ca scop pregătirea utilizatorilor finali pentru utilizarea sistemului nou implementat. Această etapă va conține activități precum dezvoltarea și livrarea programelor de instruire, organizarea sesiunilor de formare și furnizarea de materiale de referință, inclusiv manuale de utilizare și proceduri operaționale pentru a asigura o adopție eficientă și rapidă a sistemului de către utilizatori.
 5. Implementarea în producție are ca scop lansarea oficială a sistemului în mediul de producție. În cadrul acestei etape se va realiza planificarea și executarea migrării datelor, configurarea finală a sistemului în mediul live, asigurarea suportului continuu și gestionarea oricăror probleme care pot apărea în timpul primei perioade de operare.
 6. Monitorizarea și întreținerea urmărește asigurarea funcționării continue și eficiente a sistemului după implementare. În cadrul etapei de monitorizare și întreținere se va asigura implementarea unui plan de monitorizare a performanței sistemului, gestionarea actualizărilor și a patch-urilor, rezolvarea incidentelor și îmbunătățirea continuă a sistemului în funcție de feedback-ul utilizatorilor și de schimbările în cerințele business.

Este de menționat că tranziția implică adesea schimbări semnificative pentru utilizatori, procese și fluxuri de lucru. Este crucial să se gestioneze aceste schimbări prin comunicare adecvată, formare și suport pentru utilizatori și echipele implicate așa cum prevede planul de gestionare a schimbării.

Sub-plan de dezvoltare a capacității umane

Dezvoltarea capacității umane în operarea noului sistem informatic este un aspect important și obligator deoarece va rezulta în:

- Creșterea productivității – prin îmbunătățirea abilităților angajaților și familiarizarea acestora cu sistemul informatic, astfel se pot realiza sarcinile de serviciu mai eficient, conducând la niveluri mai ridicate de productivitate, totodată reducerea greșelilor și riscului de pierdere a datelor.
- Îmbunătățirea procesului decizional – deciziile vor fi bazate pe date corecte, care se pot analiza în evoluție pentru identificarea tendințelor
- Îmbunătățirea serviciului pentru clienți – angajații care sunt bine familiarizați cu sistemul informatic pot oferi un serviciu mai bun clienților, mai eficient și rapid.

Dezvoltarea capacității umane se va realiza prin:

- În cadrul etapei de planificare se vor identifica utilizatorii cheie ai sistemului informatic, se vor descrie rolurile și competențele lor pentru a determina cea mai potrivită metodă de comunicare, implicare în proiect și metodă de instruire
- Se va planifica implicarea utilizatorilor cheie în activitățile de analiză (prin intermediul workshop-urilor, sesiunilor de brainstorming, chestionare, etc.) pentru care se va stabili un grafic de lucru agreeat de către toate părțile implicate
- În cadrul etapei de dezvoltare, se va încuraja participarea utilizatorilor cheie în sesiunile de feedback, planificare a incrementului sistemului și testare de

acceptanță. Ședințele vor fi stabilite și înregistrate în calendar după clarificarea planului de livrare a incrementelor sistemului informatic

- Instruirea utilizatorilor prin selectarea celei mai potrivite metode de formare, livrarea de sesiuni de training dedicate rolului, adaptarea materiale de formare per rol (a se vedea sub-planul de instruire utilizatori pentru mai multe detalii)
- Se va livra toată documentația necesară pentru asigurarea operării eficiente și continue a sistemului, documentația se va actualiza când intervin modificări. Utilizatorii cheie vor fi implicați în revizuirea documentelor livrate de către Prestator, pentru a asigura consistența și înțelegerea materialului.
- În perioada de tranziție, se va asigura că tot personalul este suficient de instruit pentru operarea sistemului, procedurile de lucru se vor ajusta pentru a încadra operarea sistemului informatic în rutina zilnică
- În perioada de operare (mentenanță), se va asigura suportul continuu cu utilizatorii.

