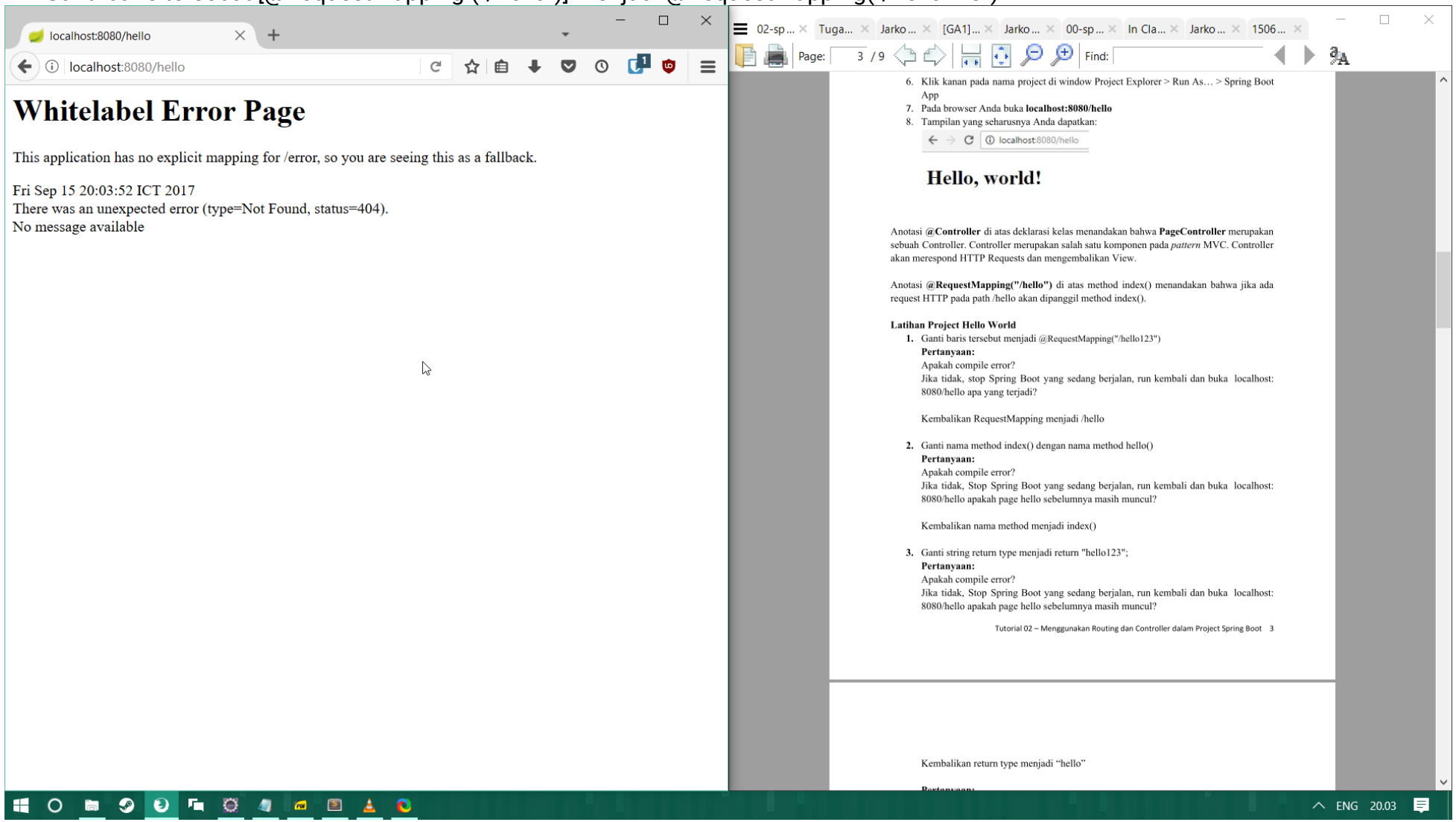
Anindito Izdihardian Wibisono
1506757466
APAP – C

Dalam tutorial kali ini, saya belajar mengenai pembuatan proyek sederhana dengan menggunakan Spring Boot pada IDE Eclipse EE. Saya memahami dan dapat mempraktekkan cara menyiapkan berkas controller serta mengatur request parameter dan path variable.

LATIHAN 1: PROJECT HELLO WORLD

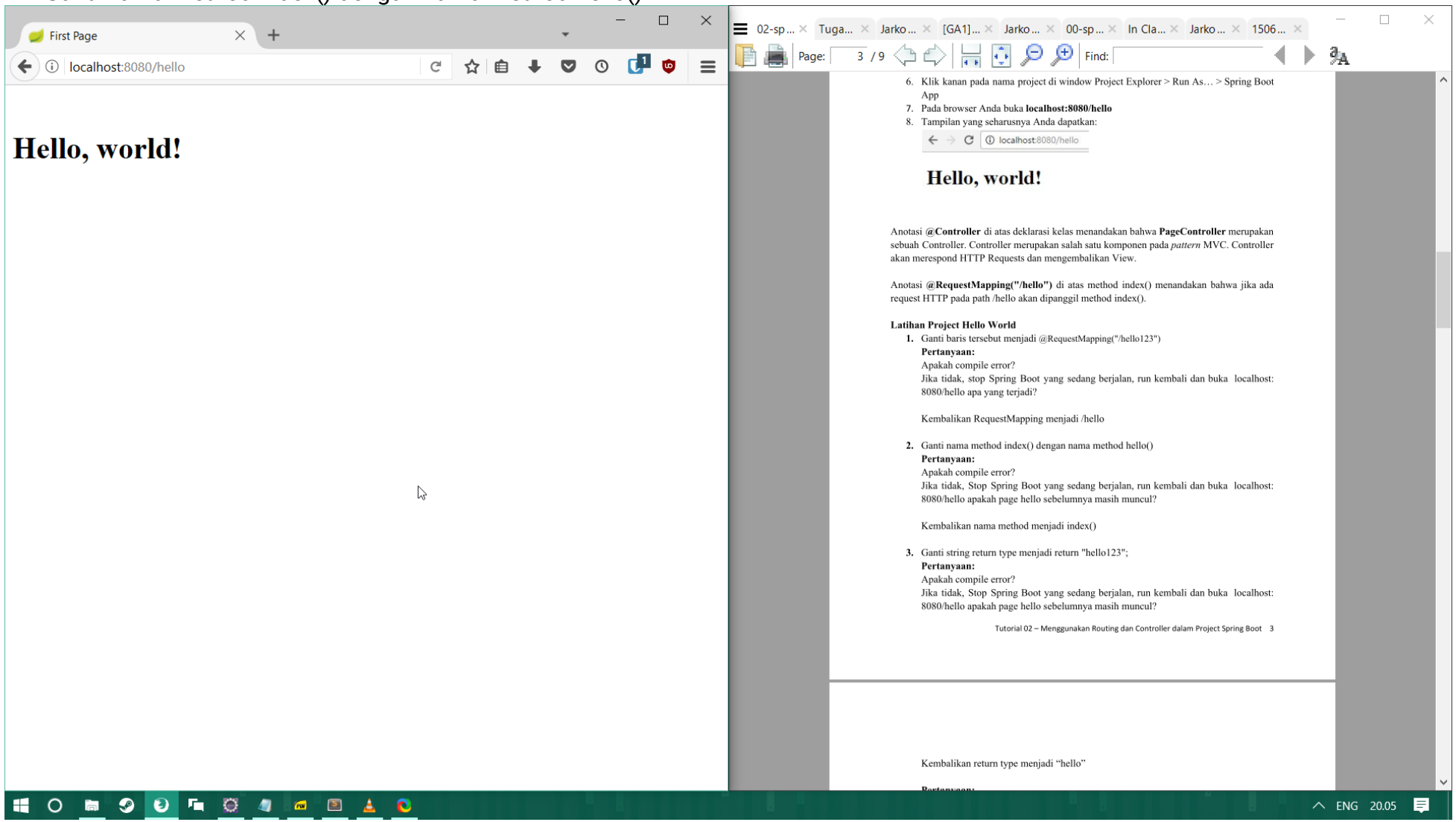
(screenshot ada di halaman-halaman selanjutnya)

