

# BEST PRACTICES IN SW TESTING



**Petr NEUGEBAUER**  
QA EVANGELIST

PV260 - SOFTWARE QUALITY

# INTRODUCTION

**Education**

Brno Business School, Brno University of Technology (MBA '12, Strategic management)  
Faculty of Informatics, Masaryk Universityzita in Brno (MSc. '99, Informatics)

**Experience**

Y Soft Corporation (2008 – 2015) | Brno (CZ) – Printing solutions

Quality Manager | R&D Manager | PMO

Siemens (2001 – 2008) | Brno (CZ), Vienna (AT), Munich (GER) – Telecommunications, ITS

PM | Quality Manager | QA | SW developer

**Professional**

Czech and Slovak Testing Board (2007 – 2015)

ISTQB – International Software Qualification Testing Board (2011 – 2015)

[pro]TEST! MORAVA (2015)

# INFLUENCERS



Gojko ADZIC



James BACH



Janet  
GREGORY



Tom GILB



Mary  
POPPENDIECK

# ISTQB CZECH AND SLOVAK TESTING BOARD

## TESTING PROFESSIONALS

350.000+ ISTQB  
certified professionals

## COVERAGE

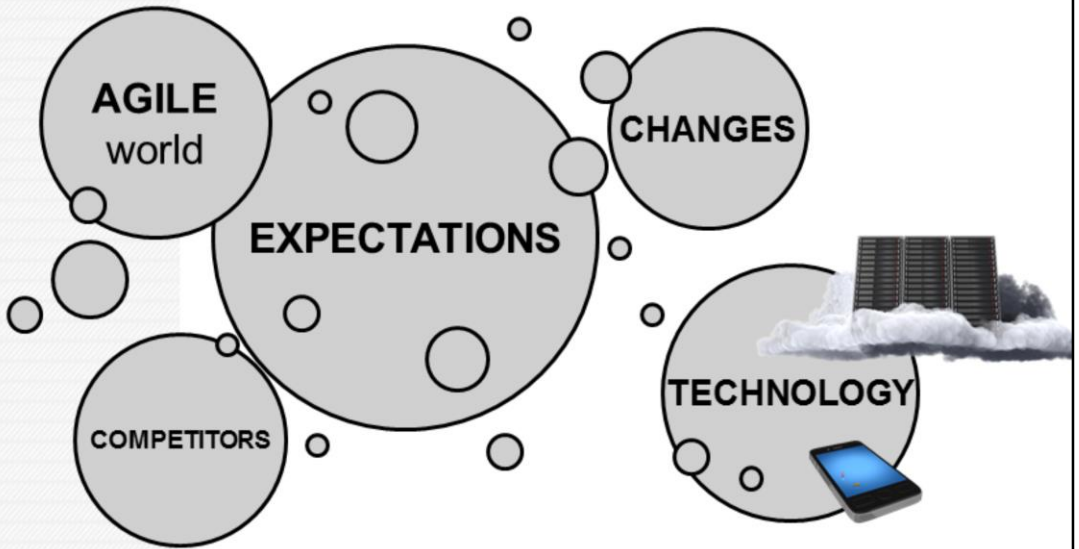
49 Member boards in  
72 countries

## International Software Qualification Testing Board



# CHALLENGES NOWADAYS

## DELIVER RESULTS



### EXPECTATIONS

- QUICK DELIVERY
- GOOD ENOUGH QUALITY

### TECHNOLOGY

- CLOUD SOLUTION – SERVICES
- MOBILE APPLICATIONS (NO WEB-based APPLICATIONS)

### COMPETITORS

- NEW BUSINESS OPPORTUNITIES
- INOVATIONS
- REFERENCES

# TESTING

- ... IS AN EXTREMELY EXPENSIVE ACTIVITY
- ... IS DOESN'T CONTRIBUTE TO BETTER QUALITY
- ... DIFFERS FROM QUALITY ASSURANCE
- ... UNREWARDED JOB

## STANDARDS AND FRAMEWORKS

- ISO/IEC 25010:2011 Software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE)
  - ISO/IEC 9126 (Standard describing typical risks)
- IEEE 829 – Standard for Software and System Test Documentation
- IEEE 1044 – Standard classification for Software Anomalies
- ISO 29119 – Software and systems engineering — Software testing
  
