|  |
| --- |
| PMF Cirilica.wmf |
| Institut za matematiku i informatiku  Prirodno-matematički fakultet  Univerzitet u Kragujevcu  Seminarski rad  Sistem za monitoring  **Mentor** **Student**  др Ana Kaplarević-Mališić Milica Živković 70/2014  Septembar 2017. |

|  |
| --- |
|  |

Sadržaj

[1 Uvod 1](#_Toc493124331)

[2 Opis sistema 1](#_Toc493124332)

[2.1 Konzolna aplikacija 2](#_Toc493124333)

[2.2 Web API 2](#_Toc493124334)

[2.3 Web aplikacija 2](#_Toc493124335)

[3 Korišćene tehnologije 3](#_Toc493124336)

[3.1 Microsoft .NET 3](#_Toc493124337)

[3.2 MS SQL 3](#_Toc493124338)

[3.3 Angular 2.4 3](#_Toc493124339)

[3.4 RegExp 3](#_Toc493124340)

[3.5 TFS 3](#_Toc493124341)

[4 Zaključak 3](#_Toc493124342)

[5 Literatura 4](#_Toc493124343)

# Uvod

Sistem za monitoring namenjen je administratorima kompanije Asseco SEE i služi za pregledniji uvid u greške koje su se dogodile u radu Captive i Documentuma. Administratorima kompanije omogućeno je ovom aplikacijom da prate greške koje pristižu sa svih računara, da ih memorišu kao bitne i da zapamte način na koji su tu grešku otklonili.

# Opis sistema

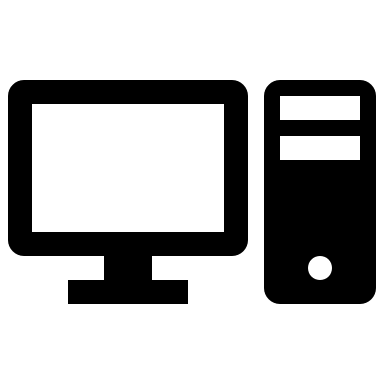
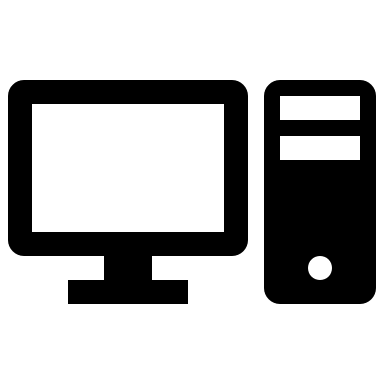
Kako bi administratori pratili greške koje su se dogodile na nekom od računara na kome se izvršava Captiva i/ili Documentum ovaj sistem je potrebno podeliti na tri komponente: konzolna aplikacija, Web API i Web aplikacija.

Konzolna aplikacija pokreće se na svakom računaru na kome se izvršava Captiva i/ili Documentum. Ovaj servis se automatski pokreće svakog dana u 8h i aktivan je do 18h. Za to vreme prati promenu svakog log fajla i beleži pronađen ispis greške u bazu podataka koja se nalazi na serveru.

Web API je servis koji je pokrenut na jednom računaru i na HTTP zahtev Web aplikacije čita podatke iz baze i prosleđuje ih Web aplikaciji.

Kada administrator pristupi Web aplikaciji sve greške koje su pronađene prikazuju se u tabeli. Korisnik ima mogućnost filtriranja po datumu, računaru sa kojeg je greška pristigla, putanji do fajla, pregled samo Captivinih ili samo Documentum grešaka itd. Takođe moguće je preuzimanje fajla u kome je ispisana greška. Grešku je moguće obrisati, arhivirati (što omogućava da se zabeleži način na koji je greška otklonjena i pri sledećem pojavljivanju primeniti isti postupak za otklanjanje iste greške) ili prebaciti na listu sistemskih ili ignore grešaka (u koliko se ista greška ponovo pojavi neće biti razmatrana). Na slici 1 prikazana je kompletna arhitektura sistema za monitoring koja je prethodno objašnjena.

Slika 1 - Arhitektura sistema



Database

Web API

API

CORE

Data

Model

ConsoleAPP

ConsoleAPP



Web APP

Model

View

Controller

## Konzolna aplikacija

Konzolna aplikacija namenjena je za rekurzivno nadgledanje log fajlova. Pri startovanju aplikacije zadaje se putanja do foldera u kome se nalaze svi log fajlovi. Pošto se aplikacija svakog dana pokreće u zadatom vremenu, fajlovi koji su zastareli (nisu današnji) ne ulaze u nadgledanje. Za svaki novo kreirani log fajl kreira se nova nit (Thread). Nit koja je zadužena za nadgledanje fajla sadrži metod koji se na događaj (Event) modifikacije fajla poziva. U koliko se pri izmeni u fajlu upisala nova greška, ona se beleži u bazu.

Log fajlovi koji se nadgledaju nemaju istu šemu, ali je sigurno da svaki log počinje timestampom. Kako bi ceo jedan log bio pročitan iz fajla (jer log može imati više linija) potrebno je parsiranje po timestampu.

