SPECIFICAŢII TEHNICE

Anexa nr. 22 la Documentația standard
aprobată prin Ordinul Ministrului Finanţelor nr. 115 din 15.09.2021

Numărul procedurii de achiziție: ***Licitație deschisă 21263282/ ocds-b3wdp1-MD-1721991666555***
Denumirea licitaţiei: ***Servicii de dezvoltare software pentru realizarea si automatizarea fluxurilor de procesare în cadrul procesului de evidență a aeronavelor înmatriculate, optimizarea cooperării dintre autoritățile publice, prin implementarea unei soluții moderne de e-guvernare pentru sporirea transparenței procesului de evidență a aeronavelor înmatriculate în cadrul Registrului Aerian al Republicii Moldova***

Denumirea serviciilor	Denumirea modelului serviciului	Ţara de origine	Produ-cătorul	Specificarea tehnică deplină solicitată de către autoritatea contractantă	Specificarea tehnică deplină propusă de către ofertant	Standarde de referinţă
1	2	3	4	5	6	7
Lotul 1						
Servicii de dezvoltare software pentru realizarea si automatizarea fluxurilor de procesare în cadrul procesului de evidență a aeronavelor înmatriculate, optimizarea cooperării dintre autoritățile publice, prin implementarea unei soluții moderne de e-guvernare pentru sporirea transparenței procesului de evidență a aeronavelor înmatriculate în cadrul Registrului Aerian al Republicii Moldova	-//-	RM	DSS	Conform prevederilor Caietului de sarcini	În conformitate cu cerințele expuse în Anunț și Caiet de sarcini și Descrierii tehnice anexate.	**Moldova Standard**

Semnat:_____
Numele, Prenumele: Sirbu Ion
În calitate de: Director
Ofertantul: DAAC Software Systems S.R.L.
Adresa: mun.Chisinau str.Calea Iesilor 10

DAAC digital.

Autoritatea Aeronautică Civilă a Republicii Moldova
Codul CPV: 72230000-6

Oferta tehnică

privind achiziționarea serviciilor de realizare si automatizare a fluxurilor de procesare în cadrul procesului de evidență a aeronavelor înmatriculate, optimizarea cooperării dintre autoritățile publice, prin implementarea unei soluții moderne de e-guvernare pentru sporirea transparenței procesului de evidență a aeronavelor înmatriculate în cadrul Registrului Aerian al Republicii Moldova (în continuare SI RA)

Cuprins

1. GENERALITĂȚI

Sistemului informațional „Registrul aerian al Republicii Moldova” (în continuare - SI RA) va fi elaborată pentru implementărea modului digitalizat de administrare a datelor referitoare la aeronavele înmatriculare prin substituirea Registrului aerian al Republicii Moldova ținut de AAC în formă manuală cu registrul automatizat de prelucrare a informației, care va permite înregistrarea, evidenţa şi generarea automată de documente, precum şi elaborarea rapoartelor statistice.

Acest sistem va reprezinta o platformă informatică comună care va interacționa cu Registrul de stat al unităților de drept şi cu alte registre de stat relevante, va genera şi de evidenţia documentele procedurale în format electronic, cu posibilitatea aplicării semnăturii electronice. SI RA urmează să simplifice procesele de lucru ale autorității de înmatriculare şi să optimizeze utilizarea resurselor umane/financiare, creșterea eficienței şi a transparenței procedurii de înmatriculare a aeronavelor.

Instituționalizarea SI RA va contribui la digitalizarea activităţii Autorității Aeronautice Civile (în continuare – AAC), în a cărei competenţă este administrarea Registrului aerian al Republicii Moldova, prevăzut la art. 7 alin. (3) subpct. 3) din Codul aerian nr. 301/2017.

Crearea şi implementarea SI RA va contribui la:

1) sistematizarea electronică a datelor despre aeronavele civile înmatriculate;

2) îmbunătățirea cadrului de evidentă a datelor;

3) asigurarea implementării mecanismelor automatizate privind schimbul de date cu alte

sisteme informaționale, care, în ansamblu, vor spori fluxul informațional automatizat de date între autorităţile publice;

4) reducerea gestionării datelor pe suport de hârtie;

5) eficientizarea şi sporirea transparenței activității în procesul de înmatriculare şi evidenţă a datelor despre aeronavele înmatriculate.

Implementarea SI RA va îmbunătăți eficiența operațională, de a răspunde prompt la nevoile cetățenilor şi va asigura o guvernanță transparentă şi responsabilă.

SI RA va asigura evidența centralizată a informației sistematizate referitoare la aeronavele civile înmatriculate în Republica Moldova, precum şi cele radiate din Registrul aerian al Republicii Moldova.

SI RA va servi un instrument de susținere și executare a tuturor activităților și proceselor de lucru la evidența aeronavelor înmatriculate, prin oferirea mijloacelor tehnice de schimb informațional, colaborare și transparentizare a activităților desfășurate de autoritatea de înmatriculare în acest proces.

Dezvoltarea SI RA urmează să asigure:

1) creşterea calităţii şi a eficacităţii procedurilor administrative demarate de către AAC;

2) îmbunătăţirea eficienței activităţii interne a autorității de certificare, supraveghere şi control în domeniul aviaţiei civile;

3) creșterea transparenței operațiunilor administrative aferente înmatriculării, modificării, suspendării, revocării și radierii aeronavelor civile din Registrul aerian al Republicii Moldova;

4) posibilitatea dezvoltării şi a îmbunătăţirii SI RA, în funcţie de solicitările utilizatorilor;

5) integrarea în cadrul SI RA a sistemelor informaționale partajate, relevante pentru a beneficia de avantajele funcționale oferite de către aceste platforme;

6) asigurarea schimbului de date, prin intermediul platformei de interoperabilitate (MConnect), cu registrele gestionate prin sistemele informatice deţinute de către autorităţile publice din Republica Moldova pentru a primi automat datele relevante necesare procedurilor administrative cu privire la înmatricularea aeronavelor.

2. SCOPUL

Scopurile SI RA sunt următoarele:

1) dezvoltarea posibilității tehnologice pentru asigurarea unei evidențe centralizate a informațiilor sistematizate cu privire la procesele referitoare la aeronavele civile înmatriculate în Republica Moldova;

2) asigurarea supravegherii respectării cerințelor privind înmatricularea aeronavelor civile și administrarea Registrului aerian al Republicii Moldova;

3) formarea resursei informaționale de stat – RA;

4) formarea resursei informaționale privind gestiunea dosarelor aeronavelor civile înmatriculate;

5) crearea opțiunilor pentru sistematizarea proceselor de lucru;

6) schimbul automatizat de date între SI RA cu alte sisteme informaționale;

7) instituirea mecanismului automatizat destinat procedurii administrative privind înmatricularea aeronavelor, în mod autonom, de către AAC;

8) asigurarea accesului la informația cu privire la aeronavele civile înmatriculate în Registrul aerian al Republicii Moldova;

9) asigurarea sistemelor informaționale de stat, cu date complete şi veridice referitoare la aeronavele civile înmatriculate, în baza schimbului de date prin intermediul platformei de interoperabilitate (MConnect);

10) asigurarea accesului la informația, serviciile şi rapoartele publice aferente aeronavelor înmatriculate;

11) crearea dosarului electronic al aeronavei înmatriculate în Registrul aerian al Republicii Moldova.

12) organizarea controlului informaţional al datelor, asigurarea acumulării informațiilor de tipul şi calitatea stabilită, cu ajutorul unor mijloace speciale tehnice şi de program, inclusiv cu implicarea subdiviziunilor delegate din cadrul AAC;

13) asigurarea protecţiei datelor şi a securităţii informaţionale în cadrul creării şi exploatării sistemului;

14) asigurarea schimbului de date cu alte sisteme informaţionale prin intermediul platformei de interoperabilitate (MConnect);

15) producerea informaţiilor statistice.

3. DESTINAȚIA

SI RA va realiza ideea implementării unei soluții IT dinamice care ar asigura suportul informatic autorității administrative de implementare și aplicare a politicilor în domeniul aviației civile să realizeze procedura administrativă cu privire la înmatricularea aeronavelor civile în Registrul aerian al Republicii Moldova, în vederea asigurării respectării legislației privind înmatricularea aeronavelor civile.

Implementarea SI RA va contribui la crearea unui mediu favorabil pentru tranziţia la societatea informaţională şi va ameliora substanţial activitatea AAC aferentă procedurii de înmatriculare a aeronavelor. Valorificarea potenţialului tehnologiei informaţiei şi comunicaţiilor şi implementarea conceptelor moderne de prelucrare a informaţiei vor îmbunătăţi calitativ procesul

de luare a deciziei, vor face posibilă transparentizarea şi eficientizarea activităţii AAC și a altor instituții publice.

Implementarea SI RA va eficientiza și va contribui la digitalizarea proceselor de înregistrare şi evidență a datelor despre aeronavele înmatriculate în Republica Moldova.

Administratorul al SI RA va putea căuta profiluri de utilizator după identificatori și/sau după parametrii de căutare introduși. Rezultatele căutării vor fi sortate după câmpurile selectate.

În timpul funcționării SI RA va asigura gestionarea drepturilor, accesului și activității utilizatorilor, iar toate acțiunile utilizatorilor care au loc în sistem vor fi înregistrate în scopul analizei și luării deciziilor ulterioare.

De asemenea, sistemul va salva automat informații privind evenimentele din sistem (apeluri la funcții și servicii, disfuncționalități, trimiterea de mesaje de sistem etc.). În aceste scopuri, va fi utilizat mecanismul intern de jurnalizare - serviciul MLog de stat.

Pentru a obține date actualizate și pentru a verifica corectitudinea informațiilor, SI RA va interacționa strâns cu resursele informaționale de stat.

Arhitectura sistemului și ordinea de organizare a obiectelor informaționale va permite, dacă este necesar, extinderea semnificativă a listei de tipuri de obiecte de evidență, precum și modificarea procedurii de susținere a ciclului de viață și de prelucrare a obiectelor de evidență.

SI RA va asigura funcționarea serviciilor și procesarea informațiilor în regim 24/7.

SI RA va sprijini lucrul cu mai mulți utilizatori în timp real.

4. NOȚIUNI ȘI ABREVIERI

În acest document sunt utilizați următorii noțiuni și abrevieri:

AAC - Autorității Aeronautice Civile
SI - Sistemului informațional
SE – Semnătura electronică
BD – Bază de date

5. ABORDAREA DE DEZVOLTARE

Dezvoltarea Sistemului informațional „Registrul aerian al Republicii Moldova” va urma metodologia ghibridă, bazată pe principiul iterativ al dezvoltării produselor software, iar aceste principii vor fi respectate în dezvoltarea și implementarea soluției software. La etapa de proiectare a sistemului va fi elaborată sarcina tehnică (SRS+SDD), în care vor fi descrise modulele și funcționalul sistemului. La etapa de dezvoltare funcționalitatea sistemului va fi împărțită în sprinturi cu posibilitatea specificării în fiecare sprint a cerințelor funcționale și prioritizarea acestora. Durata iterațiilor (sprinturilor) va fi definită și convenită cu Beneficiarul și, pentru fiecare iterație, va fi elaborat separat un document Specificații funcționale, care va defini domeniul de aplicare și detaliile sarcinii de automatizare pentru fiecare iterație. Specificațiile funcționale elaborate și alte documente tehnice vor fi aprobate de către beneficiar înainte de a fi acceptate pentru dezvoltare.

În procesul de dezvoltare se vor aplica principiile Agile pentru a permite operarea modificărilor și flexibilitate în implementare.

AAC va fi responsabil de furniza lista cerințelor generice din care urmează să fie detaliat backlog-ul soluției. Va fi efectuată analiza de business necesară. Backlog-ul soluției va fi aprobată cu AAC și va include cerințe de operare și tehnice ordonate în modul necesar.

Articolele în backlog-ul produsului va fi ordonate de către AAC după prioritate. AAC va fi liber să-și gestioneze backlog-ul, adăugând, eliminând sau schimbând ordinea articolelor din acesta la dorință. La începutul fiecărui sprint, va selecta cele mai importante articole care se încadrează într-un sprint și din ele se creează un sprint backlog.

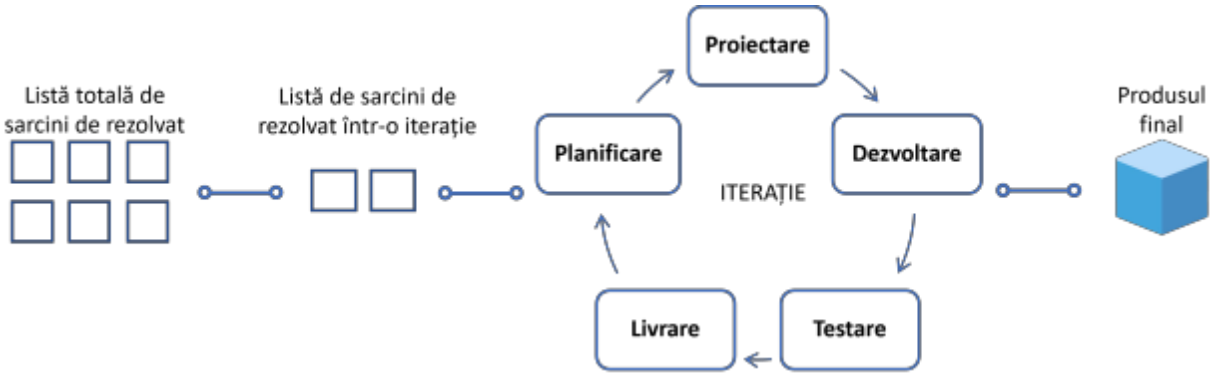


Figura 1. Descrierea indicativă a ciclului/procesului de dezvoltare Agile.

Articolele din acest sprint backlog vor fi detaliate și coordonate cu AAC pentru a aproba modelul de implementare și realizare. Sprint backlog- ul nu va fi modific în procesul de derulare a sprintului.

Produs funcțional la fiecare iterație

Fiecare sprint finalizează cu un produs funcțional, care va fi prezentat către STISC pentru acceptare în ultima (ultimele) zi (zile) ale sprintului. Produsul funcțional va îndeplini criteriile convenite – Definiția lui

„Finalizat" – produsul livrat trebuie să fie pe deplin funcțional și testat, însoțit de teste unitare relevante, însoțit de documentația relevantă dacă este cazul, codul sursă complet furnizat etc.

Produsele funcționale rezultate din diferite sprinturi pot fi combinate la discreția STISC într-o versiune lansată în producție. Orice incident raportat de Beneficiar după lansare va fi soluționat în conformitate cu Acordurile privind nivelul serviciilor (ANS).

AAC va fi responsabil de va desemna „Proprietarul produsului" (Product owner) în metodologiile Agile, care va fi permanent la dispoziția echipei pentru a răspunde la întrebările echipei, astfel evitând încetinirea procesului de implementare.

Furnizorul va avea în echipa de proiect o persoană cu rol de „Business analist" care va îndeplini totalitatea lucrărilor de analiză, proiectare de detaliu și coordonare a soluției de implementare cu AAC. Business analistul va fi responsabil de elaborarea in inițială a backlog-ului SI RA, coordonarea acestuia cu Proprietarul produsului pentru a stabili prioritățile de dezvoltare. Furnizorul împreună cu AAC va actualiza cu regularitate backlog-ul produsului pentru a reflecta lista prioritizată a funcționalităților dorite a fi implementate.

Business analistul împreună cu echipa de dezvoltare va analiza domeniul și va propune soluții optime de implementare. AAC va asigura înțelegerea corectă a cerințelor față de soluție și va ghida Furnizorul în procesul de proiectare și implementare a sistemului informatic.

Notă: descrierea metodologii de dezvoltare agilă a produselor software, care urmează să fie utilizată de compania noastră, găsiți atașat în fișierul „Metodologia de dezvoltare și implementare a produsului software _AGILE.pdf".

Implicarea AAC

Spre deosebire de modelul waterfall (cascadă) utilizat frecvent la achiziția și implementarea sistemelor informatice guvernamentale, persoana desemnată de AAC – Proprietarul produsului – va fi implicată intens în procesul de dezvoltare. Proprietarul produsului va avea trei responsabilități importante:

Va menține backlog-ul produsului. Proprietarul produsului împreună cu Business Analistul va actualiza cu regularitate backlog-ul produsului pentru a reflecta lista prioritizată a funcționalităților dorite a fi implementate.

Va răspunde la întrebările dezvoltatorilor. Proprietarul produsului va fi permanent disponibil pentru a răspunde la întrebările de clarificare ale echipei de dezvoltatori, astfel evitând comunicarea complexă și formală în cadrul proiectului. Pentru a obține un produs funcțional la sfârșitul sprintului, este extrem de important ca echipa să dispună de toate informațiile necesare la momentul oportun.

Va accepta pachetele de lucru. Pachetele de lucru livrate sunt prezentate Beneficiarului la sfârșitul fiecărui sprint pentru acceptare. Beneficiar va accepta pachetul de lucru sau va informa pe Furnizor despre defectele identificate până la finalizarea următorului sprint.

Deși nu este strict necesar, Proprietarul produsului poate participa la ședințele echipei de dezvoltare pentru a se informa despre progresele înregistrate și eventualii factori blocanți pentru o reacție promptă la aceștia.

Proprietarul produsului, de asemenea, ia decizia privind lansarea produsului în conformitate cu planul de lansare.

În conformitate cu principiile metodologiei Agile de management al proiectului, AAC va trasa viziunea de dezvoltare a produsului precum și Foaia de parcurs a acestuia pentru a monitoriza progresul și pentru a asigura dezvoltarea corespunzătoare a produsului.

6. PRINCIPII GENERALE DE DEZVOLTARE A SISTEMULUI

6.1. Principiile generale ale arhitecturii software și hardware a SI RA

Platforma WEB centralizată

Progresele în domeniul comunicațiilor moderne permit automatizarea structurilor distribuite geografic prin stocarea și prelucrarea centralizată a informațiilor, ceea ce sporește considerabil viteza și fiabilitatea informațiilor și reduce semnificativ costurile de instalare, implementare și întreținere a sistemelor.

Utilizarea internetului permite o accesibilitate largă în interiorul și în afara țării.

Utilizarea de platforme și instrumente de dezvoltare moderne și general acceptate

Soluția se va baza pe Microsoft .NET Framework, care este standardul industrial de facto pentru construirea sistemelor informatice pe mai multe niveluri și este susținut de cei mai importanți furnizori de software din lume.

Platforma tehnologică și arhitectura sa permit, dacă este necesar, extinderea funcționalității sistemului, adăugarea modulelor sau obiectelor contabile necesare.

Interfață web utilizator

Implementarea interfeței cu utilizatorul sub formă de interfață WEB permite minimizarea cerințelor față de hardware-ul și software-ul de la locul de muncă al utilizatorului și este cross-platform, ceea ce exclude legarea la un anumit sistem de operare.

Accesul la funcționalitățile Ministerului Muncii și Protecției Sociale va fi asigurat utilizând cele mai populare browsere web - Microsoft Edge, Mozilla Firefox, Google Chrome, Apple Safari din ultimele 3 versiuni.

Utilizarea componentelor open source

În dezvoltarea sistemului vor fi utilizate componente Open Source și standarde deschise. Vor fi utilizate instrumente de dezvoltare standard larg răspândite, inclusiv biblioteci cu sursă deschisă.

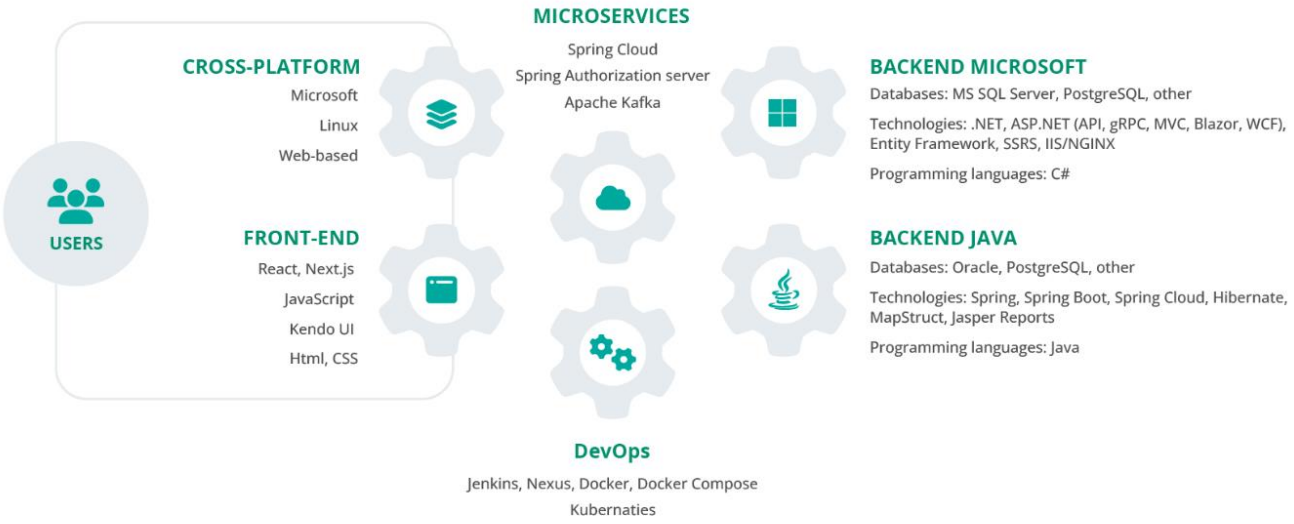
În cazul în care se utilizează componente cu sursă închisă/proprietare, acestea se vor baza pe principiul standardelor deschise și nu vor genera costuri de proprietate suplimentare.

Pentru beneficiar, aceasta înseamnă că nu sunt necesare taxe de licență suplimentare la dezvoltarea și utilizarea soluției.

7. STIVA TEHNOLOGICĂ

Suntem o companie cu o experiență largă în dezvoltarea de software, orientată spre rezolvarea sarcinilor și problemelor complexe ale clienților noștri în domeniul digitalizării și automatizării afacerilor. Putem oferi clienților noștri dezvoltarea de sisteme informatice de orice complexitate folosind următorul stack tehnologic:

Tehnologii și platforme informatice



7.1. COMPONETE CARE ASIGURĂ DESVOLTAREA SISTEMULUI SI RA

7.1.1. COMPONETE PENTRU INTERFAȚA UTILIZATORULUI

7.1.1.1 REACT.JS

React.js este o bibliotecă open source pentru limbajul de programare JavaScript pentru dezvoltarea interfețelor utilizator. Aceasta ajută la implementarea rapidă și ușoară a reactivității - fenomenul în care orice altceva se schimbă ca răspuns la o schimbare a unui element.

React este utilizat de dezvoltatorii frontend. În modelul MVC, al cărui nume vine de la Model-View-Controller, React poate fi folosit și de designeri de layout, testeri și alți specialiști implicați în crearea interfețelor web.

Utilizarea React

React este utilizat pentru a crea aplicații cu o singură pagină și cu mai multe pagini, pentru a dezvolta site-uri web mari. Biblioteca este concepută pentru:

- pentru crearea de interfețe web funcționale și interactive care nu necesită reîmprospătarea constantă a paginii;

- implementarea rapidă și ușoară a componentelor individuale și a paginilor întregi - elementele din React sunt ușor de reutilizat;
- dezvoltarea ușoară a structurilor de program complexe - este ușor să le descrii folosind abordarea din React;
- refacerea noilor funcționalități cu orice stivă tehnologică inițială: biblioteca nu depinde de restul setului de instrumente și va funcționa bine indiferent pe ce este scris codul;
- dezvoltarea de aplicații cu o singură pagină și cu mai multe pagini (SPA și PWA). Acestea sunt aplicații care funcționează ca programe și servicii web și au o interfață adecvată;
- lucrul cu partea de server a unui site web sau dezvoltarea de aplicații mobile. În astfel de cazuri, React este utilizat împreună cu instrumente care adaptează tehnologiile web pentru alte scopuri.

Caracteristici ale bibliotecii React

React are o serie de caracteristici care îl fac un instrument flexibil și puternic.

Declarative

Stilul declarativ înseamnă că este suficient să descrieți o dată cum vor arăta rezultatele codului - elemente în diferite stări. Nu trebuie să se concentreze asupra modului în care vor fi obținute rezultatele: biblioteca va face majoritatea sarcinilor. React.js va actualiza automat elementele în funcție de condiții, sarcina principală este de a le descrie corect. O abordare convenabilă și clară facilitează scrierea și depanarea codului.

Arbore virtual DOM

Orice interfață web se bazează pe un document HTML și pe stiluri CSS cu cod JavaScript atașat la acesta. Structura unui document HTML, sau mai degrabă modelul său, se numește arbore DOM (DOM vine de la Document Object Mode). Este un model arborescent în care toate elementele utilizate pe pagină sunt organizate ierarhic.

Ceea ce face React special este faptul că creează și stochează în cache un arbore DOM virtual - o copie a DOM care se modifică mai rapid decât structura reală. Acest lucru este necesar pentru a actualiza rapid paginile. Dacă un utilizator efectuează o acțiune sau are loc un eveniment, DOM-ul trebuie să se schimbe deoarece obiectele de pe pagină se vor schimba. Dar modelul real al obiectelor poate fi imens, actualizarea sa fiind un proces lent. Așa că React nu lucrează cu el, ci cu o copie virtuală în cache, care cântărește mai puțin.

Actualizarea fragmentară a DOM

Pentru a îmbunătăți performanța, React nu actualizează DOM în întregime. Aceasta păstrează două copii ușoare în memorie: cea curentă și cea anterioară. Atunci când ceva este actualizat, biblioteca compară versiunile între ele și modifică doar partea din arbore care s-a modificat efectiv. Aceasta pentru a evita reîncărcarea întregului DOM și încetinirea paginii. Abordarea pare complicată, dar este importantă pentru optimizarea încărcării.

Abilitatea de a reutiliza componente

React se bazează pe componente - elemente individuale ale unei interfețe web. Componentele sunt încapsulate, adică sunt autonome: fiecare dintre ele găzduiește toate metodele și datele necesare. În cazul componentelor React, încapsularea înseamnă, de asemenea, că starea elementului este stocată în elementul în sine, mai degrabă decât într-un obiect separat.

Componentele autonome încapsulate pot fi reutilizate, plasate în altă parte în cod, într-o secțiune diferită sau pe o pagină diferită. Datele pot fi transportate în întreaga aplicație, utilizate în afara DOM-ului unei anumite pagini. Acest lucru accelerează dezvoltarea și reduce numărul de pași pentru a crea o interfață funcțională. Prin eliminarea dependențelor complexe, încapsularea facilitează, de asemenea, depanarea.

Fluxul de date în flux descendent

Componentele pot transmite proprietăți și date între ele, dar numai într-o singură direcție - de la "părinte" la copil. Acest lucru ajută la implementarea unei ierarhii clare și facilitează depanarea. Fluxul de date unidirecțional înseamnă că programatorul poate înțelege întotdeauna de unde anume au venit datele către element.

