

A tester's guide to win developers respect

Carmen Sighiartau

The beginning





Pragmatic Programmer



from journeyman to master

Andrew Hunt David Thomas

Foreword by Ward Cunningham

"Care about your craft"

We feel that there is no point in developing testing software unless you care about doing it well.

"Think! About Your Work"

We're challenging you to think about what you're doing while you're doing it. Never run on auto-pilot. Constantly be thinking, critiquing your work in real time.



My first project



My first encounter with test cases





"The whole is greater than the sum of its parts"



Performance evaluation frameworks - the old





Performance evaluation frameworks - the new











Who would you prefer to work with?

- ★Senior tester
- ☆Very good testing knowledge
- ★No knowledge sharing in
- the team
- ★No clarifications of the tasks
- ★The best at executing well specified tasks
- ★ Hatter attitude

- ★Mid tester
- ☆Solid testing knowledge
- ☆Willing to support the client
- ★Helping the team do a better job
- ★Respects his colleagues
- ☆Gives constant positive and constructive feedback



★Understand different technologies



- ★Understand different technologies
- *Know the project architecture



- ☆Understand different technologies
- *Know the project architecture
- ★DB scripting



- ☆Understand different technologies
- *Know the project architecture
- **☆**DB scripting
- ☆Deployment model



- ☆Understand different technologies
- *Know the project architecture
- **☆**DB scripting
- ☆Deployment model
- **★TEST** environment



- ☆Understand different technologies
- ★Know the project architecture
- **☆**DB scripting
- ☆Deployment model
- **★TEST** environment
- ★Version control tools



- ★Understand different technologies
- ★Know the project architecture
- **☆**DB scripting
- ☆Deployment model
- **★TEST** environment
- *Version control tools
- *Knowledge sharing between devs and testers



- ★Understand different technologies
- ★Know the project architecture
- **☆**DB scripting
- ☆Deployment model
- **★TEST** environment
- *Version control tools
- *Knowledge sharing between devs and testers









Heuristics for recognizing professional testers

They have a **paradigm** of what testing is and they can **explain** their approach in any given situation

They consider **context first** and continuously

They know that software development and testing is a team sport

They ask questions before doing anything

They use diversified approaches

They refuse to do bad work, never fake and have the courage to tell their clients and the people they work about their ethics and values



Are you able to explain the value that you bring to your team/project?

Rate this presentation using #4792

http://www.sli.do/