Punctele de implicare a utilizatorilor cheie se vor reflecta în planul de implementare al proiectului, în cadrul etapei de planificare. Totuși, implicarea utilizatorilor cheie în fiecare etapă a proiectului este necesară pentru a înțelege necesitate proiectului, a câștiga încrederea că sistemul informatic va îmbunătăți procesul de lucru și a depăși eventualele rezistențe la schimbare.

Sub-plan de activități de mentenanță și suport tehnic

Scopul sub-planului este de a descrie procesele aferente suportului tehnic și mentenanței.

Organizarea serviciului de suport:

- Nivelul 1 – Help desk are următoarele obiective:
 - Acordare suportului inițial utilizatorilor finali
 - Culegerea datelor necesare pentru simularea problemei
 - Analiza și diagnoza problemei
 - Rezolvarea unor probleme minore și/sau escaladarea problemei/defectului la următorul nivel
- Nivelul 2 – Suport tehnic are următoarele obiective:
 - Rezolvarea problemelor la nivel de cod,
 - Asigurare livrării noii versiuni a sistemului care va include rezolvarea defectelor (patch).
 - Implementare solicitărilor de modificare și livrarea lor în cadrul unor versiuni minore a sistemului

Suportul tehnic se va realiza în baza unui SLA predefinit, în conformitate cu cerințele Caietului de sarcini pentru detalii.

Canalele de suport:

- Help desk – un sistem de tichete, care va fi utilizat pentru înregistrarea problemelor apărute. Un tichet va trata o singură problemă, pentru a corespunde criteriilor de performanță definite în SLA.
- E-mail – Prestatorul va furniza un email dedicat pentru raportarea problemelor
- Telefon – va fi un număr de telefon dedicat pentru raportarea problemelor

- Documentație – menținerea documentației actualizate la zi (exemplu – manualul de utilizare)

Canalele de suport vor fi disponibile 24/7.

Activitățile de mentenanță:

- **Mentenanța corectivă** – va cuprinde activitățile menționate în caietul de sarcini. Pentru incidentele deschise se va urma procesul:
 - Incidentul se va înregistra în aplicație help desk, un tichet va conține un singur incident raporta, în caz contrar tichetul poate fi închis.
 - Incidentul se va analiza ca prioritate și severitate
 - Prioritate – prioritatea incidentului/problemei sa va stabili conform matricei:
 - ✓ 1 – sistemul nu poate fi livrat fără o soluție confirmată aferentă defectului/problemei, necesită o reacție imediată
 - ✓ 2 - sistemul nu poate fi livrat fără o soluție confirmată aferentă defectului/problemei, dar nu necesită o reacție imediată
 - ✓ 3 – se va soluționa în funcție disponibilitatea resurselor, timpului, riscurilor
 - Severitatea unui incident se va identifica în baza matricei de mai jos:
 - ✓ Major – o funcționalitate vitală/critică nu este implementată, erori operaționale unde rezultatul obținut este diferit decât cel așteptat, manifestându-se continuu sau repetat și care duce la blocarea operațiunilor și/sau pierderi de date
 - ✓ Mediu – erori care nu sunt critice sau vitale pentru operabilitatea sistemului, acestea pot fi ușor simulate (manifestându-se continuu)
 - ✓ Minor – erori care țin de experiența de utilizare a sistemelor
 - Rezolvare incidentului – livrarea soluției și actualizare sistemului pe mediile de test și producție, inclusiv actualizarea documentației, dacă este cazul.
- **Mentenanța evolutivă** – va cuprinde activitățile menționate în caietul de sarcini.
 - Se vor implementa cererile de schimbare pentru a face sistemul să corespundă modificărilor legislative sau de business, sau să aducă la optimizarea procesului
 - Cererile de schimbare se vor depune prin email în adresa Managerului de proiect al Prestatorului și va urma procesul descris în sub-planul de management al schimbărilor. În cazul în care schimbarea este inițiată de Prestator se va depune direct o ofertă de implementare a schimbării
 - Pentru livrarea schimbărilor se va stabili un plan de livrare (release plan)
 - Actualizarea documentației, dacă este cazul
 - Controlul versiunilor sistemului

De asemenea, în perioada de suport și mentenanță, se vor monitoriza:

- Indicatorii de performanță în baza criteriilor indicate în caietul de sarcini
- Securitatea sistemului prin evaluarea periodică a securității și intervenirea cu corectări dacă se identifică anumite breșe de securitate
- Se vor elabora și menține politici de backup și recovery a datelor în caz de dezastru, care vor descrie modul de restabilire rapidă a sistemului/componentelor în cazul unor situații excepționale.

