SISTEM PENTRU ANALIZA ȘI PROCESAREA DATELOR SEMI-STRUCTURATE

Petru CERVAC

Departamentul Informatică și Ingineria Sistemelor, Școala Doctorală a Universității Tehnice a Moldovei, or. Chișinău, Republica Moldova

Autorul corespondent: Petru CERVAC, cervac.petru@iis.utm.md

Rezumat. *În lucrarea de față sunt prezentate rezultatele proiectării și cercetării a unui sistem pentru analiza și procesarea datelor semi-structurate. Obiectivul lucrării este de a oferi mediului de afaceri un mecanism simplu și eficient pentru procesarea și extragerea de date din volume mari de date semi-structurate. În lucrare sunt prezentate schema funcțională a sistemului, procesul de dezvoltare a sistemului, diagrama de instanțe, diagrama modulelor și etapele de dezvoltare a sistemului cu exemple concrete.*

Cuvinte cheie*: procesarea datelor, date semi-structurate, modele și algoritmi.*

Introducere

Omenirea trăiește în epoca informațională. În prezent datele reprezintă cea mai importantă resursă de care dispune omenirea. Actorii care exploatează eficient datele disponibile obțin avantaje pe piață, comparativ cu actorii care nu fac acest lucru. Utlizarea tehnologiilor informaționale este esențială pentru a eficientizarea procesului de exploatare a datelor extrase din volume mari de date semi-structurate.

Studiu de caz

Pentru a înțelege mai bine problema, vom utiliza cazul unei întreprinderi ce produce componente electrice. Întreprinderea produce o gamă largă de componente electrice precum și se ocupă de proiectarea acestora. Fiecare lot de componente produse este însoțit de un certificat de calitate. Verificarea calității este îndeplinită în proporție de 90% de către laboratorul intern al întreprinderii și 10% de către laboratoarele externe. Certificarea produselor se efectuează prin intermediul unui set de teste efectuat de către un set de dizpozitive. Fiecare tip de component produs necesită un set distinct de teste. Dispozitivul produce un set de date ce conțin rezultatele testării. Fiecare dispozitiv are o metodă distinctă de structurare a datelor. Testarea componentelor este finalizată prin elaborarea unui certificat de calitate. Certificatele de calitate sunt consumate atât de către departamentele interne ale înreprinderii cât și de clienții acesteia. Certificatele emise de către întreprindere trebuie să conțină și informația obținută de la laboratoarele externe. Certificatele trebuie să fie de o calitate vizuală excelentă, în caz contrar întreprinderea poate suferi pierderi de imagine și financiare.

Interpretarea datelor obținute și crearea certificatelor de calitate intră în responsabilitatea lucrătorilor din laboratoarele respective. Deseori, aceste activități pot dura mai mult timp decât efectuarea testărilor, ceea ce reduce debitul laboratorului. Tot odată, calitatea vizuală a certificatelor suferă, pe motiv că acestea sunt produse manual.

Obiectivele cercetărilor efectuate

Soluția pentru această întreprindere trebuie să fie complexă și integrală. Soluția trebuie să permită interacționarea cu o gamă largă de dispozitive pentru testare care utilizează formate diferite de prezentare a datelor. Este necesar ca soluția elaborată să poată lucra și cu dispozitivele care vor fi utilizate de către laborator în viitor. Este necesar ca soluția să oferă o gamă largă de modalități de prelucrare a datelor semi-structurate.

Elaborarea certificatelor trebuie să fie simplificată. Lucrătorii laboratorului trebuie să petreacă cât mai puțin timp în elaborarea certificatelor și cât mai mult în executarea sarcinilor lor de bază.

Calitatea vizuală a certificatelor trebuie exclusă din responsabilitățile personalului de testare. Lucrul manual în procesul elaborării certificatelor trebuie exclus. Soluția trebuie să fie simplă în utilizare.

Pentru sistemul cercetat au fost identificate următoarele cerințe funcționale:

1. Conectarea la o gamă largă de dispozitive de testare;
2. Analiza și prelucrarea datelor obținute de la dispozitivele de testare;
3. Elaborararea certificatelor de calitate;
4. Stocarea datelor;
5. Stocarea certificatelor de calitate;
6. Extensibilitatea sistemului cu noi dispozitive, formate de date, metode de prelucrare a datelor.

Spațiul de soluții pentru problemele enumerate anterior este vast și determinarea soluției optimale este o activitate complexă. La alegerea soluției este necesar de luat în considerație factori adiționali precum:

- Expertiza anterioară a companiei în dezvoltarea de produse software;
- Numărul de persoane care lucrează la implementarea soluției;
- Expertiza tehnică a persoanelor ce vor menține soluția după darea în exploatare.

