

Безмилостно тестване на веб приложения проект OpenQA

Свилен Иванов
(svilen.ivanov@gmail.com)
(<http://svilen-online.blogspot.com>)

OpenFest 2007

Какво ще представя

- Как може автоматизирано да се тества едно уеб приложение
- Създаване на тест симулиращ реален потребител на приложението
- Изпълнение на един тест върху различни браузъри и платформи
- Всичко това с помощта на Selenium
 - средство за автоматизирано тестване на уеб приложение с отворен код
 - част от групата проекти с отворен код за тестване - OpenQA

Защо автоматизиране?

- Трудоемко (и скучно)
 - различни платформи/браузъри = много варианти
- Изисква значителен човешки ресурс
 - тестери, запознати в детайли с приложението
 - „човешки“ фактор
- Не може да се повтаря лесно
 - например веднъж дневно, в началото на работния ден
- Намалява възможността за „неприятни“ изненади с приближаване на крайния срок

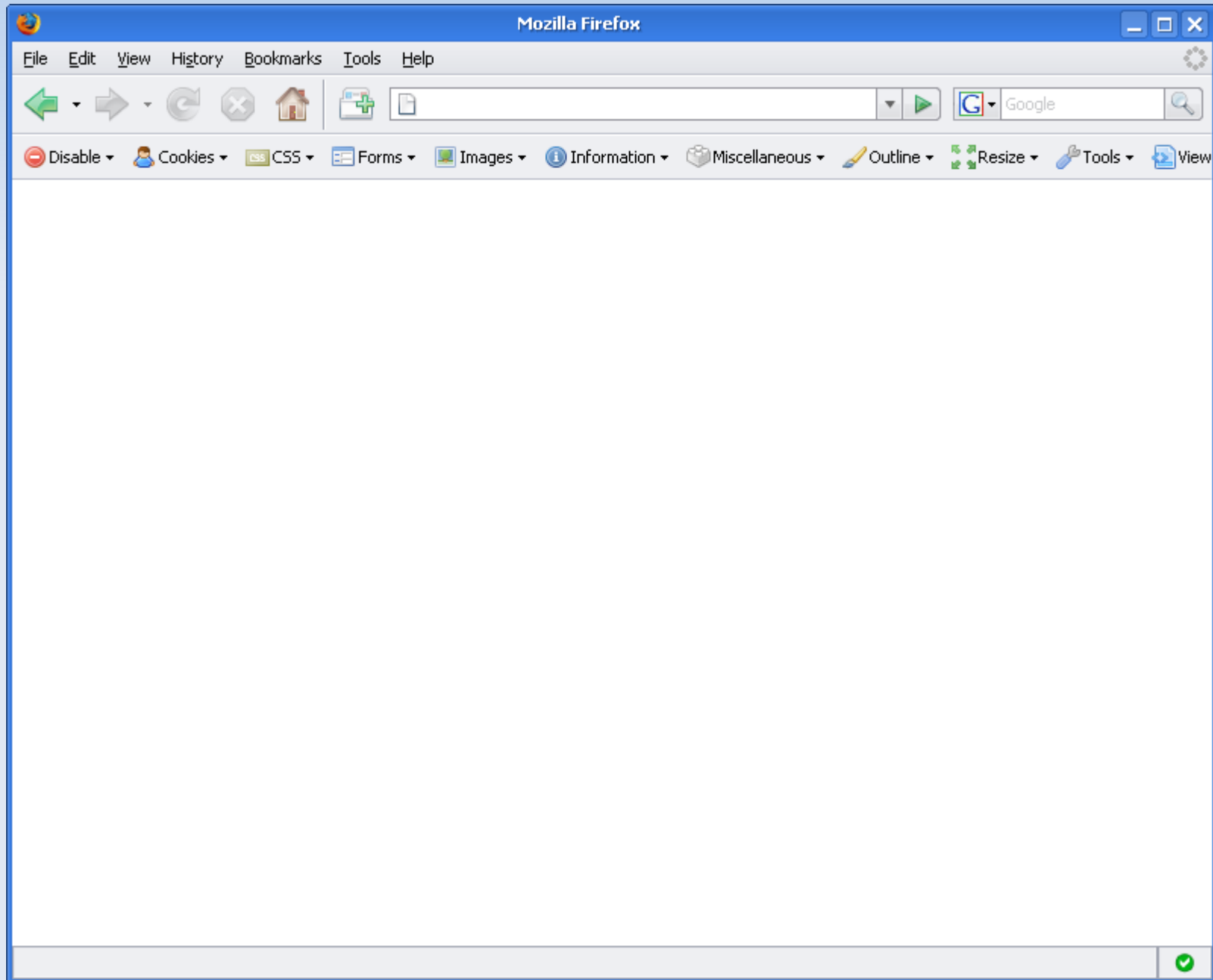
Какво е функционално тестване?