O altă caracteristică a fluxului de date este că nu este posibilă modificarea directă a proprietăților. Există funcții callback speciale în acest scop. Acestea sunt părți ale codului executabil care sunt transmise de la o componentă la alta sub forma unui parametru de funcție. Acest lucru vă permite să mențineți stabilitatea: proprietățile sunt neschimbate după creare, iar funcția callback ca și cum le recreează. Ca urmare a utilizării callback-urilor, fluxul de date arată astfel: proprietățile și informațiile sunt transmise de sus în jos, iar evenimentele au loc de jos în sus.

Sintaxa JSX

JSX înseamnă JavaScript XML. Este o extensie a limbajului JavaScript care ajută la descrierea elementelor de tip HTML utilizând codul React. Cu ajutorul sintaxei React, componentele paginii sunt create și gestionate flexibil.

Chiar dacă elementele sunt similare cu HTML, este totuși un limbaj JavaScript cu capacitatea de a modifica rapid și ușor DOM-ul cu ajutorul codului. Și totuși, JSX joacă ca HTML: în esență, dezvoltatorul descrie componenta dorită în limbajul de marcare, iar aceasta rămâne un obiect JavaScript cu funcționalitate largă. Acest lucru este convenabil și simplifică programarea, dar poate fi confuz pentru începători.

React Hooks

În versiunile mai vechi, puteai gestiona stările utilizând clase - construcții speciale. Acum React.js acceptă hooks, care sunt funcții speciale care "agață" starea unui element sau a unei metode. Schimbarea stării sau apelarea unei metode "trage" aceste funcții cu ea, iar ele sunt executate automat - acest lucru ajută la evitarea utilizării claselor, face mai ușoară și mai simplă scrierea codului.

React are cârlige încorporate, dar dezvoltatorul își poate crea și propriile cârlige - acest lucru ajută la reutilizarea codului și la reducerea cantității de cod.

Instrumente pentru dezvoltatorii React

Deoarece React.js este o tehnologie foarte populară, creatorii săi au dezvoltat extensii gratuite de browser cu instrumente pentru testare și depanare. Front-enderii folosesc adesea consola și panoul dezvoltatorului din browser pentru a testa modul în care funcționează codul lor. Instrumentele dezvoltatorului React fac acest lucru ușor și împuternicitor. De exemplu, puteți vizualiza componente cu un nivel ridicat de cuibărire chiar în browser și nu trebuie să le căutați în cod pentru mult timp. React Developer Tools există pentru noile versiuni ale browserelor populare Firefox și Google Chrome.

Beneficii ale React

- Popularitate

Este unul dintre cele mai comune trei instrumente pentru dezvoltarea front-end. Multe site-uri web și aplicații sunt scrise în React. Acesta este utilizat de companii mari precum Yandex, Uber, Sberbank, Avito, BBC, Airbnb, Netflix și altele. Locurile de muncă sunt abundente și foarte bine plătite, chiar și la pozițiile entry-level.

- Comunitate numeroasă

Aceasta este o altă consecință a popularității: există o mulțime de dezvoltatori React, inclusiv experimentați, mulți dintre ei scriu tutoriale și articole, ajutând începătorii să intre în procesul de dezvoltare. În cursul învățării, un începător poate apela întotdeauna la resurse tematice, unde vor încerca să ajute.

- Ecosistem dezvoltat

Creatorii React au încurajat activ utilizatorii să contribuie la dezvoltarea proiectului, iar entuziaștii și-au scris propriile tehnologii pentru a le împărtăși cu biblioteca. Ca urmare, acum s-a format un întreg ecosistem: zeci de biblioteci și alte instrumente care pot fi utilizate în dezvoltare. Acest lucru face procesul din ce în ce mai simplu și vă permite să rezolvați un număr foarte mare de sarcini.

- Ușor de creat

Cu ajutorul componentelor de proiect, puteți construi rapid și ușor o interfață interactivă, receptivă pentru un site web sau o aplicație de orice complexitate. Acest lucru este mult mai ușor decât scrierea manuală a reacțiilor la tot felul de evenimente și reduce numărul de erori. Deoarece componentele pot fi reutilizate oriunde în cod, sarcina devine și mai ușoară. De exemplu, pentru a crea un site sau o aplicație complexă cu mai multe pagini, nu trebuie să scrieți aceeași componentă în mod repetat.

- Reactivitate

Acesta este unul dintre avantajele cheie ale proiectului, puse în nume. Biblioteca reacționează la actualizările componentelor și afișează automat modificările acestora în arborele documentelor. Modificările pot avea loc ca răspuns la acțiunile utilizatorului, la unele modificări externe sau la alte evenimente. Ca urmare, site-urile web și aplicațiile devin mai atractive pentru utilizator.

- Eficacitate

Datorită DOM virtual, biblioteca economisește resurse. Pentru a schimba starea elementelor, nu este necesară reîncărcarea completă a întregului arbore DOM, irosind inutil traficul utilizatorului și încărcând browserul. Doar elementele specifice sunt modificate, se întâmplă prin arborele DOM virtual - site-urile și aplicațiile devin mai "ușoare" și mai convenabile.

- Viteză înaltă

Site-urile web și aplicațiile scrise în React sunt rapide și receptive datorită DOM-ului virtual. Arborele ocupă mai puțin spațiu, se actualizează mai rapid.

De asemenea, accelerează munca programatorilor. Componentele sunt ușor de creat și reutilizat, logica mesajelor dintre ele este deja gândită. Principala sarcină este de a descrie corect stările. Nu este necesar să scrieți toată logica și principiile de la zero, creatorii React au făcut-o deja pentru dezvoltator.

- Depanare ușoară

Există mai puține erori cu React decât atunci când se scrie cod în limbaje "pure" fără biblioteci. Acest lucru se întâmplă deoarece o mulțime de lucruri sunt implementate automat, astfel încât influența "factorului uman" este redusă. Dar chiar dacă un programator face o greșeală în timpul dezvoltării, este destul de ușor să o depisteze și să o corecteze datorită logicii clare a fluxului de date, instrumentelor suplimentare și sintaxei clare. Codul React este ușor de citit, de înțeles și de depanat.

7.1.2. COMPONENTS THAT ENABLE BACKEND MICROSOFT DEVELOPMENT

7.1.2.1 .NET PLATFORM

.NET Framework este o tehnologie care sprijină crearea și executarea de servicii web și aplicații Windows. La dezvoltarea .NET Framework au fost avute în vedere următoarele obiective.

– Furnizarea unui mediu de programare coerent orientat pe obiect pentru stocarea și

executarea locală a codului obiect, pentru executarea locală a codului distribuit pe internet sau pentru executarea la distanță.

- Furnizarea unui mediu de execuție a codului în care:

  - minimizează potențialul de conflict în timpul implementării software-ului și al controlului versiunii;
  - executarea sigură a codului este garantată, inclusiv a codului creat de un furnizor terț necunoscut sau care nu se bucură de încredere deplină;
  - sunt eliminate problemele de performanță în mediile de execuție a scripturilor sau a codului interpretat;

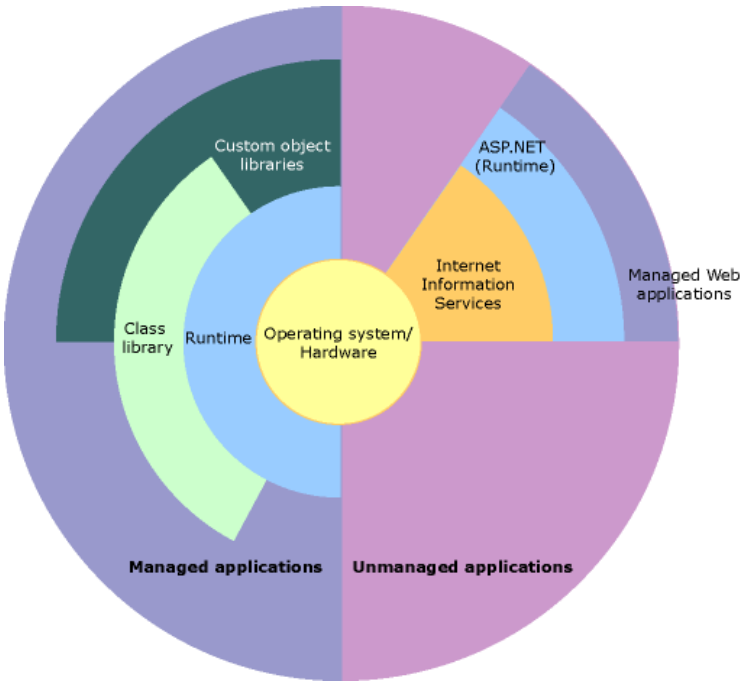
- oferă principii comune de dezvoltare pentru diferite tipuri de aplicații, cum ar fi aplicații Windows și aplicații web;
- oferă interoperabilitate standard în industrie care asigură faptul că codul .NET Framework poate fi integrat cu orice alt cod.

.NET Framework constă dintr-un mediu de execuție a limbajului comun (mediu CLR) și din biblioteca de clase .NET Framework. Mediul CLR este baza platformei .NET Framework. Mediul de execuție poate fi privit ca un agent care gestionează codul în timpul execuției și oferă servicii de bază, cum ar fi gestionarea memoriei, gestionarea firelor de execuție și comunicarea la distanță. În acest sens, mediul impune condiții stricte de tipizare și alte tipuri de verificări ale preciziei codului pentru a asigura siguranța și fiabilitatea. De fapt, principala sarcină a mediului runtime este gestionarea codului. Codul care accesează mediul de timp de execuție se numește cod gestionat, iar codul care nu accesează mediul de timp de execuție se numește cod negestionat. Biblioteca de clase este o colecție cuprinzătoare orientată pe obiecte de tipuri reutilizabile care sunt utilizate pentru a dezvolta aplicații, de la aplicații obișnuite de linie de comandă și interfață grafică cu utilizatorul (GUI) până la aplicații care utilizează cele mai recente caracteristici ale tehnologiei ASP.NET, cum ar fi formularele web și serviciile web XML.

.NET Framework poate fi găzduit de componente negestionate care încarcă mediul CLR în propriile procese și încep să execute cod gestionat, creând astfel un mediu software care permite atât facilități de execuție gestionate, cât și negestionate. .NET Framework nu numai că oferă mai multe medii de execuție subiacente, dar sprijină și dezvoltarea de medii de execuție subiacente de către furnizori independenți.

De exemplu, ASP.NET găzduiește mediul de timp de execuție și oferă un mediu scalabil pentru codul gestionat pe partea de server. ASP.NET lucrează direct cu mediul de timp de execuție pentru a asigura execuția aplicațiilor ASP.NET și a serviciilor Web XML discutate mai târziu în acest articol.

Figura următoare demonstrează relația mediului CLR și a bibliotecii de clase cu aplicațiile utilizatorului și cu întregul sistem. Figura arată, de asemenea, modul în care codul gestionat funcționează în cadrul arhitecturii mai largi.



Caracteristici ale mediului CLR

Mediul CLR gestionează memoria, execuția firelor de execuție, execuția codului, verificarea siguranței codului, compilarea și alte servicii de sistem. Aceste facilități sunt interne pentru codul gestionat care rulează în mediul CLR.

Din motive de securitate, componentelor gestionate li se atribuie diferite grade de încredere, în funcție de o serie de factori care includ originea lor (de exemplu, internetul, o rețea de întreprindere sau un computer local). Aceasta înseamnă că o componentă gestionată poate sau nu să poată efectua operațiuni de acces la fișiere, operațiuni de acces la registru sau alte funcții sensibile, chiar dacă este utilizată în aceeași aplicație activă.

Mediul de execuție asigură, de asemenea, fiabilitatea codului prin punerea în aplicare a unei infrastructuri stricte de tipizare și verificare a codului, denumită Common Type System (CTS). Sistemul de tip comun oferă autodescriere pentru tot codul gestionat. Diversele compilatoare de limbaj de la Microsoft și de la furnizori independenți creează cod gestionat care satisface sistemul de tip comun. Aceasta înseamnă că codul gestionat poate accepta alte tipuri și instanțe gestionate, asigurând în același timp corectitudinea tipului și tipizarea strictă.

În plus, un mediu de execuție gestionat elimină multe probleme software comune. De exemplu, mediul de execuție gestionează automat plasarea obiectelor și referințele la obiecte, eliberându-le atunci când nu mai sunt utilizate. Gestionarea automată a memoriei elimină două dintre cele mai frecvente erori ale aplicațiilor: scurgeri de memorie și referințe de memorie invalide.

Mediile de execuție cresc, de asemenea, productivitatea dezvoltatorilor. De exemplu, programatorii pot scrie aplicații într-un limbaj de dezvoltare familiar, profitând în același timp pe deplin de mediul de execuție, bibliotecile de clase și componentele scrise de alți programatori în alte limbaje. Acest lucru este disponibil pentru orice furnizor de compilatoare care accesează mediul de execuție. Compilatoarele de limbaje concepute pentru .NET Framework pun instrumentele .NET Framework la dispoziția codului existent scris în limbajele respective, facilitând astfel adaptarea aplicațiilor existente.

Deși mediul de execuție a fost proiectat pentru software-ul viitor, acesta suportă, de asemenea, software-ul de astăzi și de ieri. Interacțiunea dintre codul gestionat și cel negestionat permite dezvoltatorilor să utilizeze componentele COM și DLL-urile necesare.

Mediul de timp de execuție este conceput pentru a îmbunătăți performanța. Deși mediul de execuție în limbaj general furnizează multe servicii de execuție standard, codul gestionat nu este niciodată interpretat. Facilitatea de compilare la cerere (JIT) permite ca tot codul gestionat să fie executat în limbajul mașină al computerului pe care rulează. Între timp, Memory Manager elimină posibilitatea fragmentării memoriei și mărește cantitatea de memorie adresabilă pentru îmbunătățiri suplimentare ale performanței.

În cele din urmă, mediul de execuție poate fi găzduit în aplicații server de înaltă performanță, cum ar fi Microsoft SQL Server și IIS (Internet Information Services). Această infrastructură vă permite să utilizați cod gestionat pentru a scrie propria logică de program, beneficiind în același timp de performanța superioară a celor mai bune servere de producție care acceptă găzduirea mediului de execuție.

Bibliotecă de clase .NET Framework

Biblioteca de clase .NET Framework este o colecție de tipuri care sunt strâns integrate cu mediul CLR. Biblioteca de clase este orientată pe obiecte. Aceasta oferă tipuri din care codul controlat de utilizator poate moșteni funcții. Acest lucru nu numai că simplifică lucrul cu tipurile .NET Framework, dar reduce și timpul necesar pentru a învăța noi instrumente .NET Framework. În plus, componentele terților pot fi ușor combinate cu clasele .NET Framework.

De exemplu, clasele de colectare ale .NET Framework implementează un set de interfețe pentru dezvoltarea de clase de colectare personalizate. Clasele de colectare personalizate pot fi ușor combinate cu clasele .NET Framework.

După cum se așteaptă de la o bibliotecă de clase orientată pe obiect, tipurile .NET Framework vă permit să efectuați sarcini de programare comune, inclusiv manipularea șirurilor de caractere, colectarea datelor, conexiunile la baze de date și accesul la fișiere. În plus față de aceste sarcini comune, biblioteca de clase conține tipuri care susțin multe scenarii de dezvoltare specializate. Puteți utiliza .NET Framework pentru a dezvolta următoarele tipuri de aplicații și servicii:

- Aplicații de consolă. Consultați articolul Construirea aplicațiilor consolă.
- Aplicații de interfață grafică utilizator Windows (Windows Forms).
- Aplicații Windows Presentation Foundation (WPF). Aplicații ASP.NET.
- Servicii Windows;
- Aplicații orientate pe servicii care utilizează Windows Communication Foundation (WCF).
- Aplicații care susțin procesele de afaceri Windows Workflow Foundation (WF).

Clasele Windows Forms sunt un set complet de tipuri care simplifică foarte mult dezvoltarea interfețelor grafice Windows cu utilizatorul. Atunci când scrieți o aplicație web Forms ASP.NET, puteți utiliza clasele web Forms.

7.1.2.2 ASP.NET

ASP.NET este o platformă distribuită gratuit pentru dezvoltarea de site-uri și aplicații web dinamice, creată de Microsoft și parte a NET.Framework. Este o dezvoltare ulterioară a vechii tehnologii Microsoft ASP.

Având în vedere această perspectivă, dezvoltatorii companiei au decis să creeze o platformă cu propriul Common Language Runtime. Platforma cu propriul common language runtime a permis dezvoltarea de aplicații web în orice limbaj acceptat de tehnologia .NET.

Utilizarea unui timp de execuție în limbaj comun înseamnă că codul scris de un programator web în orice limbaj de programare se execută în două etape:

- este mai întâi compilat în Microsoft Intermediate Language - adică, indiferent în ce limbă a fost scrisă inițial pagina web, aceasta a fost transformată într-un document unificat;
- apoi ansamblul IL rezultat este compilat în cod mașină de nivel scăzut, care este executat în modul just-in-time, adică chiar înainte de lansarea aplicației.

Înainte de a compila în cod mașină de nivel scăzut, compilatorul determină tipul de sistem de operare și bitness-ul acestuia. Acest lucru permite programatorilor să creeze un singur cod într-un limbaj de nivel înalt pentru toate platformele comune.

Aplicațiile create în platforma ASP.NET sunt compilate în IL o singură dată, recompilarea fiind efectuată numai dacă se fac modificări ale codului sursă. Iar codul mașină este stocat în memoria cache în directorul sistemului. Acestea fiind spuse, dacă o aplicație este creată în Visual Studio, codul acesteia este rescris în IL ca parte a compilării generale a proiectului. Cu toate acestea, codul site-ului creat în afara oricărui proiect este compilat pagină cu pagină la prima solicitare a unei anumite pagini. În ambele cazuri, codul IL al aplicației este convertit în cod mașină prima dată când este executat.

ASP.NET implementează un model de programare orientat pe obiecte. Din acest motiv, oferă dezvoltatorului acces la toate obiectele din .NET Framework. De asemenea, acest model oferă:

- crearea de clase reutilizabile care simplifică și accelerează scrierea codului;
- standardizarea codului cu ajutorul interfețelor;
- extinderea claselor existente prin moștenire;
- combinarea funcțiilor utile într-o componentă compilată care poate fi distribuită separat.

O altă oportunitate care decurge din modelul de programare orientat pe obiecte este reprezentată de controalele server-side. Acestea sunt obiecte încapsulate pe care dezvoltatorii le pot manipula cu ajutorul codului pentru a le personaliza aspectul, a furniza date și a răspunde la evenimente. Marcajul HTML de nivel scăzut redat de aceste controale este ascuns. Acest lucru asigură faptul că dezvoltatorii nu trebuie să le scrie ei înșiși - în schimb, obiectele încapsulate sunt auto-scrise în elementele HTML corespunzătoare după ce pagina web este redată.

Site-uri și aplicații web

ASP.NET oferă trei platforme pentru crearea de aplicații web: formulare web, ASP.NET MVC și pagini web ASP.NET. Toate cele trei platforme sunt stabile și mature și puteți crea aplicații web excelente folosind oricare dintre ele.

- web forms

Cu ASP.NET Web Forms, puteți crea site-uri web dinamice utilizând modelul familiar de tip drag-and-drop condus de evenimente. Zona de design și sutele de controale și componente vă permit să creați rapid site-uri complexe cu interfețe utilizator și acces la date.

- MVC

ASP.NET MVC oferă o modalitate eficientă, bazată pe șabloane, de a crea site-uri web dinamice pentru a separa clar preocupările și a oferi utilizatorilor control complet asupra marcajelor pentru o dezvoltare agilă. ASP.NET MVC conține multe caracteristici pentru dezvoltarea rapidă, în conformitate cu TDD, pentru a crea aplicații complexe care utilizează cele mai recente standarde web.

- ASP.NET pagini web

Paginile web ASP.NET și sintaxa Razor oferă o modalitate rapidă, accesibilă și simplificată de a combina codul server-side cu HTML pentru a crea conținut web dinamic. Conectarea la baze de date, adăugarea de videoclipuri, conectarea la site-uri de rețele sociale și multe alte caracteristici vă ajută să creați site-uri web frumoase, conforme cu cele mai recente standarde web.

Toate cele trei platforme ASP.NET se bazează pe .NET Framework și împărtășesc caracteristicile de bază ale .NET și ASP.NET. De exemplu, toate cele trei platforme oferă un model de securitate de autentificare bazat pe membri și toate cele trei platforme utilizează aceleași instrumente pentru gestionarea cererilor, gestionarea sesiunilor etc. care fac parte din funcțiile de bază ASP.NET.

- WebAPI

Platforma ASP.NET Web API facilitează crearea de servicii HTTP pentru o gamă largă de clienți, inclusiv browsere și dispozitive mobile. ASP.NET Web API este o platformă ideală pentru crearea de aplicații REST bazate pe .NET Framework.

- Tehnologii în timp real

ASP.NET SignalR este o bibliotecă nouă pentru dezvoltatorii ASP.NET care simplifică dezvoltarea funcțiilor web în timp real. SignalR oferă comunicare bidirecțională între server și client. Serverele pot trimite instantaneu conținut către clienții conectați pe măsură ce acesta devine disponibil. SignalR suportă socket-uri web și revine la alte metode compatibile pentru browserele mai vechi. SignalR include API-uri pentru gestionarea conexiunilor (de exemplu, evenimente de conectare și deconectare), gruparea conexiunilor și autorizarea.

- Aplicații mobile și site-uri web

ASP.NET poate valorifica aplicații mobile native utilizând webAPI din partea serverului, precum și site-uri web mobile utilizând platforme de design adaptiv, cum ar fi Twitter's Initial Load. Dacă vă construiți propria aplicație mobilă, puteți crea cu ușurință un webAPI bazat pe JSON pentru a gestiona accesul la date, autentificarea și notificările push pentru aplicație. Dacă construiți un site mobil receptiv, puteți utiliza orice platformă CSS sau sistem de grilă deschis, sau puteți alege un sistem mobil puternic precum jQuery Mobile sau Sencha și aplicații mobile excelente cu PhoneGap.

- Aplicații cu o singură pagină

Aplicația ASP.NET cu o singură pagină (SPA) vă ajută să creați aplicații care includ interacțiuni semnificative pe partea de client utilizând HTML 5, CSS 3 și JavaScript. Visual Studio include un șablon pentru crearea de aplicații cu o singură pagină utilizând knockout.js și ASP.NET webAPI. În plus față de șablonul SPA încorporat, șabloanele SPA create de comunitate sunt, de asemenea, disponibile pentru descărcare.

- WebHooks

Web Interceptors este un model HTTP simplificat care oferă un model pub/sub simplu pentru combinarea webAPI-urilor și a serviciilor SaaS. Atunci când are loc un eveniment într-un serviciu, o notificare este trimisă abonaților înregistrați sub forma unei cereri HTTP POST. Cererea POST conține detalii despre eveniment care permit destinatarului să acționeze în consecință.

Interceptorii web sunt furnizați de un număr mare de servicii, inclusiv Dropbox, GitHub, Instagram, MailChimp, PayPal, Slack, Trello și multe altele. De exemplu, un interceptor web poate indica faptul că un fișier a fost modificat în Dropbox, o modificare de cod a fost efectuată în GitHub, o plată a fost inițiată în PayPal sau un card a fost creat în Trello.

Beneficiile tehnologiei ASP.NET

- Codul compilat se execută mai rapid, majoritatea erorilor fiind detectate în etapa de dezvoltare;
- Gestionarea erorilor îmbunătățită semnificativ în timpul execuției unui program gata de funcționare, utilizând blocurile try..catch;
- Controalele personalizate vă permit să evidențiați șabloanele utilizate frecvent, cum ar fi meniurile site-ului;
- Utilizarea metaforelor deja utilizate în aplicațiile Windows, cum ar fi controalele și evenimentele;
- Un set extensibil de controale și biblioteci de clase permite dezvoltarea mai rapidă a aplicațiilor;
- ASP.NET se bazează pe capacitățile multilingve ale .NET, permițându-vă să scrieți codul paginii în VB.NET, Delphi.NET, Visual C#, J# și așa mai departe;
- Posibilitatea de a memora în cache o parte sau întreaga pagină pentru a crește performanța;
- Abilitatea de a memora în cache datele utilizate pe pagină;
- Capacitatea de a separa elementele vizuale și logica de afaceri în fișiere diferite ("code behind");
- Model extensibil de procesare a interogărilor;
- Model extins de evenimente;
- Model extensibil de controale server-side;
- Disponibilitatea paginilor master pentru a seta modelele de design ale paginilor;
- Suport pentru operațiuni CRUD atunci când se lucrează cu tabele prin intermediul GridView;
- Suport AJAX încorporat;

7.1.2.3 ENTITY FRAMEWORK

Entity Framework este un set de tehnologii în ADO.NET care sprijină dezvoltarea de aplicații software centrate pe date.

Entity Framework vă permite să lucrați cu date sub formă de obiecte și proprietăți specifice domeniului (de exemplu, clienți și adresele acestora) fără a fi nevoie să luați în considerare formatul tabelelor și coloanelor bazei de date subiacente în care sunt stocate datele. Entity Framework oferă dezvoltatorilor posibilitatea de a lucra cu datele la un nivel superior de abstractizare, creând și menținând aplicații centrate pe date, reducând în același timp cantitatea de cod în comparație cu aplicațiile tradiționale. Deoarece Entity Framework este o componentă a .NET Framework, aplicațiile Entity Framework pot rula pe orice computer care rulează .NET Framework de la versiunea 3.5 cu Service Pack 1 (SP1).

Astfel, Entity Framework este o soluție de bază de date utilizată în programarea .NET. Acesta vă permite să interacționați cu SGBD utilizând entități mai degrabă decât tabele. De asemenea, este mult mai rapid să scrieți cod folosind EF.

De exemplu, atunci când lucrează direct cu baze de date, dezvoltatorul trebuie să se preocupe de conectare, pregătirea SQL și a parametrilor, trimiterea interogărilor și a tranzacțiilor. Entity Framework face toate acestea în mod automat - programatorul lucrează direct cu entitățile și îi spune doar lui EF să salveze modificările.

Entity Framework este un ORM, iar ORM-urile urmăresc să îmbunătățească productivitatea dezvoltatorilor prin reducerea sarcinii redundante de stocare a datelor utilizate în aplicații.

- Entity Framework poate genera comenzile de bază de date necesare pentru citirea sau scrierea datelor în baza de date și le poate executa pentru dvs.
- Dacă efectuați interogări, vă puteți exprima interogările către obiecte din domeniul dvs. utilizând LINQ pentru entități.
- Entity Framework va executa interogarea corespunzătoare în baza de date și apoi va materializa rezultatele în instanțe ale obiectelor domeniului dvs. pentru a le executa în aplicația dvs..

Entity Framework poate genera comenzile necesare bazei de date pentru citirea sau scrierea datelor în baza de date și le poate executa pentru dvs..

Caracteristicile

Următoarele sunt principalele caracteristici ale Entity Framework. Această listă se bazează pe cele mai importante caracteristici, precum și pe întrebările frecvente despre Entity Framework.

- Entity Framework este un instrument Microsoft.
- Entity Framework este dezvoltat ca un produs open source.
- Entity Framework nu mai este legat sau dependent de ciclul de lansare .NET.
- Funcționează cu orice bază de date relațională cu un furnizor Entity Framework valid.
- Generarea de comenzi SQL din LINQ pentru entități.
- Entity Framework va crea interogări parametrizate.
- Urmărește modificările aduse obiectelor în memorie.
- Vă permite să generați comenzi de inserare, actualizare și ștergere.
- Funcționează cu un model vizual sau cu propriile clase.
- Entity Framework stochează procedura de asistență.

