

Unit testing - videos or podcasts

[closed]

Asked 16 years ago Modified 12 years ago Viewed 6k times



41



As it currently stands, this question is not a good fit for our Q&A format. We expect answers to be supported by facts, references, or expertise, but this question will likely solicit debate, arguments, polling, or extended discussion. If you feel that this question can be improved and possibly reopened, [visit the help center](#) for guidance.

Closed 13 years ago.

I am looking for podcasts or videos on how to do unit testing.

Ideally they should cover both the basics and more advanced topics.

unit-testing

Share

Improve this question

Follow

edited Dec 9, 2011 at 18:16



Joel Coehoorn

415k ● 114 ● 577 ● 813

asked Dec 22, 2008 at 20:51

Charles Faiga

11.7k ● 27 ● 104 ● 141

11 Answers

Sorted by:

Highest score (default)



24

At [Dnr TV](#) there are two episodes with JP Boodhoo, where he gives an introduction to test driven development:



- [Test Driven Development with JP Boodhoo, Part 1](#)
- [Test Driven Development with JP Boodhoo, Part 2](#)



If you want to see unit testing and TDD used together with a whole bunch of other agile practices, I would recommend watching the screencast series [Autumn of Agile](#). This series shows the development of a fully unit tested application from start to finish.

As for podcasts, check out the following:

- [Polymorphic Podcast - Jason Bock on Unit Testing, Code Coverage & Analysis](#)
- [Hanselminutes - Scott Hanselman on Test Driven Development](#)
- [Hanselminutes - The Last Word on TDD, with Scott Bellware](#)
- [Software Engineering Radio - Gerard Meszaros on XUnit Test Patterns](#)

Since mock objects are a quite important part of unit testing, these podcast episodes might be of interest as well:

- [Hanselminutes - Scott Hanselman on Mock Objects](#)
- [Dot Net Rocks - Eli Lopian on TypeMock.Net](#)
- [Dot Net Rocks - Oren Eini on RhinoMocks](#)
- [Dot Net Rocks - Oren Eini on RhinoMocks 3.5](#)

Share Improve this answer

edited Jan 19, 2009 at 14:28

Follow

answered Dec 22, 2008 at 20:57



Erik Öjebo

10.8k ● 4 ● 55 ● 76

See also JP on The Machine.Specifications BDD framework at vimeo.com/11642767 - always interesting to see 3 years of new learning. If you merge this in and @ reply to me, I'll remove this comment – [Ruben Bartelink](#) May 13, 2010 at 13:08 ✎



9



My collected bookmark-dump

- The [Clean Code Talks](#) has some of [Miško Hevery's](#) lectures at Google. New episodes are posted at the [Google Testing Blog](#) as they arrive.
- Some of the [GTAC 2008](#) (Google Test Automation Conference 2008) videos are pretty interesting.



- [Automated Testing Patterns and Smells](#)
- [Becoming a Software Testing Expert](#) is not really about unit testing, but helps you get into the tester mindset
- [How To Design A Good API And Why It Matters](#) briefly touches on testability.
- Browsing the [testing](#) tag on se-radio reveals some interesting podcasts
- Check out the videos from the Test track of Öredev 2007 ([1](#), [2](#)), I especially liked the 'testable architecture' one.
- The [Testing](#) sub-site of InfoQ is also worth a visit for more (text, audio, video) resources

Share Improve this answer

edited Dec 23, 2008 at 10:56

Follow

answered Dec 23, 2008 at 9:42



Henrik Gustafsson

53.9k ● 9 ● 50 ● 60



6



Other hanselminutes episodes on testing:

- [#112 The Past, Present and Future of .NET Unit Testing Frameworks](#) - listened a while back, remember being slightly underwhelmed, but still worth a listen





- [#103 Quetzal Bradley on Testing after Unit Tests](#) - extremely interesting, giving deep insight into how to think about the purpose of coverage metrics etc.
- [#146 Test Driven Development is Design - The Last Word on TDD \(with Scott Bellware\)](#) - lives up to its name in that it slams home a lot of core concepts that you "always knew anyway". IIRC the podcast gives a favourable mention to the Newkirk/Vorontsov book - which I wouldn't particularly second (it's a long time since I read it -- I might just not have been ready to absorb its messages)

Other podcasts:

- [Herding Code Episode 42: Scott Bellware on BDD and Lean Development](#) - recorded after Hanselminutes #146. Again, very good discussion that helps to cement ideas around "classic tests" vs BDD vs Context Specification and various such other attempts at classifications...
- [J.B. Rainsberger: "Integration Tests Are A Scam"](#) is a recording of a presentation covering integration vs unit tests.

Other questions like this:

- [Good QA / Testing Podcast](#), which among others lists the meta podcast <http://testingpodcast.com/>

Blog posts:

- [It's Not TDD, It's Design By Example](#) - Brad Wilson, similar in vein to HC #42, attempting to get across why you're writing the tests.