- ISTQB Framework



# DEVELOPMENT PROCESS EVOLUTION

## 60's: WATERFALL

- (+ Simple and easy to manage
- (+ Applicable for small SW
- (- Big design up front
- (- Defect detected at late phases
- (- High amounts of risk and uncertain

## 70's: V-MODEL

- (+ Early testing involvement
- (+ Clear relationship between test phases and development phases
- (- Still poses limitation of sequential model
- (- Require high amount of documentation
- (- Duplication of testing effort

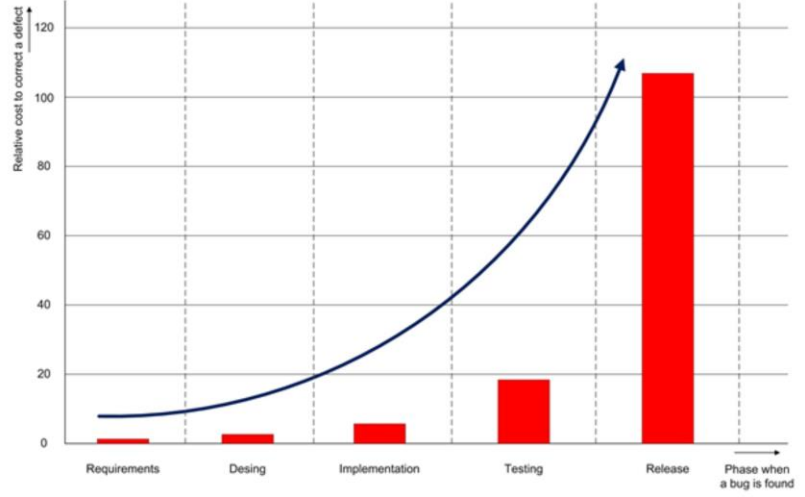
## 80's: RUP

- (+ Risk and uncertain are managed
- (+ Testing activities and process are managed
- (- Heavy documentation
- (- Late customer involvement – only UAT

## 00's: AGILE

- (+ Adaptable to changes
- (+ Early feedback
- (+ Avoid spending time on useless activities
- (- Require high-capable people
- (- Need representative from client
- (- Problem scaling up the architecture

# COSTS OF DEFECT FIXING



## MAIN OBJECTIVES

- ENSURE CUSTOMER NEEDS AND EXPECTATIONS
- ENSURE PROJECTS ARE DELIVERED ON TIME WITH HIGH QUALITY

Participates in all phases of the Product life cycle, suggests APPROVAL/REJECTION of the outputs of these phases in terms of quality.

Responsible analysis, design and measuring requirements, and managing necessary test cases to meet quality standards defined in the company.

Ensuring the highest quality by using manual functional testing, automated test suites, regression, endurance, performance and scale testing, while learning and applying testing best practices.

# MANAGING QUALITY

## TESTING VS QUALITY ASSURANCE

### Quality Control (Testing)

- Focus on finding bugs
- Does not guarantee quality

### Quality Assurance

- Focus on prevention

### Quality Analysis

# TESTER VS QA ENGINEER

## TESTER

- Executes manual tests
- Performs test scenarios review
- Uses test tool and simulators
- Analysis customer issues
- Provide summary test reports
- Participates in defect management

## QA ENGINEER

- Participates in technical analysis and review
- Interprets business requirements
- Designs and implements tests scenarios
- Focus on manual/automated tests
- Performs functional, regression, exploratory testing
- Cooperates with development team
- Focus on non-functional requirements
- Participates in Test Process Improvement

# AREAS OF EXPERTISE

## QUALITY CONTROL

- Tool support
- Test management process
- Functional testing
- Integration testing
- Regression testing

## RELEASE MANAGEMENT

- Planning
- Monitoring
- Verification

## QUALITY ANALYSIS

- Business analysis
- Formal review
- High level analysis
- Risks
- Non-functional REQs

## TEST ENVIRONMENT

- Configuration Management
- Virtualization
- Performance

## TEST PROCESS IMPROVEMENT

- Test Automation
- Standardization
- Professional development
- Academia cooperation

## INTERNAL SUPPORT

- Onboarding / trainings
- Knowledge sharing
- Remote support
- Consultations
- Documentation