Tehnologii aplicate în dezvoltarea sistemului

Soluția este dezvoltată utilizând limbajul de programare C# și platforma .Net Framework. Pentru dezvoltarea interfeței grafice a fost selectat framework-ul Windows Presentation Foundation(WPF). Interfețele grafice au fost definite utilizând limbajul XAML(Extensable Application Markup Language). Interfața grafică este dezvoltată utilizând arhitectura Model-View-ViewModel(MVVM). Pentru administratea dependențelor este utilizată librăria Microsoft.Extensions.DependencyInjections. Soluția a fost dezvoltată utilizând metodologia Proiectării bazate pe Domeniu(eng. Domain Driven Design) [1].

Modelarea sistemului

În cadrul modelării sistemului și a modului de activitate au fost determinate următoarele entități:

- **DUT** – reprezintă componenta care a fost supusă unui test. DUT-ul este o abstracție de la componenta reală care a fost testată, conținând doar informația strict necesară în cadrul domeniului de lucru;
- **Măsurare –** reprezintă o măsurare efectuată asupra unui DUT. Printre principalele componente ale măsurii se enumeră numele acesteia, parametrii utilizați pentru efectuarea măsurii și valoarea obținută. Valoarea măsurii este mereu o valoarea rațională.
- **Test –** reprezintă interpretarea măsurării. Printre componentele principale ale testului se enumeră condiția și rezultatul. Rezultatul testului este mereu o valoare de tip boolean.

Acest model poate fi utilizat pentru a reprezenta orice măsurare efectuată de către laboratorul întreprinderii. În prezent, aplicația suportă două tipuri de măsurări parvenite de la două tipuri de dispozitive, stația automată de testare W-434 și analizor de rețea vectorial (VNA). Fiecare dispozitiv extinde modelul cu caracteristicile specifice formatului de date.

Arhitectura sistemului

Pentru dezvoltarea sistemului a fost selectată arhitectura de tip strat [2]. Aceasta presupune separarea sistemului în straturi concentrice. Arhitectura impune restricții asupra comunicării dintre componentele aplicație. Fiecărui component îi este permisă comunicarea doar cu elementele ce se află în staturile inferioare. Cu cât stratul este mai aproape de centrul aplicației, cu atăt componentele aflate în acest strat vor fi mai stabile și viceversa.

În centrul sistemului sunt plasate componentele ce răspund de funcționalitatea de bază a aplicației. Stratul din mijloc conține modulele ce definesc interfețele serviciilor aplicației. Stratul exterior este destinat pentru modulele ce implementează serviciile definite în stratul din mijloc.

Structural sistemul este divizat în 7 module discrete:

- **Domain** – conține logica de domeniu al aplicației;

- **Services** – conține interfețele tuturor serviciilor utilizate în aplicație;
- **WPF.Services** – conține interfețele tuturor serviciilor utilizate de către interfața grafică;
- **Services.Default** – conține implementarea implicită a serviciilor;
- **Reporting** – conține logica legată de generarea de certificate de calitate;
- **ViewModels** – conține interfața grafică;
- **WPF** - conține punctul de intrare în aplicației precum și rădăcina de compoziție.

Modulul domain aparține stratului din interior, modulele Services și WPF.Services aparțin stratului din mijloc, iar modulele Services.Default, Reporting și ViewModels aparțin stratului exterior (Figura 1).

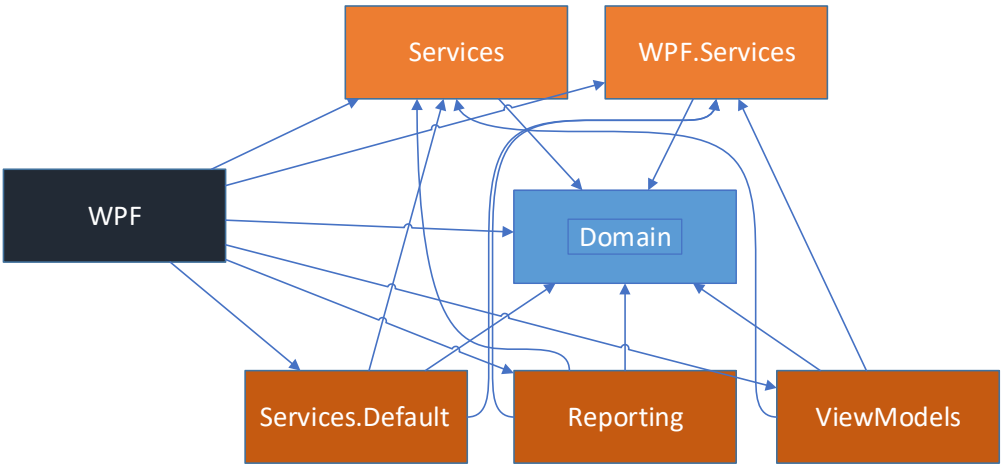

Figura 1. Diagrama modulelor

Modul de funcționare a sistemului

Modelul de funcționare a sistemului constă din 2 etape. Prima etapă (Figura 2) constă în extragerea datelor din fișierele de intrare și convertirea acestora în obiecte de domeniu. Responsabil de aceasta se face serviciul de extragere. Acesta utilizează câte un extractor specializat pentru fiecare tip de date. După ce datele au fost extrase, acestea sunt păstrate într-un spațiu de stocare pentru obiectele de domeniu.

