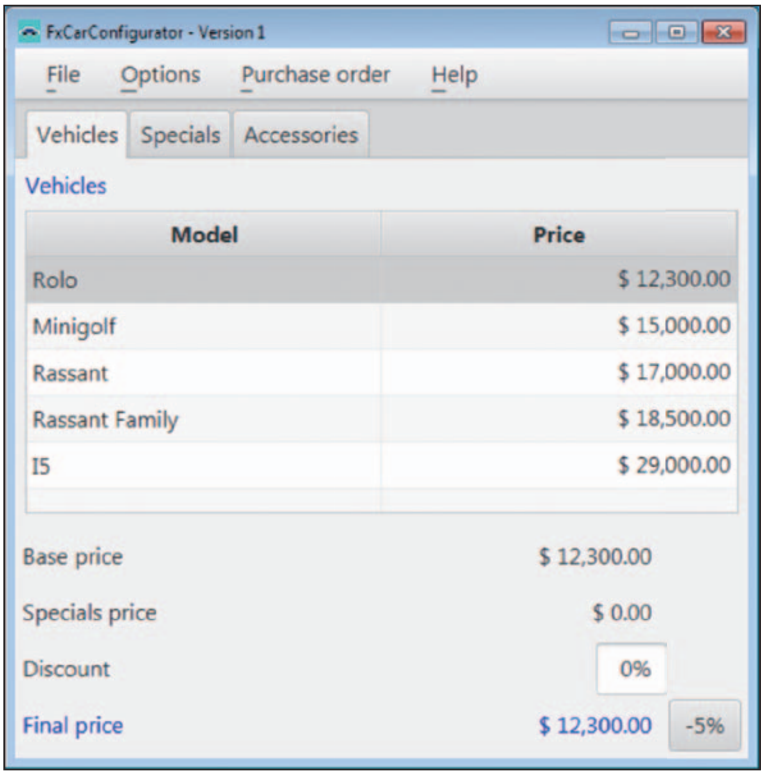
Testing JavaFX 8 UI application functionality

Automated testing of JavaFX GUI components

JavaFX is now celebrating one year in the Java8 club. As application development grows more and more complex, Oracle's Wolfgang Weigend shows us why GUI testing is so important.

by Wolfgang Weigend

As of March 2014, JavaFX has been a permanent part of JDK 8. In future, JavaFX 8 will be used to build mission critical business applications. On the one hand, this requires a degree of maturity of new JavaFX UI technology to be acceptable. On the other hand, it is necessary that user interfaces of business applications could be tested extensively for their correct application functionality. The focus is on the high data volume that is fed to automated tests into the graphical user interface. Please see below for an illustration of how such a test scenario could be set up with the QF-Test tool.



Vehicles

Model	Price
Rolo	$ 12,300.00
Minigolf	$ 15,000.00
Rassant	$ 17,000.00
Rassant Family	$ 18,500.00
I5	$ 29,000.00

Base price $ 12,300.00

Specials price $ 0.00

Discount 0%

Final price $ 12,300.00

Figure 1: Demo application "configuration of new cars" (2dimensional)

During applications development, testing plays an important role. According to the current World Quality Report, a quarter of the IT budget is already allotted for testing. The increasing complexity of the application development also need the testing requirements to increase. High test coverage can be achieved by test automation. Suitable tools help facilitate the setting up of automated tests and ensure a high integration with the developed application. The benefit of automated UI tests is that they can cover a whole process with one test in contrast to unit tests which cover only an isolated unit.

By using the correct test environment it is possible to have the bulk of tests set up by professional testers, thus freeing resources for development work. For this, the test environment should allow the tester to work with familiar terms and objects. Especially in GUI tests it is important that testers work with objects they know and recognize on the GUI of the application, even though the actual structure of the GUI is much more complex. Even in simple Java FX applications the GUI consists of many single elements in a highly complex tree structure. **Figure 1** shows a demo application for the configuration of new cars. The GUI structure of this application becomes visible in a 3D view (**Figure 2**), three-dimensionally highlighting the complexity of the nesting. However, only a small part of the elements is relevant for testing, all others are integrated on a technical level. The test tool allows the reduction of these complex GUIs to the essential.

Figure 3 shows the simplified structure as it is made available to the tester by the test tool. In specific cases it is possible to work with the full hierarchy at any time, and programmable tests are possible via scripting. Through simplification and generalization of the UI components, the support of a modular structure allows parts of the test to be reused. Thus the effort required for test creation is reduced and at the same time maintainability increased. It provides the opportunity to create modules and libraries professional testers can work with. Due to the additional integration of software drivers, data-driven testing is also possible on the GUI, and mass testing can be created.