1-1 Ganti baris tersebut [@RequestMapping ("/hello")] menjadi @RequestMapping("/hello123")



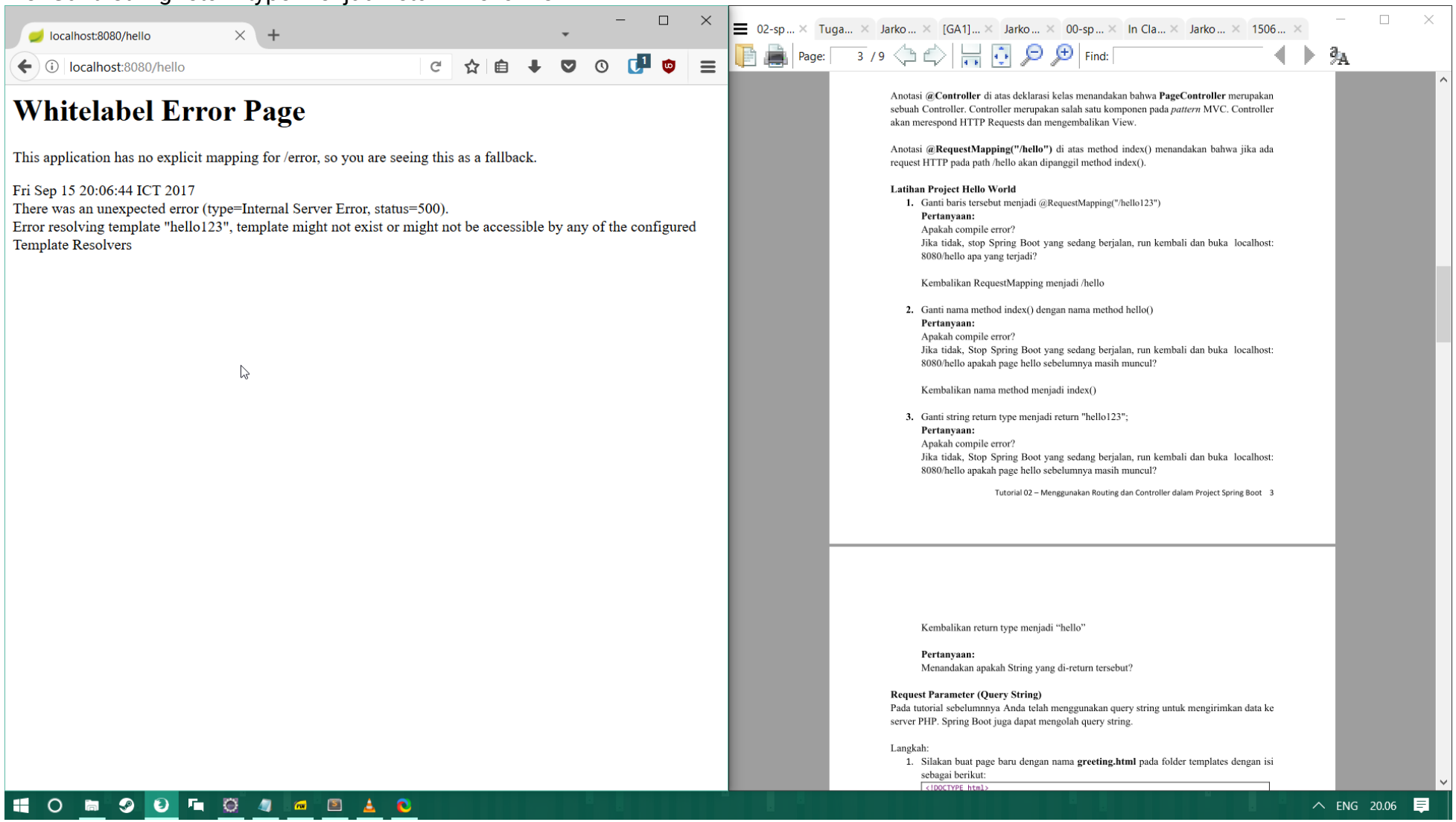
Hasil setelah file PageController.java disave dengan perubahan tersebut dan dijalankan. Tidak ada compile error, namun terjadi Whitelabel Error (404 Not Found).

1-2 Ganti nama method index() dengan nama method hello()



Hasil setelah file PageController.java disave dengan perubahan tersebut dan dijalankan. Tidak ada compile error dan halaman “hello” tetap dapat ditampilkan.