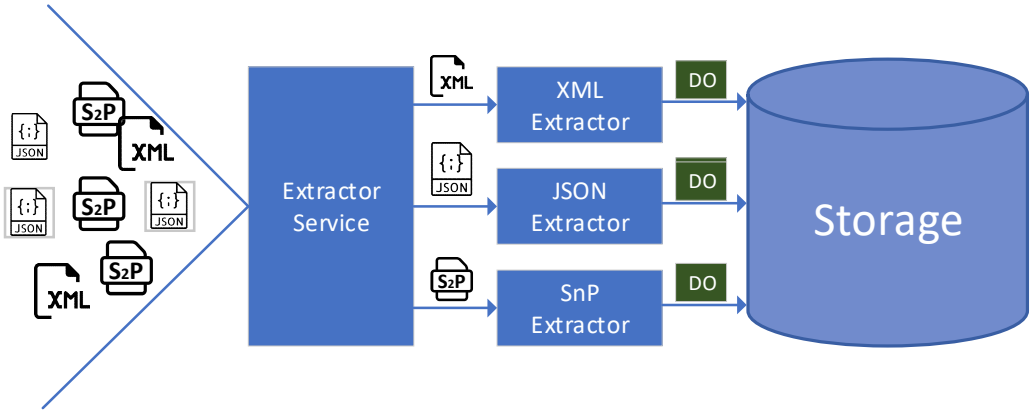

Figura 2. Etapă de extragere a datelor

La a doua etapă (Figura 3) obiectele de domeniu sunt utilizate pentru generarea certificatelor de calitate. Un certificat de calitate este compus din mai multe secțiuni. Fiecare secțiune reprezintă o vedere diferită asupra obiectelor de domeniu. Datele din fiecare secțiune sunt generate de către generatoare specializate. Fiecare generator creează un obiect specializat denumit obiect raport. Acesta conține toate datele necesare pentru reprezentarea unei secțiuni a certificatului de calitate. Un set de obiecte rapoarte reprezintă totalitatea datelor ce vor fi incluse într-un certificat de calitate. Setul de obiecte rapoarte este transmis serviciului de raportare care are sarcina de a converti datele într-un document fizic. Serviciul de raportare utilizează câte un handler pentru fiecare tip obiect raport. Fiecare handler generează o secțiune a certificatului de calitate.

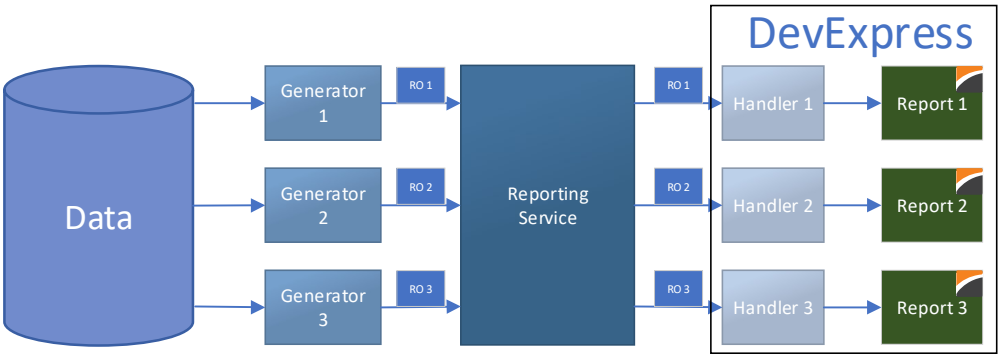

Figura 3. Etapă de procesare a datelor

Concluzii

În lucrare s-au prezentat unele rezultate obținute în procesul de dezvoltare a sistemului pentru procesarea datelor semi-structurate. Aceste structuri de date sunt specifice pentru majoritatea sistemelor de măsurare automate sau semi-automate care livrează datele în diverse formate. Extragerea și structurarea acestor date este orientată pentru a genera rapoarte informaționale și de calitate.

Referinţe:

1. EVANS, E. *Domain Driven Design: Tackling Complexity in the Heart of Software*, Boston: Addison Wesley, 2003, 529 p.
2. PALERMO, J. „*Onion Architecture,"* Clear Measure Inc., 29.07.2008. [Interactiv]. Available: https://jeffreypalermo.com/2008/07/the-onion-architecture-part-1/. [Accesat 15 02 2022].