Arhitectură

- Furnizori de date

Acestea sunt furnizori dependenți de sursă care abstractizează interfețele ADO.NET pentru a se conecta la o bază de date atunci când se programează pe baza unei scheme conceptuale.

Traduce limbajele SQL comune, cum ar fi LINQ, printr-un arbore de comenzi în propria expresie SQL și o execută pentru un anumit sistem DBMS.

- Entitatea Client

Acest nivel expune nivelul entității la nivelul superior. Entity Client oferă dezvoltatorilor posibilitatea de a lucra cu entități sub formă de rânduri și coloane utilizând interogări SQL de entități, fără a fi nevoie să creeze clase pentru a reprezenta schema conceptuală. Entity Client arată straturile structurii entității, care sunt funcțiile de bază. Aceste straturi se numesc model de date al entității.

  - Stratul de stocare conține întreaga schemă a bazei de date în format XML.
  - Stratul entităților, care este, de asemenea, un fișier XML, definește entitățile și relațiile.
  - Stratul de mapare este un fișier XML care mapează entitățile și relațiile definite la nivel conceptual cu relațiile și tabelele reale definite la nivel logic.
  - Metadata Services, care este prezent și în Entity Client, oferă un API centralizat pentru accesarea straturilor Entity, Mapping și Storage care sunt stocate în metadate.
- Stratul de stocare conține întreaga schemă a bazei de date în format XML.

Stratul de entități, care este de asemenea un fișier XML, definește entitățile și relațiile.

Un strat de mapare este un fișier XML care mapează entitățile și relațiile definite la nivel conceptual cu relațiile și tabelele reale definite la nivel logic.

Metadata Services, care este prezent și în Entity Client, oferă un API centralizat pentru accesarea straturilor Entity, Mapping și Storage care sunt stocate în metadate.

Obiectul serviciului

Stratul Object Services este un Object Context care reprezintă sesiunea de interacțiune dintre aplicații și sursa de date.

  - Principala utilizare a contextului obiectului este de a efectua diverse operațiuni, cum ar fi adăugarea, eliminarea instanțelor obiectului și salvarea stării modificate în baza de date prin intermediul interogărilor.
  - Acesta este stratul ORM Entity Framework care reprezintă rezultatul datelor pentru instanțele obiectului.
  - Aceste servicii permit dezvoltatorului să utilizeze unele caracteristici ORM bogate, cum ar fi maparea cheii primare, urmărirea modificărilor etc., prin scrierea de interogări utilizând LINQ și Entity SQL. etc., prin scrierea de interogări utilizând LINQ și Entity SQL.

Principala utilizare a contextului obiectului este de a efectua diverse operațiuni, cum ar fi adăugarea, eliminarea instanțelor obiectului și salvarea stării modificate în baza de date cu ajutorul interogărilor.

Acesta este stratul ORM Entity Framework care reprezintă rezultatul datelor pentru instanțele obiectului.

Aceste servicii permit dezvoltatorului să utilizeze unele caracteristici ORM bogate, cum ar fi maparea cheii primare, urmărirea modificărilor etc., prin scrierea de interogări utilizând LINQ și Entity SQL. etc., prin scrierea de interogări utilizând LINQ și Entity SQL.

Modalități de interacțiune cu baza de date:

Entity Framework sugerează trei moduri posibile de a interacționa cu o bază de date:

- Mai întâi baza de date: Entity Framework creează un set de clase care reflectă modelul unei anumite baze de date
- Model first: mai întâi dezvoltatorul creează un model de bază de date, pe care Entity Framework îl utilizează apoi pentru a crea baza de date reală pe server.
- Code first: dezvoltatorul creează o clasă model pentru datele care urmează să fie stocate în baza de date, iar apoi Entity Framework generează baza de date și tabelele acesteia pornind de la modelul respectiv

Beneficii:

Următoarele sunt principalele calități pozitive ale Entity Framework. Această listă include cele mai importante caracteristici ale cadrului, precum și cele despre care dezvoltatorii pun adesea întrebări.

- Entity Framework este un instrument creat de specialiștii Microsoft.
- Entity Framework este dezvoltat ca un produs open source.
- Entity Framework nu mai este legat sau dependent de ciclul de lansare .NET.
- Funcționează cu orice bază de date relațională și cu un furnizor Entity Framework valid.
- Generarea de comenzi SQL din LINQ to Entities.
- Entity Framework creează interogări parametrizate.
- Entity Framework vă permite să introduceți, să actualizați și să ștergeți comenzi.
- Funcționează cu un model vizual sau cu propriile clase.
- Entity Framework suportă proceduri stocate.

7.1.2.4 SERVERUL WEB IIS

IIS (Internet Information Services) este un server web dezvoltat de Microsoft pentru sistemele sale de operare. Produsul este complet proprietar și vine la pachet cu Windows.

Serverele web IIS au o arhitectură complet modulară care oferă trei beneficii principale:

- Componentizare
- Extinderea mediului
- Integrarea ASP.NET

Componentizare

Toate funcțiile serverului web sunt acum gestionate ca componente autonome care pot fi adăugate, eliminate și înlocuite cu ușurință. Acest lucru oferă câteva avantaje esențiale față de versiunile anterioare ale IIS:

- Protejați serverul prin reducerea zonei direcțiilor de atac. Reducerea zonei de contact este una dintre cele mai eficiente modalități de a proteja un sistem server. Cu IIS, puteți elimina toate funcțiile serverului neutilizate, obținând cea mai mică zonă de contact posibilă, menținând în același timp funcționalitatea aplicației.
- Performanță îmbunătățită și amprentă de memorie redusă. Prin eliminarea funcțiilor neutilizate ale serverului, puteți, de asemenea, să reduceți cantitatea de

memorie utilizată de server și să îmbunătățiți performanța prin reducerea cantității de cod component care este executat cu fiecare solicitare către aplicație.

- Creați servere personalizate sau specializate. Prin selectarea unui set specific de funcții de server, puteți crea servere personalizate optimizate pentru a îndeplini funcții specifice în topologia aplicației, cum ar fi cachingul de margine sau echilibrarea sarcinii. Puteți adăuga funcții personalizate pentru a extinde sau înlocui funcțiile existente utilizând componente de server proprii sau de la terți construite pe noile API-uri de extensibilitate. Arhitectura componentelor oferă beneficii pe termen lung comunității IIS: facilitează dezvoltarea de noi funcții de server pe măsură ce acestea sunt necesare, atât în cadrul Microsoft, cât și printre dezvoltatorii terți.

IIS implementează modelul eficient de activare a proceselor HTTP introdus în IIS 6.0 cu ajutorul grupurilor de aplicații. Modelul de activare a proceselor HTTP nu este disponibil numai pentru aplicațiile web care primesc cereri sau mesaje prin orice protocol. Acest serviciu independent de protocol se numește Windows Process Activation Service (WAS). Windows Communication Foundation (WCF) vine cu adaptoare de protocol care pot valorifica capacitățile WAS, sporind fiabilitatea și utilizarea resurselor serviciilor WCF.

Extinderea mediului înconjurător

Dezvoltatorii pot utiliza arhitectura modulară a IIS pentru a crea componente puternice de server care extind sau înlocuiesc caracteristicile existente ale serverului web și adaugă valoare aplicațiilor web găzduite în IIS.

Următoarele sunt aspectele pozitive pentru dezvoltarea serviciilor IIS.

1. Extinderea aplicațiilor web. Extinderea serviciilor IIS permite aplicațiilor web să beneficieze de funcționalități care, în multe cazuri, nu pot fi furnizate cu ușurință la nivelul aplicației. Folosind ASP.NET IIS sau extensibilitatea nativă C++, dezvoltatorii pot crea soluții care adaugă valoare tuturor componentelor aplicației, cum ar fi scheme de autentificare personalizate, monitorizare și logare, filtrare de securitate, echilibrare a sarcinii, redirecționare a conținutului și gestionarea stării.

2. Capacități de dezvoltare îmbunătățite. Modelul de extensibilitate C++ complet nou elimină majoritatea problemelor care afectau anterior dezvoltarea ISAPI, prezentând un API simplificat orientat pe obiect care facilitează scrierea unui cod robust pe partea de server. În plus, integrarea îmbunătățită cu Visual Studio sporește și mai mult capacitățile de dezvoltare pentru IIS.

3. Utilizarea tuturor caracteristicilor ASP.NET. Integrarea ASP.NET vă permite să dezvoltați rapid module de server utilizând interfețele familiare ASP.NET 2.0 și serviciile de aplicație ASP.NET bogate în caracteristici. Modulele ASP.NET pot furniza servicii pentru ASP, CGI, fișiere statice și alte tipuri de conținut și pot extinde complet serverul fără limitările prezente în versiunile anterioare ale IIS.

Integrare ASP.NET

Serviciile IIS permit aplicațiilor Web să profite pe deplin de caracteristicile puternice și de extensibilitatea ASP.NET 2.0. Caracteristicile ASP.NET, inclusiv autentificarea bazată pe formulare, apartenența, starea sesiunii și multe altele, pot fi utilizate în toate tipurile de conținut, oferind o singură interfață pentru întreaga aplicație Web. Dezvoltatorii pot utiliza modelele familiare de extensibilitate ASP.NET și API-urile API.NET bogate în caracteristici pentru a crea funcții de server IIS care sunt la fel de eficiente ca cele scrise cu API-uri native C++.

Modul de funcționare al IIS

Cererea utilizatorului este primită mai întâi de server și apoi transmisă către IIS pentru prelucrare. Modulul principal al serverului web este serviciul WWW. Serviciul procesează cererile utilizatorilor prin protocoalele HTTP/HTTPS.

Un singur server web gestionează mai multe site-uri în paralel. De exemplu, un server cu o adresă IP gestionează cereri pe un port TCP de la mai multe site-uri. În cadrul IIS sunt create înregistrări DNS pentru a identifica fiecare site.

În interiorul serverului web, sunt create directoare home pentru fiecare site, cu diferențierea drepturilor de acces pentru directoare. De exemplu, la prelucrarea cererilor de pe site-ul Yandex.ru, în IIS apare catalogul c:/Yandex, care primește automat toate informațiile la prelucrarea cererilor de pe acest site.

Siguranța

IIS pune la dispoziția utilizatorilor mai multe opțiuni de securitate a site-ului. Toate acestea sunt integrate în sistemul de operare Windows și diferă în ceea ce privește principiul de autentificare a utilizatorului.

- Conectare anonimă. Utilizatorul nu este identificat de sistemul de operare, deoarece nu trece prin procedura de autorizare. Utilizatorul primește un set minim de drepturi de acces.
- Autentificare de bază. Utilizatorul își introduce credențialele, care sunt transmise pe canale de comunicare deschise. De obicei, acestor clienți li se acordă un set limitat de drepturi.
- Autentificare Digest. Numele utilizatorului este transmis în clar, iar parola este criptată.
- Autentificare încorporată. Mecanismul de securitate se sincronizează cu sistemul și preia credențialele utilizatorului sub care acesta s-a autentificat cu sistemul de operare.
- Autentificare UNC. Legitimațiile utilizatorului sunt transmise către un server de la distanță unde clientul este autorizat.
- Autentificare prin certificat. Această opțiune utilizează un certificat SSL pentru validarea utilizatorului.

Există o componentă Crypto specială ca criptare pentru IIS. Aceasta este furnizată numai în sistemele de operare pentru servere începând cu Windows Server 2008. Este un utilitar complet gratuit.

Prin intermediul acestei componente, administratorul configurează criptarea pentru site folosind șabloane. Acestea vă permit să creați un set specific de reguli și apoi să le aplicați site-urilor.

IIS este conceput pentru a rula și administra site-uri web sub familia de sisteme de operare Microsoft. Serverul web este integrat ca o componentă în Windows și dispune de diverse funcții de autentificare.

7.1.2.5 NGINX WEB SERVER

nginx [engine x] este un server HTTP și un server proxy invers, un server proxy de e-mail și un server proxy TCP/UDP de uz general.

NGINX este un server web Open Source gratuit, foarte rapid și fiabil.

Principiul de funcționare

Spre deosebire de alte produse din acest segment, Nginx utilizează un principiu diferit de procesare a datelor primite. Software-ul împarte fiecare cerere a utilizatorului în mai multe cereri mai mici, simplificând astfel procesarea fiecăreia. În terminologia Nginx, acestea sunt numite conexiune de lucru.

După procesare, fiecare conexiune este colectată într-un singur container virtual pentru a fi transformată într-o singură cerere inițială și apoi trimisă utilizatorului. O singură conexiune poate gestiona simultan până la 1024 de cereri ale utilizatorului final.

Pentru a reduce sarcina RAM, serverul web utilizează un segment de memorie dedicat numit "pool". Acesta este dinamic și se extinde pe măsură ce lungimea cererii crește.

Domeniul de aplicare

Un server web este utilizat în următoarele situații:

1. Port sau adresă IP dedicată. Dacă există o cantitate mare de materiale statice (imagini, texte etc.) sau de fișiere care trebuie încărcate de utilizatori, Nginx este utilizat pentru a aloca o adresă IP sau un port separat pentru aceste operațiuni. În acest fel, sarcina pe server este distribuită.

2. Server proxy. Atunci când un utilizator încarcă o pagină cu conținut static, Nginx stochează mai întâi datele în cache și apoi returnează rezultatul. Data viitoare când această pagină este solicitată, răspunsul este mult mai rapid.

3. Echilibrarea încărcăturii. Atunci când este solicitată pagina unui site, utilizatorul primește un răspuns într-o secvență sincronă. Nginx utilizează modul asincron. Toate solicitările sunt procesate în etape diferite. Această abordare crește viteza de procesare.

4. Server de poștă electronică. Deoarece serverul web are mecanisme de autentificare încorporate, acesta este adesea utilizat pentru a redirecționa către serviciile de poștă electronică după ce un client a fost autorizat.

Nginx are mecanisme de securitate încorporate. Informațiile sunt transmise pe un canal criptat prin protocoalele SSL/TLS.

Cea mai comună practică este atunci când Nginx funcționează împreună cu Apache. Dacă solicitarea este pentru conținut static, Nginx se ocupă de aceasta. Dacă utilizatorul are nevoie de conținut dinamic (video, grafică), Apache este conectat aici.

Serverul web Nginx este ideal pentru site-urile cu conținut preponderent static. De asemenea, poate acționa ca redirecționare pentru serviciile de poștă electronică sau ca server proxy. Configurarea simplă și flexibilă vă permite să scalați produsul fără prea mult efort.

Caracteristici ale Nginx

- Viteză înaltă

Acest lucru este vizibil în special atunci când se lucrează cu conținut static care nu trebuie actualizat constant. Atunci când un utilizator încarcă o pagină web, serverul web Nginx stochează

mai întâi datele în cache și apoi returnează rezultatul. Data viitoare când este solicitată pagina, răspunsul este de câteva ori mai rapid.

- Flexibilitate

Software-ul este flexibil configurabil și personalizabil pentru a răspunde nevoilor infrastructurii.

- Consum redus de memorie

Pentru a reduce încărcarea memoriei RAM, Nginx utilizează un segment de memorie dedicat, "pool-ul". Acesta este dinamic și poate fi extins pe măsură ce lungimea cererii crește.

- Suport avantajos

Nginx are o comunitate activă și un bun suport pentru clienți, iar documentația este disponibilă în limba rusă.

- Accesibilitate

Software-ul este gratuit și distribuit sub o licență liberă. Este open source, astfel încât orice dezvoltator poate adapta Nginx la nevoile sale.

Funcționalitatea de bază a serverului HTTP

- Servirea interogărilor statice, fișiere index, crearea automată a listei de fișiere, cache cu descriptor de fișier deschis;
- proxy invers accelerat cu caching, echilibrare a sarcinii și toleranță la erori;
- Suport accelerat pentru serverele FastCGI, uwsgi, SCGI și memcached cu caching, echilibrare a sarcinii și toleranță la erori;
- Modularitate, filtre care includ compresie (gzip), intervale de octeți, răspunsuri fragmentate, filtru XSLT, filtru SSI, conversie de imagini; mai multe subinterogări pe o singură pagină prelucrate în filtrul SSI prin proxy sau FastCGI/uwsgi/SCGI sunt executate în paralel;
- Suport pentru SSL și extensia TLS SNI;
- Suport HTTP/2 cu prioritizare bazată pe ponderi și dependențe;
- suport HTTP/3. Servere virtuale identificate prin adresă IP și nume;
- Configurarea formatelor de jurnal, logare cu tampon, rotație rapidă a jurnalului, logare către syslog;
- Restricționarea accesului în funcție de adresa clientului, de parolă (autentificare HTTP Basic) și de rezultatul unei subcereri;
- Efectuarea de funcții diferite în funcție de adresa clientului;
- Limitarea vitezei la care sunt date răspunsurile;
- Limitarea numărului de conexiuni și cereri simultane de la o singură adresă;
- Geolocalizarea în funcție de adresa IP;
- Limbaj de scripting njs.

Funcționalitatea serverului proxy de e-mail

- Redirecționează utilizatorul către un server IMAP sau POP3 utilizând un server extern de autentificare HTTP;
- Verifică utilizatorul utilizând un server de autentificare HTTP extern și redirecționează conexiunea către un server SMTP intern;
- Metode de autentificare:

- POP3
- IMAP
- SMTP
- Suport SSL;
- Suport pentru STARTTLS și STLS.

7.1.2.6 MS SQL SERVER

Microsoft SQL Server este un sistem relațional de gestionare a bazelor de date (RDBMS) dezvoltat de Microsoft Corporation. Principalul limbaj de interogare utilizat este Transact-SQL, creat în comun de Microsoft și Sybase. Transact-SQL este o implementare a standardului ANSI/ISO pentru limbajul de interogare structurat (SQL) cu extensii. Este utilizat pentru a lucra cu baze de date de dimensiuni diferite, de la baze de date personale până la baze de date la scară mare de întreprindere; concurează cu alte SGBD din acest segment de piață.

Bazele de date relaționale sunt cele în care toate datele sunt stocate în tabele, iar aceste tabele sunt legate între ele: una de alta, alta de o a treia și așa mai departe.

SQL Server este baza platformei Microsoft de procesare a datelor, oferind performanțe fiabile și consecvente (inclusiv tehnologii de procesare în memorie) și ajutându-vă să extrageți mai rapid informații valoroase din orice date, indiferent dacă acestea se află pe site sau în cloud.

Avantajele SQL Server:

- Integrare strânsă cu sistemul de operare Windows.
- Performanță ridicată, toleranță la erori.
- Suport pentru mediul multi-user.
- Funcții avansate de backup al datelor.
- Lucrul cu conectivitate la distanță.
- Procesul de instalare

Instalarea Microsoft SQL Server este mai ușoară decât instalarea oricărui alt DBMS deoarece nu necesită un set special de instrumente. În plus, SQL Server este actualizat automat.

- Depozitare

Puteți gestiona datele între dispozitive, ceea ce economisește timp și simplifică foarte mult lucrul cu baza de date.

- Siguranța

Asigură securitatea și integritatea datelor. Structura tabelară cu funcții legate asigură faptul că este aproape imposibilă chiar și coruperea accidentală a datelor.

- Recuperarea datelor

are caracteristici de backup care simplifică recuperarea datelor. Acest lucru poate minimiza riscul de pierdere a datelor, indiferent de ce se întâmplă cu serverul dvs.

Fiecare versiune include mai multe ediții specializate. Acest lucru reduce complexitatea implementării și costul procesului de dezvoltare a soluțiilor personalizate adaptate pentru sarcini "înguste". Integrarea cu produsele Microsoft, de exemplu, cu platforma Visual Studio, este utilizată în mod activ la scrierea codului programului.

Funcții ale bazei de date:

- Stocarea permanentă a informațiilor.
- Căutare după criterii cheie.
- Citire și editare la cerere.

Clienții bazei de date sunt programele de aplicație, interfața acestora, diverse module interactive ale site-urilor web, cum ar fi calculatoarele și editorii online. Dar există o altă componentă a sistemului - DBMS. Acesta este conceput pentru accesul manual la informații și permite recuperarea datelor de pe disc, lucrul cu ele în memoria serverului, inclusiv utilizarea limbajului SQL structurat.

Există trei tipuri de baze de date - client-server, file-server și embedded. MS SQL Server face parte din prima categorie. În plus, sistemul este relațional, adică adaptat pentru stocarea datelor fără redundanță, cu riscuri minime de anomalii și încălcare a integrității tabelelor interne.

Extensii ale limbajului SQL

Limbajul SQL este un standard care unifică prelucrarea datelor de către toate bazele de date relaționale. Această abordare simplifică referințele încrucișate și face posibilă trecerea la o "platformă diferită" fără o reproiectare majoră a proiectului.

Transact-SQL (T-SQL) - utilizat în MS SQL Server.

Principalele componente și servicii ale serverului SQL:

Componentă a motorului bazei de date. Oferă stocare, procesare rapidă a tranzacțiilor și protecție a datelor.

SQL Server. Responsabil pentru pornirea, întreruperea și, în general, rularea unei instanțe de Microsoft SQL Server. Numele fișierului executabil este sqlservr.exe.

Agent SQL Server. Îndeplinește rolul de programator de sarcini. Numele fișierului executabil este sqlagent.exe.

Browser SQL Server. Ascultă cererea de intrare și se conectează la instanța SQL Server necesară. Numele fișierului executabil este sqlbrowser.exe.

SQL Server Full-text Search (Căutare full-text). Permite utilizatorului să efectueze interogări full-text asupra datelor cu caractere din tabelele SQL. Numele fișierului executabil este fdlauncher.exe.

SQL Writer Service (Serviciul SQL Writer). Vă permite să faceți copii de rezervă și să restaurați fișiere chiar și atunci când SQL Server este oprit. Numele fișierului executabil este sqlwriter.exe.

SQL Server Analysis Services (SSAS). Oferă capabilități pentru analiza datelor și învățarea automată. Numele fișierului executabil este msmdsrv.exe.

SQL Server Reporting Services (SSRS). Oferă funcții pentru raportare. Numele fișierului executabil este ReportingServicesService.exe.

SQL Server Integration Services (SSIS). Ajută la extragerea, transformarea și încărcarea diferitelor tipuri de date de la o sursă la alta. Cu alte cuvinte, pot fi considerate drept un convertor de informații brute. Numele fișierului executabil este MsDtsSrvr.exe.

7.1.2.7 POSTGRESQL

PostgreSQL este un sistem de gestionare a bazelor de date obiect-relațional

PostgreSQL este un popular sistem gratuit de gestionare a bazelor de date obiect-relaționale. PostgreSQL se bazează pe limbajul SQL și suportă numeroase caracteristici.

Pentru ce este PostgreSQL

- Acces, organizare și stocare flexibilă a bazelor de date.
- Gestionarea înregistrărilor în bazele de date: crearea, editarea și ștergerea, actualizarea versiunilor etc.
- Vizualizarea la cerere a informațiilor necesare din baza de date, de exemplu, pentru a le trimite către un site web sau o interfață de aplicație.
- Trimiterea de tranzacții, cereri secvențiale asamblate sub forma unui scenariu.
- Stabilirea și controlul accesului la anumite informații, gruparea utilizatorilor în funcție de nivelul de drepturi.
- Controlul versiunilor și organizarea accesului simultan la baza de date din surse diferite, astfel încât să se evite eșecurile.
- Protejarea informațiilor de posibile scurgeri și pierderi.

– Monitorizarea stării bazei ca întreg.

Avantajele PostgreSQL:

1. Oferă acces gratuit. Orice specialist poate descărca și instala SGBD gratuit și poate începe imediat să lucreze cu baze de date. Dacă plasați baza de date și DBMS în cloud storage, va trebui să plătiți.

2. Suportă cross-platform. PostgreSQL poate fi instalat pe orice platformă și poate rula pe orice sistem de operare: Linux, macOS, Windows. Utilizatorul primește sistemul "out of the box" - nu sunt necesare instrumente suplimentare pentru a instala și utiliza programul.

3. Suportă modelul relațional al obiectelor. În mod tradițional, SGBD-urile populare sunt relaționale. Aceasta înseamnă că datele stocate în ele sunt reprezentate ca înregistrări conectate între ele prin relații. Se formează liste conexe, care pot avea unele relații între ele - așa se formează un tabel. Modelul obiect suportă caracteristicile despre care am vorbit în articolul despre OOP, cum ar fi moștenirea.

PostgreSQL este un DBMS obiect-relațional. Aceasta înseamnă că suportă atât abordarea obiect, cât și cea relațională.

4.Suportă diferite formate de date. PostgreSQL acceptă multe tipuri de date și structuri diferite, inclusiv adrese de rețea, date în format text JSON și date geometrice pentru coordonatele geopoziției. De asemenea, are suport pentru baze de date XML și NoSQL. Toate aceste formate pot fi stocate și procesate în RDBMS.

Atunci când lucrați cu PostgreSQL, vă puteți crea propriile tipuri de date, denumite tipuri de date personalizate. Tipurile de date personalizate sunt necesare pentru a simplifica lucrul cu baza de date sau pentru a stabili constrângeri.

5. Vă permite să lucrați cu dimensiuni mari ale datelor Dimensiunea unei baze de date în PostgreSQL este nelimitată și depinde de cantitatea de memorie liberă disponibilă la locul de stocare: pe server, pe computerul local sau în cloud.

Dimensiunea maximă a tabelelor este de 32 terabytes. Acest lucru este mai mult decât suficient pentru a stoca datele unor companii precum Amazon. Un rând din baza de date nu poate depăși 1,6 terabytes, iar dimensiunea maximă a unei celule este de 1 gigabyte.

6. În conformitate cu cerințele ACID

Acronimul ACID vine de la:

- atomicitate,
- coerență,
- izolare,
- durabilitate.

Acestea sunt patru cerințe pentru funcționarea fiabilă a sistemelor care prelucrează date în timp real. Dacă toate cerințele sunt îndeplinite, datele nu vor fi pierdute din cauza erorilor tehnice sau a defecțiunilor echipamentelor.

Atomicitatea sistemului este capacitatea de a efectua tranzacții. Tranzacțiile sunt operațiuni sau acțiuni care au loc complet sau deloc.

Consistența înseamnă că o tranzacție va fi executată numai dacă nu încalcă consistența datelor din baza de date.

Izolarea sistemului înseamnă că acțiunile paralele nu se afectează reciproc. De exemplu, un transfer de bani între două conturi nu trebuie să afecteze un al treilea cont.

Reziliența sistemului înseamnă că o tranzacție care a fost deja executată nu va fi anulată din cauza unor probleme tehnice, de exemplu, dacă se sting luminile în camera serverului.

PostgreSQL îndeplinește toate cele patru cerințe ACID și asigură integritatea datelor pentru tranzacții și alte activități.

7. Acesta acceptă toate funcțiile disponibile în bazele de date moderne. De exemplu, PostgreSQL are funcții fereastră, tranzacții imbricate și declanșatoare.

Funcțiile fereastră vă permit să selectați anumite înregistrări dintr-un tabel și să efectuați calcule cu acestea într-o coloană separată. De exemplu, puteți adăuga o coloană cu data primei vizite a unui utilizator la un tabel cu date despre un magazin online. Această coloană va fi utilă dacă trebuie să calculați LTV (customer lifetime value).