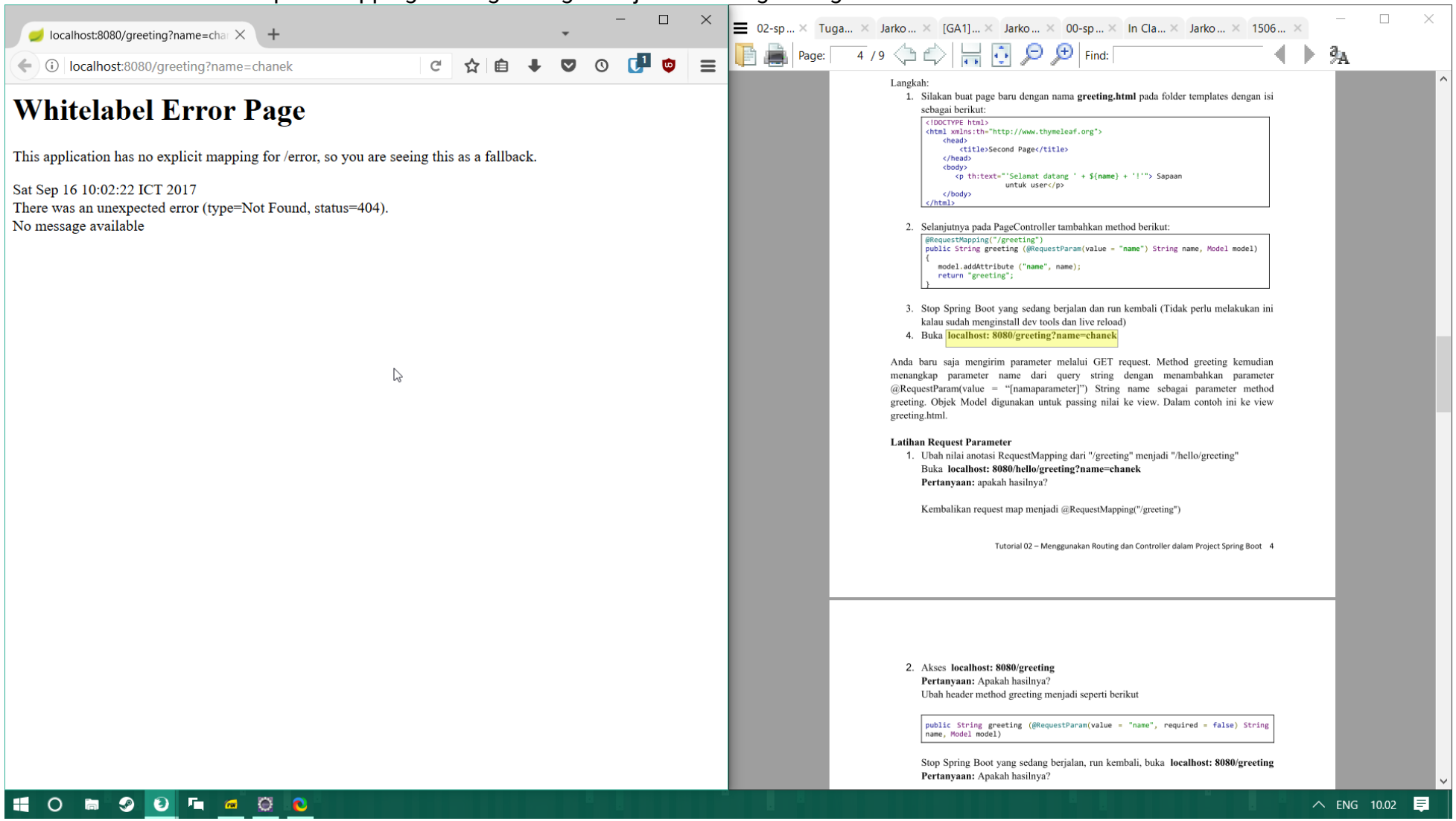
1-3 Ganti string return type menjadi return "hello123"



Hasil setelah file PageController.java disave dengan perubahan tersebut dan dijalankan. Tidak ada compile error, namun terjadi Whitelabel Error (500 Internal Server Error). Hal ini karena berdasarkan return string “hello123”, proyek mencari “hello123.html” untuk ditampilkan di browser, namun file tersebut tidak ada di templates.

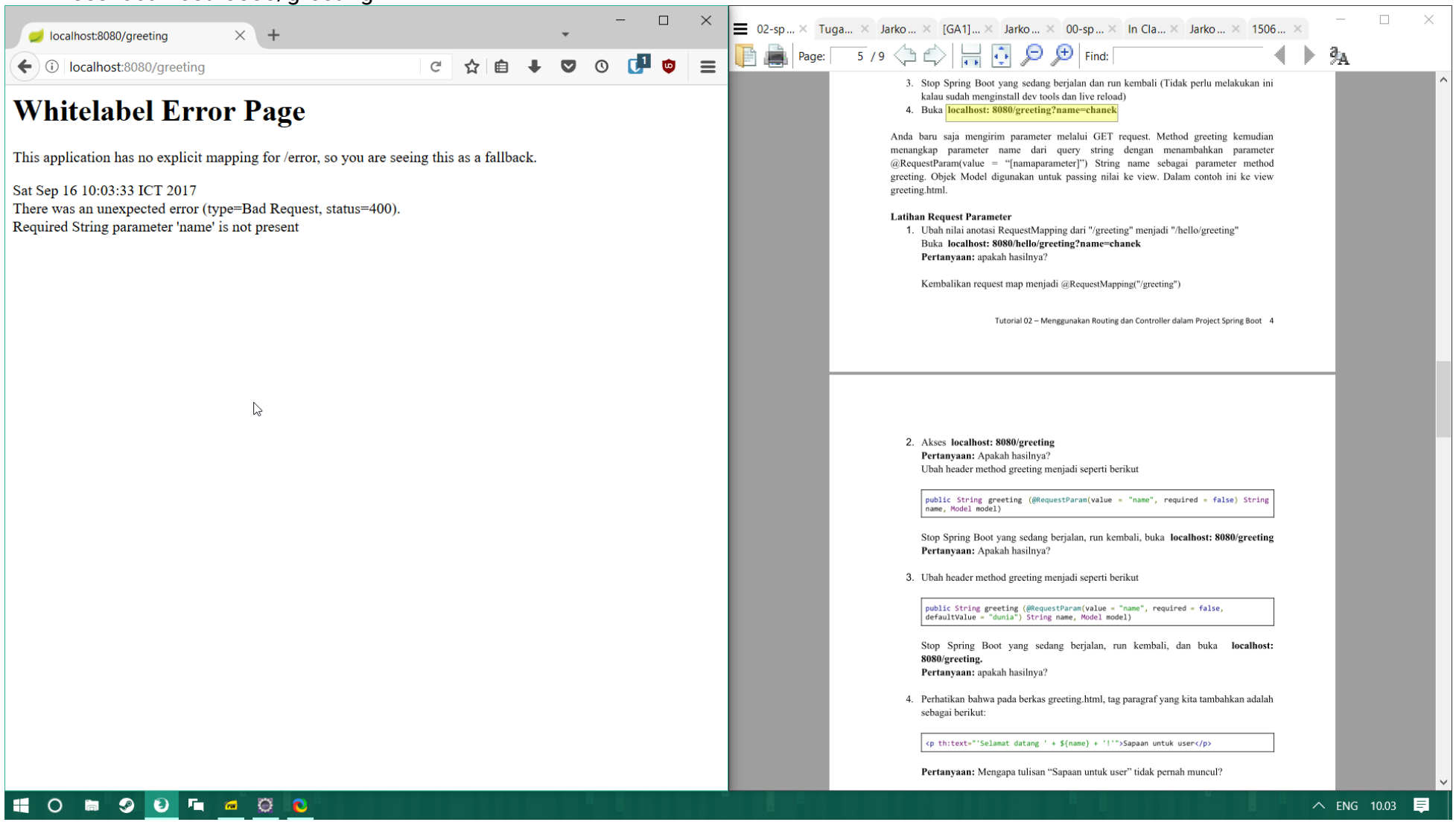
LATIHAN 2: REQUEST PARAMETER/QUERY STRING