Comunicarea se va realiza prin intermediul rapoartelor lunare de activitate care vor conține cel puțin lista de defecte rezolvate, în curs de rezolvare, livrate, etc., lista de schimbări aferente sistemului informatic sau procesului de suport și mentenanță (conținutul raportului și termenul de prezentare se vor agreea în etapa de planificare).

Sub-plan de instruire utilizatori

Obiectivul activității – instruirea utilizatorilor sistemului informatic are drept scop familiarizarea lor cu noul sistem, asigurarea că ei au cunoștințele și suportul necesare pentru a opera sistemul informatic în cea mai eficientă și productivă manieră.

Instruirea utilizatorilor se va realiza în urma acceptanței finale a sistemului informatic, atunci când Prestatorul și Beneficiarul sunt încrezuți că acesta corespunde normelor legislative și proceselor interne.

Procesul de instruire va fi divizat în următoarele activități:

- Pregătirea instruirii, în cadrul acestei activități se vor realiza următoarele:
 - identificarea participanților la sesiunile de instruire, rolurile lor, necesitățile lor, competențele curente. Această activitate de va realiza de către Prestator și Beneficiar
 - elaborarea materialelor de training – se vor elabora manuale dedicate pentru fiecare curs și rol, prezentări, etc.
 - determinarea graficului de instruire – ziua, ora și durata
 - determinarea modului de livrare al cursului - on-line sau cu prezență fizică
 - determinarea condițiilor logistice și tehnice
 - determinarea numărului maxim de participanți la curs (recomandăm grupuri de cel mult 10 persoane)
 - creare agendei cursului și transmiterea către participanți (pentru fiecare curs se va crea propria agendă)
- Desfășurare instruirii conform graficului stabilit, agendei și recepționarea de feedback. Pentru cursurile desfășurate cu prezență fizică, fiecare participant va semna fișa de prezență. De asemenea, se va planifica și o sesiune de întrebări și răspunsuri pentru a clarifica întrebările rămase. La finalul cursului, participanții la instruire vor primi un certificat de participare, și vor completa un chestionar de feedback, pentru a clarifica dacă cursul a fost util, a acoperit toate ariile necesare de utilizare a sistemului. În anumite cazuri se pot pregăti cursuri adiționale în limita timpului și bugetului proiectului.
- Revizuirea rezultatelor, în urma desfășurării cursurilor se vor crea rapoartele de instruire (conținutul raportului se va stabili în perioade de planificare), raportul va fi prezentat în termen de cel mult 5 zile lucrătoare de la data livrării cursului.

Sub-plan de comunicare

Obiectivul planului de comunicare este de a se asigura că persoanele implicate în proiect sunt informate în timp util, și implicate în funcție de rol în activitățile proiectului. În etapa de planificare, se vor identifica persoanele implicate în proiect, persoanele interesate de proiect, rolul fiecăruia, datele de contact și nivelul de implicare. Se va crea

organigrama proiectului, cu indicarea clară a căilor de escaladare a problemelor.

Rolurile și responsabilitățile fiecărui grup de participanți la un nivel înalt:

- Managerul de Proiect: Gestionează activitățile zilnice ale proiectului, comunicarea și asigurarea alinierii cu obiectivele proiectului.
- Membrii Echipei: Responsabili pentru raportarea la timp a progresului, a problemelor și pentru colaborarea eficientă în cadrul rolurilor lor.
- Părțile Interesate: Responsabile pentru a se menține informate, pentru a oferi feedback și pentru a susține obiectivele proiectului.

