DEZVOLTAREA SISTEMULUI DE SERVICII ON-LINE PENTRU SALOANELE INTERNAŢIONALE SPA

Ilie CÎRCIUMARU, Facultatea de Matematică şi Informatică

The paper provides an investigation and practical implementation of new informational technologies best suitable for the developing of complex systems of on-line services with inclusion of databases. The information system is focused on the on-line services offering by international spa salons. It consists of front-end and back-end parts and provides the large diversity of functionalities: spa registration, advanced search with visualization of specific information, booking, paying and more.

În societatea modernă are loc tendinţa de transferare în masă a diferitelor servicii în domeniul Internet. Utilizatorii de Internet pot achita impozitele şi serviciile comunale, face achiziţii, procura bilete de avion etc. Deci internetul a pătruns aproape în orice sferă a vieţii, iar sfera de turism şi odihnă nu este o excepţie. Astăzi jumătate din hoteluri şi staţiuni au site-uri proprii de unde pot fi văzute poze, inclusiv rezervate servicii. Există şi site-uri care grupează mai multe hoteluri, saloane spa, staţiuni şi oferă posibilitatea de a efectua rezervarea şi achitarea serviciilor chiar pe aceste site-uri. Însă astfel de site-uri, ca booking.com, propun o gamă largă de oferte care adesea nu este comodă pentru căutare, iar filtrarea se face în baza criteriilor prea reduse. De aceea a fost luată decizia de a crea un sistem de servicii on-line, care se va concentra numai pe saloanele spa din toată lumea.

Scopul principal al acestui sistem (numit „Esparadise”) constă în furnizarea posibilităţii oamenilor de a găsi un spa salon conform cerinţelor şi de a efectua diverse rezervări.

Obiectivele propuse sunt:

1. Realizarea înregistrării saloanelor internaţionale spa cu adăugarea preţurilor, pozelor şi indicarea serviciilor prestate de acest salon.
2. Implementarea căutării sofisticate.
3. Posibilitatea de a vizualiza informaţia despre spa salon.
4. Posibilitatea de a rezerva camera în spa şi obţinerea sumei pentru achitare.

Esparadise se planifică în calitate de proiect comercial, profitul căruia se calculează ca o pondere de 5% din suma pe care utilizatorii

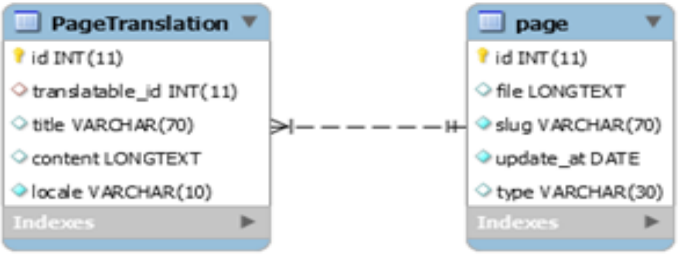
vor cheltui-o pe servicii (rezervarea camerelor, servicii adăugătoare etc.). Pentru aceasta, toate informaţiile despre rezervări sunt stocate în baza de date pe care administratorul în fiecare lună le va factura şi le va trimite saloanelor spa pentru achitare.

Implementarea sistemului este separată pe două părţi: „Back-end" şi „Front-end". În dezvoltarea aplicaţiilor web „Front-end" înseamnă partea aplicaţiei cu care interacţionează clientul, adică este interfaţa grafică pe care o reproduce navigatorul. „Back-end" înseamnă dezvoltarea software pe partea serverului, în care se include lucrul cu baza de date, realizarea logicii de business al aplicaţiei şi generarea paginilor, care apoi vor fi trimise navigatorului pentru reproducerea pe partea clientului. De obicei, realizarea sistemelor începe cu „Back-end", şi anume, cu proiectarea bazei de date.

Pentru gestionarea bazei de date, a fost aleasă soluţia open source – MySQL. MySQL are abilităţi destul de bune – susţine tranzacţiile, indexarea, crearea procedurilor şi funcţiilor, astfel, acoperind toate necesităţile aplicaţiei. Proiectul bazei de date se prezintă în formă de diagramă Entitate-Relaţie [1], care include în sine toate entităţile cu relaţii specificate între ele.

Fig. 1. Tabele pentru entitatea Pagina

O parte de diagramă este prezentată în Figura 1 – două tabele



pentru entitatea Pagina (*page*), care sunt legate prin relaţia de tip unu (*page*) la mai multe (*PageTranslation*). Conform cerinţelor, sistemul trebuie să fie internaţionalizat şi tabela PageTranslation va conţine traduceri pentru câmpurile traductibile (*title, content*) cu specificarea limbii în câmp „locale".