2-1 Ubah nilai anotasi RequestMapping dari "/greeting" menjadi "/hello/greeting"

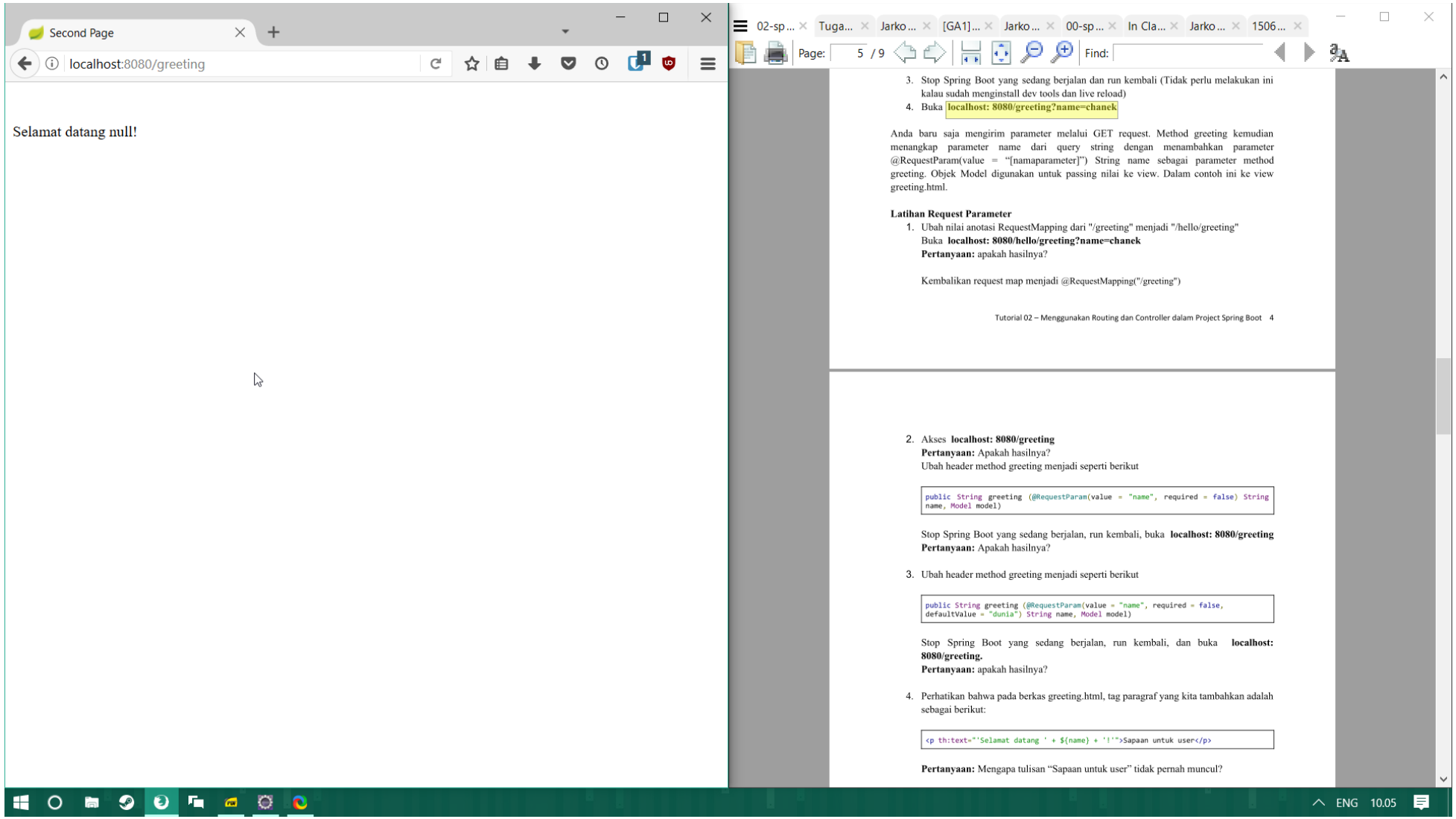


Hasil setelah file PageController.java disave dengan perubahan tersebut dan dijalankan. Tidak ada compile error, namun terjadi Whitelabel Error (404 Not Found).

