

Головна сторінка білінг-панелі наведена на рисунку 2.

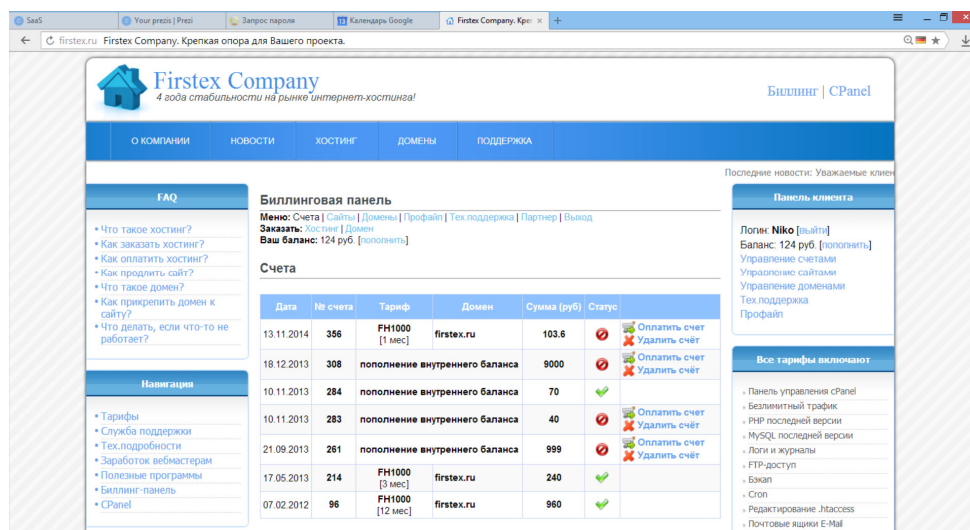


Рисунок 2 – Поля для вводу на сторінці заповнення бази знань

### Висновок

Здійснено підвищення ефективності надання хостинг-послуг для малого і середнього бізнесу за рахунок розробки web-орієнтованої системи білінг-панелі для хостинг-компанії, яка забезпечує можливість самостійного обслуговування клієнтів: управління рахунками, сайтами, доменними іменами, а також має додаткові функції, такі як партнерська реферальна система, можливість здійснювати переписку в чаті з службою підтримки. Також реалізована адміністративна панель управління, що дозволяє виконувати наступні функції: керування рахунками, сайтами, доменними іменами, DNS-зонами, тарифними планами, користувачами та новинами хостинг-компанії.

Вперше одержано доступну багатофункціональну програмну систему, що задовільняє всі основні вимоги, що висуваються до білінг-панелі в сфері надання хостинг-послуг в Україні.

### Список використаних джерел

1. Вікіпедія : LAMP [Електронний ресурс]. — Режим доступу : <http://uk.wikipedia.org/wiki/LAMP>.
2. Вікіпедія : PHP [Електронний ресурс]. — Режим доступу : <http://ru.wikipedia.org/wiki/PHP>.
3. Вікіпедія : MySQLi [Електронний ресурс]. — Режим доступу : <http://ru.wikipedia.org/wiki/MySQLi>.

УДК 004.9

## ТЕХНОЛОГІЯ РОЗРОБКИ ЕФЕКТИВНОГО ВЕБ-ДОДАТКУ НА МОВІ JAVA

Шумик В.В.

МТУ «Миколаївська політехніка», студент

Веб-додаток - клієнт-серверний додаток, в якому клієнтом виступає браузер, а сервером - веб-сервер. Логіка веб-додатку розподілена між сервером і клієнтом, зберігання даних здійснюється, переважно, на сервері, обмін інформацією відбувається по мережі. Однією з переваг такого підходу є той факт, що клієнти не залежать від конкретної операційної системи користувача, тому веб-додатки є міжплатформними сервісами. Істотна перевага побудови Web додатків для підтримки стандартних функцій браузера полягає в тому, що функції повинні виконуватися незалежно від операційної системи даного клієнта. Замість того щоб писати різні версії для Microsoft Windows, Mac OS X, GNU / Linux та інших операційних систем, додаток створюється один раз для довільно вибраної платформи і на ній розгортається. Однак різна реалізація HTML, CSS, DOM та інших специфікацій в браузерах може викликати проблеми при розробці веб-додатків і подальшій підтримці. Крім того, можливість

користувача налаштовувати багато параметрів браузера (наприклад, розмір шрифту, кольору, відключення підтримки сценаріїв) може перешкоджати коректній роботі додатка [1].