Tranzacțiile imbricate sunt tranzacții în cadrul altor tranzacții. De exemplu, efectuarea unei serii de transferuri în tranșe în cadrul unui contract. Să presupunem că au fost executate cinci tranzacții, iar cea de-a șasea are probleme. Toate tranzacțiile anterioare care se aflau în interiorul unei tranzacții mari trebuie să fie anulate. Derularea înapoi a unei tranzacții înseamnă anularea tuturor modificărilor de date cauzate de această tranzacție.

În PostgreSQL, puteți crea declanșatoare - funcții care se execută automat în anumite condiții. De exemplu, puteți crea un declanșator care se execută atunci când datele despre o companie închisă sunt șterse din baza de date. Declanșatorul creat va adăuga automat o înregistrare în câmpul necesar al unui alt tabel: "datele companiei sunt șterse, compania este închisă".

8. Există un dialect propriu al limbajului SQL. Aproximativ 80% din operațiunile din SGBD sunt efectuate cu ajutorul interogărilor în limbajul clasic SQL. PostgreSQL are propriile interogări pentru anumite operații. Acestea sunt mult mai scurte, deci este mai ușor să lucrați cu ele.

9. Suportă interogări complexe. PostgreSQL lucrează cu interogări complexe, compuse. Sistemul face față sarcinilor de parsare și de efectuare a operațiunilor laborioase care implică în același timp citirea, scrierea și validarea. Este mai lent decât analogii săi atunci când vine vorba doar de citire, dar în alte aspecte își depășește concurenții.

10. Suportă funcții de scriere în mai multe limbi. În PostgreSQL, puteți scrie propriile funcții - blocuri de cod personalizate care efectuează anumite acțiuni. Această caracteristică este disponibilă în aproape orice DBMS, dar PostgreSQL acceptă mai multe limbi decât analogii săi. În plus față de SQL standard, PostgreSQL poate fi scris în C și C++, Java, Python, PHP, Lua și Ruby. Acesta suportă V8, unul dintre motoarele JavaScript, astfel încât JS poate fi, de asemenea, utilizat împreună cu PgSQL. Este implementat suport pentru Delphi, Lisp și alte limbaje rare. Dacă este necesar, puteți extinde sistemul pentru alte limbaje.

Modificarea SQL care este utilizată în PostgreSQL se numește PL/pgSQL. Este o extensie procedurală care acceptă calcule complexe și completează SQL "clasic" cu noi caracteristici.

11. puteți configura duplicarea sincronă a datelor Lucrul cu PostgreSQL acceptă replicarea logică. Replicarea este salvarea unei copii a bazei de date. Copia poate locui pe un server diferit. Cu replicarea logică, orice modificări sunt sincronizate în toate copiile bazei de date, indiferent de locul în care sunt stocate. Aceasta înseamnă că aceeași versiune a bazei de date va fi stocată peste tot.

12. Puteți migra date de la un alt SGBD fără pierderi Volumul de date al companiilor mari poate ajunge la 10 terabytes. Migrarea acestora va lua timp și va suspenda operațiunile. Companiile mici sau start-up-urile pot "migra" rapid la PostgreSQL de la un alt DBMS fără a pierde nimic în acest proces. Toate datele pot fi migrate folosind instrumente speciale.

13. Modificarea simultană a bazei de date. O caracteristică importantă a PostgreSQL este posibilitatea de accesare simultană a bazei de date de pe mai multe dispozitive. SGBD-ul implementează arhitectura client-server atunci când baza de date este stocată pe server și accesată de pe computerele client. De exemplu, acesta este modul în care sunt implementate diverse site-uri web. Una dintre dificultățile posibile este situația în care mai multe persoane modifică baza de date în același timp și trebuie evitate conflictele.

PostgreSQL utilizează în acest scop tehnologia MVCC (Multiversion Concurrency Control). Fiecare utilizator primește un snapshot - un "instantaneu" al bazei de date, în care sunt efectuate modificările. Numai după ce tranzacția este confirmată, acestea ajung în baza de date originală. În timp ce o persoană face modificări, acestea nu sunt vizibile pentru ceilalți utilizatori. Nu există conflicte și nu este necesar să se blocheze citirea sau scrierea.

Limitările actuale ale PostgreSQL:

- Nu există restricții privind dimensiunea maximă a bazei de date
- Nicio limită privind numărul de înregistrări din tabel
- Nu există restricții privind numărul de indexuri din tabel
- Dimensiunea maximă a tabelului este de 32 Tbytes
- Dimensiunea maximă a înregistrărilor este de 1,6 Tbytes
- Dimensiunea maximă a câmpului este de 1 Gbyte
- Numărul maxim de câmpuri într-o înregistrare - 250-1600 (în funcție de tipurile de câmpuri)

7.1.2.8 SERVICII DE RAPORTARE

SQL Server Reporting Services (SSRS) sunt servicii pentru proiectarea, crearea, livrarea și vizualizarea rapoartelor. Cu ajutorul acestor servicii, puteți crea rapoarte tabulare, interactive, grafice și alte rapoarte mai complexe utilizând diagrame și alte elemente de raportare.

SSRS este implementat ca serviciu web, iar administrarea sa și gestionarea rapoartelor se realizează prin intermediul unei interfețe web. Accesul standard la rapoarte este, de asemenea, oferit utilizatorilor prin intermediul interfeței web. SSRS oferă posibilitatea de a integra rapoartele dezvoltate în aplicații terțe, adică există unele funcționalități API. De exemplu, SSRS poate fi integrat cu SharePoint.

Pentru ca Report Server să poată stoca rapoartele publicate, modelele de rapoarte și ierarhia dosarelor, este necesară o bază de date SQL Server.

În SSRS puteți gestiona drepturile de acces la rapoarte, adică, de exemplu, puteți acorda unui grup (sau unui anumit utilizator) drepturi de acces pentru a vizualiza un raport, dar nu și altuia.

Reporting Services suportă stocarea în cache a rapoartelor, adică, de exemplu, aveți un raport a cărui execuție durează mult timp, dar ale cărui date se modifică rar, iar pentru ca raportul să fie executat mai rapid, acesta poate fi stocat în cache.

SSRS oferă posibilitatea de a se abona la rapoartele publicate, adică, de exemplu, după un program, utilizatorul va primi rapoarte noi prin poștă sau într-un dosar public.

Toate rapoartele create cu Reporting Services pot fi exportate în formate diferite (păstrând în același timp aspectul vizual), de exemplu:

- Excel;
- Word;
- PDF;
- CSV;
- XML.

7.1.2.9 DOCKER COMPOSE

Docker Compose este un set de instrumente inclus în Docker. Acesta este conceput pentru a rezolva sarcini legate de implementarea proiectelor.

Docker-compose este un add-on pentru Docker, o aplicație scrisă în Python care vă permite să rulați simultan mai multe containere și să direcționați fluxuri de date între acestea.

Pentru fiecare proiect (cluster de containere), Docker își creează propria rețea, în care containerele se pot referi unele la altele prin numele pe care le specificăm în docker-compose.yml. Toate setările de pornire ale clusterului de containere sunt în același fișier, care se află în directorul rădăcină al proiectului.

Tehnologia Docker Compose, în termeni simpli, vă permite să rulați mai multe servicii cu o singură comandă.

Diferențe între Docker și Docker Compose

Docker este utilizat pentru a gestiona containerele individuale (servicii) care alcătuiesc o aplicație.

Docker Compose este utilizat pentru a gestiona simultan mai multe containere care fac parte dintr-o aplicație. Acest instrument oferă aceleași caracteristici ca Docker, dar vă permite să lucrați cu aplicații mai complexe.

7.1.2.10 KUBERNETES

Kubernetes (K8s) este o platformă portabilă, open source, de orchestrare a containerelor. Acest instrument este utilizat pentru a automatiza implementarea de aplicații containerizate pe diferite gazde, precum și scalarea și gestionarea planificată a acestora.

Cu K8s, puteți gestiona un întreg cluster de containere, în loc să fiți distras de întreținerea unei singure mașini virtuale sau fizice cu containere implementate. În cadrul unui cluster K8s, containerele (Docker, containerd și CRI-O) pot fi lansate în mod programat, în funcție de cantitatea de putere de calcul disponibilă și de nevoile individuale de resurse ale fiecărui container.

K8s este inclus în multe distribuții populare, inclusiv Red Hat OpenShift, Rancher / SUSE, VMWare Tanzu și IBM Cloud. Acest suport larg permite Kubernetes să evite să fie legat de anumiți furnizori de software. Acest lucru oferă inginerilor DevOps care îl utilizează mai multe oportunități de a se concentra pe propriul lor produs.

Activitatea Kubernetes

Containerele din Kubernetes sunt combinate în obiecte logice numite poduri - seturi de una sau mai multe unități de bază gata să fie implementate pe noduri ("noduri" de mașini fizice sau virtuale).

Podurile sunt configurate și gestionate utilizând trei agenți sau utilități de bază: kubectl, kubelet și kubeadm. În plus față de agentul principal de control kubelet, fiecare "nod" rulează un proxy de rețea kube-proxy, care configurează regulile de rețea.

Ciclul de viață al podurilor poate fi ușor gestionat și scalat la starea dorită. Acest lucru oferă un nivel ridicat de control asupra stabilității și calității aplicațiilor implementate.

Avantajele Kubernetes

- Gestionarea simultană a containerelor pe mai multe gazde.
- Optimizarea resurselor echipamentelor utilizate.
- Implementarea automată și actualizarea aplicațiilor.
- Conectarea și adăugarea de stocare pentru a rula aplicații cu stare.
- Scalarea aplicațiilor containerizate și a resurselor acestora din mers.
- Gestionarea declarativă a serviciilor, care asigură controlul deplin asupra aplicațiilor implementate.
- Monitorizarea automată a sănătății și recuperării aplicațiilor cu ajutorul funcțiilor autorun, autoreplace, autoreplication și autoscale.

7.1.3. INSTRUMENTE DE INTEGRARE

7.1.3.1 SOAP

SOAP (Simple Object Access Protocol) este un protocol simplificat pentru schimbul de informații într-un mediu distribuit fără control centralizat. Mesajele SOAP sunt utilizate pentru a transfera informații între emițător și receptor. Prin combinarea mai multor mesaje SOAP, se pot crea modele cerere-răspuns.

Deși nu există nicio dependență de un protocol de transport specific, în majoritatea cazurilor protocolul SOAP este utilizat împreună cu protocolul HTTP pentru a oferi suport pentru infrastructura Internet existentă. SOAP vă permite să creați legături și să utilizați serviciile web descoperite prin descrierea căilor de rutare a mesajelor. SOAP este utilizat pentru a căuta servicii web în registrele UDDI. Spațiul de lucru acceptă SOAP 1.1.

SOAP este un protocol bazat pe XML care descrie cele trei părți ale fiecărui mesaj:

- Pachet. Un pachet definește un suport pentru descrierea conținutului unui mesaj și a modului în care acesta este prelucrat. Un mesaj SOAP este un pachet care conține un număr arbitrar de antete și un singur corp. Pachetul este elementul de nivel superior al unui document XML și acționează ca un container care conține informații de control,

adresa mesajului și textul mesajului. Antetele transmit informații de gestionare, cum ar fi atributele QoS. Corpul conține identitatea și parametrii mesajului. Antetele și corpul sunt elemente inferioare ale unui pachet.

- Reguli de codificare. Un set de reguli de codare descrie instanțele tipurilor de date ale aplicațiilor. Regulile de codificare specifică un mecanism de serializare pentru schimbul de instanțe ale tipurilor de date de aplicație. SOAP specifică o schemă de tipuri de date bazată pe XSD, independentă de limbajul de programare, precum și un set de reguli de codificare pentru toate tipurile de date. Deoarece codificarea SOAP este incompatibilă cu WS-I, se recomandă utilizarea tipului literal (fără codificare) pentru a implementa servicii web interoperabile și a oferi suport pentru WS-I.
- Stiluri de interacțiune. Operațiunile de interacțiune pot fi implementate într-un format de apel de procedură la distanță (RPC) sau de schimb de mesaje (document). Aceste stiluri de interacțiune sunt descrise mai jos.

Modele de legare

SOAP suportă două stiluri de interacțiune:

Apel de procedură la distanță (RPC): Un apel la o operațiune care returnează un rezultat. Utilizat de obicei împreună cu codificarea SOAP, care este incompatibilă cu WS-I.

Schimbul de documente: Acest stil implică schimbul de documente sau mesaje; descrie un nivel inferior de abstracțiuni care necesită mai multă programare.

Stiluri de codare

În mediile informatice distribuite, stilurile de codare descriu modul în care datele sunt convertite între formatul intern al aplicației și formatul unui anumit protocol. Procesul de conversie se numește serializare și deserializare.

Specificația SOAP specifică stilul de codificare SOAP:

Codificare SOAP: Stilul de codificare SOAP asigură serializarea/deserializarea valorilor tipurilor de date din modelul de date SOAP. Acest stil de codificare este descris în

WSDL descrie un tip literal de codificare:

Tip literal: Termenul literal indică faptul că documentul este citit în forma sa originală, adică fără codificare. Serializarea documentului se face ca XMI (codul sursă al mesajului XML este compilat în conformitate cu o schemă din WSDL). În cazul tipului literal, fiecare fragment de mesaj este asociat cu o definiție specifică a schemei. Tipul literal este compatibil cu WS-I.

Model de date

Modelul de date SOAP este menit să furnizeze tipuri abstracte de date independente de limbaj utilizate de tipuri comune de limbaje de programare. Acesta constă din următoarele tipuri:

- Tipuri XSD simple. De exemplu, int, string și date.
- Tipuri compuse. Tipurile compozite sunt împărțite în două categorii: structuri și matrici. Structurile sunt tipuri compozite numite, fiecărui element al cărora i se atribuie un nume unic (etichetă XML). Array-urile conțin elemente ale căror identificatori sunt poziția lor, nu numele lor.

Toate elementele și identificatorii care alcătuiesc modelul de date SOAP sunt definiți în URI-urile spațiului de nume. Standardul SOAP descrie regulile de creare a tipurilor de date. Schema XML a unui proiect trebuie să specifice tipurile reale de date.

Implementări SOAP

În prezent sunt disponibile diverse implementări ale protocolului SOAP. De exemplu, Fundația Apache furnizează Apache SOAP, bazat pe proiectul IBM SOAP4J, precum și mediile de execuție Apache Axis și IBM WebSphere.

Legături

O relație specifică legătura dintre numele complet al unui element XML, numele unei clase Java și un stil de codare. Maparea descrie metoda de transformare înainte și înapoi între un element XML de intrare cu un nume complet și o clasă Java în cadrul unei anumite codificări.

7.1.3.2 REST

REST API este o modalitate prin care site-urile și aplicațiile web pot interacționa cu un server. Este denumit și RESTful.

Termenul constă din două acronime, care sunt descifrate după cum urmează.

Un API (Application Programming Interface) este codul care permite două aplicații să facă schimb de date de la un server. REST (Representational State Transfer) este o modalitate de a crea un API utilizând protocolul HTTP.

Tehnologia REST API este utilizată ori de câte ori utilizatorul unui site web sau al unei aplicații web trebuie să furnizeze date de la server. De exemplu, atunci când un utilizator face clic pe o pictogramă video pe un site de găzduire video, API REST efectuează operațiuni și lansează videoclipul de pe server în browser. Acesta este în prezent cel mai comun mod de organizare a API-urilor. Ea a înlocuit metodele SOAP și WSDL, populare anterior.

RESTful nu are un singur standard de funcționare: este denumit un "stil arhitectural" pentru operațiunile serverului.

Principiile REST API

RESTful are 7 principii pentru scrierea codului de interfață.

Separarea clientului de server (client-server). Clientul este interfața cu utilizatorul a unui site web sau a unei aplicații, cum ar fi o casetă de căutare pentru găzduire video. Într-un API REST, codul de solicitare rămâne pe partea de client, în timp ce codul de acces la date rămâne pe partea de server. Acest lucru simplifică organizarea API-ului, facilitează adaptarea interfeței cu utilizatorul la o altă platformă și permite o mai bună scalabilitate a stocării datelor pe partea serverului.

Nicio înregistrare a stării clientului (Stateless). Serverul nu trebuie să stocheze informații despre starea clientului (operațiile efectuate). Fiecare cerere din partea clientului ar trebui să conțină numai informațiile necesare pentru a primi date de la server.

Casheable. Datele cererii trebuie să specifice dacă datele ar trebui să fie puse în cache (stocate într-un buffer special pentru cererile frecvente). Dacă acest lucru este specificat, clientul va avea dreptul de a accesa acest buffer dacă este necesar.

Interfață uniformă. Toate datele trebuie solicitate prin intermediul unei singure adrese URL, utilizând protocoale standard precum HTTP. Acest lucru simplifică arhitectura site-ului sau a aplicației și face interacțiunea cu serverul mai ușor de înțeles.

Sistem stratificat. În RESTful, serverele pot fi situate pe niveluri diferite, iar fiecare server interacționează numai cu cel mai apropiat nivel și nu este conectat cu solicitările de la alte niveluri.

Cod la cerere (Code on Demand). Serverele pot trimite cod către un client (de exemplu, un script pentru a rula un videoclip). În acest fel, codul general al unei aplicații sau al unui site web devine mai complex doar atunci când este necesar.

Începând cu stilul Null. Clientul cunoaște un singur punct de intrare către server. Alte posibilități de comunicare sunt furnizate de server.

Standarde și metode

RESTful în sine nu este un standard sau un protocol. Dezvoltatorii sunt ghidați de principiile API-urilor REST pentru a crea servere eficiente pentru site-urile și aplicațiile lor. Principiile permit ca arhitectura serverului să fie construită folosind alte protocoale: HTTP, URL, JSON și XML.

Acest lucru diferențiază API REST de metoda Simple Object Access Protocol (SOAP) creată de Microsoft. În SOAP, interacțiunea pentru fiecare protocol trebuie să fie scrisă separat numai în format XML. De asemenea, în SOAP nu există cacheabilitatea cererilor, documentație mai extinsă și o implementare a dicționarului separată de HTTP. Acest lucru face ca stilul REST API să fie mai ușor de implementat decât standardul SOAP.

Deși nu există standarde, există bune practici comune la crearea API REST, cum ar fi:

- utilizarea protocolului securizat HTTPS
- utilizarea instrumentelor de dezvoltare API Blueprint și Swagger
- aplicarea aplicației de testare Get Postman
- aplicarea cât mai multor coduri HTTP posibile (listă)
- arhivarea datelor mari

Arhitectură

REST API se bazează pe HTTP (Hypertext Transfer Protocol). Acesta este un protocol standard pe internet, creat pentru transferul de hipertext. Acum, orice alte tipuri de date sunt trimise folosind HTTP.

Fiecare obiect de pe server în HTTP are propriul său URL unic într-un format secvențial strict.

7.2. PREZENTAREA GRAFICĂ A STIVA TEHNOLOGICĂ PROPUSĂ PENTRU SI RA

În figură este prezentată topologia stivei tehnologice a soluției software SI RA (luând în considerare componentele, modulele și blocurile funcționale necesare).

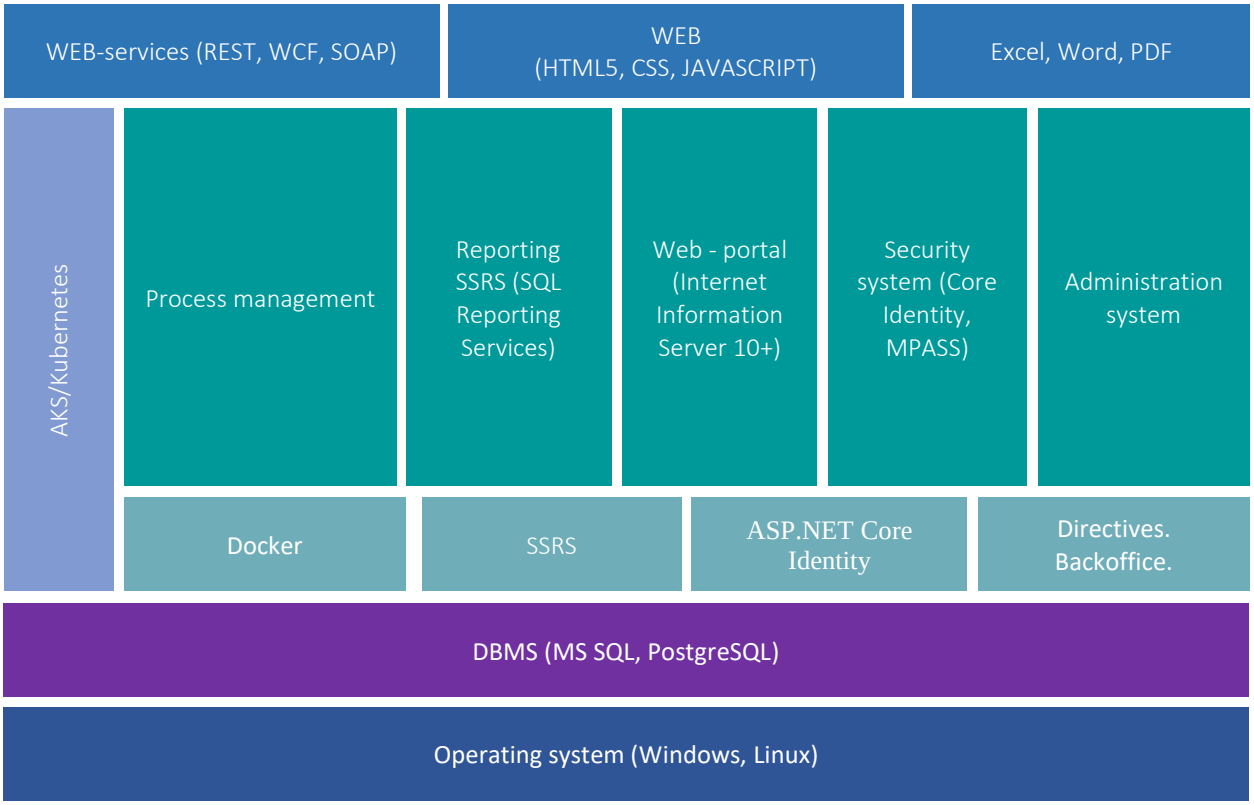


Figura 1 Arhitectura software și hardware, componente și blocuri funcționale

Componente software, standarde, platforme, tehnologii posibile utilizate:

Platforme
- Windows Server 2019+, SQL Server 2019+, Linux Debian 9+
- MCloud, VMWare, Kubernetes, Docker.
- Third-party government services - (MPass, MSign, MLog, MNotify).

Stocarea datelor
SQL Server 2019+, PostgreSQL

Căutare după conținut
- SQL Server, PostgreSQL Full Text Search

Interfața utilizatorului
- ReactJS, HTML, CSS,

Programare calendaristică
- Quartz .NET, Hangfire

Instrumente de integrare
- xUnit, Google POSTMAN

Instrumente și medii de dezvoltare
- Visual Studio 2022 Community, JetBrains Rider, Visual Studio Code
- SQL Server Management Studio, SQL Profiler, DevArt SQL Compare Tools.
- Telerik Kendo UI.
- SQL Reporting (SSRS).

Gestionarea configurației
- TFS Git
- TFS Agile Workflow

Elaborarea de rapoarte
- SQL Server Reporting Services (SSRS)

8. ”BUSINESS MODEL” AL DOMENIULUI AUTOMATIZĂRII

Funcționalitatea principală a sistemului

Sistemul va fi împărțit logic în mai multe module funcționale:

- Modul de registrul electronic de aeronave
- Modul de interacțiune și schimb de date
- Modul de raportare
- Modul de securitate și administrare a sistemului
- Modul de gestionare a cabinetului personal
- Modul de jurnalizare
- Modul de audit

Blocul de migrare inițială a datelor

SI RA oferă o soluție pentru următoarele sarcini funcționale:

8.1.1. Modul de registrul electronic de aeronave

– Sistemul va oferi posibilitatea de a înregistra aeronave;
– Sistemul va oferi efectuarea verificări automate ale datelor (fără duplicare, format, corectitudine etc.);
– Sistemul va oferi furnizarea informațiilor despre aeronave;
– Sistemul va oferi imprimarea:
  - certificatelor de înmatriculare;
  - certificatelor de radiere;
– Sistemul va oferi introducerea datelor privind:
  - modelele și tipurile de aeronave;
  - producătorilor de aeronave ,aeronavelor
  - deținătorilor și proprietarilor de aeronave
  - numerelor de înmatriculare
– Sistemul va oferi vizualizarea:
  - istoricului înmatriculărilor;
  - înmatriculării la nivel de proprietar de aeronavă;
  - înmatriculării la nivel de deținător de aeronavă.

8.1.2. Modul de interacțiune și schimb de date

– Sistemul va oferă posibilitatea de a realiza un schimb de date cu sistemele informaționale de administrare a autorităților publice prin intermediul serviciului de stat MConnect;
– Toate interfețele externe, vor fi realizate în sistem la bază standardelor deschise;
– Interacțiunea cu serviciile guvernamentale (MPass, MSign, MLog, MNotify, MPower) se va realiza direct, prin intermediul API oferite de acestea;

– Recepționarea datelor din Registrul de stat al unităţilor de drept se va realiza în baza identificatorului – IDNO.

8.1.3. Modul de raportare

– Sistemul va oferi utilizatorilor posibilitatea să recepționeze date statistice în format de rapoarte predefinite (numărul și componența rapoartelor va fi coordonat cu beneficiarul, nu mai mult decăt 5 rapoarte);

– Setul rapoartelor accesibile pentru utilizatori va fi stabilit în funcție de rolurile desemnate;

– Rapoartele vor fi elaborate în baza șabloanelor și algoritmilor preliminare , aprobate de către beneficiarul;

– Sistemul va permite generarea rapoartelor în următoarele formate: .xlsx, .docx, .pdf.

– Sistemul va furniza rapoarte generate pe baza datelor din baza de date a SI RA, în conformitate cu parametrii prestabiliteși din tabelele create prealabilă în baza de date la etapa de dezvoltare. Tabelele vor fi formate de către dezvoltatori, datele pentru tabele vor fi coordonate cu beneficiarul. Odată ce sistemul va fi implementată, date la aceste tabele pot fi modofocate sau adăugate numai de către dezvoltatori.

Notă: descrierea constructorului de rapoarte, care va fi implementat în sistem vezi în Anexa 2.

8.1.4. Modul de securitate și administrare a sistemului

8.1.4.1 Gestionarea utilizatorilor și controlul accesului

– Administratorii de sistem din cadrul SI RA, vor avea posibilitatea de a gestiona profilurile utilizatorilor, vizualizarea listei utilizatorilor;

– Căutarea profilului necesar al utilizatorului conform parametrilor de căutare;

– Adăugarea unui utilizator nou;

– Gestionarea accesului utilizatorului (activ/ anulat/ sistat);

– Variante de autentificare a utilizatorului (cu ajutorul nume de utilizator + parolă, cu ajutorul serviciului Mpass);

– Acțiunile tuturor administratorilor cu privire la gestionarea profilurilor utilizatorilor, vor fi înregistrate în registrul de acțiuni al utilizatorilor cu ajutorul serviciului guvernamental MLog.