2-2 Akses localhost: 8080/greeting

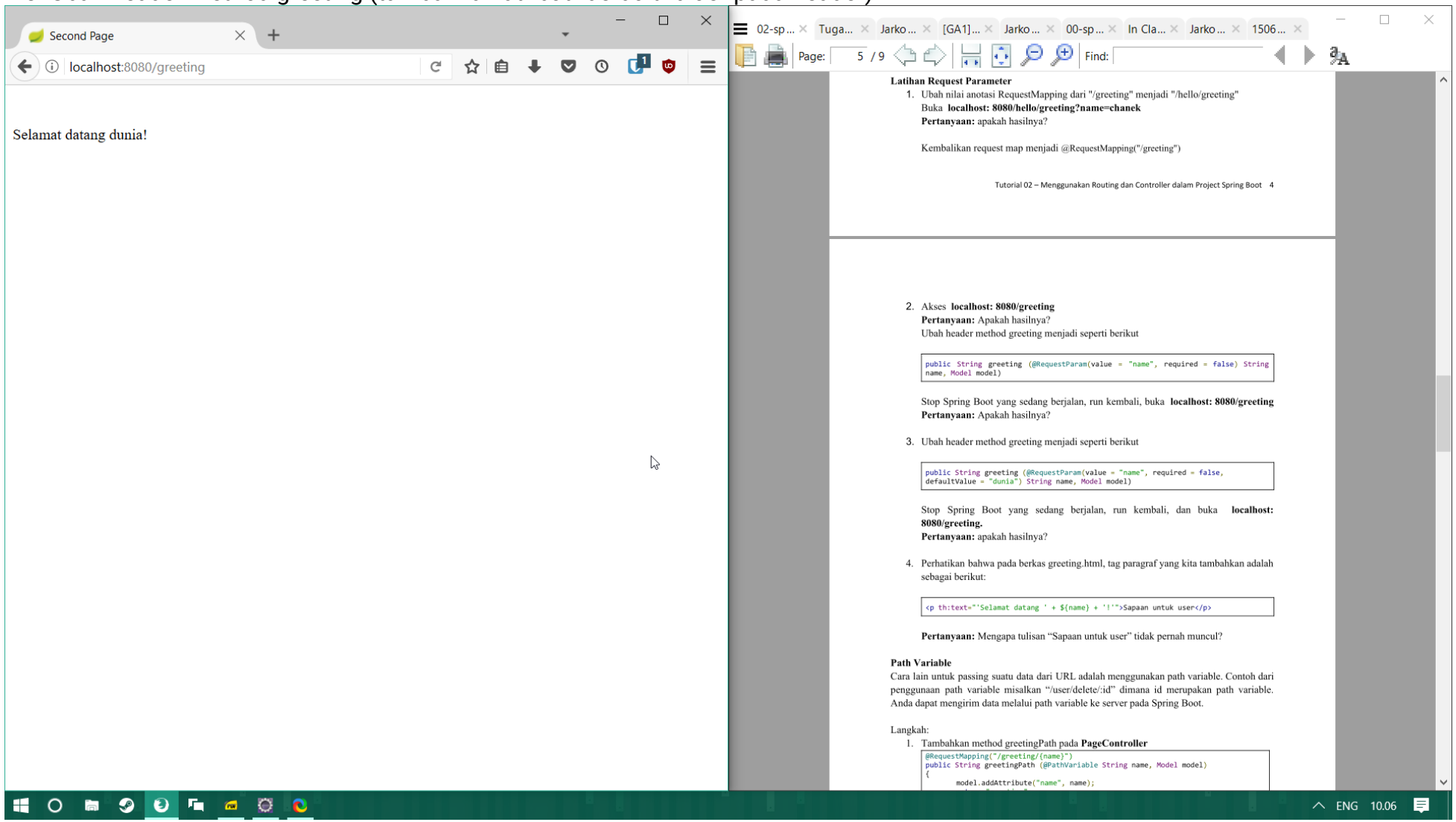


Hasil setelah file PageController.java disave dengan perubahan tersebut dan dijalankan. Tidak ada compile error, namun terjadi Whitelabel Error (400 Bad Request).



Hasil setelah file PageController.java disave dengan perubahan header method greeting dan dijalankan. Tidak ada compile error, dan ketika browser menampilkan halaman greeting, nilai "null" menggantikan tempat yang seharusnya diisi nama pada parameter.

2-3 Ubah header method greeting (tambahkan atribut “defaultValue” pada header)



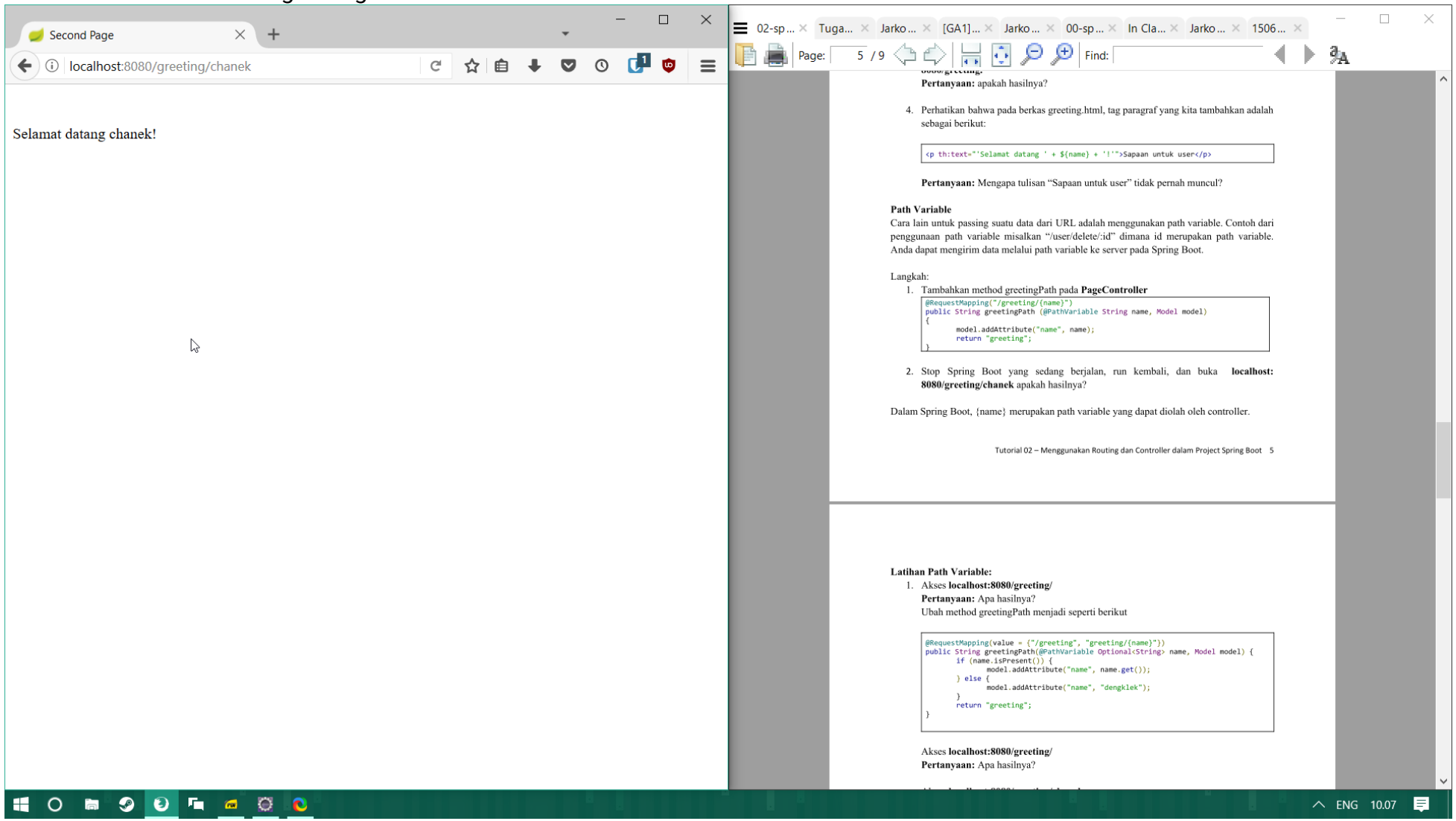
Hasil setelah file PageController.java disave dengan perubahan tersebut dan dijalankan. Tidak ada compile error, dan teks “dunia” sebagai default/placeholder mengisi tempat yang seharusnya diisi parameter nama.