Se vor defini canalele și instrumentele de comunicare:

- Ședințele. În etapa de planificare se vor agree ședințele care urmează să fie ținute – periodicitatea, ziua, ora și durata lor. O lista preliminară a ședințelor este:
 - Ședința de kick-off, se va organiza la început de proiect pentru a prezenta experții cheie din partea Prestatorului și Beneficiarului, a prezenta metodologia și instrumentele de lucru, lista preliminară de riscuri, graficul de lucru.
 - Ședințele privind statutul proiectului, se vor organiza săptămânal (ziua, ora și durata se va agree în cadrul etapei de planificare). Recomandarea noastră este ca această ședință să fie organizată joi, în prima jumătate a zilei, on-line (acest fapt va face mai eficientă logistica ședinței).
Participanți: managerii de proiect.
 - Ședințele lunare privind progresul se vor organiza lunar, și vor avea scopul de prezentare a rapoartelor tehnice de progres lunar. Participanții sunt membri din echipa prestatorului și persoane desemnate din partea beneficiarului pentru a discuta stadiul activităților derulate.
 - Ședințele ad-hoc se vor organiza la necesitate pentru decizii care nu pot fi amânate și pot fi organizate de către managerii de proiect atât a Prestatorului, cât și a Beneficiarului.
 - Ședințele de planificare a incrementelor sistemelor informatice se vor organiza la începutul proiectului (etapa de planificare) și periodic în etapa de dezvoltare pentru a revizui scopul următorului increment de sistem. Scopul ședințelor este de a agree ce funcționalități se vor livra în cadrul fiecărui increment și nivelul de testare aferent. La aceste ședințe vor participa experți atât din partea Prestatorului, cât și a Beneficiarului.
 - Ședințele de progres ale echipei de dezvoltare sunt ședințele echipei de dezvoltare care se vor organiza zilnic în cadrul sprintului și unde se va analiza progresul. Participanți este numai echipa de dezvoltare.
 - Ședințele de clarificare se vor organiza cel puțin o dată în cadrul sprintului. În cadrul lor se vor prezenta, discuta sarcinile din backlog. Participanți sunt echipa de dezvoltare, analistul business/system și opțional - reprezentanți ai Beneficiarului.
 - Ședințele de revizuire a sprint-ului se vor organiza în ultima zi a sprint-ului. În cadrul ședinței se vor prezenta sarcinile executate în decursul sprint-ului. Participanți sunt echipa de dezvoltare, analistul business și reprezentanți ai Beneficiarului.
 - Ședințele de revizuire și feedback a incrementului se vor organiza conform planului de livrare a incrementului. În cadrul ședinței se va prezenta incrementul sistemului realizat. Participanți sunt echipa de dezvoltare, analistul business și reprezentanți ai Beneficiarului.

- Ședințele de lucru (workshop-uri, brainstorming, etc.) se vor organiza ședințe care vor avea scopul de clarificare și detalierea cerințelor față de sistem, ședințele vor avea atât subiecte business, cât și tehnice.

Cu cel puțin o zi lucrătoare înainte de ședință se va prezenta agenda ședinței.

- Rapoartele. Rapoartele prezintă instrumentul de bază pentru informa despre progresul proiectului la diverse etape. În etapa de planificare se vor agreea lista de rapoarte, conținutul, destinarii. O lista preliminară a rapoartelor este:
 - Raportul de Început (Inițial)
 - Rapoartele de activitate lunare (de progres)
 - Raportul de activitate intermediar
 - Raportului de activitate final
- E-mail. E-mailul va fi utilizat pentru anunțuri, transmiterea documentației, rapoartelor, și altor informații aferente proiectului. E-mailul va avea un subiect tipizat (de ex. <nr contract><titlul>) pentru a face mai simplă cautarea mesajului de interes, dar și a diferenția comunicare în cadrul proiectului de alte mesaje.
- Microsoft Teams, se va utiliza pentru comunicare și actualizări rapide, de asemenea se va crea canalul proiectului pentru un acces mai rapid la documentație
- Microsoft Sharepoint, va fi utilizat pentru stocarea documentației proiectului
- Azure DevOps sa va utiliza pentru gestiunea backlog-ului, gestionarea sarcinilor echipei și comunicare.