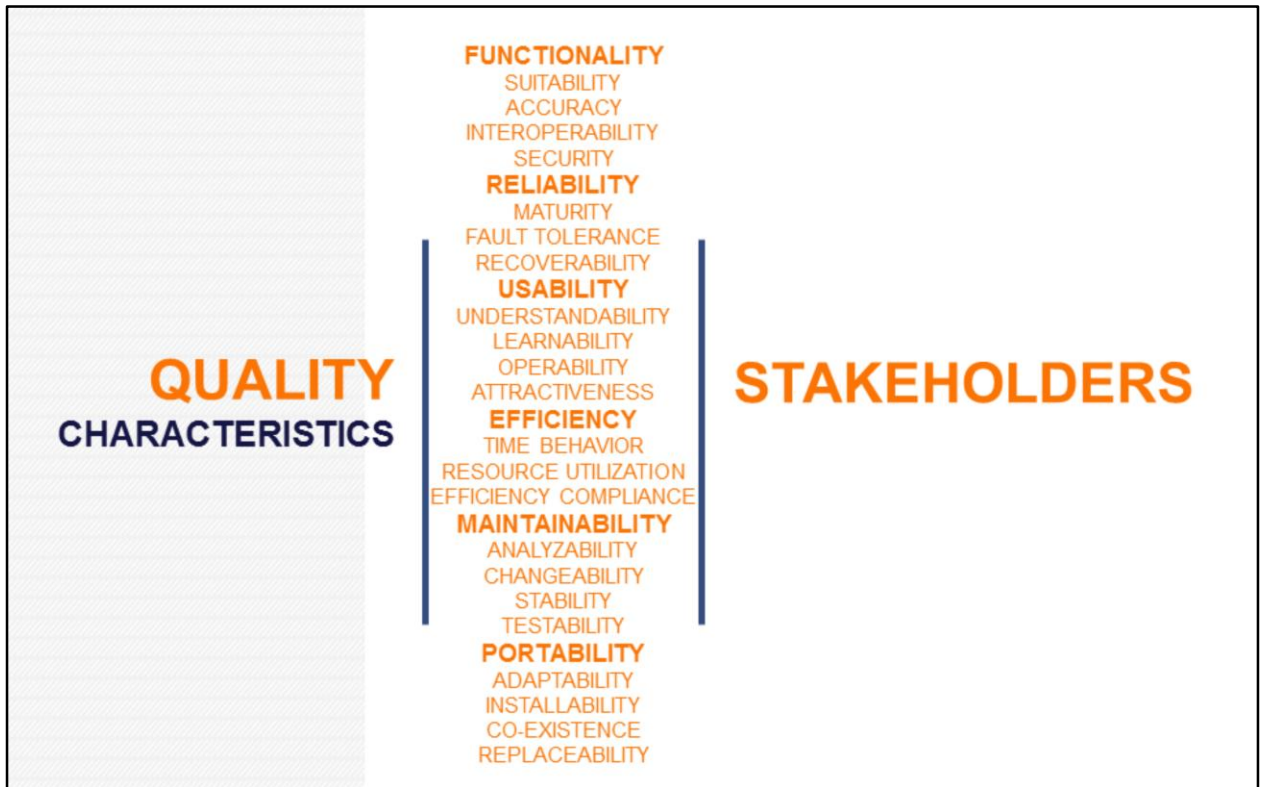
# WHAT IS QUALITY?

Quality is about meeting the needs and expectations of customers.

Stakeholder is any person, group or object, which has some direct or indirect interest in a system.







ISO 9126 QUALITY CHARACTERISTICS



# QUALITY IS ...

... THE LEVEL OF CONFORMANCE OF THE FINAL DELIVERABLE(S) TO THE **REQUIREMENTS**.

REQUIREMENTS ARE DEFINED BY ALL **STAKEHOLDERS!**

Quality is about meeting the needs and expectations of customers.

Stakeholder is any person, group or object, which has some direct or indirect interest in a system.