- Проверява коректността на системата от гледна точка на потребителя
- Обхваща почти всички подсистеми на приложението:
 - потребителски интерфейс, съобщения за грешки
 - манипулиране на данни, бизнес процеси
 - сигурност и контрол на достъпа
 - ...
- (Обикновено) се прилага се към в края на процеса на разработка

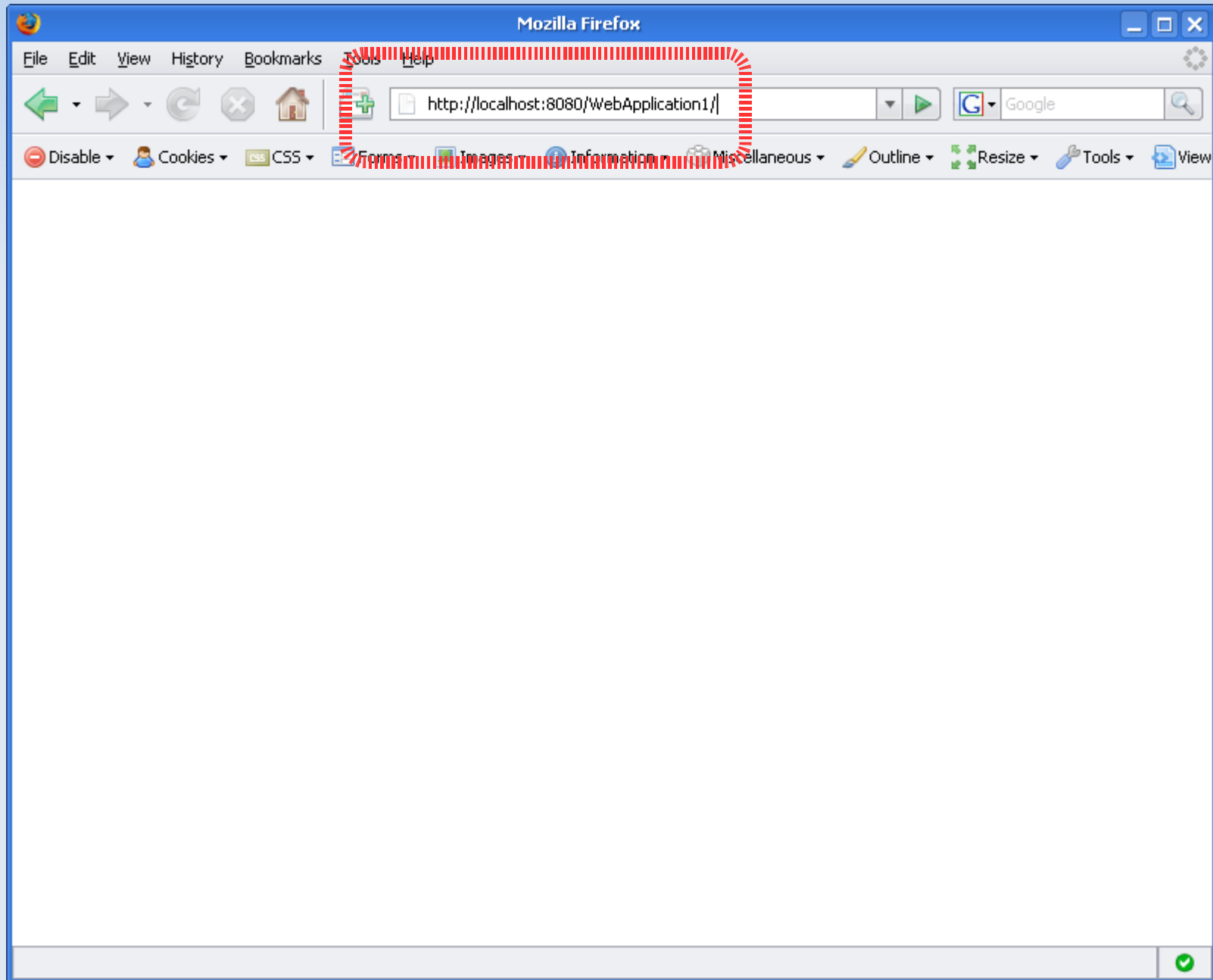
Какво е Selenium?