În etapa de planificare se vor agreea și documenta detaliat toate aspectele menționate mai sus, ceea ce va asigura că toate părțile interesate sunt informate, implicate pe în proiect, facilitând colaborarea și crescând șansele de succes ale proiectului.

Sub-plan de management al problemelor

Probleme apărute în derularea proiectului vor fi gestionate conform procesului:

- Identificarea problemei. Orice parte interesată poate identifica și raporta o problemă. Problema se va raporta către Managerul de proiect, preferabil scris prin email cu oferirea de detalii, ulterior problema va fi înregistrată în registrul de probleme. Registrul de probleme se va menține într-un fișier Excel, conținutul se va agreea în faza de planificare.
- Prioritizarea. Problema se va prioritiza în baza impactului asupra graficului, scopului, calității, bugetului proiectului.

Grila de priorități:

- Urgentă – Proiectul nu poate continua până când problema nu este imediat rezolvată
- Înaltă – Proiectul nu poate continua dacă problema nu este rezolvată până la o anumită dată
- Medie – Problema ar bloca proiectul în viitor
- Joasă – nu afectează proiectul

- Escaladarea. Managerul de proiect va escala imediat problema conform liniilor de escaladare, dacă este cazul
- Rezoluția. Managerul de proiect, în colaborare cu echipa de dezvoltare vor identifica un plan de rezolvare a problemei, și va numi un responsabil de implementarea planului.
- Monitorizare și control. Până la rezolvarea problemei, repusabilul de rezolvare problemei va raport prin email despre acțiunile întreprinse și progresul realizat. Se va actualiza și registrul de probleme cu datele oferite.
- Comunicare. Registrul de probleme se va include în rapoartele lunare de progres.
- Închidere. Înainte de a considera problema ca rezolvată și închisă, soluția se va verifica. Se va actualiza și registrul de probleme corespunzător.

Sub-planul de management al riscurilor

Sub-planul de management al riscurilor are scopul de a identifica, evalua, întreprinde măsuri de minimizare a riscurilor, monitorizare, ceea ce va asigura o abordare proactivă a riscurilor și evitare unor eventuale probleme care ar avea impact negativ asupra proiectului.

O listă inițială a riscurilor va fi prezentată și discutată în ședința de kick-off.

- Identificarea riscurilor:
 - Se va realiza în cadrul ședințelor tehnice sau de management, prin interviuri, sesiuni de brainstorming.
 - Orice parte interesată în proiect poate identifica riscuri, în orice forma – scrisă sau verbală
 - Riscurile vor fi înregistrate în registrul riscurilor, unde se va completa categoria, descrierea riscului, impactul. Responsabilitatea de gestiune a registrului este a Managerului de proiect al Prestatorului.
 - Se va identifica categoria riscului - de exemplu: management, comunicare, extern, tehnic
- Evaluarea riscurilor. Evaluarea riscului se va realiza de către managerul de proiectul împreună cu echipa de dezvoltare (dacă este necesar se va solicita și participarea experților în domeniu din partea Beneficiarului) pentru a determina:
 - Impactul riscului asupra obiectivelor proiectului: buget, calitate, timp.
 - Rating-ul se va determina în baza următoarei matrice:
 - Critic ar duce la întârzierea semnificativă a proiectului, creșterea bugetului, scăderea calității, imposibilitatea de a implementa scopul integrat al proiectului. Reprezentând o abatere de peste 20% din datele de referință.
 - Mediu ar reprezenta o abatere care poate varia de la 5% până la 20% inclusiv față de valorile de referință agreeate
 - Minor ar avea un impact nesemnificativ care prezintă o abatere de cel mult 5% față de valorile de referință
 - Prioritate de producere se va stabili pentru analiză și mitigare suplimentară, prioritate se va stabili în baze matricei:
 - Înaltă: probabilitate de producere este de peste 70%.

- Medie: între 30% și 70%.
- Joasă: mai mică de 30%.
- Identificarea planului de răspuns, se vor lua în considerație următoarele strategii:
 - Evitare: se vor dezvolta strategii pentru a evita activitățile sau situațiile cu risc ridicat care ar putea duce la rezultate negative.
 - Reducere: se vor identifica și implementa măsuri pentru a reduce probabilitatea și/sau impactul riscurilor identificate.
 - Transfer: transferul riscurilor către terțe părți
 - Acceptare: decizia de acceptare a anumitor riscuri dacă impactul acestora este minim sau dacă costurile de mitigare depășesc consecințele potențiale.