8.1.4.2 Gestionarea rolurilor

– Sistemul de securitate și acces la date este proiectat în baza rolurilor;

– Fiecărui rol stabilit preliminar îi este atribuit un set de funcții și restricții;

– Acțiunile tuturor administratorilor privind gestionarea rolurilor vor fi înregistrate în registrul de acțiuni ale utilizatorilor cu ajutorul serviciului guvernamental MLog.

8.1.5. Modul de gestionare a cabinetului personal

– Sistemul va oferi un spațiul specific, care va oferi utilizatorilor date și funcționalitate doar lor accesibile – Cabinetul personal;

– La cabinetul personal va avea acces doar utilizatorul curent înregistrat;

– Din Cabinetul personal se solicită funcțiile de sistem accesibile unui anumit utilizator în conformitate cu drepturile acestuia, nivelul de acces și restricțiile stabilite pentru acest utilizator;

– În Cabinetul personal vor fi stocate date personale ale utilizatorului, setări specifice personale de sistem.

8.1.6. Modul de jurnalizare

– Sistemul va avea o listă de evenimente, ale căror informații pot fi salvate pentru analize ulterioare;

– Înregistrarea informațiilor privind acțiunile utilizatorilor și evenimentele sistemului va permite salvarea vizualizarea informațiilor necesare într-o formă convenabilă;

– Lista de evenimente și acțiuni ale utilizatorilor din sistem va fi coordonată cu beneficiarului.

8.1.7. Modul de audit

– Introducere manuală a inspecțiilor de audit;

– Vizualizarea istoricului inspecțiilor;

– Sistemul va audita activitățile și monitorizarea a activității utilizatorilor și acțiunilor dubioase.

– Sistemul va înregistra și va urmări alertele în scopuri de audit.

9. OBIECTELE INFORMAȚIONALE ALE SI RA

Pe baza funcționalității sistemului și a principalelor procese care au loc în cadrul acestuia, se disting următoarele obiecte informaționale:

a) Utilizatori;
b) Dosarul electronic.

10. ROLURI ÎN SISTEM

Principalele roluri ale sistemului vor fi următoarele:

c) Conducător
d) Inspector aeronautic
e) Destinatari ai datelor
f) Administrator

11. REALIZAREA CERINȚELOR GENERALE ALE SISTEMULUI

11.1. ARHITECTURĂ

Arhitectura SI RA va reprezenta o structură standard pe trei niveluri:

- BD;
- server de aplicații (server de logică de afaceri);
- client subțire (server web, aplicații web);

de asemenea, modulele de servicii web, care asigură interacțiunea cu sistemele informatice externe.

Serverul de aplicații asigură executarea logicii operaționale de bază a sistemului și servește ca element de interacțiune între partea utilizatorului (client) și sistemele externe cu baza de date a sistemului. Din punct de vedere funcțional, serverul de aplicații este format din 5 părți principale:

- sistemul "Business Logic" (toate funcționalitățile de bază ale sistemului, și anume sprijinirea și gestionarea ciclului de viață al obiectelor de înregistrare, cererile de servicii web, asigurarea interacțiunii dintre client și baza de date, efectuarea de verificări, eșantionarea, generarea și trimiterea de notificări, comunicarea cu sistemele externe);
- sistemul de securitate (gestionarea profilurilor, rolurilor, drepturilor și activității utilizatorilor);
- sistemul de logare (înregistrarea acțiunilor și evenimentelor utilizatorilor în sistem);
- sistemul de gestionare a clasificatorilor (întreținerea și actualizarea ghidurilor și a clasificatorilor);

- sistemul de gestionare a setărilor (gestionarea parametrilor și a modelelor comune la nivelul întregului sistem).

Baza de date este structurată logic din două părți - baza de date operațională și baza de date istorică. Împărțirea obiectelor de înregistrare în părți ale bazei de date se bazează pe statutul acestora pe parcursul ciclului lor de viață.

SI RA interacționează cu alte sisteme informatice prin intermediul serviciilor WEB și cu utilizatorii prin intermediul interfeței WEB.

Arhitectura SI RA va permite în timp dezvoltarea funcționalității sale și lucrul cu diferite tipuri de probe și surse de informații. De asemenea, va permite, dacă este necesar, scalarea produsului software atât în cazul unei creșteri a numărului de utilizatori, cât și în cazul unei creșteri a volumului de informații gestionate de acesta.

Pentru a dezvolta cu succes componente pentru soluția portalului actual, SI RA va utiliza soluții deschise, neproprietare, utilizate pentru aplicații WEB convenționale.

Baza de date, serverele care implementează business logica și furnizează interfața de utilizator a sistemului sunt localizate într-o locație centrală stabilită de client (este furnizat MCloud).

La sistem au acces doar utilizatorii înregistrați, cărora li se acordă drepturi în funcție de rolurile lor.

Interacțiunea utilizatorului cu sistemul se face prin intermediul interfeței WEB. Oferă o soluție atât pentru calculatoarele staționare, cât și pentru dispozitivele mobile (cu adaptarea rezoluției).

Schimbul de informații între stațiile de lucru ale utilizatorilor și serverele SI RA vor fi făcute în timp real pe internet, prin canale securizate.

Figura 2 demonstrează arhitectura software a SI RA.

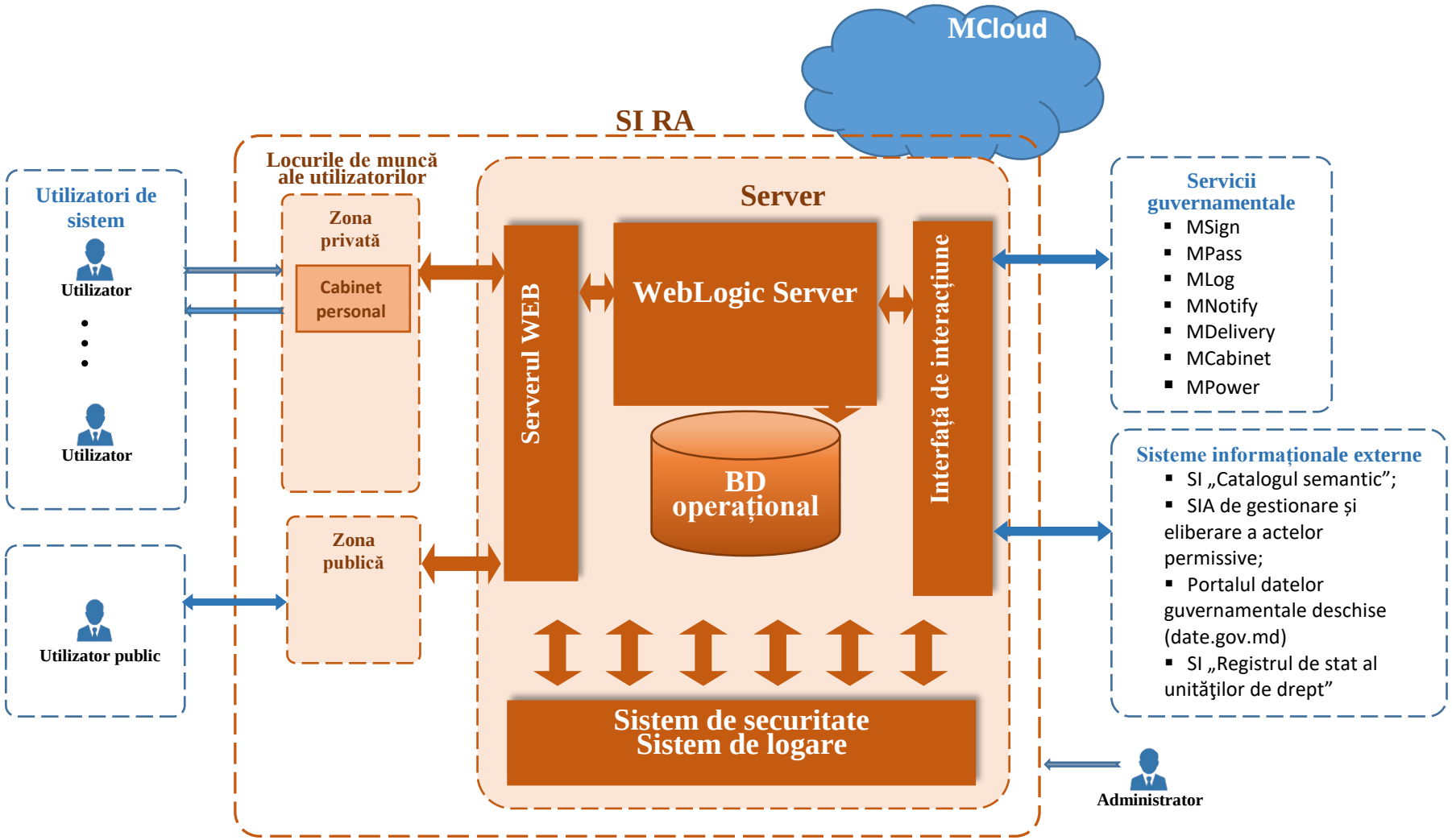


Figura 2 – Arhitectura SI RA

Cerinte de Arhitectură

– SI RA va fi construit folosind standarde deschise adecvate și nu va utiliza standarde proprietare;
– arhitectura va utiliza soluții de integrare bazate pe cele mai bune practici din industria IT;
– Si RA va fi construit pe baza soluțiilor orientate către servicii, va utiliza o arhitectură modulară bazată pe componente reutilizabile și interfețe abstracte;
– arhitectura sistemului va fi pe mai multe niveluri, cu o delimitare clară a nivelurilor;
– componentele sistemului vor fi relativ independente și vor interacționa între ele prin interfețe dedicate;
– sistemul va putea fi găzduit pe platforma guvernamentală comună MCloud;
– soluția software va fi flexibilă, inclusiv în cazul adăugării/înlăturării instanțelor container;
– acolo unde este posibil, modificările și parametrii sistemului nu vor afecta activitatea curentă a sistemului, sesiunile active, cererile etc.;
– arhitectura soluției software va oferi un grad ridicat de disponibilitate la introducerea de noi versiuni și va avea, de asemenea, capacitatea de a rula mai multe instanțe simultan;
– SI RA va fi compatibil cu cele mai noi versiuni de web-browsers - Chrome, Safari, FireFox, Edge;
Modelul de date va fi descris și pus la dispoziția beneficiarului într-un format convenit.

11.2. INTEROPERABILITATEA

Va fi realizată interacțiunea cu următoarele sisteme:

1. Platforma tehnologică guvernamentală comună (MCloud) – pentru găzduirea sistemului.
2. Platforma de interoperabilitate (MConnect) – pentru schimbul de date cu alte sisteme informaționale și registre de stat.
3. Serviciul electronic guvernamental integrat de semnătură electronică (MSign) – pentru semnarea documentelor electronice.
4. Serviciul electronic guvernamental de autentificare și control al accesului (MPass) – pentru autentificarea și controlul accesului în cadrul sistemului.
5. Serviciul electronic guvernamental de jurnalizare (MLog) – pentru asigurarea evidenței operațiilor (evenimentelor) produse în cadrul SI RA.
6. Serviciul electronic guvernamental de notificare (MNotify) – pentru notificarea registratorilor.
7. Sistemul informațional „Catalogul semantic”– pentru evidența metadatelor ca resurse informaționale.
8. serviciul guvernamental de livrare (MDelivery) – pentru livrarea fizică a rezultatelor prestării serviciilor publice sau private către persoane fizice și juridice.
9. Portalul guvernamental al unităților de drept (MCabinet) – pentru oferirea persoanelor juridice ce desfășoară activitate de întreprinzător a informațiilor oficiale de interes public și a informațiilor documentate despre sine, disponibile în cadrul registrelor, inclusiv al SI RA, și sistemelor informaționale ale furnizorilor de date.
10. Sistemul informațional automatizat de gestionare și eliberare a actelor permisive – pentru gestionarea și eliberarea actelor permisive.
11. Sistemului informațional „Registrul resurselor și sistemelor informaționale de stat”:
- sistemul informaţional „Registrul de stat al unităţilor de drept” – oferă acces la date cu privire la unităţile de drept.
12. Portalul datelor guvernamentale deschise (date.gov.md) – oferă date şi informaţii cu caracter deschis despre aeronavele înmatriculate în Registrul aerian al Republicii Moldova.
13. Registrul împuternicirilor de reprezentare în baza semnăturii electronice (MPower) – oferă împuterniciri de reprezentare în baza semnăturii electronice.

Interoperabilitatea cu sistemele informatice externe va fi realizată numai dacă sunt îndeplinite următoarele condiții:

1. Dacă un sistem informațional extern este pus în funcțiune sau sprijină posibilitățile de realizare (executare) a integrării cu SI RA.
2. În cazul în care sistemul informatic extern necesită primirea/trimiterea de date de la SI RA, atunci se implementează un API pe partea SI RA, dar nu în sistemul informatic extern, pentru a trimite sau primi date.
3. Beneficiarul este responsabil pentru furnizarea documentației API necesare pentru integrare. precum și de modificările din sistemele externe pentru a realiza integrarea cu SI RA.

11.2.1. Interfețe pentru interacțiune

Toate interfețele sistemului pentru interacțiunea cu sistemele externe se vor baza pe standarde deschise. Toate fluxurile de mesaje dintre sistem și entitățile externe vor utiliza standarde deschise.

Sistemul va fi integrat cu platforma de interoperabilitate MConnect pentru a utiliza datele din sistemele informatice de stat externe. Schimbul de date cu sistemele de stat externe va fi realizat prin intermediul MConnect.

Sistemul va oferi posibilitatea de a defini noi interfețe pentru a accesa sistemele externe utilizând standarde deschise. Aceste interfețe vor fi disponibile pentru apelarea funcțiilor sistemului atunci când se va realiza nevoia de interacțiune.

Toate funcțiile sistemului și serviciile furnizate utilizatorilor vor fi neutre din punct de vedere tehnologic - sisteme de operare standard, browsere de internet comune, nu vor fi necesare componente suplimentare (plug-in-uri etc.).

Toate interfețele sistemului vor fi documentate corespunzător.

11.3. RAPORTARE

– SI RA va furniza rapoarte standard privind situația cererilor și activitatea utilizatorilor, exportabile în diferite formate;

– SI RA va oferi posibilitatea de a crea rapoarte personalizate pe baza unor criterii și filtre specifice, utilizând un instrument de creare a rapoartelor;

– Sistemul va oferi instrumente de analiză pentru vizualizarea tendințelor și perspectivelor datelor, cu tablouri de bord personalizabile;

– Sistemul va oferi rapoarte generate pe baza datelor din baza de date, în conformitate cu parametrii prestabiliți și cu tabelele din baza de date. În baza de date, în stadiul de dezvoltare, tabelele vor fi formate de către dezvoltatori. Datele pentru tabele vor fi vor fi coordonate cu clientul. Clientul va putea să selecteze parametrii și să genereze rapoarte bazate pe datele din aceste tabele. Odată ce sistemul este implementat, va depinde de dezvoltatori să adauge alte date la aceste tabele sau să creeze altele noi.

Notă: descrierea constructorului de rapoarte, care va fi implementat în sistem, se vezi în Anexa 2.

11.4. SECURITATE

– SI RA va respecta cerințele Ordinului nr. 201 din 28-03-2017 privind aprobarea Cerințelor minime obligatorii de securitate cibernetică;

– Beneficiarului i se va furniza un set complet de documentație tehnică pentru sistem cu descrierea specificațiilor, configurației, amplasării componentelor, regulilor de acces la rețea și la componente etc. conform;

– Toate procesele din sistem vor funcționa cu un nivel minim de privilegii (documentația va specifica nivelul minim de privilegii pentru fiecare componentă);

– SI RA va utiliza MPass ca mecanism de autentificare a utilizatorilor pentru exploatarea interfeței web de către utilizatorii de e-mail;

– Pentru cazurile de utilizare a aplicațiilor client de e-mail prin SMTP/IMAP, se va utiliza un alt mecanism de autentificare și autorizare a accesului, care urmează să fie definit în procesul de proiectare a AIS pentru Ministerul Muncii și Protecției Sociale;

– SI RA va reduce la minimum datele de identificare personală care sunt stocate;

– SI RA va expune API-ul de pregătire și de verificare a stării de sănătate prin intermediul cererilor HTTP GET. Verificarea stării de sănătate va verifica starea cât mai multor componente ale e-IS către Ministerul Muncii și Protecției Sociale. Dacă se identifică o eroare, va fi trimis un mesaj de eroare lizibil de către om;

– SI RA va include un mecanism de timeout al sesiunii care va solicita utilizatorului să se conecteze din nou după o anumită perioadă de inactivitate. Perioada de inactivitate va putea fi configurată, iar valoarea implicită este de 15 minute;

– Toate datele de intrare vor fi validate atât de client, cât și de server;

– Toate caracterele UNICODE vor putea fi introduse/afișate în cadrul componentelor SI RA;

– Atunci când se încearcă accesul neautorizat, sistemul va genera un eveniment cu un nivel de eroare și va trimite un mesaj utilizatorilor că accesul nu este autorizat și va fi investigat;

– Sistemul nu va stoca acreditările în text clar.

Securitate și protecția datelor

Sarcina principală a securității informaționale a sistemului este de a proteja resursele și infrastructura informațională de acțiuni deliberate sau accidentale de natură naturală sau artificială, care conduc la prejudicierea utilizatorilor - participanți la procesul de schimb de informații și direct la date - informații despre obiectele contabilității.

Asigurarea securității informaționale a sistemului include o serie de activități și acțiuni, cum ar fi:

- măsuri care să asigure prevenirea amenințărilor,
- răspuns și contramăsuri,
- elaborarea politicii de securitate,
- aplicarea de tehnologii sigure,
- asigurarea unei infrastructuri sigure,
- utilizarea de canale de comunicare sigure și criptarea datelor,
- diferențierea accesului la funcții și date în sistem,
- înregistrarea acțiunilor utilizatorilor în sistem.

Furnizorul va identifica pericolele și va recomanda măsuri de protecție și controale pentru a asigura și implementa securitatea informațiilor din sistem.

Sistemul va oferi următoarele caracteristici de securitate:

- autentificare - asigură că sistemul este accesibil numai utilizatorilor cu o identitate validată și restricționată

- autorizare - utilizatorii autentificați pot accesa serviciile și datele corespunzătoare drepturilor lor de acces

- confidențialitate - asigură că terții neautorizați nu pot accesa datele înregistrate

- integritate - asigură că datele nu au fost modificate sau schimbate de către o terță parte neautorizată.

Furnizorul va lua în considerare, va preveni și va minimiza efectele periculoase ale următorilor factori în implementarea proiectului:

- utilizarea de tehnologii periculoase de prelucrare și stocare a datelor;

- introducerea de componente în produsele hardware și software care îndeplinesc funcții care nu sunt prevăzute în documentația referitoare la aceste produse;

- utilizarea de module și biblioteci care afectează funcționarea normală a sistemelor informatice și de telecomunicații, precum și a sistemelor de protecție a informațiilor;

- expunerea sistemului (cu parole de autentificare) la falsificarea sau autentificarea credențialelor;

- colectarea și utilizarea ilegală a datelor (prin acces diferențiat);

- scurgerea de informații prin canale tehnice neprotejate;

- accesul neautorizat la funcțiile sistemului și la resursele informaționale ale bazei de date;

Sistemul va oferi următoarele mecanisme tehnologice pentru a asigura protecția și securitatea datelor:

- accesul la date numai prin intermediul unei interfețe obiect unice;

- diferențierea accesului utilizatorilor la date în funcție de rolurile lor;

- gestionarea centralizată a datelor și controlul accesului;

- controlul acțiunilor utilizatorilor;

- utilizarea de canale de comunicare sigure și criptarea celor mai sensibile date.

În dezvoltarea sistemului se va utiliza o abordare de tip "siguranță prin proiectare".

Nivelul de completare obligatorie a câmpurilor de date va îndeplini cerințele clientului și va fi asigurat prin mecanisme de control logic și verificare a datelor, precum și prin structura bazei de date.

Toate procesele din sistem vor rula cu un nivel minim de privilegii.

Sistemul nu va stoca date sensibile (parole, date personale etc.).) în domeniul public.

Comunicarea între componentele sistemului va fi asigurată prin utilizarea interfețelor interne. Sistemul va oferi acces la interfețele sale externe folosind metode de autentificare securizate.

Sesiunea de utilizator din sistem va fi încheiată fie de către utilizator, fie automat la expirarea sesiunii. Toate datele despre sesiunea utilizatorului (ora de începere, ora de încheiere, ordinea de conectare și deconectare) vor fi salvate în Jurnalul de activitate al utilizatorului.

Sistemul va avea mecanisme eficiente pentru a preveni interceptarea neautorizată a sesiunilor active inițiate de utilizatori legitimi.

Sistemul de securitate pentru diferențierea accesului la funcții și date este bazat pe roluri.

Rolurile sunt create în sistem în funcție de responsabilitățile funcționale ale utilizatorilor. Fiecărui rol predefinit i se atribuie o listă disponibilă de funcții și limitări.

Unui utilizator i se pot atribui mai multe roluri, atunci drepturile sale vor fi egale cu suma drepturilor tuturor rolurilor atribuite. De asemenea, sistemul vă permite să delegați temporar drepturile unui utilizator altuia.

În plus, utilizatorilor li se pot acorda drepturi sau restricții speciale în funcție de funcțiile lor în sistem, afilierea teritorială, proprietățile obiectelor contabile, identificatorul utilizatorului etc.

Accesul la sistem este acordat numai utilizatorului care a trecut cu succes autentificarea (prin serviciul MPass sau Login-Password) și identificarea (pe bază de rol).

Definirea utilizatorilor și a rolurilor și atribuirea drepturilor anumitor utilizatori se face de către Administrator. Administratorul poate gestiona, de asemenea, listele de funcții și restricții disponibile pentru fiecare rol. În acest scop, el este prevăzut cu interfața corespunzătoare. În plus, administratorul poate gestiona stările contului de utilizator (stări permise: active, blocate, suspendate, anulate), poate actualiza datele de identificare din PGR sau poate limita perioada de valabilitate a drepturilor delegate utilizatorului.

Acțiunile administratorului privind gestionarea rolurilor vor fi înregistrate în Jurnalul acțiunilor utilizatorului utilizând serviciul MLog de stat

Clientului i se va pune la dispoziție un set complet de documentație tehnică privind sistemul cu descrierea funcțiilor și prevederilor care asigură securitatea informațiilor, specificații, configurare, plasarea componentelor, reguli de acces la rețea și componente etc.

11.5. MENTENAȚĂ

– va înregistra acțiunile și evenimentele într-un mod structurat. Înregistrarea va fi configurabilă și bazată pe un cadru de înregistrare extensibil. Cadrul de logare va suporta cel puțin formatul JSON.
– SI RA va diferenția evenimentele și acțiunile pe care le înregistrează în funcție de cel puțin următoarele niveluri: Critic, Eroare, Avertisment, Info, Depășire.
– Evenimentele de nivel Critic și Eroare vor fi înregistrate numai pentru erori nerecuperabile care necesită intervenție umană.
– Înregistrările jurnalului de evenimente vor include cel puțin:
- tipul evenimentului;
- marca temporală (ștampila de timp) a momentului în care s-a produs evenimentul;
- nivelul evenimentului;
- componenta de sistem care a generat evenimentul;
- utilizatorul/agentul utilizator, IP care a declanșat evenimentul;
- identificatorul obiectului informațional afectat;

– SI RA va implementa închiderea grațioasă, adică închiderea unei instanțe de container de aplicații nu va afecta activitățile în curs, cum ar fi sesiunile active, cererile, jurnalele de evenimente etc.
– Codul sursă va utiliza manageri de pachete pentru dependențele de biblioteci terțe. Toate programele suplimentare necesare vor fi incluse în definiția imaginii containerului și se vor baza pe depozitul public de containere;
– Furnizorul va aplica procedura și instrumentele de implementare necesare. Procedura de implementare va acoperi toate condițiile prealabile înainte de începerea instalării sistemului. Implementarea va fi automatizată și va include inițializarea bazei de date;
– Actualizările sistemului vor fi automatizate, inclusiv scripturi sau coduri de actualizare/dezactivare a bazei de date. Pentru a permite actualizărilor să ruleze în mediul de producție, se recomandă efectuarea unor modificări minore ale bazei de date.

11.6. FIABILITATE

Sistemul va rămâne operațional și se va asigura că funcțiile sale sunt restabilite în cazul următoarelor situații de urgență:

- în caz de defecțiuni în partea hardware sau software a stației de lucru a utilizatorului, rezultând repornirea sistemului de operare (în acest caz, restabilirea operabilității va avea loc după repornirea dispozitivului);
- în caz de defecțiuni tehnice ale stațiilor de lucru ale utilizatorilor (restaurarea funcției aplicației client este atribuită sistemului de operare al dispozitivului);
- în caz de erori legate de software-ul instalat pe stația de lucru a utilizatorului (recuperarea este responsabilitatea sistemului de operare).

La selectarea arhitecturii hardware a sistemului, o soluție de echilibrare a sarcinii va fi aplicată nodurilor hardware, inclusiv capacitatea serverului și stocarea datelor.

Toate bazele de date ale sistemului vor fi salvate pe un server extern cu o frecvență de o dată pe zi. Copiile de rezervă se păstrează cel puțin 3 luni. Hardware-ul va fi implementat ținând cont de redundanța activă/standby pentru a asigura transferul de sarcină către serverul standby în cazul în care serverul de lucru eșuează. Datele privind nodurile de lucru și de așteptare vor fi complet sincronizate.

Modul de așteptare la cald va fi aplicat bazei de date pentru a asigura integritatea datelor. Nodul primar și nodul Standby sunt utilizate în acest scop. SGBD utilizează înregistrarea înainte de scriere (Wal) pentru a arhiva continuu tranzacțiile bazei de date. Pentru fiecare modificare făcută fișierelor de date, WAL produce o intrare de fișier jurnal. Sistemul utilizează aceste intrări

de jurnal pentru a efectua restaurări punctuale din arhive și pentru a menține serverul de rezervă actualizat. În cazul unei defecțiuni, lucrul va fi comutat pe serverul de rezervă.

Arhitectura sistemului și interfața cu utilizatorul vor elimina apelurile accidentale către proceduri, funcții, comenzi. Toate apelurile către funcții, metode, proceduri vor fi verificate cu atenție pentru invocarea accidentală.

Verificarea blocului de condiții logice va asigura procesarea corectă a situațiilor cauzate de apelarea funcțiilor premature (eronate) sau introducerea valorilor nevalide ale datelor de intrare. În aceste cazuri, sistemul va suspenda funcționarea și va emite mesaje de alarmă adecvate utilizatorului, după care va reveni la starea de funcționare care a precedat comanda incorectă (nevalidă) sau introducerea incorectă a datelor.

11.7. STANDARDIZARE ȘI UNIFICARE

Armonizarea informațiilor și a datelor se va realiza prin:

- utilizarea unui sistem unificat de clasificare și codificare a obiectelor informaționale și a componentelor sistemului;
- utilizarea clasificatorilor naționali, sectoriali și a altor clasificatori standard utilizați în sisteme similare de funcționare a obiectelor contabile;
- utilizarea formularelor standard de documente (rapoarte) și limitarea rațională a compoziției tipului acestora (conform acordului cu clientul);
- aplicarea metodelor și mijloacelor unificate de colectare, pregătire, control și stocare a matricelor de informații ale sistemului.