- Проект специализиран за уеб приложения
- Управлява браузър, симулирайки потребителски действия
 - щракане върху бутони/връзки, попълване на полета, drag & drop
- Мултиплатформен
 - Linux, Windows, Mac OS X
 - Firefox, Internet Explorer, Safari, Opera, Konqueror
- Проект с отворен код
 - стартира 2004, активна общност

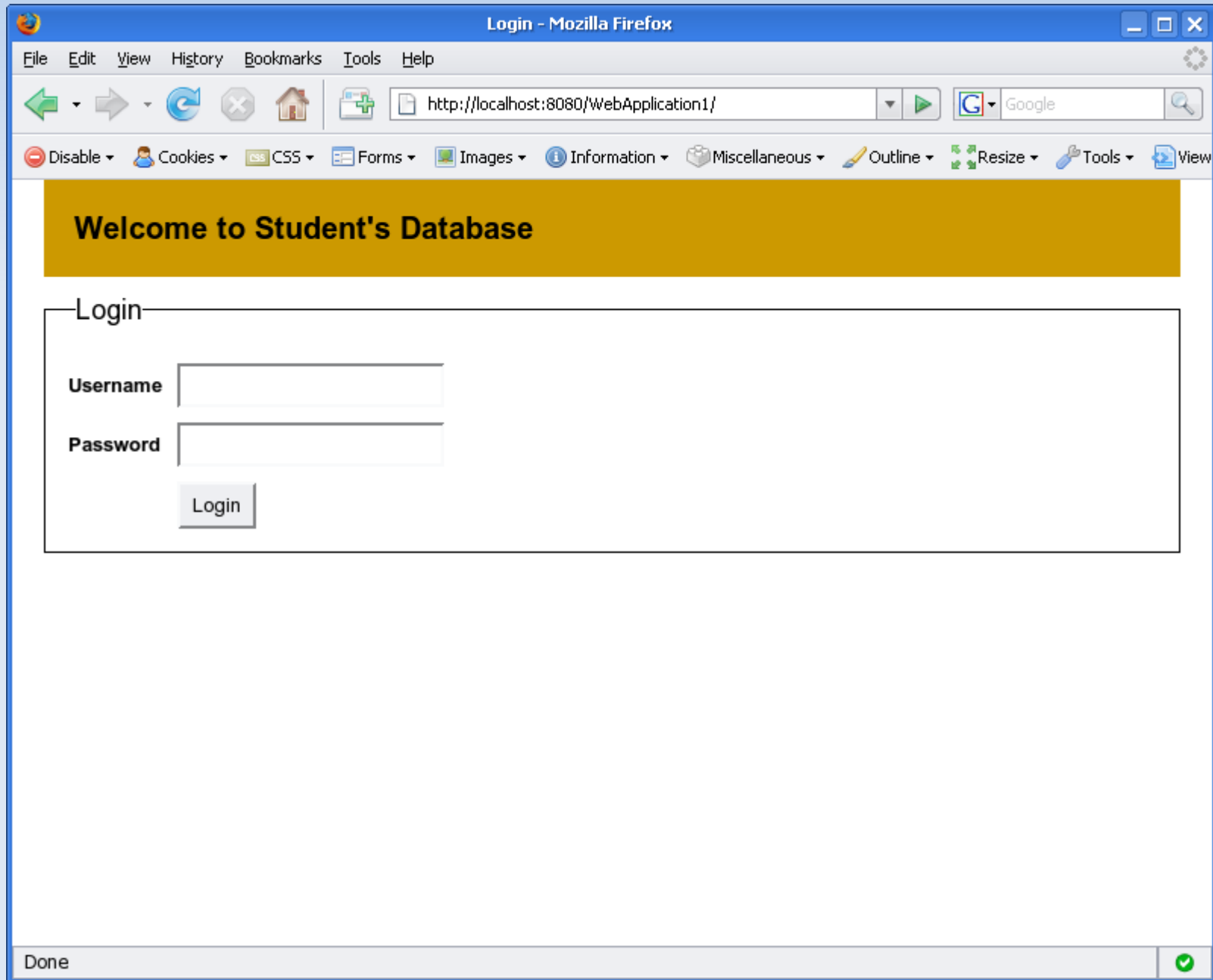
Пример: Тестване на регистрация



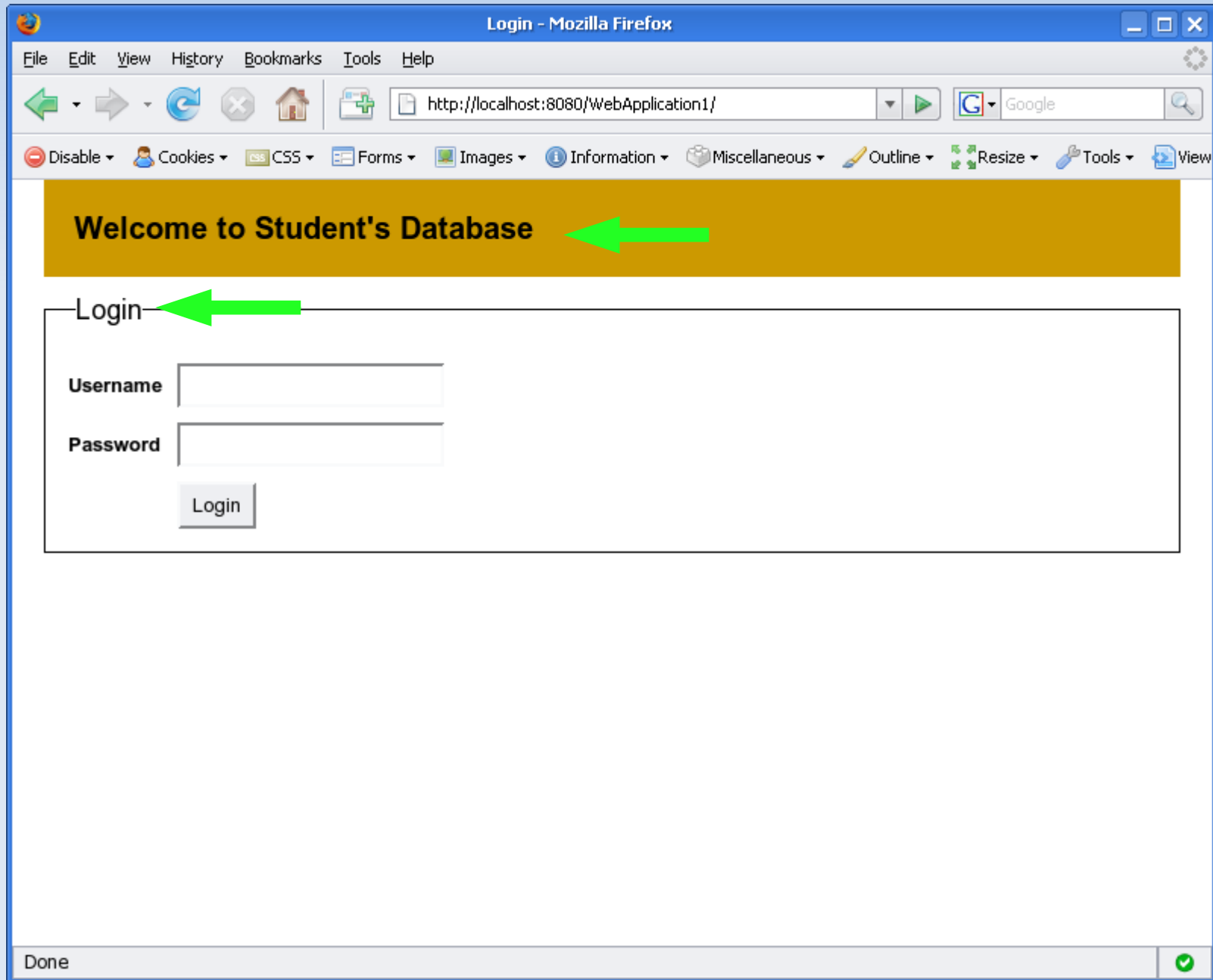
Пример: Тестване на регистрация



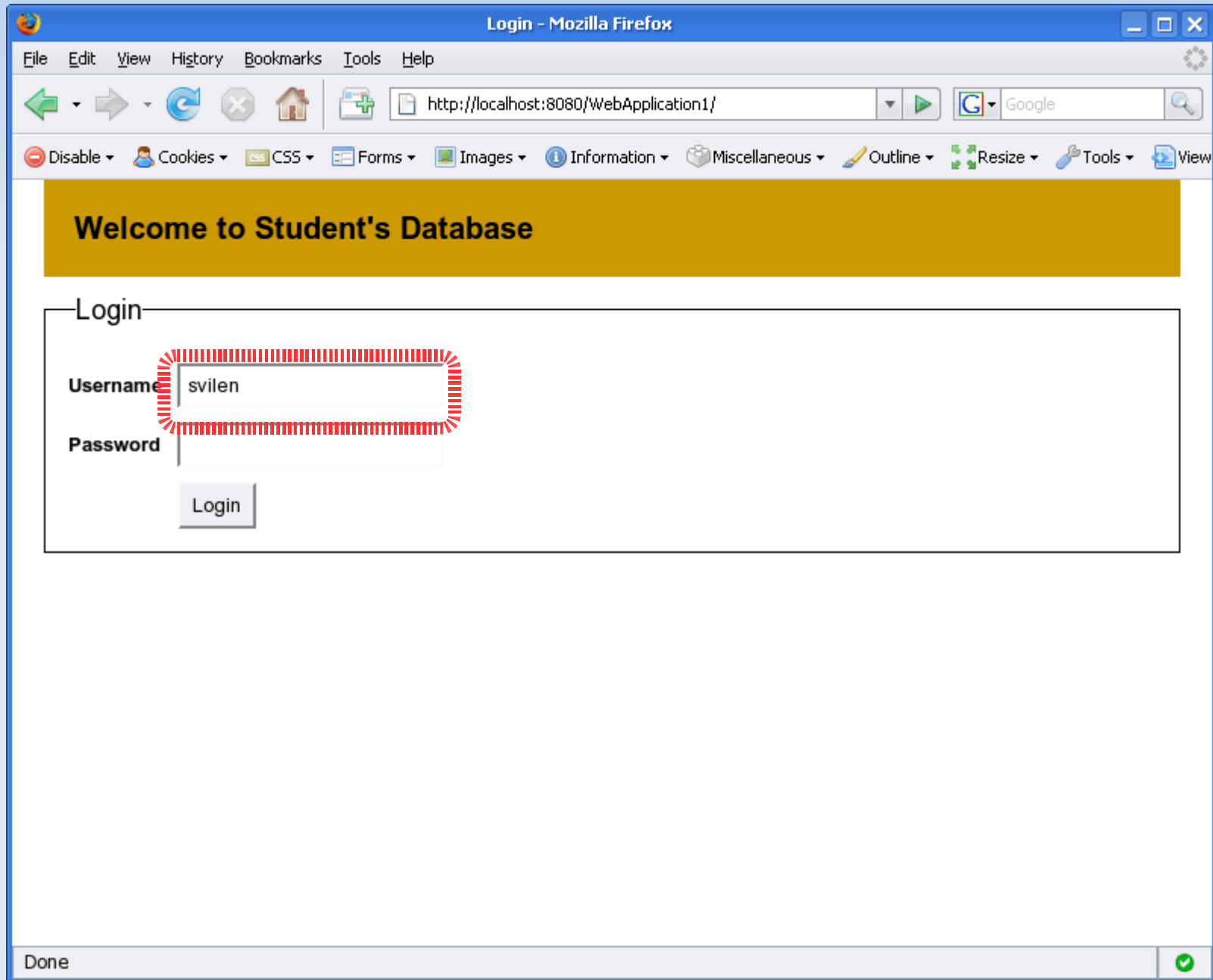
Пример: Тестване на регистрация



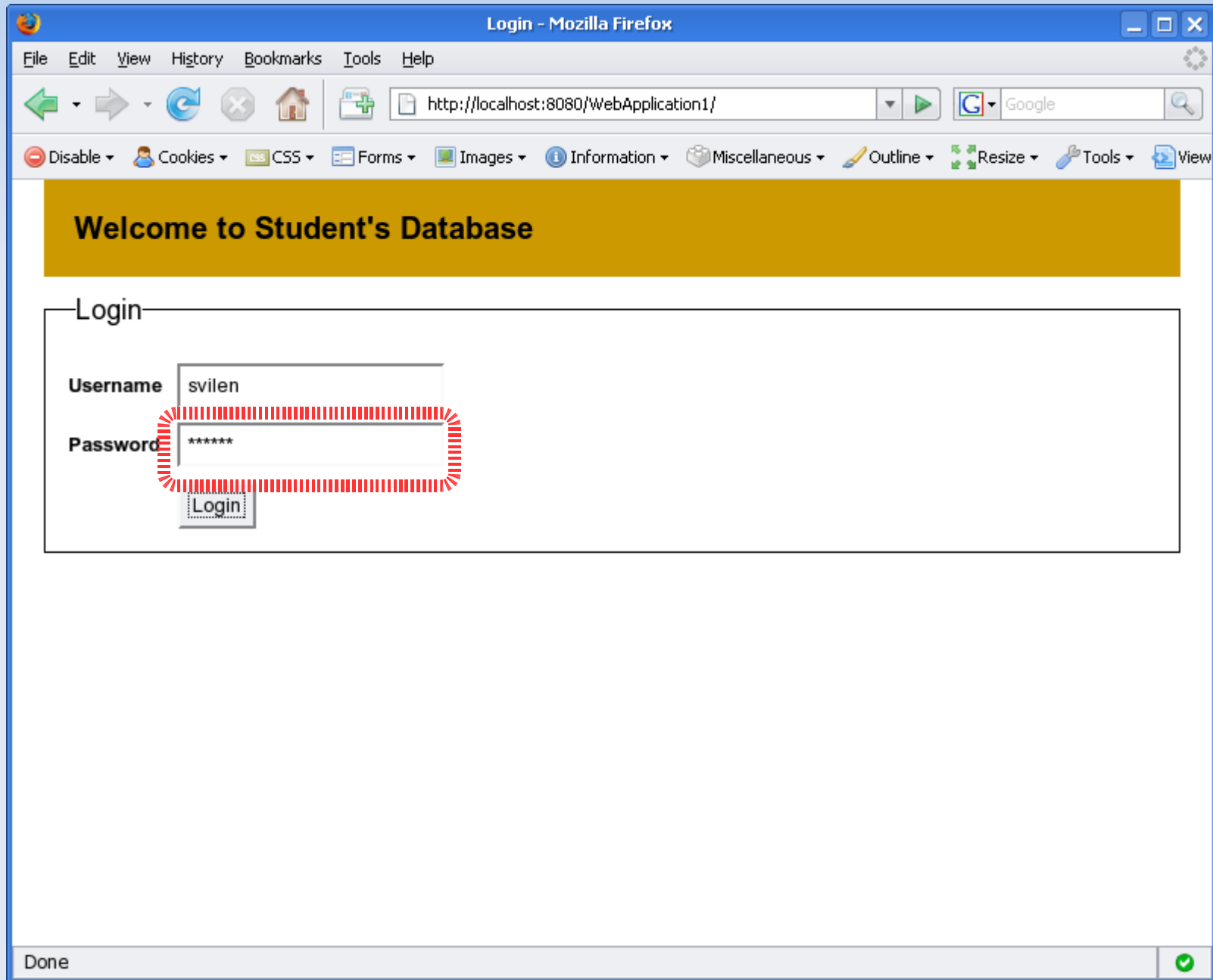
Пример: Тестване на регистрация