Швидкий розвиток інформаційного Web - середовища призвів до того, що вимоги до Web-додатків суттєво змінилися. Зокрема спостерігається тенденція до створення багатих Web-додатків, тобто додатків, інтерфейс яких надає можливості, що не відрізняються від можливостей звичайного додатку, який призначений для настільної системи [2]. Але при роботі програм, що підтримують мережеву взаємодію, усунути затримку відповіді, пов'язану з передачею даних через мережу Інтернет, принципово неможливо. Пом'якшити негативний ефект від затримки даних дозволяє технологія з використанням мови Java.

Після установки JDK - комплекта розробника додатків на Java та JRE - комплекта реалізації віртуальної машини, необхідної для виконання Java-додатків, які для більшості платформ загрузаються з сайту ORACLE, вибирається середовище розробки IDE Eclipse, яке на відміну від інших IDE повністю безкоштовне та має потужну ком'юніті-підтримку. В якості збирача проекту вибирається Maven, який забезпечує декларативне, а не імперативне збирання. Він створює визначену ієрархію в проекті та сам додає всі потрібні додаткові інструменти, а в кінці збирає все в один цільний проект, який можна вже використовувати або переносити на інше місце з подальшою доробкою або переробкою.

Вибір Веб-сервера залежить від розміру та можливостей проекту, тому пропонується встановити легкий, простий і безкоштовний варіант Tomcat (рис. 1), який можна легко налаштувати як в проект, так і в середовище розробки IDE Eclipse з можливістю їх з'єднання і контролю, оскільки надаються всі потрібні функціональні можливості.

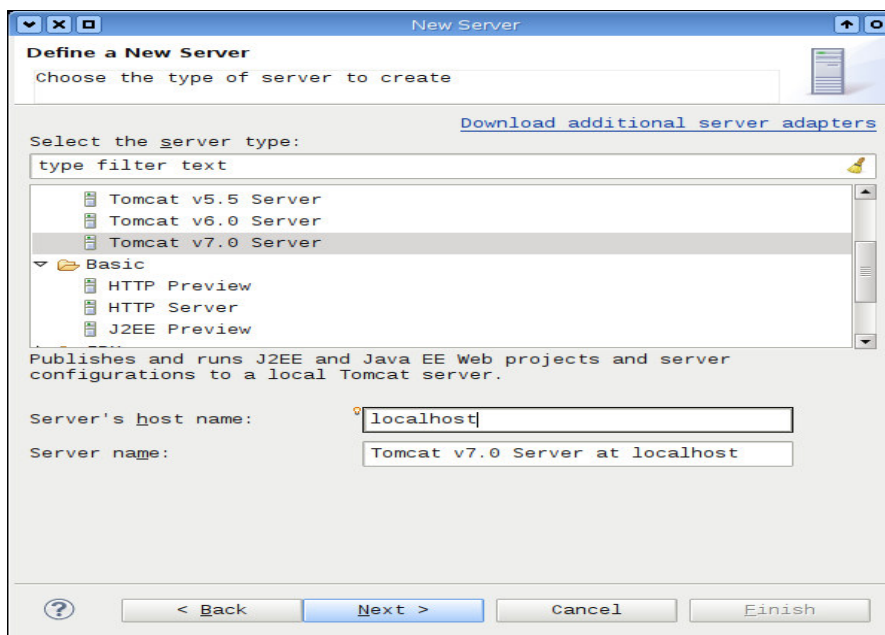


Рисунок 1 - Вибір Tomcat в якості встановленого Веб-сервера.