I know you didn't ask for books but... Can I also mention that Beck's TDD book is a must read, even though it may seem like a dated beginner book on first flick through (and [Working Effectively with Legacy Code](#) by Michael C. Feathers of course is the bible). Also, I'd append Martin(& Martin)'s Agile Principles, Patterns & Techniques as really helping in this regard. In this space (concise/distilled info on testing) also is the excellent [Foundations of programming ebook](#). Goob books on testing I've read are [The Art of Unit Testing](#) and [xUnit Test Patterns](#). The latter is an important antidote to the first as it is much more measured than Roy's book is very opinionated and offers a lot of unqualified 'facts' without properly going through the various options. Definitely recommend reading both books though. AOUT is very readable and gets you thinking, though it chooses specific [debatable] technologies; xUTP is in depth and neutral and really helps solidify your understanding. I read [Pragmatic Unit Testing in C# with NUnit](#) afterwards. It's good and balanced though slightly dated (it mentions RhinoMocks as a sidebar and doesn't mention Moq) - even if nothing is actually incorrect. An updated version of it would be a hands-down recommendation.

More recently I've re-read the [Feathers](#) book, which is timeless to a degree and covers important ground.

However it's a more 'how, for 50 different wheres' in nature. It's definitely a must read though.

Most recently, I'm reading the excellent [Growing Object-Oriented Software, Guided by Tests by Steve Freeman and Nat Pryce](#). I can't recommend it highly enough - it really ties everything together from big to small in terms of where TDD fits, and various levels of testing within a software architecture. While I'm throwing the kitchen sink in, [Evans's DDD](#) book is important too in terms of seeing the value of building things incrementally with maniacal refactoring in order to end up in a better place.

Share Improve this answer
Follow

edited May 23, 2017 at 12:33



Community Bot
1 ● 1

answered Dec 23, 2008 at 9:27



Ruben Bartelink
61.8k ● 31 ● 194 ● 260



You could check out the [Unit Testing episodes](#) of Dime Casts.Net.

3



Share Improve this answer
Follow

answered Feb 13, 2009 at 13:19



RuudKok
5,302 ● 2 ● 28 ● 28





The following videos are by the author of The Art Of Unit Testing by Roy Osherove:

3

Understanding Test Driven Development:



<http://osherove.com/videos/2009/8/25/understanding-test-driven-development.html>



Unit Testing Best Practices:



<http://osherove.com/videos/2009/8/25/unit-testing-best-practices.html>

Understanding Mock Objects

<http://osherove.com/videos/2009/8/25/tdd-understanding-mock-objects.html>

Share Improve this answer

edited Dec 5, 2012 at 0:39

Follow

answered Jun 13, 2011 at 15:04



User

65.7k ● 74 ● 194 ● 253

hi, all links are not working ;(– [manuelBetancurt](#) Dec 5, 2012 at 0:34

@MaKo: Just updated them – [User](#) Dec 5, 2012 at 0:40



2

[Google employee](#) and testing evangelist [Misko Hevery](#) is definitely an authority on unit testing. I'm surprised he hasn't been mentioned yet. He's one of the best lecturers



out there, unit testing or no. In fact, I didn't really give a hoot about unit testing until I watched one of his 'casts. He gives clear, professional, to-the-point lectures and 9 times out of ten has something interesting to say. In my book, he's up there with Yegge in terms of influential developers. Plus, he makes subtle nerd jokes as an added bonus too which I find hilarious.



Share Improve this answer

answered Dec 23, 2008 at 9:10

Follow



Andrew

1,217 ● 10 ● 17



2

[JP Boodhoo](#) is a great guy on unit testing and BDD. He made several videos on how to do unit testing on mvp patterns.



I agree with Andrew. [Misko Hevery](#) is quite thoughtful on testing. His project [testability-explorer](#) is an innovation to measure your code as an evidence for further refactoring, based on some existing metrics.



Share Improve this answer

edited Dec 23, 2008 at 9:23

Follow

answered Dec 23, 2008 at 9:18



Sun Liwen

1,222 ● 1 ● 15 ● 21



Here are 2 really good introductory talks on TDD,BDD and unit testing. The speakers are both authors on the

2

subject and both books are great.



[Kent Becks's talk developer testing](#)

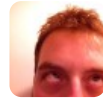
[David Astel's video on BDD](#)



Share Improve this answer

Follow

answered Jan 19, 2009 at 13:56



[Johnno Nolan](#)

29.6k ● 19 ● 113 ● 160



0

Unit Testing in Delphi 2005

This [BDNtv](#) episode shows the unit testing integration for DUnit for a Delphi/Win32 application in Delphi 2005. Unit testing for C#, Delphi for .NET, and Delphi for Win32 with both NUnit and DUnit are supported in Delphi 2005



Share Improve this answer

Follow

answered Dec 23, 2008 at 9:32



[Charles Faiga](#)

11.7k ● 27 ● 104 ● 141



0

Rob Conery covers a bit of this in his [Store Front](#) series. But probably not worth it unless your into the ASP.Net MVC stuff.



Share Improve this answer

Follow

answered Jan 19, 2009 at 14:00



[tpower](#)

56.8k ● 19 ● 70 ● 101





0



Here is a link to video by Hadi Hariri on how [Mocks make uniting testing easier](#)

Make Tests Easier with Mocks Mocking allows functionality to be stubbed out during tests allowing you to concentrate on the system under test and less on other subsystems. This session covers the idea behind mocks and shows some of the existing mocking frameworks available. See how to effectively use mocks and what the differences are between state and behavior tests.

Here is a link to video by Robert Love on how to [Building Unit Tests with DUnit](#) for delphi

Building Unit Tests with DUnit Get up to speed with unit testing in DUnit. No prior knowledge of Unit testing or DUnit is required.

Share Improve this answer

Follow

answered Oct 10, 2009 at 19:13



Charles Faiga

11.7k ● 27 ● 104 ● 141