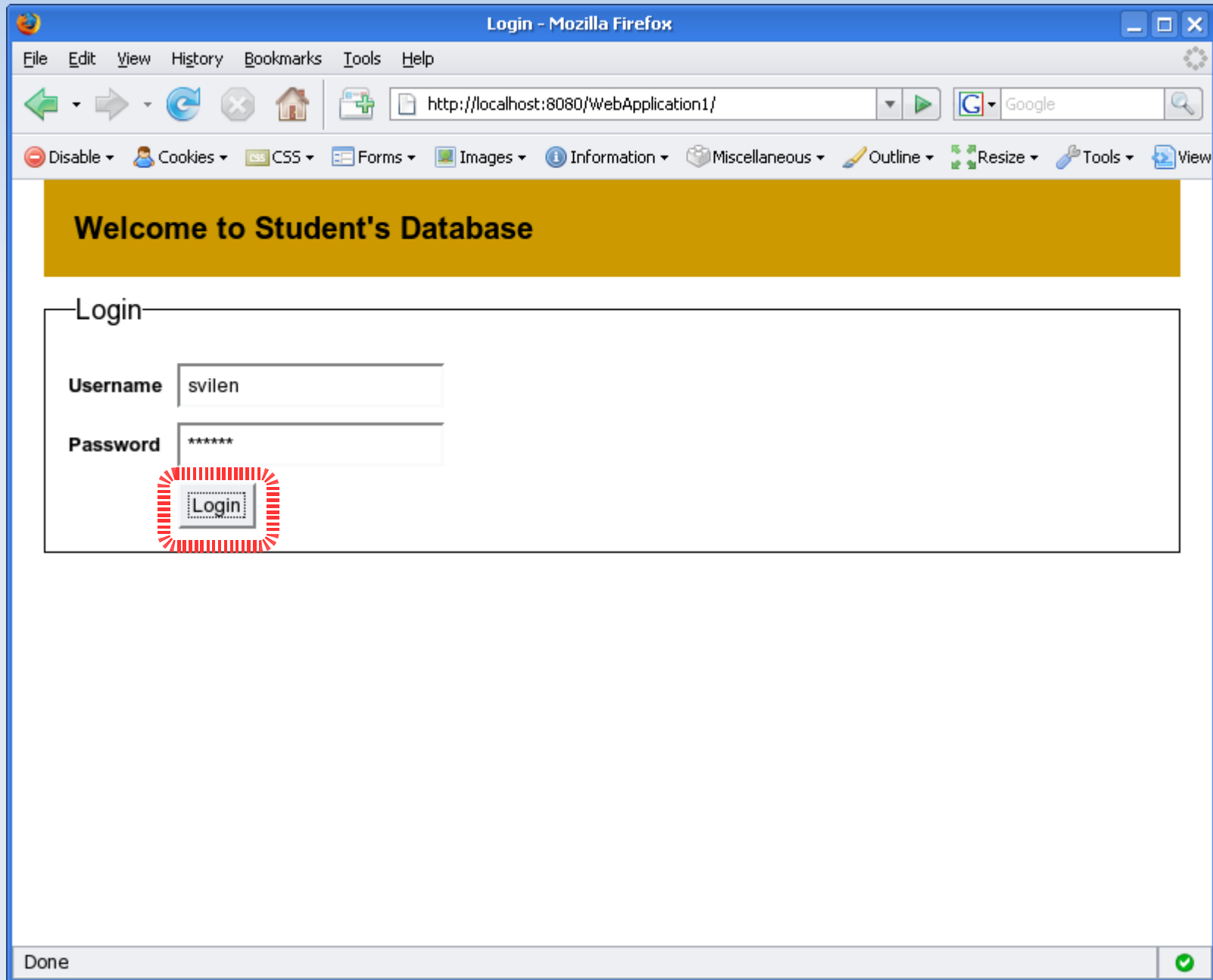
Пример: Тестване на регистрация



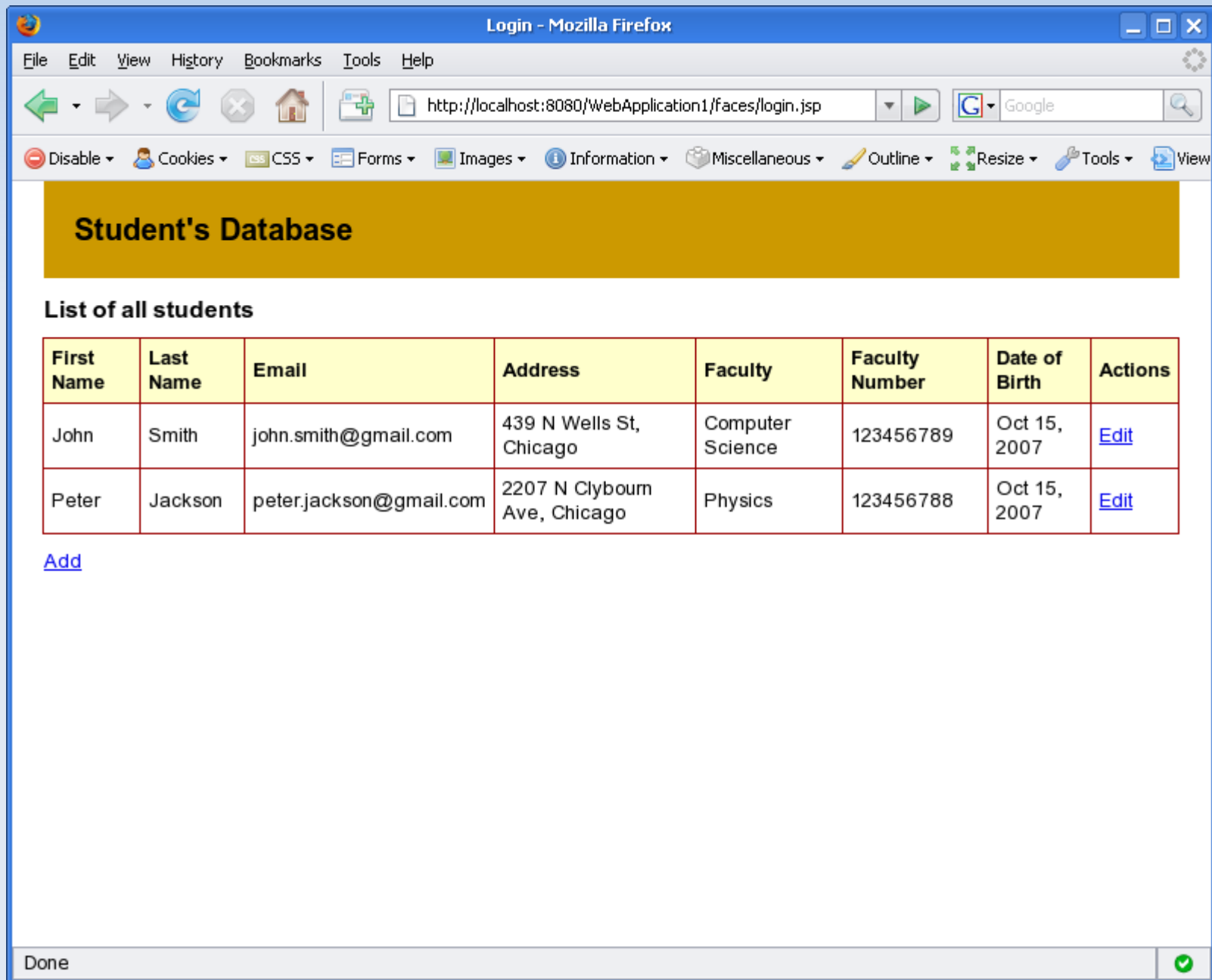
Пример: Тестване на регистрация



Пример: Тестване на регистрация



Пример: Тестване на регистрация



The screenshot shows a Mozilla Firefox browser window with the address bar displaying `http://localhost:8080/WebApplication1/faces/login.jsp`. The page content includes a yellow header with the text "Student's Database", a section titled "List of all students", and a table with the following data:

First Name	Last Name	Email	Address	Faculty	Faculty Number	Date of Birth	Actions
John	Smith	john.smith@gmail.com	439 N Wells St, Chicago	Computer Science	123456789	Oct 15, 2007	Edit
Peter	Jackson	peter.jackson@gmail.com	2207 N Clybourn Ave, Chicago	Physics	123456788	Oct 15, 2007	Edit

Below the table, there is a blue link labeled "Add". The browser's status bar at the bottom shows "Done" and a green checkmark icon.