2-4 Mengapa tulisan “Sapaan untuk user” tidak pernah muncul?

Atribut “th:text” pada file html akan mengatur isi dari atribut tersebut sebagai body/isi tag bersesuaian yang berisi tag “th:text”. Oleh karena itu, tulisan body yang sebenarnya selalu digantikan oleh isi atribut “th:text”.
(ref: http://www.thymeleaf.org/doc/tutorials/2.1/usingthymeleaf.html#using-thtext-and-externalizing-text)

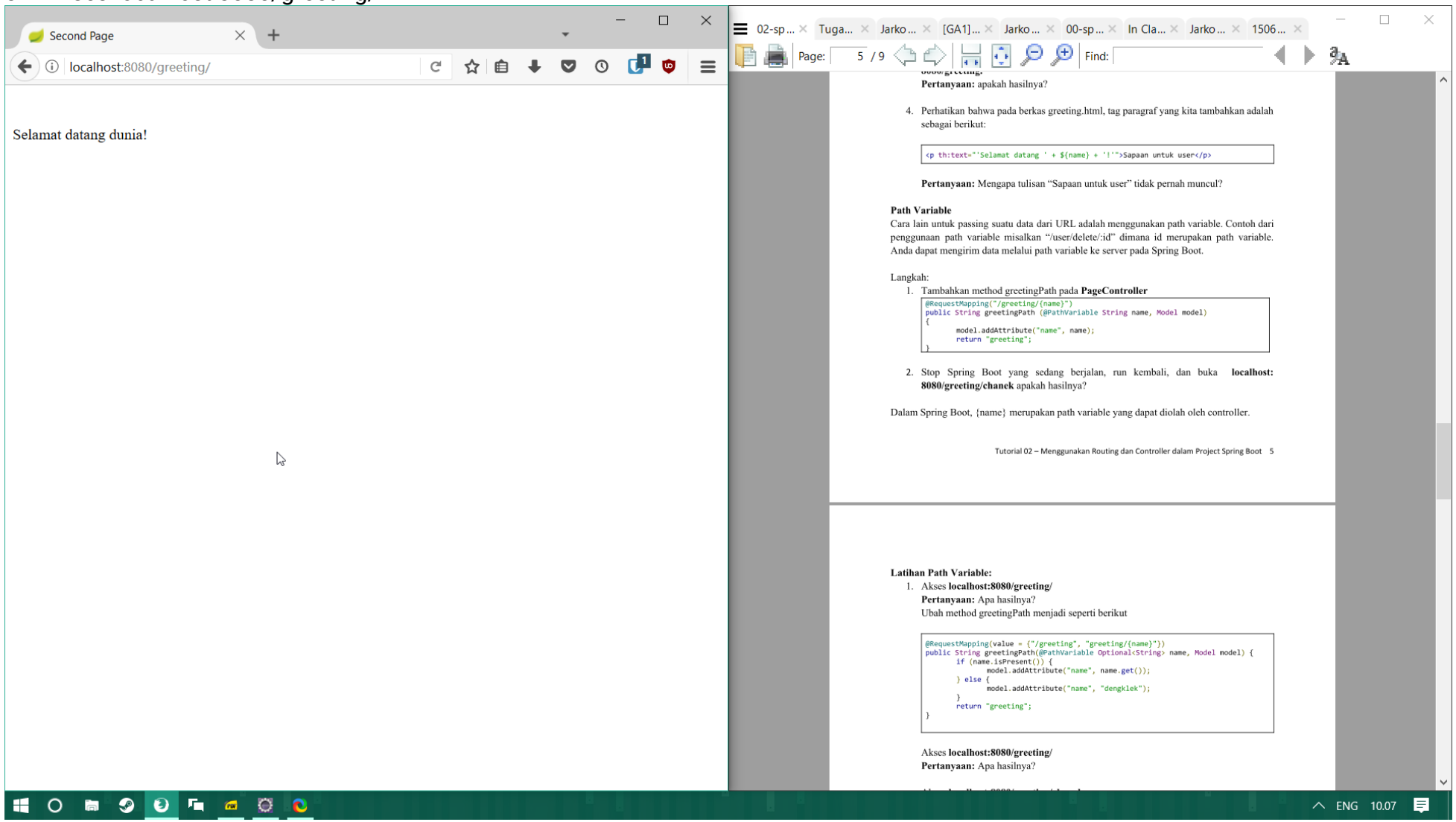
LATIHAN 3: PATH VARIABLE

3-1 Akses localhost:8080/greeting/chanek



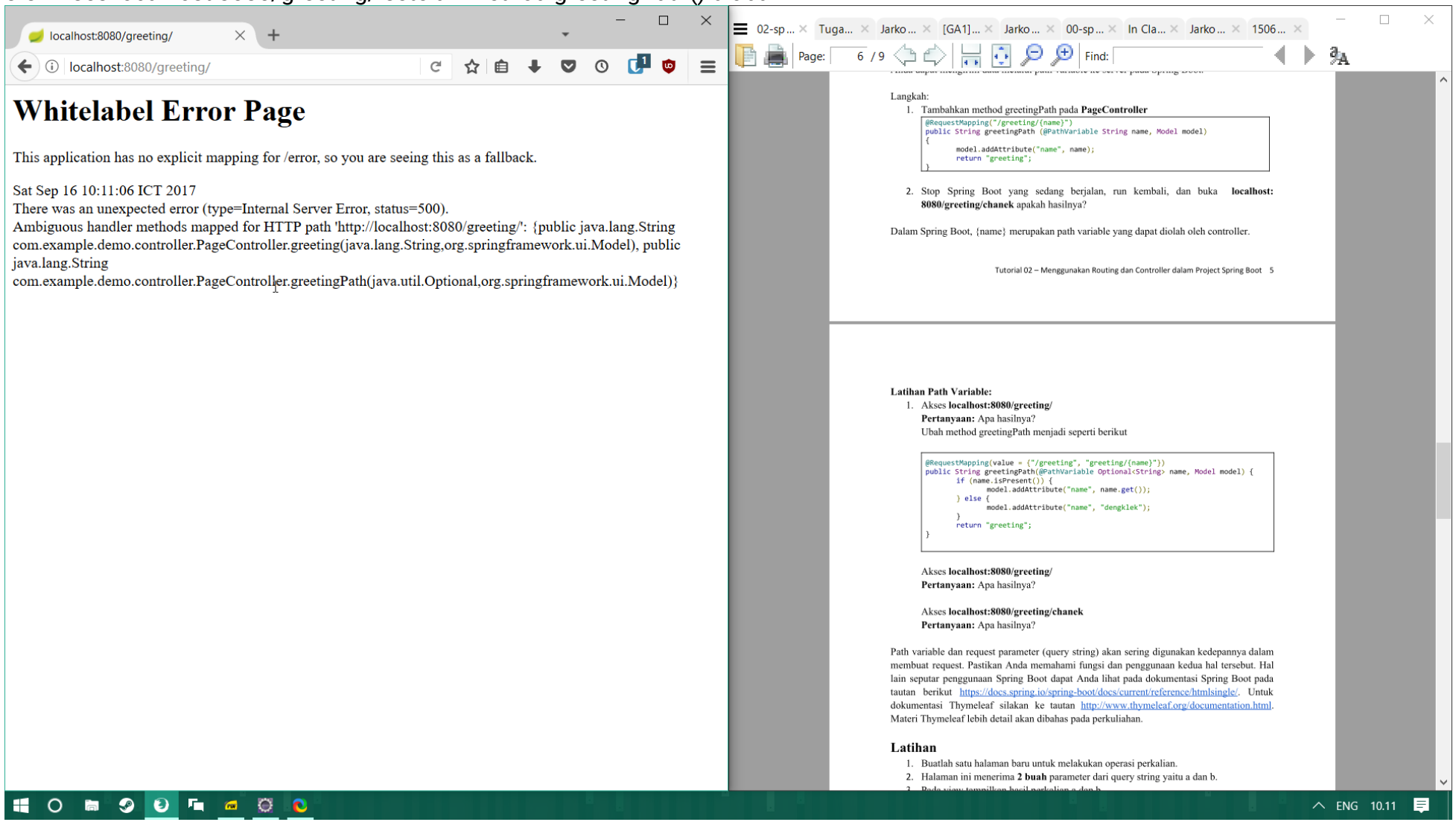
Hasil setelah file PageController.java disave dengan tambahan method greetingPath() dan dijalankan.