Unificarea structurii se va realiza prin principiul modular al construcției arhitecturii și tastarea modulelor algoritmice.

Unificarea software-ului va fi realizată:

- utilizarea instrumentelor software standard pe cât posibil;
- utilizarea modulelor de programe unificate în dezvoltarea programelor de aplicații.
- utilizarea platformei DAAC implementate cu succes

Indicatori care stabilesc gradul necesar de utilizare a metodelor standard, unificate de implementare a funcțiilor sistemului, instrumente software furnizate, metode și modele matematice standard, soluții standard de proiectare:

- suport pentru Protocoale moderne de transport: TCP / IP, HTTP(S) ;
- suport pentru Internet-standarde: API RESTful;
- sprijinirea standardelor pentru implementarea mecanismelor de căutare;
- suport pentru cele mai comune formate de documente: Json, XML, HTML, Javascript;
- utilizarea arhitecturii microservice;
- suport pentru soluții de cluster cu echilibrare a sarcinii;
- abilitatea de a funcționa pe diverse platforme de sistem de operare, cum ar fi Windows, Linux.

Sistemul de codificare și clasificare utilizat pentru formarea informațiilor normative și de referință va îndeplini cerințele naționale pentru clasificarea și codificarea obiectelor contabile, precum și va lua în considerare experiența internațională în crearea unor astfel de sisteme.

Soluția dezvoltată va asigura unificarea sarcinilor funcționale, a operațiunilor și a interfețelor de utilizator.

12. COMPOZIȚIA ȘI CONȚINUTUL LUCRĂRILOR PRIVIND ELABORAREA SISTEMULUI

12.1. MONITORIZAREA PROIECTULUI

Monitorizarea implementării proiectului

Monitorizarea poate fi definită ca un proces continuu și metodic de colectare a datelor și informațiilor pe tot parcursul proiectului. Informațiile colectate sunt utilizate pentru evaluarea periodică a proiectului; astfel, ajustăm progresul proiectului fără a opri lucrările.

Monitorizarea implementării unui proiect implică măsurarea modului în care un anumit parametru sau set de parametri se modifică sau nu se modifică în timpul proiectului (de exemplu, timpul alocat unei etape, numărul de funcții îndeplinite etc.).

Caracteristici distinctive ale monitorizării:

- Desfășurat în mod continuu;
- Proiectul este acceptat ca atare, fără modificări. Rezultatele și activitășile planificate sunt comparate cu rezultatele și activitîțile reale;
- Informații obțimute sunt utilizate pentru a îmbunătăți activitatea proiectului

Obiectivele prinicpale ale monitorizării sunt de a se asigura că:

- activitățile planificate sunt implementate la timp. Abaterii sunt înregistrate. Activitățile corective se desfășoară pentru a se conforma calendarului activităților.
- metodologiile, tehnologiile și metodele implementate au un impact pozitiv asupra eficienței procesului de dezvoltare.
- există o tendință pozitivă în ceea ce privește indicatorii de performanță.
- obiectivele au fost îndeplinite.

Principii de monitorizare:

- Visualibilitatea - prezentare grafica a informatiei.
- Informativ - reducerea timpului și efortului necesar pentru a analiza datele diagramei.
- Comparabilitate - monitorizarea este regulată, dinamica îmbunătățirilor poate fi observată.
- Identificarea «valorilor aberante» - posibilitatea de a identifica tipuri speciale de procese care ar trebui prelucrate într-un anumit mod.
- Monitorizarea realizării obiectivelor - capacitatea de a evalua progresul către atingerea unui obiectiv într-un anumit interval de timp.

Elaborarae unui plan de monitorizare a implementării înainte de implementarea proiectului, împreună cu calendarul de implementare (sau în cadrul acestuia). După ce planul de monitorizare este pregătit și am început direct să implementăm proiectul, supraveghetorul (Project Manager) efectuează în mod regulat monitorizarea.

Monitorizarea necesită*:*

1. Identificarea indicatorii relevanți pentru obiective.
2. Stabilirea surselor de informații pentru calcularea indicatorilor.
3. Selectarea metodelor de colectare a informațiilor (chestionar, interviu, observație, studiu de documentare).

4. Determinarea frecvenței și calendarului colectării informațiilor și calculării indicatorilor.
5. Numirea persoanelor responsabile de colectarea, analizarea informațiilor și calcularea indicatorilor.
6. Alegerea tehnologiei de procesare și analiză a informațiilor.
7. Determinarea cine să transmită și cum să utilizați rezultatele analizei

Monitorizarea progresului proiectului se bazează pe standarde internaţionale de management de proiect, software adecvat şi bazat pe controlul planului de implementare.

Monitorizarea progresului proiectelor include următoarele activități:

- elaborarea calendarului-reţea (timetabil) pentru crearea obiectelor proiectului de investiţii şi explicaţiile acestora;
- controlul asupra alocării țintă a plăților, respectarea termenelor limită, etapelor și parametrilor de cost, precum și volumul real al lucrărilor efectuate și costurile suportate în toate etapele implementării proiectului;
- analiza abaterilor identificate de la planul de implementare a proiectului, inclusiv evaluarea impactului acestora asupra cadrului de timp și bugetului pentru îmbrăcăminte;
- elaborarea (pe baza rezultatelor analizei abaterilor) a propunerilor de activități care vizează implementarea proiectului în intervalul de timp planificat și cu un buget fix;
- previzionarea finalizării proiectului și a costurilor totale;
- organizarea supravegherii operaţionale asupra indicatorilor tehnico-metodologici ai proiectului, analiza documentaţiei tehnice de raportare;
- analiza rapoartelor auditorilor externi din partea Clientului sau a companiilor de supraveghere;

Automatizarea proceselor de schimb de informații între Furnizor și Client, precum și alte părți, rămâne o condiție esențială pentru monitorizarea eficientă a progresului proiectelor și realizarea termenelor și bugetului planificate, produse software bazate pe proiecte pentru managementul proiectelor (cum ar fi TFS, Microsoft Project etc.).

Compania noastră utilizează Microsoft Team Foundation Server pentru a gestiona întârzierile de produse, pentru a crea încărcări de lucru, pentru a descompune întârzierile de produse în activități și pentru a atribui sarcini membrilor echipei. Compania noastră consideră că este instrumentul perfect și adaugă multe îmbunătățiri utile la panourile/bachetele Kanban și Scrum. În plus față de managementul de proiect (inclusiv agil și cascadă), acesta oferă gestionarea codului sursă, raportare, managementul cerințelor, construcții automate, capacități de testare și de gestionare a lansărilor etc.

TFS sprijină echipa de dezvoltare în timpul planificării sprintului și a activităților zilnice Scrum; ajută la urmărirea stării activităților; verifică starea erorilor și rezultatele testelor; și au informații fiabile privind prezentarea generală a proiectului obținute din tablouri de bord personalizate.

12.2. DEZVOLTAREA SISTEMULUI SOFTWARE

Etapa este inițiată și executată numai în cadrul proiectului.

Managerul de proiect este responsabil pentru implementarea fazei de dezvoltare a sistemului software.

Următoarele artefacte sunt create în timpul fazei de dezvoltare:

a) Sarcina tehnică / Componența modificărilor;
a) modificări ale produsului software;
b) instrucțiuni de operare, actualizat;

c) ghidul administratorului, actualizat;

Sarcina tehnică/componența modificărilor se semnează de către contractant, de acord cu funcționarii implicați în proiect și cu reprezentantul autorizat al Beneficiarului. Sarcina tehnică se aprobată de către Beneficiar.

Specificațiile tehnice/compoziția modificărilor reprezintă singurul criteriu de evaluare a conformității produsului software.

Modificările instrucțiunilor de operare sunt convenite de către funcționarii implicați în proiect, de către șeful serviciului de suport (dacă este disponibil) și de către reprezentantul autorizat al Beneficiarului.

Modificările instrucțiunilor de operare vor fi aprobate de către conducătorul Prestătorului.

Modificările în Ghidul administratorului sunt convenite de către funcționarii implicați în proiect, de către șeful serviciului de suport (dacă este disponibil) și de către reprezentantul autorizat al Beneficiarului.

Modificările în Ghidul administratorului sunt aprobate de către conducătorul Prestătorului.

Toate procesele și activitățile principale ale fazei de dezvoltare sunt însoțite de testarea internă a codului programului.

12.3. TESTAREA SISTEMULUI SOFTWARE

Etapa de dezvoltare se încheie cu un proces de verificare, în timpul căruia se efectuează un test de calificare. Beneficiarul este responsabil pentru executarea acestui proces, iar managerul de proiect este responsabil pentru asistență.

La testarea de calificare participă reprezentantul autorizat al Beneficiarului, managerul de proiect, testerii, alți reprezentanți ai Beneficiarului și Prestătorului. Reprezentantul autorizat al Beneficiarului și managerul de proiect convin asupra compoziției participanților la testarea de calificare și asigură notificarea acestora.

Înainte de începerea testarea de calificare, reprezentanții Beneficiarului și ai prestătorului dezvoltă în comun scenarii de testare. Scenariile sunt aprobate de Beneficiar.

Erorile și comentariile identificate în timpul testării de calificare sunt analizate de participanții la test și sortate în trei categorii:

I – erori critice - erori care au dus la oprirea procesului tehnologic sau o defecțiune în funcționarea produsului software;

II - erori moderate - erori care cauzează neplăceri în funcționare sau au un impact negativ asupra performanței sau siguranței sistemului, precum și limitând ușor funcționalitatea sistemului (fără a întrerupe procesul) în termenii de referință;

III - erori în procesul de proiectare - erori sau comentarii care necesită extinderea sau modificarea funcționalității sistemului software care depășesc sarcina tehnică.

După analizarea problemelor, sunt posibile următoarele soluții:

- Erorile din prima și a doua categorie sunt eliminate înainte de trecerea în exploatare comercială/ operațiune de probă. Prin acord, se alocă timp suplimentar și se ajustează planul calendaristic de implementare a proiectului. După ce erorile sunt eliminate, produsul software este re-calificat de testare;
- Comentariile si erorile din categoria a treia sunt analizate de către reprezentanții Prestătorul împreuna cu reprezentanții Beneficiarului. Pentru fiecare articol din acest grup se ia o decizie separată, care este întocmită într-un protocol comun.

În absența erorilor critice și a comentariilor care nu permit începerea etapei de implementare, Prestătorul pregătește produsul software pentru actualizare și implementare a modificărilor.

12.4. SERVICII DE INSTRUIRE

Pe parcursul implementării proiectului, Prestătorul va oferi instruire personalului al Beneficiarului, numai în cadrul funcționalității a sistemului SI RA.

Numărul de cursanți din grup, timpul de pregătire și volumul de materiale pentru instruire vor fi coordonate cu Beneficiarul.

Notă: Instruirea va fi realiza conform planul de instruire, vezi Anexa 1.

12.5. SERVICII ÎN PERIOADA DE GARANȚIE A SITEMULUI SI RA

12.5.1. Lista serviciilor în perioada de garanție

Prestătorul oferă următoarele servicii în perioada de garanție a sitemului SI RA:

- Consultare a Beneficiarului în aspecte ce deține de identificare, analiza și formularea sarcinilor de mentenanța;
- Eliminarea erorilor din software (dezvăluite în timpul funcționării SI RA)
- Consultări ale personalului care administrează SI RA (asociate cu imposibilitatea utilizării sistemului);
- Linia telefonică de gestionare a incidentelor (analize, recomandări, depanare).
- Asistență în restabilirea performanței sistemului (în caz de defecțiune a componentelor software).

12.5.2. Planul de suport în perioada de garanție a sistemului SI RA

SI RA va fi acoperit servicii de garanție timp de 12 luni de la acceptanță. Garanția include toate modulele care au fost dezvoltate și furnizate în acest sistem informatic. Condițiile și volumul lucrărilor al serviciilor de mentenanță extinsă vor fi negociate de părți pentru fiecare an suplimentar Termenii și domeniul de aplicare al serviciilor de întreținere extinse vor fi convenite de părți pentru fiecare an suplimentar și se va încheia fie un contract de mentenanță, fie un act adițional la contract.

Furnizorul asigură servicii de garanție pentru utilizator și funcționalitatea sistemului informațional. Serviciul de garanție include următoarele grupuri de măsuri de activități:

- să repare defectele raportate de beneficiar;
- rezolvarea incidentelor raportate de beneficiar în conformitate cu SLA-urile stabilite;
- asistență pentru utilizatori/ linie fierbinte pentru utilizatori.

Furnizorul garantează furnizarea următoarelor servicii de garanție pentru software-ul dezvoltat și furnizat pe o perioadă de 12 luni de la data de acceptanță:

- fixarea erorilor, până la 3 zile lucrătoare;
- patch-uri (modificarea produsul software după livrare pentru a detecta și corecta defecțiunile latente ale produsului software înainte ca acestea să devină defecțiuni efective);
- Serviciul Call Center în zilele lucrătoare 5/8 (răspunde la orice apel de întreținere efectuat în termen de 1 zi lucrătoare);
- Consultanță a personalului al beneficiarului prin telefon sau e-mail 5/8;
- Serviciu la fața locului dacă este necesar, în următoarea zi lucrătoare;
- Vizită la fața locului, dacă este necesar, în următoarea zi lucrătoare.

12.5.3. Acord privind nivelul serviciilor (SLA)

Următoarele cinci (5) categorii de SLA-uri trebuie măsurate în mod auditabil de către furnizor, iar nivelurile de conformitate trebuie raportate beneficiarului în mod periodic:

SLA privind timpul de funcționare: Furnizorul va lua toate măsurile rezonabile pentru a asigura disponibilitatea sistemului și a aplicațiilor pentru toți utilizatorii. "Timpul de funcționare" al sistemului este un parametru măsurat prin scăderea intervalului de timp programat pentru întreținere din numărul de ore din luna în curs. "Timpul de funcționare" este apoi calculat ca procentaj de ore din luna respectivă în care sistemul este complet disponibil (utilizatorii pot efectua toate lucrările în aplicație, astfel cum sunt definite în cerințele aplicației, fără nicio degradare a serviciului).

SLA-uri privind timpul de răspuns: Furnizorul va fi responsabil pentru asigurarea faptului că timpul de încărcare a paginii aplicației/timpul de răspuns al sistemului respectă standardele industriale acceptabile pentru aplicații web utilizabile și receptive. Având în vedere natura distribuită a sistemului și multiplele puncte de integrare cu sistemele externe, timpii de încărcare a paginilor vor depinde în mod necesar de capacitatea de reacție a aplicațiilor externe. Pe parcursul implementării, furnizorul va colecta date privind timpii de răspuns ai sistemelor externe și timpii de răspuns ai paginilor care depind de aceste servicii externe. Pe baza acestor date, furnizorul și clientul vor elabora un set de timpi de răspuns acceptați de comun acord, având în vedere diferite scenarii de conectivitate a utilizatorului, pentru a susține o experiență rezonabilă a utilizatorului.

SLA privind monitorizarea sistemului: Furnizorul va fi responsabil de stabilirea procedurilor de monitorizare a resurselor sistemului, cum ar fi utilizarea CPU a serverului, utilizarea memoriei, spațiul pe disc și orice alte măsurători necesare pentru a asigura disponibilitatea sistemului și timpii rapizi de răspuns la întreruperile serviciilor. Aceste măsurători ar trebui, de asemenea, să ofere o notificare adecvată cu privire la faptul că resursele hardware ale sistemului se apropie de epuizare, astfel încât să poată fi luate măsuri de atenuare.

Furnizorul trebuie să notifice clientul că orice prag depășit de utilizarea sistemului va necesita modificări ale configurației sistemului și ale hardware-ului cu un preaviz de trei (3) luni.

Furnizorul trebuie să notifice Clientul că metricile au depășit limitele stabilite (propuse de Furnizor și acceptate de Client) care vor afecta accesibilitatea, comportamentul și stabilitatea aplicației în termen de o (1) oră de la depășirea acestor praguri.

Furnizorul trebuie să notifice clientul în termen de treizeci (30) de minute de la indicarea de către serviciile de monitorizare a faptului că aplicația nu este accesibilă.

Estimarea SLA-urilor pentru îmbunătățirea sistemului: În cazul în care, în perioada de performanță, Clientul identifică noi funcționalități necesare care nu au fost specificate inițial în cerințele tehnice, Consultantul va colabora cu Clientul pentru a defini cerințele funcționale pentru orice noi caracteristici. Odată ce o specificație funcțională pentru o nouă funcționalitate a fost dezvoltată și convenită, Consultantul trebuie să prezinte o estimare pentru dezvoltarea și livrarea completărilor sistemului în termen de o (1) săptămână.

Notă: elaborarea unor funcționalități noi care nu au fost specificate inițial în cerințele tehnice, va fi realizată după încheierea unui contract nou.

SLA-uri de asistență pentru clienți

- Tabelul - Clasificarea cererilor de servicii de asistență

Nivel de dificultate	Descriere	Timp de răspuns	Timp de rezoluție
Un grad de dificultate (critic)	O problemă de gravitate unu (1) este o problemă de producție catastrofală care poate afecta grav disponibilitatea soluției. Într-un astfel de caz, o parte sau toate componentele soluției sunt indisponibile sau nu funcționează;	imediat	2 ore

Nivel de dificultate	Descriere	Timp de răspuns	Timp de rezoluție
	pierderi de date de producție și nu există nicio soluție procedurală. Exemple de cazuri de gravitate unu: DB-ul devine corupt sau inaccesibil.		
A doua grad de dificultate (High)	O problemă de gravitate 2 (2) este o problemă în care Soluția funcționează, dar într-o capacitate foarte redusă. Situația provoacă un impact semnificativ asupra unor părți ale operațiunilor comerciale și asupra productivității Soluției. Sistemul este expus la o potențială pierdere sau întrerupere a serviciului. Exemplu de cazuri de gravitate doi: un nod al clusterului devine inactiv sau indisponibil, incapacitatea de a actualiza BD de către reprezentanții entităților sau administratorii soluției sau incapacitatea de a sincroniza datele între nodurile BD.	1 oră	4 ore
Gradul de dificultate trei (obișnuite) erori necritice	O problemă de gravitate trei (3) este o problemă cu impact mediu spre scăzut care implică o pierdere parțială de funcționalitate necritică, care afectează unele operațiuni, dar permite utilizatorilor/administratorilor soluției să continue să funcționeze. Aceasta poate fi o problemă minoră cu pierdere limitată sau fără pierdere de funcționalitate sau impact asupra funcționării clientului și probleme în care există o ocolire sau evitare ușoară de către utilizatorul final. erori critice	1 zi lucrătoare	3 zile lucrătoare (orice întrerupere necesară pentru rezolvare va fi luată în considerare la calcularea indicatorilor de disponibilitate a sistemului)
Gradul de dificultate patru (Low) erori non-critice	Problemă importantă, dar care poate aștepta, fără pierderi de funcționalitate sau impact asupra funcționării clientului și probleme care pot fi ușor ocolite sau evitate de către utilizatorul final.	1 zi lucrătoare	5 zile lucrătoare (orice întrerupere necesară pentru rezolvare va fi luată în considerare la calcularea indicatorilor de disponibilitate a sistemului)

-

12.5.4. Organizarea serviciului de asistență (service desk)

Pentru a asigura implementarea eficientă a software-ului și a serviciilor de garanție, compania va utiliza un serviciu de asistență automatizat (ServiceDesk) bazat pe standardele bibliotecii de infrastructură IT (ITIL) și ale gestionării serviciilor IT (ITSM).

Biblioteca ITIL descrie cele mai bune practici pentru gestionarea serviciilor corporative. Aceste abordări ajută companiile din întreaga lume să îmbunătățească calitatea și eficiența serviciilor lor.

Metodologia ITIL identifică trei aspecte principale ale gestionării serviciilor:

- Primul dintre acestea este reprezentat de organizații și oameni. Este necesar ca fiecare angajat și întreaga companie să fie ghidate de obiective comune. În același timp, personalul trebuie să aibă competențe suficiente pentru a îndeplini sarcinile de afaceri.
- Al doilea aspect este informația și tehnologia. Acestea sunt obiecte și instrumente de automatizare care sunt utilizate în gestionarea serviciilor IT.

- Al treilea aspect este reprezentat de partenerii și furnizorii companiei care contribuie la crearea și furnizarea de servicii către utilizatori.

Metodologia ITIL recunoscută la nivel internațional pentru software și suport tehnic descrie cele mai bune practici de organizare a activității și interacțiunii dintre departamentele IT ale companiei.

Avantajele unei soluții bazate pe ITIL/ITSM:

- Furnizarea de servicii ICO devine mai orientată către client, iar acordul privind nivelul de servicii contribuie la îmbunătățirea relațiilor;
- Serviciile sunt descrise mai bine, în limba clientului și cu detaliile necesare.
- Controlul calității și al costurilor serviciilor este îmbunătățit.
- Relația dintre client și contractant este îmbunătățită.

Serviciul Service Desk integrat în compania noastră îndeplinește funcțiile de primire și procesare a cererilor clienților, a cererilor de consultanță și a reclamațiilor privind funcționarea sistemului, precum și a cererilor de dezvoltare a sistemului sau de adaptare la schimbarea condițiilor de funcționare.

Avantajele utilizării sistemului Service Desk:

- punct de intrare unic pentru clienți - toate solicitările sunt trimise la un număr de telefon multicanal sau la o adresă de e-mail unică.
- procedura de restabilire a furnizării normale a serviciilor este simplificată, cu pierderi minime pentru clienți, în cadrul nivelurilor de servicii și al priorităților de afaceri convenite.

Orice solicitare primită prin intermediul Service Desk trece prin următorii pași:

- Primirea cererii este prima interacțiune cu clientul.
- Înregistrarea unui incident sau a unei cereri.
- Efectuarea unei evaluări inițiale a cererii, încercarea de a o rezolva sau determinarea persoanei care o poate rezolva.
- Numirea unui specialist care va fi responsabil de rezolvarea problemei.
- Identificarea problemei.
- Rezolvarea problemei.
- Determinarea nevoii de formare a clienților;
- Închiderea incidentului și notificarea clienților.

12.5.5. Condiții de suport

1. Dacă există probleme în funcționarea sistemului, beneficiarul formează și trimite o cerere către furnizor. Furnizorul de sistem trebuie să primească cereri numai de la persoana responsabilă desemnată în numele beneficiarului. Solicitările nu ar trebui să fie dublate, iar condițiile și cerințele din solicitare ar trebui să fie descrise în detaliu.

2. Timpul de răspuns și de execuție - (a se vedea Termenii și condițiile de răspuns), adică executarea acțiunilor de trimitere a unui mesaj de confirmare către beneficiar cu privire la înregistrarea acestei cereri și soluționarea directă a problemei depind de momentul depunerii cererii și de complexitatea soluționării acesteia. Timpul maxim pentru soluționarea unei probleme poate fi mărit în funcție de complexitatea problemei sau de alte condiții.

3. Programul de livrare a serviciului standard în zilele lucrătoare de la 9: 00 la 18: 00.

4. Furnizorul de servicii oferă clientului posibilitatea de a-l contacta în următoarele moduri:

- expediere de e-mail la adresa de e-mail convenită a furnizorului;
- apel telefonic la numărul corporativ al furnizorului.
- utilizarea serviciului Service Desk.

5. Toate cererile sunt înregistrate în sistemul de informații al prestatorului - Service Desk. Informațiile privind cererea înregistrată sunt trimise Prestatorului prin e-mail. O listă a cererilor deschise cu statusuri actualizate este trimisă Beneficiarului la cerere.

6. Atunci când cererea Beneficiarului este închisă, Beneficiarului i se trimite o notificare. Beneficiarul trebuie să verifice rezultatul și să își dea avizul (acceptată/neacceptată).

7. Nivelul de prioritate (”mic”, ”mediu”, ”înalt”) și nivelul de criticitate sunt stabilite pentru fiecare cerere. Acești indicatori sunt stabiliți de Beneficiar la momentul depunerii cererii. Prioritatea este acordată cererilor cu un nivel de criticitate ”Înalt”. Dintre cererile cu un nivel de criticitate, se acordă prioritate cererilor cu un nivel de prioritate ”Înalt”.

8. În cazul în care cerințele nu sunt clare sau suficient de detaliate, sau cerințele sunt prea generale (de exemplu, " adăugarea unei noi coloane "Divizie" la toate rapoartele - în acest caz, nu se specifică în ce rapoarte trebuie adăugată această coloană și care sunt regulile de completare a acestei coloane), cererea este returnată Beneficiarului pentru clarificări și detalieri suplimentare ale cerințelor, iar executarea acestei cereri este suspendată. Pentru cererile pentru care nu se primesc clarificări și explicații în mai mult de 5 zile lucrătoare, prestatorul trimite Beneficiarului o notificare de 3 zile privind închiderea cererii. Astfel de cereri trebuie închise la 3 zile de la data notificării.

9. Metode de furnizare a serviciilor de mentenanță: acces la distanță prin VPN sau RDP sau alte instrumente, consultații telefonice sau prin e-mail.

Notă: descrierea metodologiei de prestare a serviciilor privind suportului tehnic pentru realizarea proiectului utilizată în cadrul companiei noastre poate fi găsită în fișierul atașat ”Metodologia de prestare a suportului tehnic.pdf”.