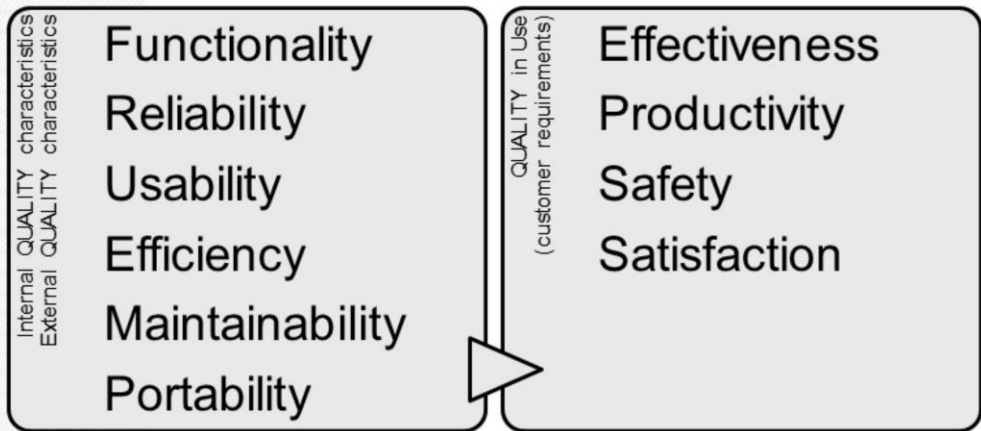
## REQUIREMENTS

No stakeholder → No Requirements

No Requirements → Nothing to do

No Requirements → Nothing to test

## ISO/IEC 25000:2005



ISO/IEC 25000:2005: Software engineering — Software product Quality Requirements and Evaluation (SQuaRE)

The goal is to harmonize two standards - old ISO/IEC 9126 and the ISO/IEC 14598.

align software quality requirements specification and software quality evaluation, supported by a software quality measurement process

Internal Quality | External Quality | Quality in use - Taken together, these requirements address the needs of relevant stakeholders, including those pertinent to various product life-cycle phases.

AREA -> AREA (influences)

AREA <- AREA (depends on)

**Functionality**

Suitability  
Accuracy  
Interoperability  
Security  
Compliance

**Reliability**

Maturity  
Fault tolerance  
Recoverability  
Compliance

**Usability**

Understandability  
Learnability  
Operability  
Attractiveness  
Compliance

**Efficiency**

Time behavior  
Resource utilization  
Compliance

**Maintainability**

Analyzability  
Changeability  
Stability  
Testability  
Compliance

**Portability**

Adaptability  
Instability  
Co-existence  
Replaceability  
Compliance

## MANAGING EXPECTATIONS

MANAGE EXPECTATIONS

**NEEDS vs REQUIREMENTS**

DESIGN MUST MEET THE BUSINESS NEEDS

**No unintentional design in the requirements**

CUSTOMER vs STAKEHOLDER

**Identify stakeholders**

QUALITY

**Expectations of ALL stakeholders**

# **AUTOMATION**

... WHY / WHEN / WHAT

# TEST AUTOMATION

## INTRODUCTION

### Why:

- Reduce amount of manual testing activities (motivation)
- Early feedback
- Sanity tests

