Test Promotion Tip: Understand the developer

I wrote a list of tips for promoting software testing. One of these was: “ Understand the developer.”

Try to understand the developer and know how to deal with developers.
Soft skills are vital for a tester in the process of creating test awareness in his organization. The following list can help a little with your empathy to developers and guide you on how to interact with developers.

The list is already 11 years old, but still going strong. And it is perhaps even more important now agile development is practiced more and more!

Good tester	**Good developer**
Domain knowledge	Knowledge of the inside of the product
Quickly up to speed	Thorough understanding
Ignorance is important	Expertise is important
Model is user behaviour	Model is system design model
Focus on what can go wrong	Focus on how it can work
Focus on the severity of the problem	Focus on the interest in a problem
Empirical	Theoretical
What is observed	How is it designed
Sceptical	Believer
Tolerate boredom	Automates boredom
Comfortable with conflicts	Avoid conflict
Reporting problems	Understanding problems

(From Pettichord, "Testers and Developers Think Differently: Understanding and Utilize Different Traits of the Key Players on Your Team," Software Testing & Quality Engineering, Vol. 2, no. 1, 2000, pp. 42-45.)

Now a few additional tips to get you on your way:

- Convince the developer that their software quality is their first and foremost priority.
- Conferring with the developer on the test strategy and test cases is a good way to start this process.
- Communicate continuously with the developer - for example, let them do a check on any major findings before you log them into your findings database. This will give the developer time to make / or think about the solution before the next issue meeting starts.

Show intelligence:

- Know the functional and technical specifications.

- Make notes when you ask questions, so you're not always asking the same question over and over again.
- Log your issues with reproducible steps and give a clear regression analysis of the impact of the issue (if applicable)

More tips? Search for agile on the Internet. As agile has focus on collaboration you can get a lot of ideas concerning this topic, even if you're not in an agile development trajectory.

Also search for the complete article of Pettichord on the internet for more background info.