Entităţile care au câmpuri pentru traducere implementează strategia descrisă anterior. Ca limbaj de bază utilizat în dezvoltarea „Back-end" a fost ales PHP, deoarece el susţine POO, este *open source* şi are

multe instrumente („Frameworks”) create de comunitate pentru efectuarea rapidă a aplicaţiilor web. Pe lângă aceea că „Framework” [2] automatizează multe operaţii rutine, el impune o arhitectură efectivă, care disciplinează programatorul şi permite scalarea uşoară a aplicaţiei. Pentru a realiza Esparadise, a fost ales framework-ul Symfony. Symfony [3] este compus din module (*bundle*), pe care utilizatorul poate să le utilizeze sau nu. Printre cele mai importante module utilizate au fost: ORM (*object relaţional mapping*) Doctrine, Twig (şablonizator), Forms, Routing (modul pentru rutare), Translation. Trebuie de evidenţiat că, de facto, Doctrine este o bibliotecă separată care permite interacţiunea cu baza de date în stil orientat pe obiecte şi furnizează un limbaj propriu DQL pentru interogări. Symfony implementează MVC [4] – şablon de proiectare, care divizează programul logic pe trei nivele. Schema interacţionării între nivele este desenată pe Figura 2.

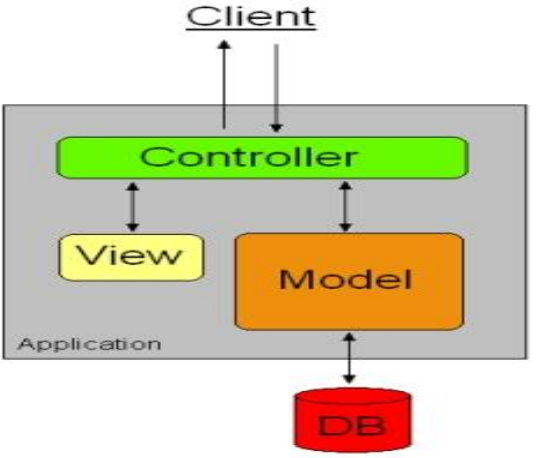


Fig.2. Diagrama pe nivele

Aici Controller – nivel care primeşte interogări de la utilizator, le prelucrează, conform necesităţilor, trimite cerere la model pentru informaţie. Informaţia primită o trimite la nivelul View (reprezentare). Model – nivel care lucrează cu baza de date şi conţine business-logica aplicaţiei. Sarcina principală este primirea cererilor de la Controller şi întoarcerea informaţiei necesare. View – nivel de vizualizare a

paginilor. El nimic nu ştie despre baza de date sau business-logică, sarcina lui constă în primirea şi afişarea datelor. Se poate spune că logic Controller şi Model aparţin dezvoltării „Back-end”, iar View aparţine dezvoltării „Front-end”.

Pe partea „Front-end” au fost utilizate următoarele tehnologii:

1. HTML (HyperText Markup Language) – un limbaj de marcare utilizat pentru crearea paginilor web.

2. Twig – şablonizator, adică un instrument care uşurează prezentarea datelor în HTML din PHP. Twig furnizează programatorului stil POO la accesarea obiectelor şi structurilor standarde, astfel ca ciclu, operator condiţional if-else, variabile, macro-uri.

3. CSS – o tehnologie web, utilizată pentru formatarea documentelor HTML. Deoarece CSS nu are moştenire, variabile şi stiluri imbricate care foarte mult ar accelera dezvoltarea, a fost utilizat SCSS [5] – aşa-numitul preprocesor. Codul scris în el va fi compilat în CSS pur.

4. JavaScript – limbaj de programare scriptual incorporat în navigator. Este folosit pentru adăugarea interactivităţii pe pagina web, de exemplu: galeria de imagini, calendar pentru indicarea datei, harta pentru indicarea locaţiei.

Cu efectuarea părţii „Front-end” se termină realizarea sistemului. Însă lucrul asupra sistemului nu se termină la acest punct. În continuare va fi o cale lungă de adaptare la cerinţele utilizatorilor şi adăugarea posibilităţilor noi, aşa ca achitarea prin PayPal sau integrarea cu alte platforme. Pentru a vedea sistemul realizat, accesaţi http://esparadise.com/.

Referinţe:

1. Diagrama ER – https://en.wikipedia.org/wiki/Entity-relationship_model [Accesat 27.06.16].
2. PHP Frameworks - http://www.hongkiat.com/blog/best-php-frameworks [Accesat 22.06.16].
3. Symfony - https://symfony.com/ [Accesat 22.06.16].
4. MVC - https://en.wikipedia.org/wiki/Model-view-controller [Accesat 22.06.16].
5. Pagina oficială SCSS - http://sass-lang.com/ [Accesat 27.06.16].

Recomandat
Ion ANDRIEŞ, dr., conf. univ.