### Limitations:

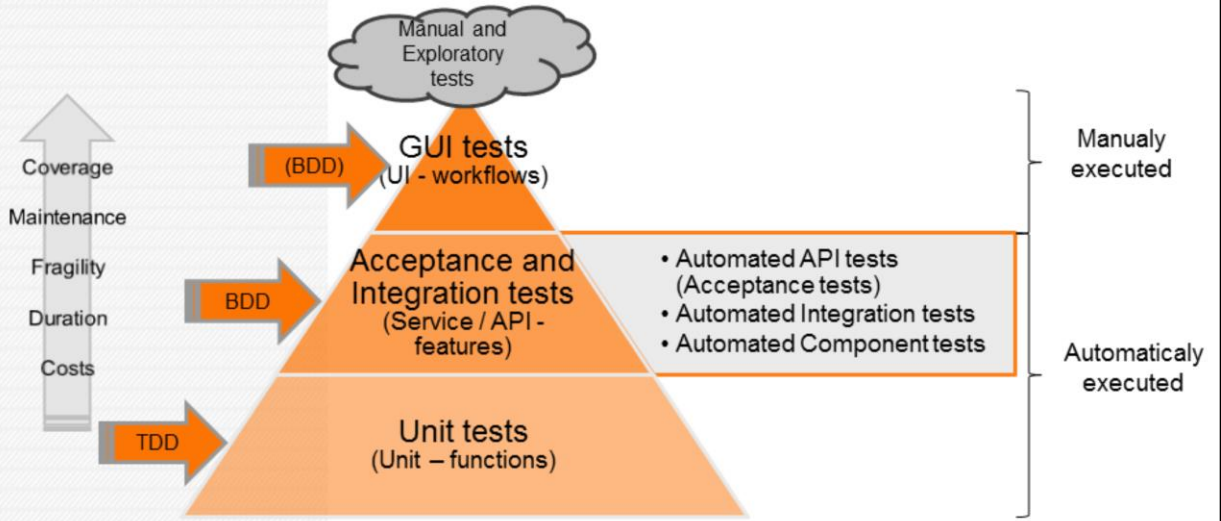
- Automation does not detect bugs

### Agile approach:

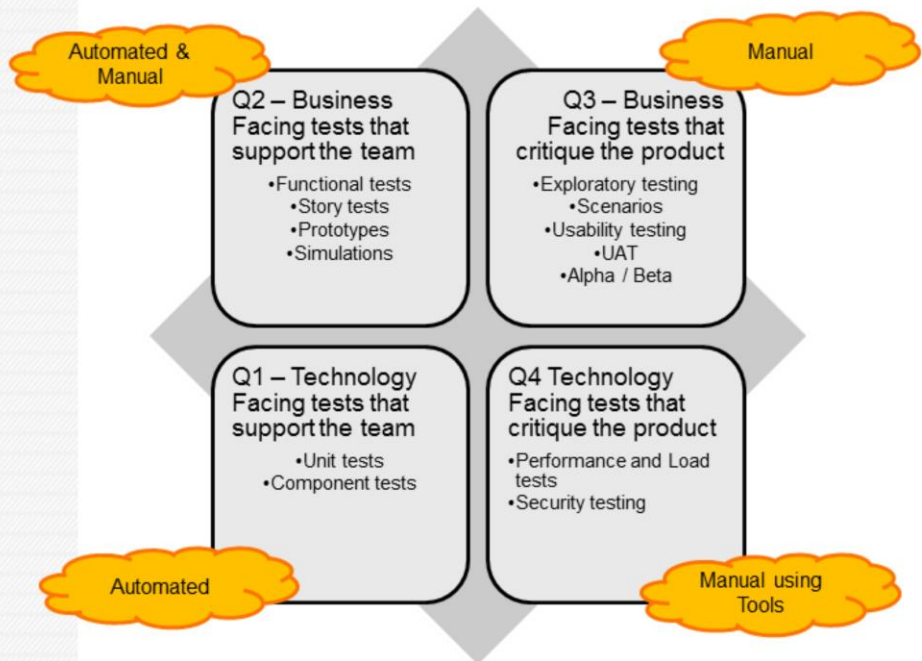
- Test Driven Development (TDD)
- Behavioral Driven Development (BDD)
- Acceptance Test Driven Development (ATTD)



# TEST AUTOMATION PYRAMID



# TESTING QUADRANTS



- Brian Marick introduced the quadrants
- Lisa Crispin and Janet Gregory gave Agile Quadrants the wings
- With shorter iterations and continuous delivery, it's difficult to draw the line between activities that support the team and those that critique the product. Why would performance tests not be aimed at supporting the team? Why are functional tests not critiquing the product? Why would exploratory tests be only for business stuff? Why is UAT separate from functional testing?
- The Quadrants don't fit well with all the huge changes that happened in the last five years, including the surge in popularity of continuous delivery, devops, build-measure-learn, big-data analytics obsession of product managers, exploratory and context driven testing.

