

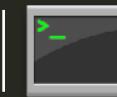
Testing 101

Software Systems – Programming – 2W2

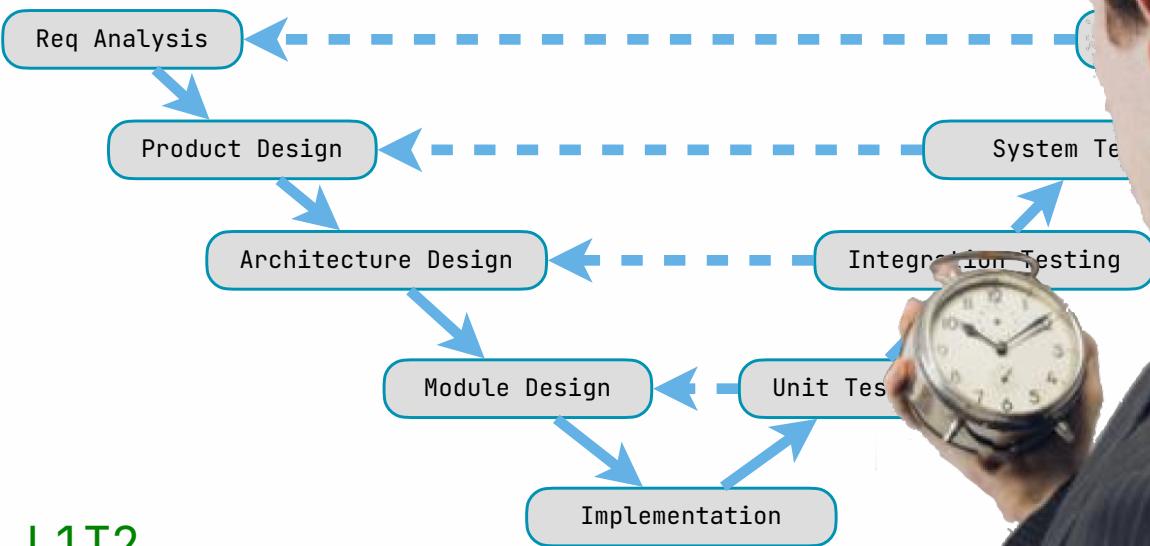
Dr. Vadim Zaytsev aka @grammarware, November/December 2020

UNIVERSITY
OF TWENTE.

Fm
Formal
Methods
& Tools



V-Model



L1T2

Unit Testing

- aka Component Testing
- Choose your **unit**
 - class, method, program, service, function, block...
- Encode **expectations**
 - invariants, preconditions, postconditions, ...
- Use a **framework**
 - @ParameterizedTest, @RepeatedTest...
- *Never go to sea with two chronometers*

Integration Testing

- Focus on **units** working **together**
- **Bottom-up**
 - best if communications are known
- **Top-down**
 - mock it till you rock it
- **Architecture-driven**
 - layers? services? client-server?
- Coverage is hard!

System Testing

- aka **Functional Testing** (verification, not validation)
- Software system **as a whole**
- e.g. for a **compiler**
 - run a compiled program, verify its output
- Popular techniques:
 - **stress testing**
 - **NFR testing**
 - **differential testing**

Acceptance Testing

- Prototypes
 - made to test ideas, not to evolve
- MVP
 - functionality, design, reliability, usability
- PoC
 - to demonstrate feasibility
- Pilot
 - controlled selective deployment
- Alpha/beta testing

White Box Techniques

- We *know* how the system **works**
- Anything **coverage-based**
- **Control flow** testing
- **Data flow** testing
- **Boundary value** analysis
- **Equivalence** partitioning
- **Grammar-based** test data generation

Black Box Techniques

- We *know* how the system is **supposed** to behave
- **Differential** testing
- **Stress** testing
- **Fuzz** testing
- **Random** testing
- **Scenario-based** testing
- **Model-based** testing

Deployment Tiers

- aka Deployment Environment/Server/Stage/...
- **local**
 - on your machine, while developing
- **dev** or **trunk**
 - in a sandbox
- **test**
 - full stack of tests
- **prod**
 - production / operation

Conclusion

- Granularity
 - unit \Rightarrow integration \Rightarrow system \Rightarrow acceptance
- Separation
 -  vs 
- Tiers
 - dev \Rightarrow test \Rightarrow prod
- All models are *wrong!*

Topics/slides Disclaimer

- Good ✓

- watch before Q&A
- embrace reality
- try out at labs
- ask for feedback
- apply to project
- dig deeper
- recall from slides

- Bad ✗

- slides over videos
- assumptions
- blanks
- timing