3-2 Akses localhost:8080/greeting/

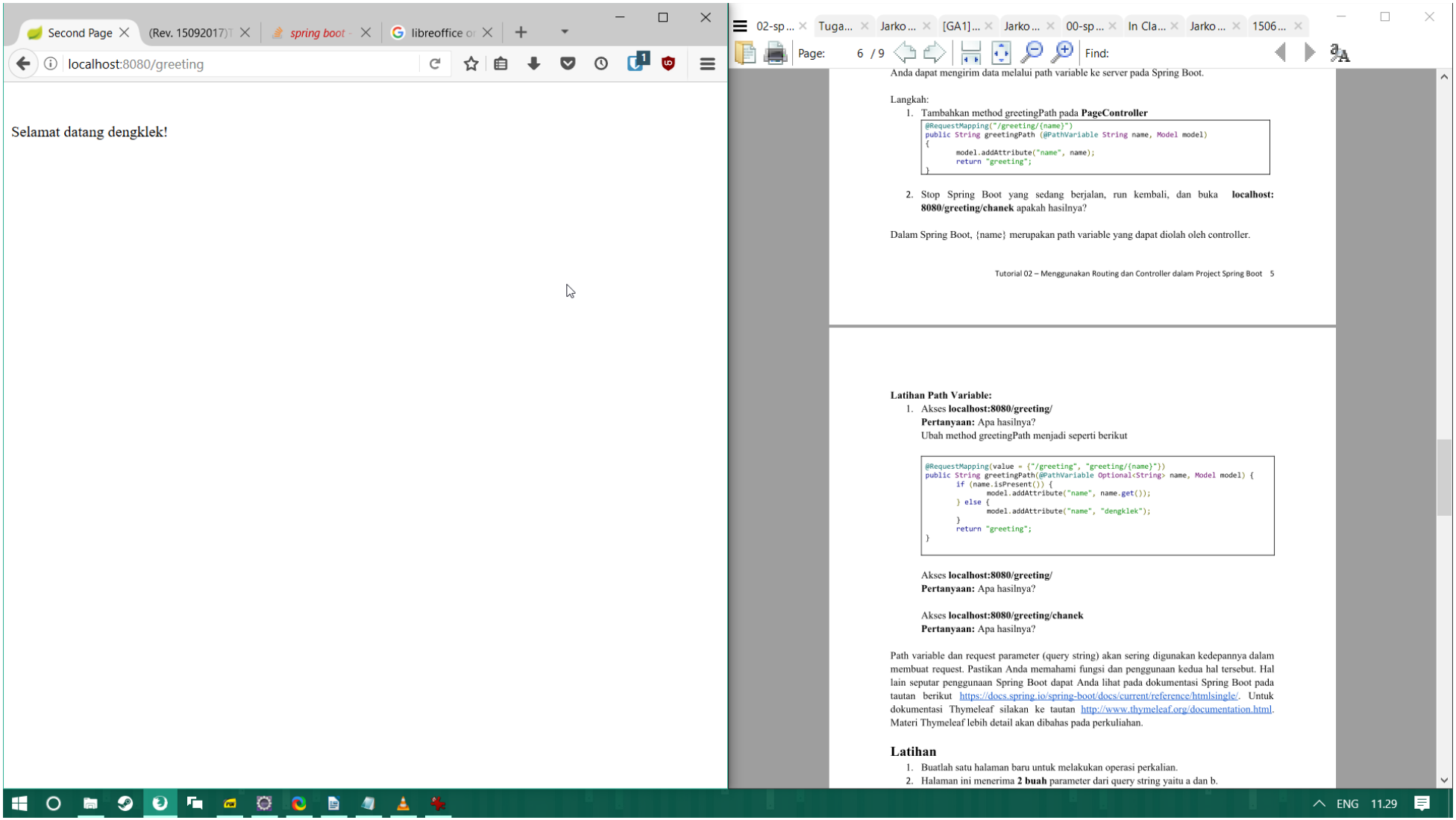


Teks yang ditampilkan menggunakan placeholder dari method greeting().

3-3 Akses localhost:8080/greeting/ setelah method greetingPath() diubah

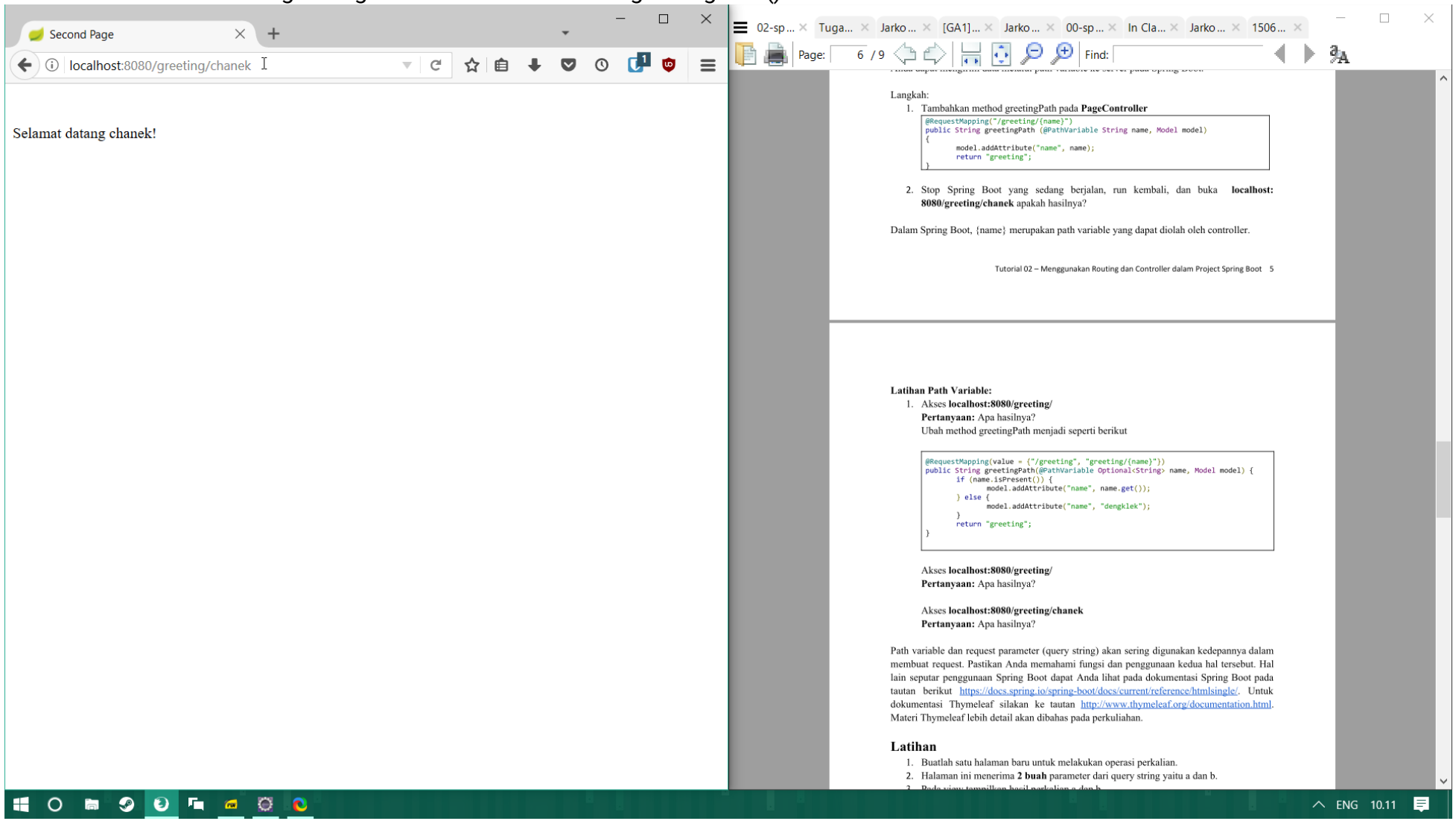


Hasil setelah file PageController.java disave dengan perubahan tersebut dan dijalankan. Tidak ada compile error, namun terjadi Whitelabel Error karena adanya ambiguitas method (greeting atau greetingPath) yang semestinya digunakan Spring Boot.



Apabila method greeting() di-comment, maka akses localhost:8080/greeting akan menggunakan placeholder dari method greetingPath(), yaitu “dengklek”.

3-4 Akses localhost:8080/greeting/chanek setelah method greetingPath() diubah



Hasil setelah file PageController.java disave dengan perubahan tersebut dan dijalankan. Tidak ada compile error, dan halaman greeting ditampilkan sesuai dengan input path nama (“chanek”).