## BEHAVIOR DRIVEN DEVELOPMENT

### BDD

#### Scenario X: Account is in credit+

**Given** *the account is in credit*

*And the card is valid*

*And the dispenser contains cash*

**When** *the customer requests cash*

**Then** *check that the account is debited*

*And ensure cash is dispensed*

*And check that the card is returned.*

# BEHAVIOR DRIVEN DEVELOPMENT

## BDD

**Scenario X: Account is in credit+**

**Given** *the account is in credit*

*And the card is valid*

*And the dispenser contains cash*

**When** *the customer requests cash*

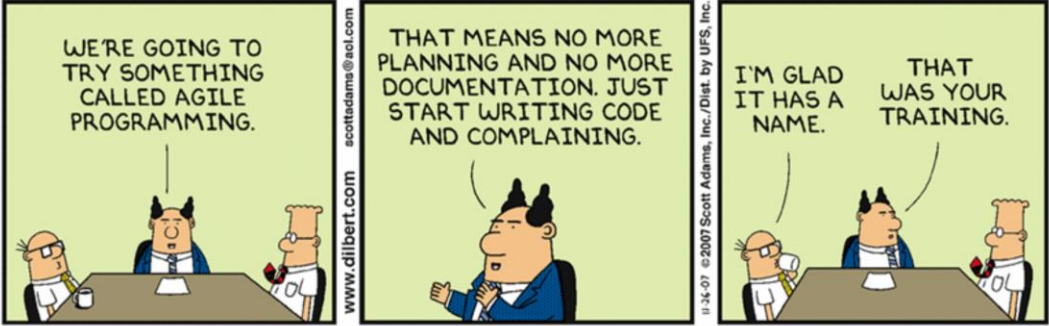
**Then** *check that the account is debited*

*And check that cash is dispensed*

*And check that the card is returned*

*And check that nothing happens that shouldn't happen and everything else happens that should happen for all variations of this scenario and all possible states of the ATM and all possible states of the customer's account and all possible states of the rest of the database and all possible states of the system as a whole, and anything happening in the cloud that should not matter but might matter.*

# COFFEE BREAK



# **BUILDING QA TEAM**

## MOTIVATION

People are the most important in an organization  
People are not predictable



# MOTIVATION

Motivation

- From the Latin word 'movere' – move to action.

Internal factors (motive) vs external factor (stimulus)

- 3 dimension
- Direction (choice) | Intensity (effort) | Persistence (duration)

**Stimulus** – easier to be introduced

**Motives** – stronger and far more effective



## **MOTIVATION – THE MANAGERIAL POINT OF VIEW**

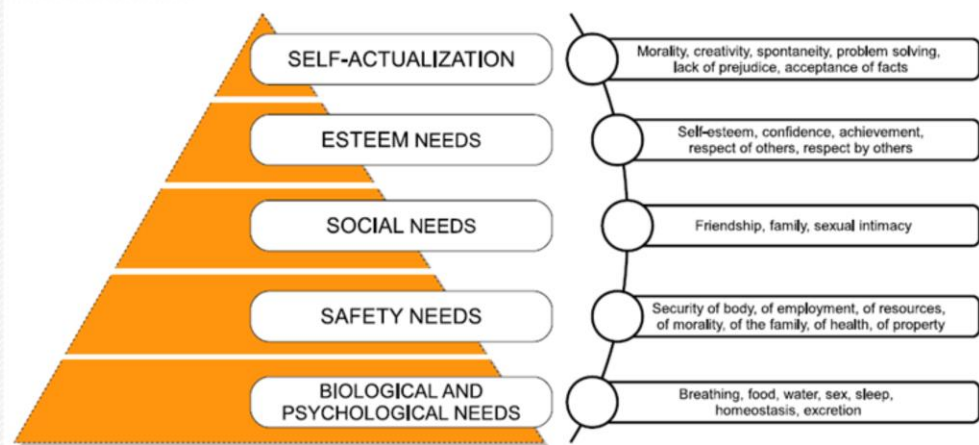
### **Intrinsic motivation**

- responsibility, status, recognition, personal and professional development, opportunities, and other similar factors