Kada je log parsiran potrebno je proveriti da li sadrži jednu od ključnih reči koje određuju da li je nešto greška a to su: ERROR, EXEPTION, FAIL itd. U koliko je pronađena greška proverava se da li se nalazi na listi sistemskih grešaka i samo u slučaju da nije sistemska beleži se u bazu zajedno sa putanjom do fajla, linijom u kojoj je greška pronađena, datum ispisivanja greške i nazivom računara sa kojeg je greška pristigla.

## Web API

Web aplikacija na HTTP zahtev poziva određenu rutu. U zavisnosti od rute, Web API prosleđuje različite podatke Web aplikaciji. Završni deo rute određuje koji se kontroler poziva, na osnovu imena kontrolera. Svaki kontroler ima metode get, post i delete koje se mogu pozivati sa ili bez argumenata. U metodu get svakog kontrolera izvršava se konekcija sa bazom i čitanje podataka u zavisnosti od uloge kontolera (čitanje svih grešaka, samo Captivinih ili samo Documentum, čitanje arhiviranih grešaka ili ignore liste…).

## Web aplikacija

Web aplikacija izgrađena je od prikaza pet komponenti: prikaz svih grešaka, prikaz Captiva grešaka i njihova statistika, prikaz Documentum grešaka sa statistikom, prikaz arhiviranih grešaka i prikaz liste sistemskih grešaka sa formom dodavanja novih. Ove stavke navedene su u meniju. Pri pokretanju aplikacije prikazuju se statistike grešaka (procenat Captivinih i Documentim grešaka, ukupan broj grešaka koje su pristigle danas kao i statistike grešaka za svaki računar zasebno). Sve greške je moguće sortirati po datumu, ili po fajlu u kome su pronađene. U tabeli pored ispisa jedne greške nalazi se i dugme za arhiviranje greške nakon čega je potrebno uneti komentar za otklanjanje greške, dugme za ignorisanje takvog tipa grešaka nakon čega se greška briše iz tabele i prebacuje u listu ignore, dugme za brisanje, dugme za preuzimanje fajla u kome je greška pronađena i dugme za detaljan pregled greške. Moguća su filtriranja po svakom atributu u tabeli kao i brisanje grešaka od-do zadatih datuma. Klikom na sistemske greške kao stavku menija pored prikaza svih ignore grešaka postoji i forma koja administratoru omogućava dodavanje dela grešaka koja ne treba da se prikazuje i time podešava da se svaka greška koja u sebi sadrži takav deo ignoriše.

Web aplikacija se na svaki zahtev korisnika za modifikaciju sistema osvežava i ponovo šalje zahtev Web API-ju koji prosleđuje izmenjene podatke ili ih menja i vraća nove.

# Korišćene tehnologije

## Microsoft .NET

Microsoft .NET Framework je softverska platforma koja može biti instalirana na računarima koje pokreće Windows operativni sistem. On uključuje veliki broj gotovih biblioteka kodova za uobičajene probleme u programiranju.

Sistem za procesiranje logova je konzolna aplikacija napisana u C#. C# je objektno-orijentisani programski jezik i sastavni je deo .NET frameworka.

OWIN predstavlja standardni interfejs između .NET web servera i web aplikacija upotrebljen za pravljenje Web API-a.

Katana je open source projekat Microsofta koji se sastoji od skupa komponenata za pravljenje i hostovanje aplikacija zasnovanih na OWIN standardu (hostovanje Web API-a).

## MS SQL

MS SQL je relaciona baza koja podržava Transact SQL, što znači pored postojećih (SELECT) upita, omogućava i složenije stvari, kao na primer izmenu programskog toka.

## Angular 2.4

Angular je JavaScript framework za izradu single-page aplikacija. Aplikacija je izgrađena od komponenti koje se sastoje od pogleda (.html, .css) i kontrolera (.ts) i servisa koji ih opslužuje.

Za izradu web aplikacije korišćen je Angular zbog lake realizacije ideja, modularnosti i razvijenog community dela koji umnogome olakšava rešavanje potencijalnih problema.

## RegExp

Regularni izraz je niz karaktera, određen sintaksnim pravilima, koja služi za proveru validnosti nekog podatka. Uobičajeno se koriste za davanje tačnog opisa skupa, bez potrebe za nabrajanjem svih njegovih elemenata. (Wikipedia, Regular Expression)

## TFS

Microsoft-ovo rešenje za upravljanje izvornim kodom, izveštavanje, upravljanje zahtevima, upravljanje projektom i testiranje. Pokriva ceo životni vek aplikacije. Prilagođen je za upotrebu u Visual Studio i Eclipse razvojnim okruženjima. (Wikipedia, Team Foundation Server)

# Zaključak

U kompaniji Asseco South Estern Europe, u periodu od tri nedelje, radila sam na projektu “Sistem za monitoring”. Pored upoznavanja sa novim tehnologijama imala sam prilike da vidim kako izgleda rad u firmi i upoznala se sa projektima na kojima firma radi. Znanje koje sa stekla na predmetima „Baze podataka 1“, „Softverski inženjering 1“, „Vizualno programiranje“, „Web programiranje“ su mi pomogla da završim projekat uspešno.

# Literatura

.NET Framework https://sr.wikipedia.org/sr-el/.NET\_Framework

OWIN, Katana https://codeopinion.com/self-host-asp-net-web-api/

Wikipedia *Regular Expression* https://en.wikipedia.org/wiki/Regular\_expression

Wikipedia *TFS* https://en.wikipedia.org/wiki/Team\_Foundation\_Server