Кістяк проекту – це ядро основного універсального фреймворка SpringMVC з відкритим вихідним кодом на Java-платформі, який надає прості в користуванні, добре задокументовані засоби вирішення проблем, що виникають при створенні додаткових корпоративних масштабів. В його основі використовується шаблон проектування MVC (model view controller), який ділиться на 3 частини: моделювання, представлення і контроль. Оскільки любий проект неможливий без бази даних, то для спрощення роботи програміста створюється фреймворк Hibernate – ORM представлення для мови програмування Java, який вирішує завдання об'єктно-реляційного відображення та представлення вільних програмних засобів з відкритим вихідним кодом.

Вибір самої бази даних залежить від завдань і кількості даних. Нам підійде безкоштовна MySQL, яка використовується в якості сервера, до якого звертаються локальні чи віддалені об'єкти, але в її дистрибутив входить бібліотека внутрішнього сервера, що дозволяє автономне програмування.

Замість того, щоб застосовувати мовний стандарт SQL, можна поставити анотації в класах і об'єкти автоматично будуть зберігатися в базі даних.

Для синхронізації роботи потрібно застосувати технології систем керування версіями VCS, які полегшують роботу з інформацією, що змінюється, дозволяють зберігати декілька версій одного документу, а при необхідності повертатися до більш ранніх версій. Вони тримають проект на віддаленому сервері і дають розробникам можливість вносити свої зміни в проект, не заважаючи іншим. З таких технологій найпопулярнішою є Git – розподілена система керування версіями файлів, ядро якої представляється набором утиліт командного рядка з параметрами, а всі налаштування зберігаються в текстових файлах певних конфігурацій.

Перевірка правильної роботи додатку проводиться з допомогою інструменту Junit, як бібліотеки для модульного тестування програмного забезпечення на мові Java, з метою встановлення відповідності програми заданим вимогам та виявлення ситуацій, в яких поведінка програми є небажаною або не відповідає специфікаціям.

#### **Список використаних джерел**

1. Беллиньясо М. Разработка Web-приложений: задача - проект - решение. — М.: «Диалектика», 2007. — 640 с.
2. Олишук А. В. Разработка Web-приложений. — М.: «Вильямс», 2006. — 352 с.

УДК 681.3

## **ПРОГРАМНА СИСТЕМА ОПТИМІЗАЦІЇ ВИМІРЮВАНЬ НА ОСНОВІ ПЛАНУВАННЯ ЕКСПЕРИМЕНТУ**

**Ганущак В.М.<sup>1)</sup>, Пукас А.В.<sup>2)</sup>**

*Тернопільський національний економічний університет*

*<sup>1)</sup> магістрант, <sup>2)</sup> к.т.н., доцент*

### **I. Постановка задачі**

У багатьох сферах діяльності людини доводиться мати справу з експериментальними дослідженнями [1]. Як результат експерименту дослідник отримує набори числових даних, на основі яких необхідно показати існуючу закономірність чи явище, знайти оптимальне значення певної характеристики, виявити важливість кожного впливу на результуючу характеристику та ін. [2]. Важливе місце при цьому має процес планування експерименту, який забезпечує проведення експерименту в такий спосіб, щоб отримати оптимальний результат з мінімальними витратами, оскільки не завжди можливо забезпечити як технічно чи технологічно (космічна галузь, атомна енергетика, авіація, ін.), так і фінансово необхідну кількість спостережень. З математичної точки зору задача планування експерименту полягає в знаходженні математичної моделі об'єкту дослідження із максимальною точністю при мінімальній кількості спостережень [3].

### **II. Мета роботи**

Метою дослідження є оптимізація експериментальних досліджень на основі розробки програмної системи, яка забезпечує можливість отримання оптимального плану експерименту для вибраної математичної моделі.

### **III. Особливості програмної реалізації системи**

На мові ASP.NET розроблено програмне забезпечення у вигляді web-додатку, що дозволяє на основі задання параметрів експерименту, зокрема виду та структури моделі, критерію оптимізації моделі, виду області експерименту отримати оптимальний план для проведення спостережень [4, 5].