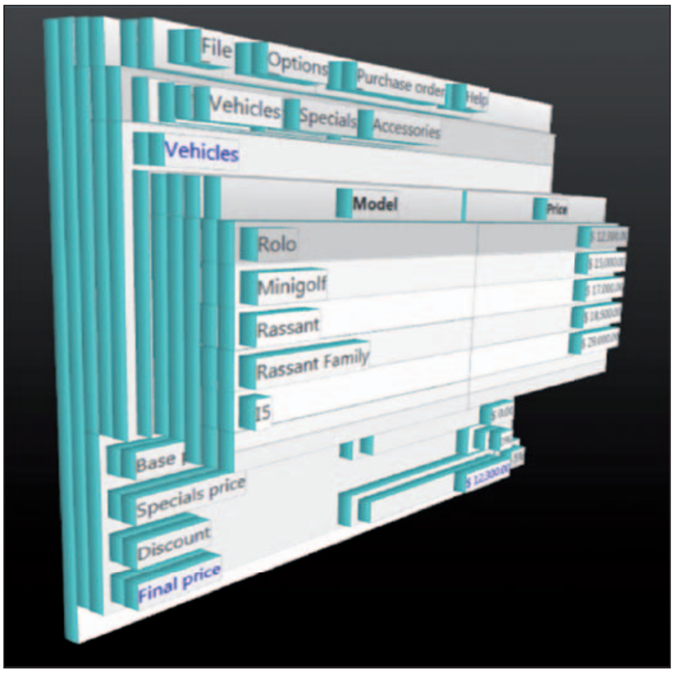


Figure 2: Demo application “configuration of new cars” (3dimensional)

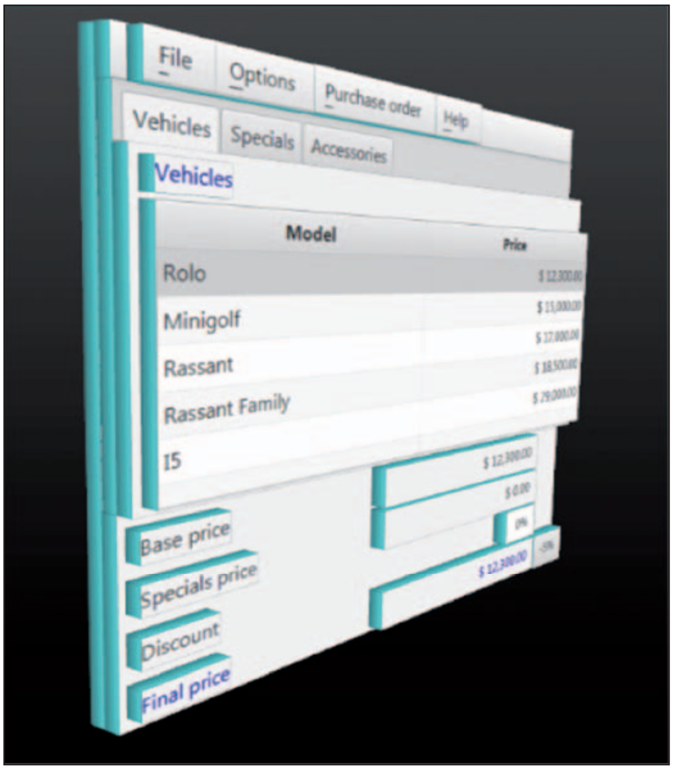


Figure 3: 3dimensional structure simplified by test tool QF-Test

Conclusion

GUI testing of an application is essential to the corresponding development. By directly integrating a test tool into the development process, a continuous integration scenario could be set up. Test tools are facilitating the JavaFX 8 applications testing by simplifying the component hierarchy and due to its modular structure. Also this is a valid option for testers, who haven't any programming knowledge.

Wolfgang Weigend is a senior leading systems consultant at Oracle Germany B.V. & Co. KG. He is chiefly involved in Java technology and architecture development for company-wide applications.

References

[1] http://docs.oracle.com/javase/8/javase-clienttechnologies.htm

[2] http://docs.oracle.com/javase/8/javafx/api/toc.htm

[3] http://www.capgemini.com/thought-leadership/world-quality-report-2014-15

[4] http://www.qfs.de/de/qftest/index.html