Пример: Тестване на регистрация

Student's Database

List of all students

First Name	Last Name	Email	Address	Faculty	Faculty Number	Date of Birth	Actions
John	Smith	john.smith@gmail.com	439 N Wells St, Chicago	Computer Science	123456789	Oct 15, 2007	Edit
Peter	Jackson	peter.jackson@gmail.com	2207 N Clybourn Ave, Chicago	Physics	123456788	Oct 15, 2007	Edit

[Add](#)

Done

Пример: Тестване на регистрация

- Тестов сценарий:
 - 1.Отваряме приложението
 - Виждаме форма за въвеждане
 - 2.Въвеждаме валидни потребителски име и парола
 - 3.Натискаме „Login“
 - Ако видим списък със студенти, всичко е наред

Как изглежда един тест

Команда	Аргумент 1	Аргумент 2
open	<i>/WebApplication1/</i>	
verifyTextPresent	<i>Welcome to Student's Database</i>	
verifyTextPresent	<i>Login</i>	
type	<i>form:username</i>	<i>svilen</i>
type	<i>form:password</i>	<i>s3cre3t</i>
clickAndWait	<i>form:login</i>	
verifyTextPresent	<i>Student's Database</i>	
verifyTextPresent	<i>List of all students</i>	

Как изглежда един тест

```
01 <table>
02   <tr>
03     <td>open</td>
04     <td>/WebApplication1</td>
05     <td></td>
06   </tr>
07   <tr>
08     <td>verifyTextPresent</td>
09     <td>Welcome to
10       Student's Database</td>
11     <td></td>
12   </tr>
13   <tr>
14     <td>verifyTextPresent</td>
15     <td>Login</td>
16     <td></td>
17   </tr>
18   <tr>
19     <td>type</td>
20     <td>form:username</td>
21     <td>svilen</td>
22   </tr>
23     <td>type</td>
24     <td>form:password</td>
25     <td>s3cr3t</td>
26   </tr>
27   <tr>
28     <td>clickAndWait</td>
29     <td>form:login</td>
30     <td></td>
31   </tr>
32   <tr>
33     <td>verifyTextPresent</td>
34     <td>Student's
35       Database</td>
36     <td></td>
37   </tr>
38   <tr>
39     <td>verifyTextPresent</td>
40     <td>List of all
41       students</td>
42     <td>svilen</td>
43   </tr>
44 </table>
```

Как да стартираме теста (1/3)

- Чрез Selenium Test Runner
 - ядрото: Javascript/HTML за симулиране на действията на потребителя
 - позволява стартиране на тестовете в произволен браузър: Firefox, Internet Explorer, Opera, ...
 - необходим е достъп до уеб сървъра, където работи приложението
 - скриптове от други домейни нямат достъп до DOM на текущата страница

Как да стартираме теста (2/3)

- Чрез Selenium IDE
 - разширение към Firefox
 - стартиране на тест върху произволна инсталация на приложението без да е необходимо някакво допълнително инсталиране
 - подпомага разработването на нови тестове като записва действията на потребителя
 - идеален за бързо експериментиране с възможностите на Selenium
 - може да записва тестове като - HTML, Java, Perl, Python, Ruby, C#

Как да стартираме теста (3/3)

- Чрез Selenium Remote Control
 - прокси сървър между браузъра и приложението
 - „инжектира“ в кода на приложението необходимия Selenium код
 - командва браузъра дистанционно
 - позволява да се пишат тестове на различни езици: Java, Ruby, Perl, PHP, Python, C#.
 - дава възможност да се създават сложни тестове
 - стартиране на тестовете по график, като част от процеса на разработка
 - например при промяна на кода в SVN/CVS

ВЪЗМОЖНОСТИ

- Потребителски действия
 - Навигация
 - Попълване на входни полета
 - бутони, текстови полета, падащи менюта...
 - Щракване, влачене и пускане с мишка
 - Работа с cookies
- Проверка
 - наличие/отсъствие на текст /DOM елементи
 - alert(), confirm(), изкачащи прозорци
- Разширяване

Заклучение

- Selenium предлага автоматизиране на голяма част от ръчното тестване
- Преносимост на тестовете от едни браузър/ОС на други
- Предлага богат избор на езици за програмиране на тестовете
- Може да се стартира без намеса на тестера
- Предвидена е възможност за разширяване с функционалност специфична за конкретния проект