12.5.6. Termeni și condiții de suport

Termeni și condiții de răspuns

Tabel. Condiții și termeni de răspuns pentru suport de garanție a sistemului

Nivelul de neconformitate	***Descrierea discrepanței***	***Timp maxim de răspuns/timp maxim pentru rezolvarea unei probleme (business hours, days)***
Înalt	*Probleme care au un impact semnificativ asupra asigurării funcționalității sistemului. Astfel de inconsecvențe pot limita utilizarea întregului sistem sau a unei părți a acestuia.*	*Timp maxim de răspuns: 8 ore.* *Timp maxim pentru rezolvarea unei singure probleme: 24 de ore*
Mediu	*Probleme legate de încălcări ale funcționalității sistemului care pot fi rezolvate fără riscul pierderii de date sau al deteriorării funcționalității sistemului.*	*Timp maxim de răspuns: 2 zile.* *Timp maxim pentru rezolvarea unei singure probleme: 5 zile*
Mic	*Probleme cu impact redus sau inexistent asupra funcționalității sistemului. De exemplu, probleme cu afișarea mesajelor sau modificări ale documentației.*	*Timp maxim de răspuns: 3 zile.* *Timp maxim pentru rezolvarea unei singure probleme: 10 zile*

13. CHECKLIST

13.1. CERINȚE FUNCȚIONALE ȘI TEHNICE PENTRU SI RA

Soluția tehnică propusă implementează cerințele descrise în caiet de sarcini pentru dezvoltarea Sistemului informațional „Registrul aerian al Republicii Moldova”, după cum urmează:

ID	Cerință funcțională	Corespunde Da/Nu
31.	**DOCUMENTELE SISTEMULUI SI RA** În cadrul SI RA se folosesc următoarele categorii de documente: 1) documentele de intrare, care sunt fundamentale pentru introducerea datelor în sistem; 2) documentele de ieşire, obţinute în urma funcționării sistemului; 3) documentele tehnologice.	da
45.	**SI RA conține următoarele categorii de informații:** 1) Registrul aerian al Republicii Moldova; a) Furnizarea informațiilor despre aeronave; b) Imprimarea certificatelor de înmatriculare; c) Imprimarea certificatelor de radiere; d) Introducerea modelelor de aeronave e) Introducerea tipurilor de aeronave f) Introducerea producătorilor de aeronave, g) Introducerea aeronavelor h) Introducerea deținătorilor de aeronave i) Introducerea proprietarilor de aeronave j) Introducerea numerelor de înmatriculare k) Introducerea înmatriculărilor l) Introducerea înmatriculărilor m) Vizualizarea istoricului înmatriculărilor n) Vizualizarea înmatriculării la nivel de proprietar de aeronavă o) Vizualizarea înmatriculării la nivel de deținător de aeronavă Audit/inspecții AAC.	da
	2) Rezultatele auditurilor și inspecțiilor de supraveghere efectuate de AAC a) Introducere manuală a inspecțiilor de audit b) Vizualizarea istoricului inspecțiilor	da
	3) Următoarele sisteme informaționale partajate sunt integrate în cadrul SI RA: a) MConnect – platforma guvernamentală de interoperabilitate care facilitează integrarea SI RA cu alte sisteme și resurse informaționale pentru realizarea schimbului de date; b) MPass – pentru autentificarea utilizatorilor autorizați SI RA; c) MSign – pentru semnarea formularelor electronice completate și documentelor/rapoartelor generate de fluxurile de lucru implementate de SI RA; d) MLog – pentru a înregistra evenimentele critice în timpul funcționării SI RA; e) MNotify – pentru a notifica utilizatorii autorizați SI RA.	da
	4) SI RA interacționează și realizează schimbul de date cu Registrul de stat al unităților de drept – care furnizează date oficiale de înregistrare a persoanelor juridice;	da
	Toate funcționalitățile disponibile sunt grupate pe roluri și disponibile din cadrul interfețelor SI RA. Selectarea rolului curent are loc automat conform rolului atribuit utilizatorului din interfața de administrare, partajarea mai multor roluri de una și aceeași persoana nu este disponibilă.	da
	5) Rolurile atribuite în SI RA a) conducător – este îndeplinit de către director/directorii adjuncți.	da

	Interfaţa permite accesul la următoarele funcționalități: • căutarea dosarelor; • căutarea documentelor aferente procedurilor administrative inițiate; • rapoartele; • profilul de date privind aeronava înmatriculată; • vizualizarea datelor despre utilizatorii SI RA; • generare/creare/modificare/editare/eliminare dosar; • și altele, după caz. b) inspector aeronautic – este îndeplinit de către persoanele calificate corespunzător, responsabile de înmatricularea aeronavelor și supravegherea respectării cerințelor privind navigabilitatea acesteia din cadrul AAC. Interfaţa permite accesul la următoarele funcționalități: • elaborarea și editarea documentelor privind înmatricularea, radierea aeronavei din registru, consemnarea și modificarea datelor din registru, suspendarea, revocarea și emiterea duplicatului certificatului de înmatriculare; • căutarea dosarelor; • modificarea/Crearea documentelor/editarea; • generare rapoarte; • și altele, după caz. c) destinatari ai datelor – persoane, din cadrul entităților publice, care au rolul de vizualizare a informațiilor furnizate de SI RA, fără dreptul de a crea, a introduce, a redacta, a radia, a arhiva datele din SI RA. Subiecții respectivi au următoarele roluri distincte: • dispun de acces la seturi de date din SI RA; • dispun de acces la rapoarte generate de SI RA în regim de vizualizare. d) administrator - este îndeplinit de persoana responsabilă de introducerea utilizatorilor în SI RA din cadrul AAC. Interfaţa permite accesul la următoarele funcționalități: • adăugarea/ștergerea/modificarea datelor utilizatorilor; • administrarea utilizatorului și controlul accesului; • configurare SI RA; • monitorizarea SI RA.	
	6) Obiectele informaționale ale SI RA Dosarul electronic reprezintă un obiect informațional ce conține totalitatea metadatelor și actelor aferente salvate în format electronic. Utilizarea noțiunii de dosar electronic va permite consolidarea unei arhive electronice detaliate care va conține documentele și informația detaliată aferentă.	da
	7) SI RA cuprinde următoarele funcționalități: a) Furnizarea informațiilor despre aeronave; b) Imprimarea certificatelor de înmatriculare; c) Imprimarea certificatelor de radiere; d) Introducerea modelelor de aeronave, indicând: • Denumirea • Producătorul e) Introducerea tipurilor de aeronave, indicând denumirea; f) Introducerea producătorilor de aeronave, indicând denumirea; g) Introducerea aeronavelor indicând: • Numărul de fabricație (Serial Number) • Anul de fabricație • Tipul Aeronavei (prin selectare din catalog existent) • Producătorul Aeronavei (prin selectare din catalog existent)	da

	• Modelul Aeronavei (prin selectare din catalog existent) h) Introducerea deținătorilor de aeronave, indicând. Datele ar trebui să fie obținute în mod automatizat prin intermediul platformei de interoperabilitate (MConnect): • Denumirea • Adresa • IDNO i) Introducerea proprietarilor de aeronave indicând. Datele ar trebui să fie obținute în mod automatizat prin intermediul platformei de interoperabilitate (MConnect): • Denumirea • Adresa • IDNO j) Introducerea numerelor de înmatriculare, indicând marca (numărul va fi generat automat după aceea); k) Introducerea înmatriculărilor indicând: • Aeronava (prin selectarea din catalogul existent) • Numărul de înmatriculare (prin selectarea din catalogul existent) • Proprietarul (prin selectarea din catalogul existent) • Deținătorul (prin selectarea din catalogul existent) • Dată emitere inițială Certificat de Înmatriculare • Dată înmatriculare • Număr Certificate de Înmatriculare • Volum registru • Dată sfârșit valabilitate l) Introducerea înmatriculărilor indicând: • Aeronava (prin selectarea din catalogul existent) • Numărul de înmatriculare (prin selectarea din catalogul existent) • Proprietarul (prin selectarea din catalogul existent) • Data de emitere inițială a Certificatului de Înmatriculare • Dată de înmatriculare • Numărul Certificatului de Înmatriculare • Volum registru • Data de sfârșit a valabilității • Data suspendării • Data revocării • Data radierii • Numărul de Certificat de Radiere • Observații (opțional); m) Vizualizarea istoricului înmatriculărilor utilizând diferite criterii de filtrare stabilite de comun acord de către Beneficiar împreună cu Prestatorul; n) Vizualizarea înmatriculării la nivel de proprietar de aeronavă; o) Vizualizarea înmatriculării la nivel de deținător de aeronavă	
	8) Audit / inspecții AAC cuprinde rezultatele auditurilor și inspecțiilor de supraveghere efectuate de AAC la agenții aeronautici civili și urmărirea acțiunilor subsecvente (planuri de măsuri acceptate de AAC); a) Conținutul se referă la: • Introducere manuală a inspecțiilor de audit indicând: • Domeniul/domeniile inspectat(e)/auditat(e) • Data inspecției • Echipa • Organizația inspectată/auditată	da

	• Neconformități identificate • Clasa/Gradul/Tipul neconformității • Planul de măsuri • Acțiuni întreprinse • Stadiul acțiunilor b) Vizualizarea istoricului inspecțiilor utilizând diferite criterii de filtrare stabilite de comun acord cu beneficiarul.	
	c) Normalizarea datelor existente din fișierele Excel și import date existente în noul sistem.	Da, Beneficiarul trebuie să furnizeze fișierele Excel conform structura prezentată de la Furnizorul
	SPAȚIUL TEHNOLOGIC AL SI RA	
46.	SI RA este găzduit pe platforma tehnologică guvernamentală comună (MCloud).	da
47.	Serviciul WEB SI RA interacționează extern pentru a asigura acces la informația cu caracter public și va fi accesat prin intermediul adresei web (https://ra.gov.md);.	da
48.	Sistemul informațional va furniza utilizatorului o interfață Web de acces la Registrul aerian al Republicii Moldova, accesibilă prin intermediul unui explorator Internet (MS Internet Explorer, Mozilla FireFox, Google Chrome, Opera, Safari, etc.). Din punct de vedere funcțional, se va realiza o soluție fiabilă și scalabilă atât în cazul creșterii numărului de utilizatori ce folosesc resursele sistemului informatic, cât și în cazul creșterii volumului de informație gestionată de acesta	da
49.	Utilizatorul are acces prin intermediul rețelei globale Internet la compartimentele publice ale SI RA .	da
50.	Subiectul antrenat în procesele de operare a SI RA este Autoritatea Aeronautică Civilă;	da
51.	Accesul utilizatorilor autorizați la SI RA se va face limitat, autorizarea fiind obținută prin intermediul serviciului MPass. Toate conexiunile utilizatorilor la sistem se vor efectua prin intermediul sesiunilor criptate	da
	ASIGURAREA SECURITĂȚII INFORMAȚIONALE A SI RA	
52.	Securitatea informațională se subînțelege ca starea de protecție a sistemului la etapele procesului de creare, procesare, stocare și transmitere a datelor de la acțiuni accidentale sau intenționate cu caracter natural sau artificial, care au drept scop crearea prejudiciului participanților în procesul de schimb informațional. Sistemul complex al securității informatice reprezintă totalitatea mijloacelor legislative, organizatorice și economice precum și a mijloacelor tehnologice și metodelor de protecție softwarehardware și criptografică a informației, orientate spre asigurarea unui nivel necesar de integritate, confidențialitate și accesibilitate a resurselor informaționale. Principalele sarcini ale asigurării securității informaționale sunt: 1) asigurarea integrității informației - protecția împotriva modificării sau ștergerii datelor; 2) asigurarea confidențialității – protecția împotriva accesului neautorizat la date; 3) asigurarea accesibilității – protecția împotriva blocării accesului utilizatorilor autorizați la resursele informaționale.	da
53.	SI RA sprijină mecanisme de securitate adecvate în scopul protejării și securizării datelor din sistem prin implementarea	da

	următoarele obiective de securitate: 1) Autentificarea: Garantează faptul că zonele restricționate ale serviciului sunt accesibile numai utilizatorilor cu identitatea verificată prin MPass. 2) Autorizarea: Garantează că utilizatorii autentificați pot accesa numai serviciile și datele care corespund rolurilor și drepturilor de acces ale acestora. 3) Confidențialitatea: Garantează că datele nu pot fi interceptate sau accesate de o terță parte neautorizată și că datele nu pot fi accesate într-un moment necorespunzător. 4) Integritatea: Garantează că fluxul de date nu a fost modificat sau manipulat de o terță parte neautorizată sau datele nu au fost accesate înainte de un termen anumit sau un timp anumit. 5) Non-repudierea: Măsură prin care se asigură faptul că, după emiterea/recepționarea unei informații, expeditorul/destinatarul nu poate nega, în mod fals, că a expediat/primit informații.	
54.	În cadrul Sistemului SI RA se asigură generarea și păstrarea înregistrărilor de audit ale securității pentru operațiunile de prelucrare a datelor cu caracter personal, în condițiile cadrului normativ în materie de protecție a datelor cu caracter personal, precum și prin integrarea în cadrul SI RA a serviciului MLog.	da
55.	Utilizatorii interni sunt autorizați să acceseze doar blocurile funcționale și datele pentru care au permisiunile necesare, conform rolurilor fiecăruia.	da
56.	O necesitate importantă legată de securitate constă în păstrarea înregistrărilor de audit pentru analiza integrității sistemului și pentru monitorizarea activității utilizatorilor.	da
57.	Infrastructura sistemului necesită funcționarea non-stop a SI RA și include circuite cu noduri de rezervă ce preiau încărcătura sistemului pe perioada cât acesta se află în proces de mentenanță sau restaurare.	da, va fi realizată de către Specialiștii infrastructurii Mcloud.
	CERINȚE PRIVIND ARHITECTURA RA	
	a. Arhitectura sistemului va fi client - server. b. Sistemul trebuie sa fie scalabil. c. Actualizarea datelor trebuie să se facă on-line, în timp real. d. Toate datele, inclusiv fișiere atașate, vor fi centralizate pe server, pentru exploatarea din Intranetul AAC; a) Soluția informatică va utiliza o arhitectură de tip client-server pentru dezvoltarea sistemului. Această structură trebuie să asigure separarea clară între logica de prezentare, accesibilă utilizatorilor prin interfața client, și logica de procesare, situată pe server. Implementarea trebuie să faciliteze comunicarea eficientă între client și server, suportând scalabilitatea și flexibilitatea în gestionarea cererilor utilizatorilor și a procesării datelor. b) RA trebuie să realizeze și să permită auditarea, monitorizarea și regăsirea tuturor tranzacțiilor efectuate în sistem; c) RA va oferi mecanisme proprii (log) pentru: - controlul operațiunilor importante efectuate asupra datelor - activitatea utilizatorilor - modificări drepturi de acces - consistența bazelor de date d) RA trebuie să asigure mecanisme împotriva pierderii	da

	accidentale de date. e) RA să permită integrarea cu alte baze de date, integrarea cu email, fax, scanări inclusiv OCR f) Arhitectura aplicației server dar și clienții trebuie să fie independente de platforma și să poată rula pe sistemele de operare Linux, Windows, Unix. Această cerință va respecta alinierea la standardele stabilite pentru MCloud și se va realiza în coordonare cu Serviciul Tehnologia Informației și Securitate Cibernetică. g) Ca și suport de date, sa suporte baze de date moderne inclusiv MySQL, MariaDB, SQL Server, PgSQL - sau superior. h) Soluția informatică va include suport pentru servicii de tip SMTP/IMAP și, în conformitate cu prevederile HG nr.376/2020 pentru aprobarea Conceptului serviciului guvernamental de notificare electronică MNotify și a Regulamentului privind modul de funcționare și utilizare a serviciului guvernamental de notificare electronică MNotify, va utiliza exclusiv pentru colectarea datelor de contact ale destinatarilor necesare expedierii notificărilor electronice serviciul guvernamental de notificare electronică Mnotify. m) Să permită securizarea datelor cu drepturi de tip citire, creare, actualizare, ștergere pentru fiecare element din structura organizatorică n) Să includă servicii web REST pentru toate datele stocate în aplicație o) Securitatea sa se aplice pentru toate datele indiferent cum sunt interogate	
	i) Să se poată adăuga câmpuri noi în interfață de către beneficiar fără intervenția furnizorului (și fără a scrie cod), să se poată reordona sau modifica câmpuri existente (pentru câmpurile noi sistemul va genera coloane fizice în baza de date pentru a păstra normalizarea bazei de date și posibilitățile de indexare native în RDBMS)	Toate formele de interacțiune a beneficiarului cu sistemul trebuie să fie convenite și aprobate înainte de dezvoltare, în timpul fazei de proiectare a sistemului. Sistemul va avea numai interfețe preprogramate.
	i) Să se poată configura direct în interfața sistemului orice validări pe câmpuri	Funcția de validare a câmpurilor de date ar trebui să fie definită și convenită în prealabil, în faza de proiectare a sistemului, pentru a asigura o dezvoltare eficientă și pentru a preveni erorile ulterioare. Această logică de validare va include reguli și constrângeri specifice pentru fiecare câmp de date, cum ar fi tipul de date, lungimea maximă/minimă, formatele acceptabile și alte condiții specifice. Aceste reguli vor fi documentate și aprobate de a începe faza de dezvoltare, pentru a se asigura o înțelegere clară a cerințelor. Această abordare asigură că toate validările sunt

		bine înțelese și corect implementate.
	j) Să includă un mecanism configurabil de fluxuri de lucru, care să poată fi adaptat după eventualele schimbări în procedurile interne	Beneficiarul trebuie să definească și să aprobe toate fluxurile de lucru în sarcina tehnică, pe durata fazei de proiectare a sistemului.
	k) Să includă un sistem de raportare integrat (fără licențe suplimentare) l) Generarea de rapoarte să se poată face în baza mai multor criterii selectate de către utilizator, cu posibilitatea preluării datelor din două module simultan	Formularele de rapoarte trebuie să fie furnizate de către beneficiarul (până la 10 rapoarte), precum și parametrii pe baza cărora vor fi generate rapoartele. Pentru a genera rapoarte cu parametri selectabile, predefinite în BD sistemului, se va utiliza mecanismul de raportare AdHoc (a se vedea anexa 2).
	p) Să funcționeze pe orice browser modern inclusiv pe dispozitive mobile	Aplicația va funcționa în cea mai populară rezoluție la momentul actual - 1366*768 și mai mult. Se recomandă să se folosească ecrane mari pentru a îmbunătăți utilizarea a aplicației.
	q) Configurația sistemului va fi stocata într-un instrument de configuration management si deployment. Astfel aplicația va putea fi reinstalata, mutata pe alt server sau actualizata oricând in mod automatizat. r) Să permită instalarea, dezinstalarea și actualizarea de module fără oprirea aplicației	Desfășurarea și administrarea serverului de aplicații și a bazei de date vor fi efectuate de către administratorul sistemului, cu ajutorul documentației tehnice
	INSTRUIREA	
	Servicii de instruire prestate de furnizor vor include: a) Instruirea directă a administratorilor sistemului; b) Instruirea directă a utilizatorilor; c) Materiale electronice tip eLearning pentru utilizatori. Scopul programului de instruire este de a asigura administrarea şi operarea sistemului. Toate programele de instruire trebuie să fie însoţite de activităţi practice, documentaţii şi manuale. Materialele de instruire se pun la dispoziţia cursanţilor cu cel puţin 10 zile înainte de data de desfăşurare a cursurilor. Materialele de instruire vor fi livrate în format electronic, în limba română. Ofertanţii trebuie să prezinte procedura după care va realiza instruirea utilizatorilor. Procedura va conţine cel puţin următoarele informaţii: a) Descrierea cursurilor şi a rezultatelor aşteptate; b) Modalitatea de evaluare a cursurilor; c) Formulare utilizate. Ofertanţii vor prezenta un plan de instruire care să conţină toate serviciile solicitate. De asemenea, Ofertantul trebuie să prezinte în cadrul propunerii tehnice curricula cursurilor de instruire	Da, vezi anexa 1
	MENTENANȚĂ, PLANUL DE MENTENANȚĂ ȘI SUPORT	
	1. Mentenanța soluției pe durata contractului va include toate update-urile de la producător pentru modulele contractate,	Da, conform punctul

	complianța legislativă, precum și inspecția periodică a bazei de date. Prestatorul va efectua pe o perioadă de 1 lună servicii de monitorizare, prevenție și remediere erori (mentenanță corectivă, adaptivă) cu scopul remedierii și evitării problemelor de funcționare în viitor. Acestea pot include monitorizarea log-urilor soluției, monitorizarea parametrilor de funcționare (utilizare procesor, spațiu alocat, memorie utilizată etc), optimizarea indecșilor folosiți de bazele de date, executarea periodică a diverselor proceduri critice sau monitorizarea performanței aplicației. Prestatorul va remedia toate erorile apărute și va efectua operațiuni de parametrizare, configurare și optimizare, în scopul îmbunătățirii funcționării, acestea regăsindu-se în raportul final. Pentru funcționalitățile dezvoltate și implementate în cadrul contractului, după lansarea în producție, prestatorul va asigura activități de suport, asistență tehnică și mentenanță perfectivă, corectivă, adaptivă, în timpul programului de lucru și în afara acestuia, la cererea Beneficiarului până la încheierea contractului. Acestea vor lua în calcul cerinţele naționale. Mentenanța corectivă se referă la totalitatea operațiunilor necesare pentru remedierea erorilor de funcționare ale sistemului informatic. Mentenanța adaptivă se referă la totalitatea operațiunilor necesare adaptării sistemului informatic la schimbările mediului tehnic și de bussiness în care acesta funcționează, fără a aduce funcționalități noi. Mentenanța perfectivă se referă la totalitatea operațiunilor necesare adaptării sistemului informatic la schimbările mediului (tehnologic, legislativ, business, etc.) în care acesta funcționează, ce au ca rezultat funcționalități noi. Prestatorul trebuie să asigure în timpul programului de lucru și în afara acestuia, după lansarea în producție, suportul necesar pentru a remedia incidentele semnalate de operatorii economici și utilizatorii interni. Prestatorul va realiza monitorizarea activă a platformei tehnice, pe care rulează aplicația, semnalarea incidentelor către Beneficiar și remedierea acestora, după caz. Caracterul acestei activități va fi unul proactiv, având scopul depistării și rezolvării problemelor, înainte ca acestea să afecteze negativ funcționalitatea sistemului. După implementarea sistemului, este necesară asigurarea întreţinerii acestuia printr-un proces de mentenanţă corectivă, adaptivă și perfectivă, care să ia în calcul noile cerințe strâns legate de sistemele și funcționalitățile dezvoltate în cadrul contractului. Rezolvarea fiecărei solicitări de mentenanţă corectivă, adaptivă și perfectivă va fi agreată cu Beneficiarul, iar Prestatorul se obligă să furnizeze Beneficiarului o notă de implementare care va cuprinde descrierea detaliată a modificărilor efectuate (funcționalități afectate, modificări ale obiectelor bazei de date, modalitatea de implementare, etc). Pentru toate modificările aduse sistemului după lansarea în producție, Prestatorul are obligația de a actualiza documentația tehnică și funcțională, după caz.	”12.5 SERVICII ÎN PERIOADA DE GARANȚIE A SITEMULUI SI RA”

	2. Suportul și asistența tehnică se vor face via telefon, prin e-mail sau la sediul clientului. Numărul de solicitări de asistență tehnică telefonică sau prin e-mail va fi nelimitat în limita a 8 ore lunar. Numărul de ore pentru suport tehnic nu se reportează de la o luna la alta. Numărul de deplasări la sediul achizitorului va fi de minim 1 data/trimestru și se vor realiza numai la solicitarea achizitorului pe bază de e-mail. Furnizorul va asigura asistența tehnică a beneficiarului, monitorizarea și administrarea aplicației în regim 7 din 7 zile, 24 din 24 ore, 365 zile/an. Ofertantul trebuie să prezinte in cadrul propunerii tehnice metodologia de testare după care se vor realiza activitățile de testare în timpul desfășurării proiectului. Planul detaliat de testare, însoțit de scenariile/cazurile de testare, va fi realizat de către Prestator și aprobat de Beneficiar înainte de fiecare etapă de testare agreată prin planul de proiect. Planul de mentenanță trebuie să pună la dispoziția Beneficiarului un instrument de management al cererilor de mentenanță corectivă, adaptivă și cererilor de schimbare evolutivă. Instrumentul de management va oferi cel puțin următoarele informații: 1) ID cerere 2) Tipul cererii (corectivă/ adaptivă, evolutivă) 3) Descrierea pe scurt a cererii 4) Detaliile cererii 5) Cine a solicitat (din partea Beneficiarului) 6) Data şi ora preluării cererii (câmp completat atunci când se primește cererea) 7) Data şi ora deschiderii în sistem a cererii 8) Data şi ora rezolvării cererii 9) Durata de rezolvare (se va calcula ca diferență între data şi ora rezolvării şi data şi ora deschiderii în sistem) 10) Ataşamente (plan de realizare, document de specificatii, etc.) 11) Starea cererii 12) Efort estimat (se va completa de către Prestator) Ofertantul va descrie în Oferta tehnică modalitatea de îndeplinire a cerințelor de servicii de mentenanță şi va descrie instrumentul de management a cererilor de mentenanță.	

ANEXA 1
CURRICULA CURSURILOR DE INSTRUIRE

Planul de instruire

Scopul: Instruirea utilizatorilor pentru utilizarea eficientă și sigură a software-ului dezvoltat.

Durata cursului: durata cursului este de cel puțin 3 sesiuni pentru grupul de utilizatori și grupul de administratori. Un grup de studiu nu trebuie să depășească 10 persoane. Dacă este necesar, numărul de sesiuni va fi mărit. Toți cursanții vor primi materialele de instruire necesare în conformitate cu cursurile de instruire planificate.

Analiza necesităților și a publicului țintă

Toți utilizatorii care urmează să fie instruiți trebuie să fi primit o instructaj informatic de bază înainte de a participa la cursurile relevante.

Toți utilizatorii trebuie să fie familiarizați cu procesele operaționale din domeniul în care urmează să fie instruiți.

Adaptarea materialului de instruire

Pe baza analizei publicului țintă, vor fi pregătite materiale care sunt cele mai relevante pentru nevoile utilizatorilor.

Elaborarea materialelor de formare

Va fi elaborat și furnizat un manual de utilizare bazat pe text, care va include descrieri detaliate ale funcțiilor software și instrucțiuni pas cu pas.

Prezentări

Vor fi elaborate prezentări pentru a fi utilizate în timpul sesiunilor de instruire, cu accent pe principalele caracteristici, funcționalități și scenarii de utilizare.

Organizarea cursurilor de instruire

Seminarii directe/la distanță: Desfășurarea unei serii de sesiuni (directe sau online) pentru a instrui utilizatorii în aspectele de bază ale utilizării software-ului.

Prima sesiune: introducere în software, funcții de bază, navigare.

Al doilea sesiune: prezentarea funcțiilor cheie, scenarii tipice de utilizare.

Al treilea sesiune: întrebări și răspunsuri despre software-ului.

După fiecare sesiune, utilizatorii vor verifica materialul primit.

Testarea finală: La sfârșitul cursului de instruire, se va efectua un test final și va fi întocmit un act privind finisarea cursurului de instruire cu succes.

Specificații privind condițiile de curs

Locația preferată pentru instruire este sediul beneficiarului. Beneficiarul este responsabil să organizeze facilitățile de instruire necesare, accesul personalului la sistem și la internet în conformitate cu programul de instruire care urmează să fie convenit între Beneficiar și Furnizor. Ar trebui să fie stabilit un plan de instruire provizoriu pentru desfășurarea cursurilor de instruire.

Planul de instruire detaliat livrat va include următoarele:

- echipamentele necesare pentru fiecare sesiune;
- programul cursului.

După finalizarea cursurilor de instruire, managerul de proiect va întocmi, într-o perioadă de timp rezonabilă (de exemplu, 5 zile), un document de acceptare a instruirii care să confirme că cursurile planificate au avut într-adevăr loc.

ANEXA 2
Motor de generare a rapoartelor customizabile (raportare ad-hoc) - vizualizare de date și configurator de rapoarte personalizate.

GENERAL INFORMATION

Digital data surrounds us everywhere. According to Forbes, we create about 2.5 quintillion bytes of such data every day.

When utilized properly, data opens up a wealth of opportunities for individuals and companies looking to improve their performance, operational efficiency, profitability, and growth over time. In this day and age, failing to use digital data to your advantage can be disastrous for your business - it's like walking down a busy street with a blindfold on.

As the amount of data available grows exponentially, it's critical to work with the right Ad Hoc reporting tools to not only segment, visualize and monitor large data sets, but also find answers to new questions you didn't even know existed. And when it comes to finding actionable answers to specific questions, ad hoc analysis and reporting is a must.

Data is becoming increasingly important, reinforcing the need for organizations to use their data effectively. When used properly, organizations' data offers significant opportunities to improve operational efficiency, profitability, and growth.