Identificarea planului de răspuns se va realiza de către echipa de dezvoltare. Opțional se pot invita persoanele interesate din partea Beneficiarului.

- Monitorizarea, controlul și raportarea riscurilor.
- Toate părțile interesate sunt responsabile să monitorizeze, actualizeze informațiile aferente riscurilor. Pentru fiecare risc va fi numit un responsabil care va asigura implementarea planului de răspuns (dacă este cazul), dar acest fapt nu privează de responsabilitate celelalte părți interesate de monitorizarea riscurilor.
- Roluri și responsabilități
 - Managerul de proiect din partea Prestatorului este responsabil de supravegherea și evaluarea procesului, documentare și comunicare
 - Echipa de dezvoltare este responsabilă de identificarea, evaluare, monitorizare și controlul riscurilor
 - Părțile interesate sunt responsabile de identificarea, monitorizarea și controlul riscurilor

Registrul de riscuri va fi ținut într-un fișier Excel și va conține și instrucțiunile de completare, care va face claritate pentru fiecare parte interesată despre semnificația datelor. Un sumar al registrului de riscuri va fi inclus în rapoartele lunare de progres, și eventual în alte rapoarte dacă este cazul.

Sub-plan de management al schimbărilor

Prin definiție, schimbarea afectează scopul proiectului, calitatea, livrabilele, resursele, etc. Astfel este important ca fiecare parte interesată în proiect să fie informată cum schimbările se procesează, dar și despre implementare lor.

Prezentul sub-plan detaliază modul în care schimbările se vor procesa, pentru a minimiza apariția problemelor și a maximiza adoptarea.

Procesul de management al schimbărilor se va derula conform următorului proces:

- Inițierea schimbării. Orice parte interesată poate iniția o schimbare. În acest scop se va transmite către Prestator o cerere de schimbare (modelul va fi agreat în perioada de planificare) prin canalele agreate de comunicare. Este important ca cererea să fie intern aprobată.

- Evaluarea schimbării. Cererea de schimbare se va înregistra în registru cererilor de schimbare. Echipa de dezvoltare (la necesitate cu participarea experților în domeniu din partea Beneficiarului) va evalua schimbarea din perspectiva:
 - Impactul asupra scopului, timpului, bugetului, calității
 - Riscurile asociate implementării schimbării sau dacă schimbarea nu va fi implementată.
 - Fezabilitatea tehnică și resursele necesare

Managerul de proiect va transmite Beneficiarului o ofertă de implementare care va include rezultatele evaluării (modelul de sa agreea în cadrul etapei de planificare).

- Aprobarea schimbării se va realiza de către Beneficiar conform nivelelor de competențe specificate în Caietul de sarcini.
- Implementarea schimbării.
 - Schimbarea se va analiza și dacă este cazul se va întocmi un Raport de analiză suplimentar (detalii a se vedea în Caietul de sarcini).
 - Se va actualiza backlog-ul cu sarcini specifice schimbării, la necesitate de vor organiza ședințe de clarificare.
 - Se va actualiza documentația de proiect aferentă care este impactată de schimbare (de exemplu manualele de utilizare).
 - Implementarea schimbării va fi validată/acceptată și de către Beneficiar (înainte de fi pusă în producție, dacă merge vorba de etapa de mentenanță).
 - Actualizarea registrului
- Rolul și responsabilitate
 - Managerul de proiect din partea Beneficiarului este responsabil de gestiunea procesului de schimbare, comunicare, evaluarea lui
 - Echipa de dezvoltare este responsabilă de inițierea schimbărilor, documentare lor, și implementare
 - Managerul de proiect din partea Beneficiarului este responsabil de asigurarea aprobării schimbărilor
 - Părțile interesate sunt responsabile de inițierea schimbărilor conform planului, oferirea de detalii și feedback.