**Conclusion**

In conclusion, we succeed to develop a working application with Microservice Architecture and Spring Boot in Enterprise Java. We implemented all the proposed essential features (Must-haves) and most of optional features (Nice-to-have) mentioned at Part 4.4. We now have an excellent understanding of all the technologies (can be found at Appendix 8.1) involved to make this Online Cash Register Application. We gained a very clear understanding regards microservices architecture and what advantages or disadvantages it has over more traditional web development. Besides, we know how to implement Eureka discovery server and Zuul gateway within microservices architecture. We also gained solid knowledge about Spring Security and how to implement it in different situations. Despite some small drawbacks, Microservice Architecture is a good choice in this context where developers are building a server-side enterprise application and it must support a variety of different clients including desktop browsers, mobile browsers and native mobile applications. (<http://microservices.io/patterns/microservices.html>)

We believe with all the advantages, Microservices Architecture and Spring Boot framework will be more and more widely used in web development and Enterprise Java.