With Ad Hoc tools, users of the system can create list reports, on data without specialized IT skills. Anyone with computer skills can create their own report.

Flexible Customizable Reporting

Ad Hoc reporting is a branch of business intelligence used to generate one-off and ongoing reports in the form of list tables containing various data such as numbers, statistics, and product or service information, verifying the relevance of the data when the report is called up.

Reports that are not standard for an organization and for which the data warehouse or storefront is not optimized for performance improvement are generated using dynamic queries (sql query) against the database, data warehouse or storefront.

Ad Hoc report is a special type of report that is generated on demand or in response to a specific user need. This type of report is characterized by the fact that it is developed without any prior plans or templates and can be quickly created and used to solve specific problems or analyze data.

Typically, this type of reporting is required for decision support systems in exceptional situations where regular reports cannot provide enough information to make a decision.

It is a convenient means of visualizing tabular data and producing the necessary report forms. The main feature is that its structure is not rigidly defined, the user can achieve the most informative reporting in a convenient form.

Ad Hoc reports are often used to make operational decisions or identify hidden trends and patterns in data. They help to quickly assess the current situation and define a development strategy. Due to their flexibility and timeliness, Ad Hoc reports are a valuable tool for managers, analysts and other professionals who need to quickly obtain and analyze information.

Ad Hoc reports can be created in various fields and areas of activity. They can be useful in business, finance, marketing, logistics and other industries. Often Ad Hoc reports are used to make operational decisions or to analyze a specific situation or problem.

While reports are usually developed by the IT department, which can take days, this platform allows non-technical users to access valuable insights using a report generator.

Advantages of using Ad Hoc reports

The advantage of Ad Hoc reports is their flexibility and adaptability to different users. They provide quick access to information that can be collected and presented in different formats from PDF to HTML, taking into account the needs of a particular user or request.

Ad Hoc reports provide a number of benefits that make them an indispensable tool for businesses:

1. Flexibility and customizability (configurability)

Ad Hoc reports allow the user to self-select data, analyze it and generate reports according to their needs and requirements. This allows to customize the presentation of information to meet specific business goals and objectives.

2. Saving the company time and money

Thanks to the flexibility and customizability of Ad Hoc reports, users can more quickly and efficiently get the information they need without the need to contact IT specialists and save money on development. It also saves time and allows for faster decision making.

3- Ease of Use

Ad Hoc reports have a simple and intuitive interface, which makes them accessible even for users without special knowledge in the field of analytics and programming. Thanks to this, even beginners will find it easy to master the tools for creating and analyzing Ad Hoc reports.

4. Data relevance and accuracy

Ad Hoc reports provide up-to-date data. This allows you to quickly analyze data and react quickly to changes in external conditions and market situations, which in turn helps you make more informed and informed decisions, improves management efficiency, and allows you to better track key metrics and indicators. In addition, with easy access to data and easy modification, Ad Hoc reports provide more accurate information and less likelihood of errors.

All these benefits are combined in Ad Hoc reports to help you better understand goals and objectives, make the right decisions, and analyze information efficiently

FUNCTIONALITY OF THE CUSTOMIZABLE REPORTS MECHANISM

Our company, following the needs of clients and IT market trends, has developed a specialized tool for generating Ad Hoc reports Customizable Reports Mechanism.

The mechanism provides an opportunity to create new reports, create templates for each type using metadata from directories and classifiers of the developed software product.

Due to integration of the Customizable Reports Mechanism into the WEB-application, users can easily and conveniently interact with the data of the software product. This significantly increases the efficiency of work with information, allowing to quickly and accurately generate the necessary reports. This mechanism has the following functionality:

- Selecting information from multiple data source tables. The Analytical Reporting development system allows loading data from various DBMS tables using SQL queries.
- Processing of the received data, their harmonization to a uniform form and standard. Combining and linking data from various classifiers/reference books into a single data table, which allows visualizing data on the state of the business at any level of information detail.
- Modeling of own data set, indicators for control and analysis of necessary parameters for creating reports.
- Visualization of data in tabular form. Visualization improves and at the same time speeds up the process of tracking, comparing and analyzing operational information.
- Offload reports in various formats. Allows you to upload reports to various file formats like PDF, MS Excel, HTML and also allows you to print reports instantly.
- Intuitive technology. Understandable technology refers to tools that are easily and quickly understood and utilized by end users. If the technology is too complex, or appears to be, the likelihood of using it is reduced due to potential learning and comprehension issues.
- Scalability. The ad hoc reporting tool is relevant to companies of all sizes, both corporate and small family businesses. It is also capable of scaling with growth-oriented businesses.

The customizable reporting engine is multi-lingual. This feature allows users to work with reports in different languages, which is especially important for international companies or organizations with multilingual audiences. This functionality can be expanded or reduced depending on the needs of a particular project.

The mechanism is embedded in software products based on the following technologies:

- Backend - .NET 4.8+ and .NET Core 6+;
- Frontend - .NET Razor Views and jquery 2+.
- Extension for React and Blazor, which allows our product to stay up-to-date and ready to work with the latest technology solutions.
- Extension to work with different types of data sources including My SQL and Oracle. This extends our product's data warehousing and processing capabilities, providing customers with maximum flexibility and efficiency in managing their information.

These perspectives enable our product to provide customers with state-of-the-art technology solutions and the ability to effectively adapt to changing information technology requirements and trends.

Basic steps for creating Ad hoc reports

In order to create an Ad hoc report, you should:

1. Identify the purpose of the report: First of all, it is necessary to define the purpose of the report. What exactly is to be learned or addressed by the report.
2. Identify the data needed: You need to determine what data is required for the report. You should identify their sources and make sure that you have access to the data.
3. Select the required data or data source: Selecting all the required data from available sources, includes conducting surveys, analyzing statistical data or collecting information from various sources.
4. Format the report: Select the appropriate format for how the report will be displayed.
5. Submit report generation request: Generate Ad Hoc report in the desired format using the selected data.
6. Save or modify the report: If you periodically need to retrieve data for the selected parameters, you should save the generated report. If new information or parameters become available, you should update the report based on the new data that may become available after the report is generated.

By following these steps, you can create an Ad Hoc report that will contain relevant and useful information, and recall the saved version at any time.

STRUCTURE OF THE CUSTOMIZABLE REPORTING MECHANISM

During development, the customizable reporting engine is integrated directly into the application being developed. Thus, users have access to data management and analysis directly within the web interface.

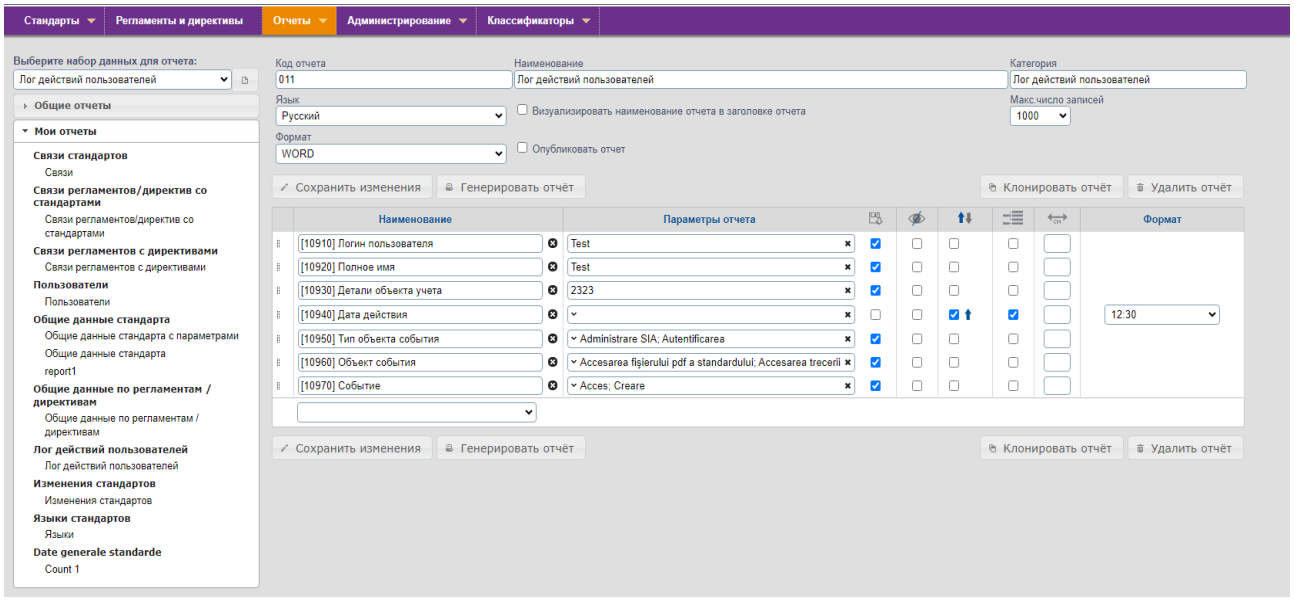
The settings of the customizable reporting engine are stored in the main entities *Interface*, *Data Tables and Query*. This model provides flexible use of metadata on which data collection and display is based.

The *interface* is a convenient tool for selecting the necessary data, and controlling - the order of displaying columns, filtering.

We have created a user-friendly interface for managing the visualization of dynamic fields, combining them on a single screen, taking into account the principle of “user friendly” to make the process of selecting parameters and setting filters as simple as possible for our users. Now they

can easily specify the parameters and filters they need to instantly generate new reports that meet their needs.

Data tables contain metadata on the basis of which reports will be generated and which can



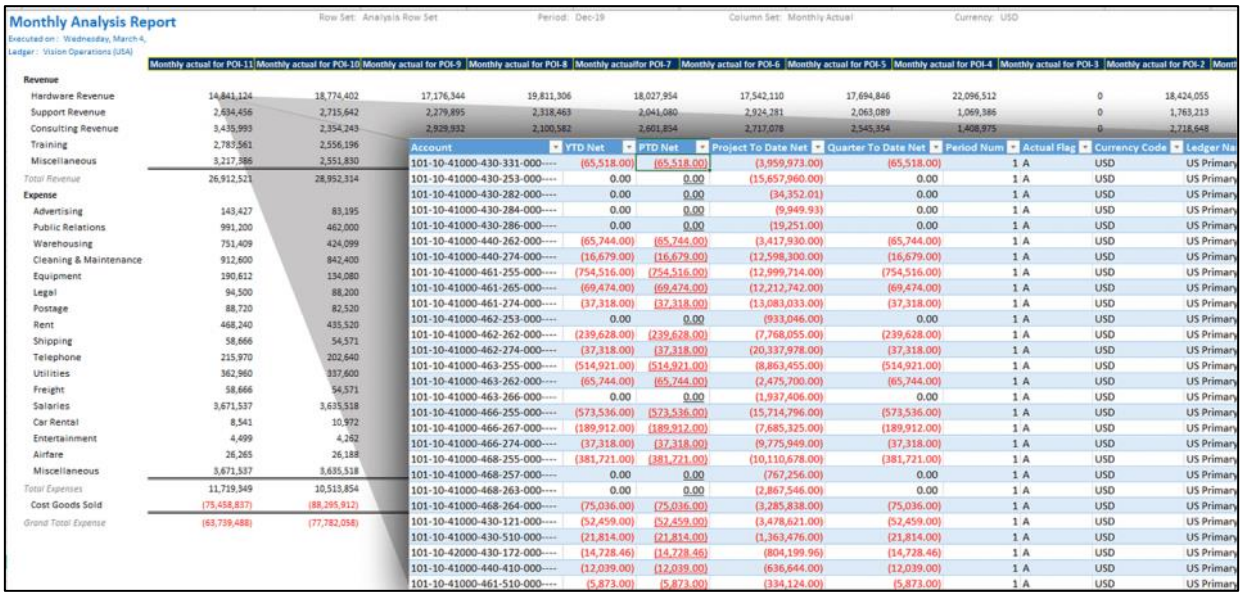
participate in the formation of queries to be executed within the framework of report creation.

AdHoc is based on SQL Views. Thanks to its flexibility and the ability to work with data from SQL Views, AdHoc provides a high level of comfort when working with data, data is displayed directly from the source, without intermediate steps, a high level of security, because it works with pre-approved data structures, respecting the access rights and security policies configured at the database level. All this together - presents a complete solution of a company's reporting system.

A *query* is a dynamic XML that is sent to the Microsoft Reporting Services report server including a dynamic SQL query that is used to retrieve the data. Each query also contains information about its name, a brief description, the method of data filling and the mode of data loading. Also, within the XML Query there is Metadata: name, data type, timestamp formats, and other parameters.

Data visualization

Data visualization is carried out in tabular form, which simplifies, accelerates and makes convenient the process of tracking, comparing and analyzing relevant information.

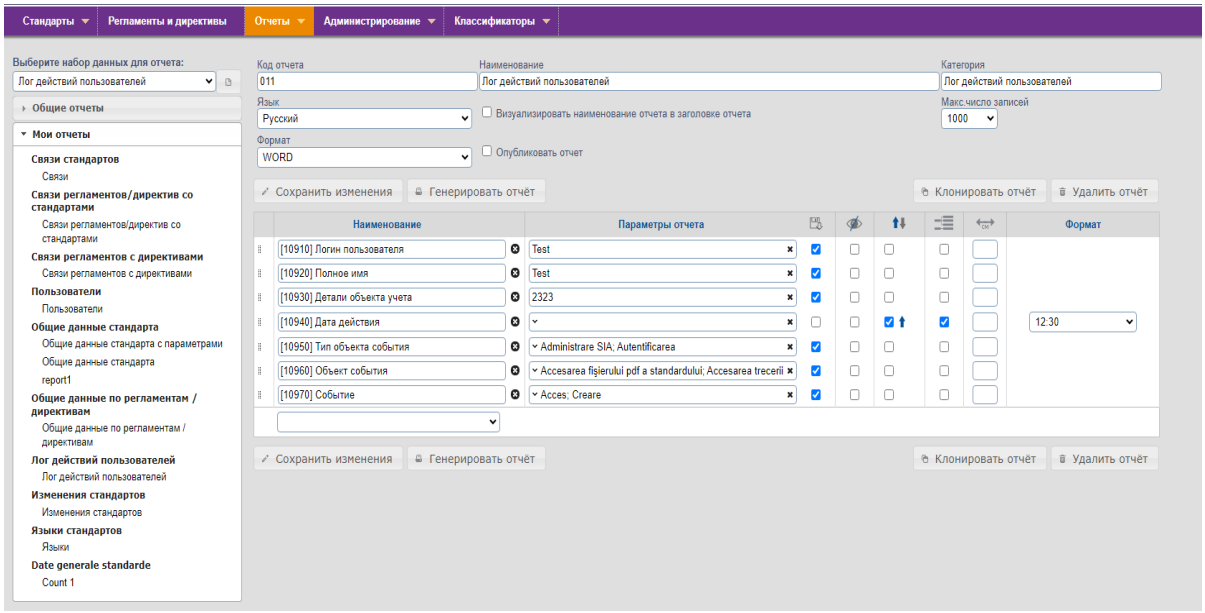


Monthly Analysis Report

Account	YTD Net	PTD Net	Project To Date Net	Quarter To Date Net	Period Num	Actual Flag	Currency Code	Ledger Na
101-10-41000-430-331-000----	(65,518.00)	(65,518.00)	(3,959,973.00)	(65,518.00)	1	A	USD	US Primary
101-10-41000-430-253-000----	0.00	0.00	(15,657,960.00)	0.00	1	A	USD	US Primary
101-10-41000-430-282-000----	0.00	0.00	(34,352.01)	0.00	1	A	USD	US Primary
101-10-41000-430-284-000----	0.00	0.00	(9,949.93)	0.00	1	A	USD	US Primary
101-10-41000-430-286-000----	0.00	0.00	(19,251.00)	0.00	1	A	USD	US Primary
101-10-41000-440-262-000----	(65,744.00)	(65,744.00)	(3,417,930.00)	(65,744.00)	1	A	USD	US Primary
101-10-41000-440-274-000----	(16,679.00)	(16,679.00)	(12,598,300.00)	(16,679.00)	1	A	USD	US Primary
101-10-41000-461-255-000----	(754,516.00)	(754,516.00)	(12,999,714.00)	(754,516.00)	1	A	USD	US Primary
101-10-41000-461-265-000----	(69,474.00)	(69,474.00)	(12,212,742.00)	(69,474.00)	1	A	USD	US Primary
101-10-41000-461-274-000----	(37,318.00)	(37,318.00)	(13,083,033.00)	(37,318.00)	1	A	USD	US Primary
101-10-41000-462-253-000----	0.00	0.00	(933,046.00)	0.00	1	A	USD	US Primary
101-10-41000-462-262-000----	(239,628.00)	(239,628.00)	(7,768,055.00)	(239,628.00)	1	A	USD	US Primary
101-10-41000-462-274-000----	(37,318.00)	(37,318.00)	(20,337,978.00)	(37,318.00)	1	A	USD	US Primary
101-10-41000-463-255-000----	(514,921.00)	(514,921.00)	(8,863,455.00)	(514,921.00)	1	A	USD	US Primary
101-10-41000-463-262-000----	(65,744.00)	(65,744.00)	(2,475,700.00)	(65,744.00)	1	A	USD	US Primary
101-10-41000-463-266-000----	0.00	0.00	(1,937,406.00)	0.00	1	A	USD	US Primary
101-10-41000-466-255-000----	(573,536.00)	(573,536.00)	(15,714,796.00)	(573,536.00)	1	A	USD	US Primary
101-10-41000-466-267-000----	(189,912.00)	(189,912.00)	(7,685,325.00)	(189,912.00)	1	A	USD	US Primary
101-10-41000-466-274-000----	(37,318.00)	(37,318.00)	(9,775,949.00)	(37,318.00)	1	A	USD	US Primary
101-10-41000-468-255-000----	(381,721.00)	(381,721.00)	(10,110,678.00)	(381,721.00)	1	A	USD	US Primary
101-10-41000-468-257-000----	0.00	0.00	(767,256.00)	0.00	1	A	USD	US Primary
101-10-41000-468-263-000----	0.00	0.00	(2,867,546.00)	0.00	1	A	USD	US Primary
101-10-41000-468-264-000----	(75,036.00)	(75,036.00)	(3,285,838.00)	(75,036.00)	1	A	USD	US Primary
101-10-41000-430-121-000----	(52,459.00)	(52,459.00)	(3,478,621.00)	(52,459.00)	1	A	USD	US Primary
101-10-41000-430-510-000----	(21,814.00)	(21,814.00)	(1,363,476.00)	(21,814.00)	1	A	USD	US Primary
101-10-42000-430-172-000----	(14,728.46)	(14,728.46)	(804,199.96)	(14,728.46)	1	A	USD	US Primary
101-10-41000-440-410-000----	(12,039.00)	(12,039.00)	(636,644.00)	(12,039.00)	1	A	USD	US Primary
101-10-41000-461-510-000----	(5,873.00)	(5,873.00)	(334,124.00)	(5,873.00)	1	A	USD	US Primary

CompanyName	OrderID	UnitPrice	Quantity	Freight	Quantity - Reverse Rank
QUICK-Stop	10345	$7.30	80	$249.06	1
QUICK-Stop	10345	$32.00	70	$249.06	2
Frankenversand	10396	$17.20	60	$135.35	3
Frankenversand	10342	$10.00	56	$54.83	4
QUICK-Stop	10361	$27.20	55	$183.17	5
QUICK-Stop	10361	$14.40	54	$183.17	6
Lehmanns Marktstand	10343	$26.60	50	$110.37	7
Frankenversand	10342	$15.20	40	$54.83	8
Königlich Essen	10325	$27.80	40	$64.86	8
Frankenversand	10342	$19.20	40	$54.83	8
Frankenversand	10337	$7.20	40	$108.26	8
Frankenversand	10396	$7.20	40	$135.35	8
Die Wandernde Kuh	10356	$10.00	30	$36.71	13

Dynamic fields of the Customizable Reporting Engine



Input - text field. It is used to set the time range (Fig. 1), to enter identifiers (Fig. 2). In other words, for everything, variations of which are many.

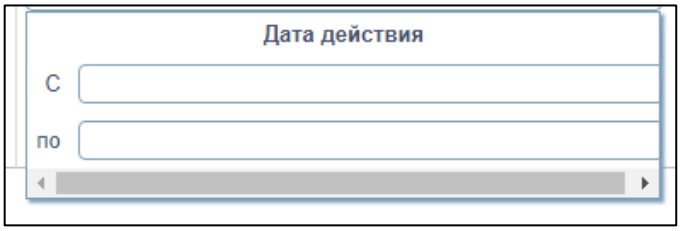


Fig.1 - time range

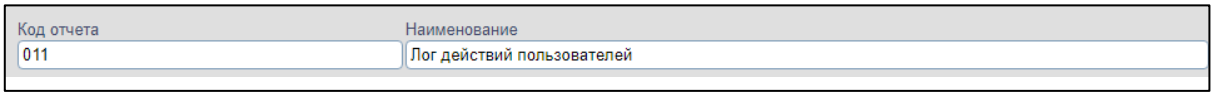
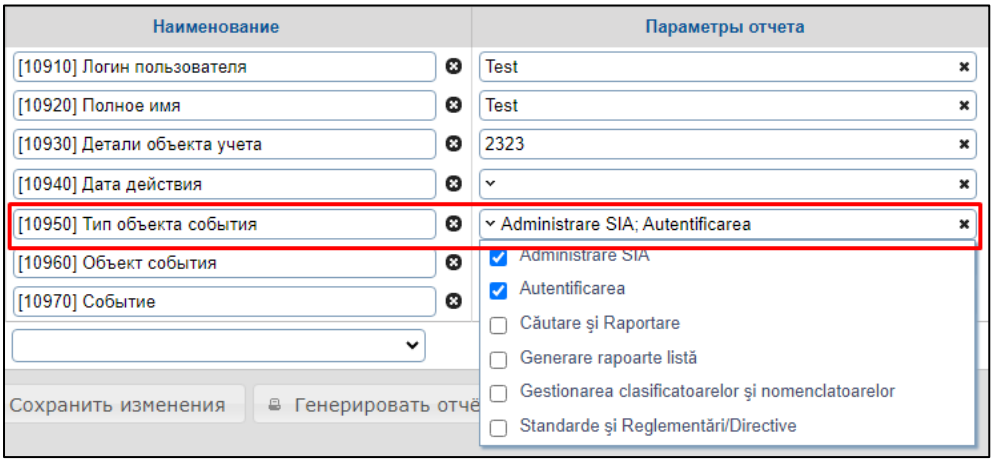
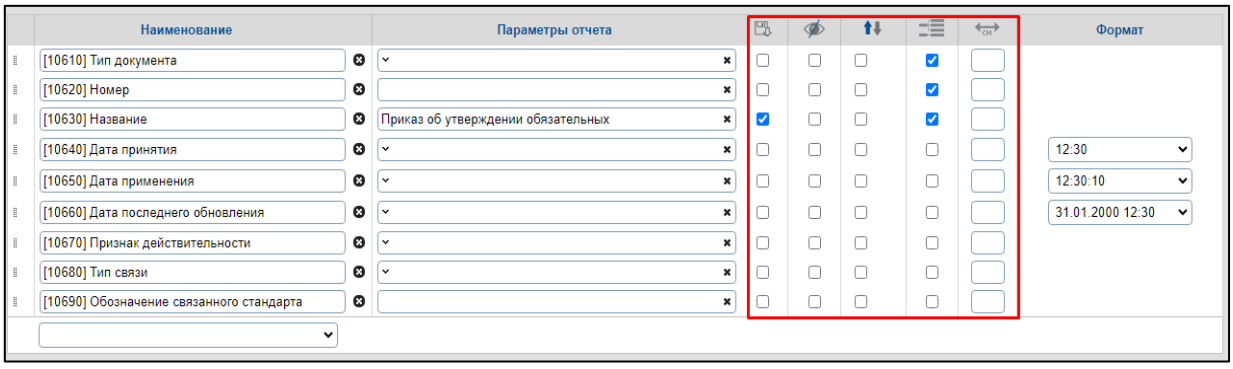


Fig.2 - data input

Select - drop-down list. You can specify one or more data grouping criteria to the selected report parameter.

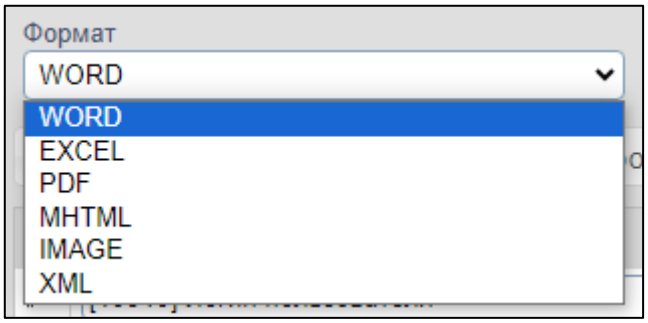


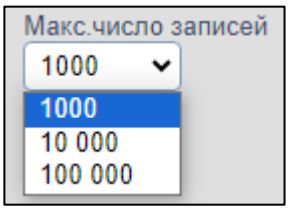
Checkbox – form for multiple selection of variations. The user can select a list of required fields in the upload.



Simple and complex uploads, using filters

Created reports can be uploaded in different formats. The user can upload the same set of data in different formats and with different maximum number of lines.





The created reports can be saved in the personal cabinet of the user who created it, and if necessary, he can transfer them to the “General Reports” section for general access.

WHAT DATA CAN BE INCLUDED IN AN AD HOC REPORT?

Ad Hoc report can include various data for analysis and decision support. Some of them are summarized below:

Statistical data - such data typically includes numerical indicators and metrics that can be useful for evaluating and analyzing the performance, efficiency, and results of various business processes.

Financial data - such data includes accounting statements, revenue and expense data, cash flow and other information related to the financial aspects of the business.

Customer data - such data includes information about customers, their profiles, preferences, purchases and interactions with the company. This data can be used to segment customers, create personalized offers and improve customer service.

Product data - such data includes information about the company's products or services, their features, prices, sales, etc. They can be useful for analyzing sales, identifying popular products, and forecasting demand.

Marketing data - Such data includes results of marketing campaigns, promotional activities, website visit analytics and other information related to marketing efforts. They can help analyze the effectiveness of marketing strategies and determine optimal promotional channels.

Any other data - such data stored in relational tables and views in Source Databases.

OPTIONS FOR USE

- Data sampling for advanced diagnostics and its analysis. There are tasks when you need to check how this or that component of the system functions. Or the whole base is in the extended mode of metrics collection. And you need to quickly get more data than usually used;
- Monitoring. There is a functionality of control of actual data. In general for the tasks of monitoring systems or specialized cases when you need the speed of obtaining results;
- Organizing local work with data in the form of pre-saved reports.

EXPERIENCE OF IMPLEMENTATION

To date, our Customizable Reporting Engine has been implemented in the following government information systems:

- „Accesul Activ la Registrul Bunurilor Imobile” for the State Services Agency, Cadastre Department - is designed to create a unified information space for processes related to active access to the Real Estate Register. The system provides storage, updating and provision of documented information on these processes using information and communication technologies, ensuring timeliness, authenticity and reliability of information.
- “e-Standard” for the Standardization Institute of Moldova - designed for efficient management and updating of the database of standards belonging to the National Institute of Standardization.