

Bibliographie pour le cours MGL7460

Réalisation et maintenance de logiciels

Guy Tremblay

Automne 2016

Manuel suggéré (mais non obligatoire)

- [1] A. Hunt and D. Thomas. *The Pragmatic Programmer—From Journeyman to Master*. Addison-Wesley, 2000. (En vente à la COOP-UQAM.)

Quelques livres particulièrement intéressants

- [1] K. Beck. *Test-Driven Development—By Example*. Addison-Wesley, 2003.
- [2] J.A. Campbell and P.P. Papatreou. *SonarQube In Action*. Manning, 2013.
- [3] J. Humble and D. Farley. *Continuous Delivery—Reliable Software Releases Through Build, Test, and Deployment Automation*. Addison-Wesley, 2011.
- [4] A. Hunt. *Pragmatic Thinking & Learning—Refactor Your “Wetware”*. The Pragmatic Bookshelf, 2008.
- [5] J. Rasmusson. *The Agile Samurai—How Agile Masters Deliver Great Software*. The Pragmatic Bookshelf, 2010.
- [6] J. Richardson and W.A. Gwaltney. *Ship it!: A Practical Guide to Successful Software Projects*. The Pragmatic Bookshelf, 2005.
- [7] J.F. Smart. *BDD In Action*. Manning, 2015.
- [8] V. Subramaniam and A. Hunt. *Practices of an Agile Developer—Working in the Real World*. The Pragmatic Bookshelf, 2006.
- [9] J. Visser. *Building Maintainable Software—Ten Guidelines for Future-Proof Code (Java Edition)*. O’Reilly, 2016.

Bibliographie détaillée (autres livres)

- [1] K. Beck. *Extreme Programming Explained—Embrace Change*. Addison-Wesley, 2000.
- [2] K. Beck. *Test-Driven Development—By Example*. Addison-Wesley, 2003.
- [3] K. Beck. *Implementation Patterns*. Addison-Wesley, 2008.
- [4] P. Bourque and R.E. Fairley, editors. *Guide to the Software Engineering Body of Knowledge (Version 3.0)*. IEEE Computer Society, 2014. <http://www.computer.org/web/swebok/v3>.
- [5] G.T. Brown. *Ruby Best Practices*. O’Reilly, 2009.
- [6] S. Chacon. *Pro Git*. Apress, 2009.
- [7] D. Chelimsky, D. Astels, Z. Dennis, A. Hellesoy, B. Helmkamp, and D. North. *The RSpec Book: Behaviour Driven Development with RSpec, Cucumber, and Friends*. The Pragmatic Bookshelf, 2010.
- [8] M. Cohn. *User Stories Applied: For Agile Software Development*. Addison-Wesley Professional, 2004.
- [9] D.B. Copeland. *Build Awesome Command-Line Applications in Ruby: Control Your Computer, Simplify Your Life*. The Pragmatic Bookshelf, 2012.
- [10] E. Evans. *Domain-Driven Design—Tackling Complexity in the Heart of Software*. Addison-Wesley, 2004.
- [11] M.C. Feathers. *Working Effectively with Legacy Code*. Prentice-Hall, 2005.
- [12] N. Ford. *The Productive Programmer*. O’Reilly, 2008.
- [13] M. Fowler. *Refactoring—Improving the Design of Existing Code*. Addison-Wesley, 1999.
- [14] M. Fowler. *Domain-Specific Languages*. Addison-Wesley, 2011.
- [15] A. Fox and D. Patterson. *Engineering Software as a Service—An Agile Approach Using Cloud Computing*. Strawberry Canyon LLC, 2013.
- [16] H. Fulton. *The Ruby Way, Second Edition*. Addison-Wesley, 2007.
- [17] D. Ghosh. *DSLs in Action*. Manning, 2011.
- [18] P. Goodliffe. *Code Craft—The Practice of Writing Excellent Code*. No Starch Press, 2006.
- [19] A. Hunt. *Pragmatic Thinking & Learning—Refactor Your “Wetware”*. The Pragmatic Bookshelf, 2008.
- [20] A. Hunt and D. Thomas. *Pragmatic Unit Testing In Java with JUnit*. The Pragmatic Bookshelf, 2003.

- [21] R. Jeffries. *The Nature of Software Development—Keep it Simple, Make it Valuable, Build it Piece by Piece*. The Pragmatic Bookshelf, 2015.
- [22] B.W. Kernighan and R. Pike. *The Practice of Programming*. Addison-Wesley, 1999.
- [23] A. Koleshko. *Rake Task Management Essentials*. Packt Publishing, 2014.
- [24] J. Kutner. *The Healthy Programmer—Get Fit, Feel Better, and Keep Coding*. The Pragmatic Bookshelf, 2013.
- [25] B. Marick. *Everyday Scripting with Ruby—For Teams, Testers, and You*. The Pragmatic Bookshelf, 2006.
- [26] R.C. Martin. *Agile Software Development—Principles, Patterns, and Practices*. Prentice-Hall, 2003.
- [27] R.C. Martin. *Clean Code—A Handbook of Agile Software Craftmanship*. Prentice-Hall, 2009.
- [28] R.C. Martin. *The Clean Coder—A Code of Conduct for Professional Programmers*. Prentice-Hall, 2011.
- [29] S. McConnell. *Code Complete—A Practical Handbook of Software Construction (Second Edition)*. Microsoft Press, 2004.
- [30] G. Meszaros. *xUnit Test Patterns—Refactoring Test Code*. Addison-Wesley, 2007.
- [31] B. Meyer. *Agile!: The Good, the Hype and the Ugly*. Springer, 2014.
- [32] N. Middleton and R. Schneeman. *Heroku: Up and Running*. O’Reilly, 2013.
- [33] R. Olsen. *Design Patterns in Ruby*. Addison-Wesley, 2008.
- [34] R. Olsen. *Eloquent Ruby*. Addison-Wesley Professional Ruby Series, 2011.
- [35] P. Perrotta. *Metaprogramming Ruby*. The Pragmatic Bookshelf, 2010.
- [36] T.M. Pigoski. *Practical Software Maintenance—Best Practices for Managing Your Software Investment*. John Wiley & Sons, Inc., 1997.
- [37] E.S. Raymond. *The Art of UNIX Programming*. Addison-Wesley, 2004.
- [38] J.F. Smart. *Jenkins: The Definitive Guide—Continuous integration for the masses*. O’Reilly, 2011.
- [39] V. Subramaniam. *Programming Groovy—Dynamic Productivity for the Java Developer*. The Pragmatic Bookshelf, 2008.
- [40] A.A. Takang and P.A. Grubb. *Software Maintenance—Concepts and Practice*. International Thomson Computer Press, 1996.
- [41] B.A. Tate. *Seven Languages in Seven Weeks: A Pragmatic Guide to Learning Programming Languages*. Addison-Wesley, 2010.
- [42] D. Thomas and A. Hunt. *Programming Ruby: The Pragmatic Programmer’s Guide*. Addison-Wesley, 2001.

- [43] V. Vernon. *Implementing Domain-Driven Design*. Addison-Wesley, 2013.
- [44] M. Wynne and A. Hellesoy. *The Cucumber Book: Behaviour-Driven Development for Testers and Developers*. The Pragmatic Bookshelf, 2012.
- [45] A. Zeller. *Why Programs Fail—A Guide to Systematic Debugging*. Morgan Kaufmann Publishers and dpunkt.verlag, 2006.