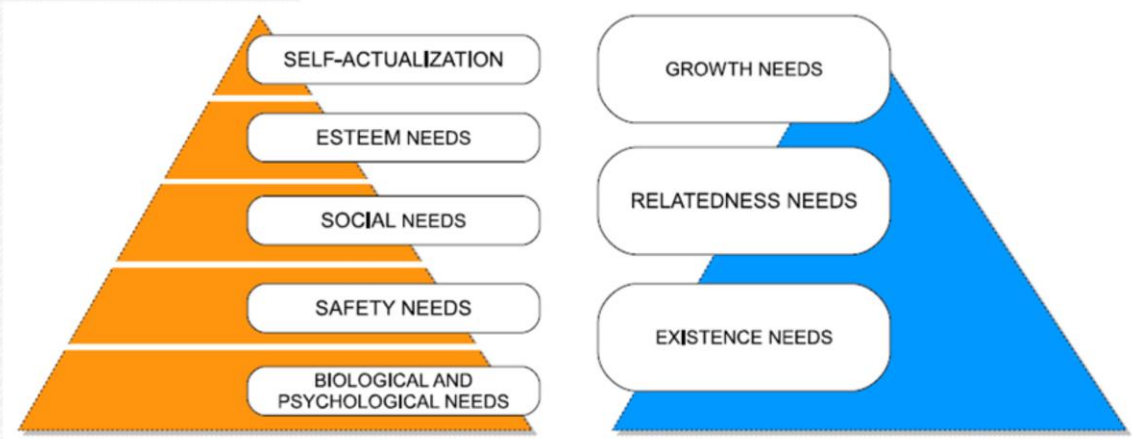
### **Extrinsic motivation**

- salary, wages, benefits and bonuses, work condition, fringe, security, promotion, contract of service, the work environment and conditions of work

# MASLOW'S HIERARCHY OF NEEDS

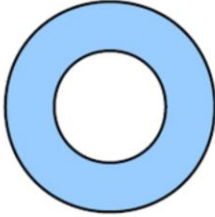


# ALDERFER'S ERG THEORY

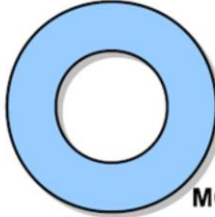


# HERZBERG'S TWO FACTORS THEORY

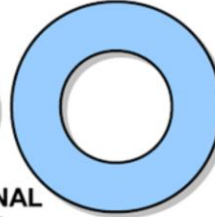
DISSATISFACTION and  
DEMOTIVATION



NOT DISSATISFACTION  
BUT NOT MOTIVATION



POSSITIVE SATISFACTION  
AND MOTIVATION



HYGIENE  
FACTORS

MOTIVATIONAL  
FACTORS

SATISFACTION <-> NO SATISFACTION  
DISSATISFACTION <-> NO DISSATISFACTION

## MANAGEMENT 3.0 – 10 INTRINSIC DESIRES

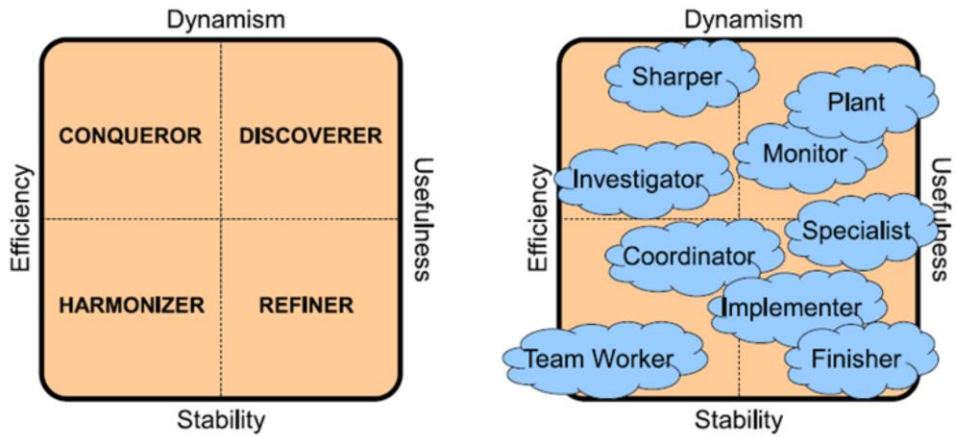
Curiosity	The need to think
Honor	Being loyal to a group
Acceptance	The need for approval
Mastery / Competence	The need to feel capable
Power	The need for influence of will
Freedom / Independence / Autonomy	Being an individual
Relatedness / Social Contact	The need for friends
Order	Or Stable environments
Goal / Idealism / Purpose	The need for purpose
Status	The need for social standing

## **MOTIVATION SUMMARY**

What motivates one demotivates others

Motivating people is NOT the same as NOT demotivating people

# TEAM ROLES (PERSONALITY TYPOLOGY)



Comparison of Jiri Plaminek's typology and Belbin team roles



# **IMPROVING PROCESSES**

CONTINUOUS IMPROVEMENT



## PROCESS IMPROVEMENT

**It means success**

Requires commitment from management

Involves monitoring and measurement

People do not like changes

(people like changes, they do not like uncertainty)

It is about processes, not people

# CMMI

## Level 1: Initial

## Level 2: Managed

Requirement Management, Project Planning, Process and Product QA, Configuration Management, ...

## Level 3: Defined

Requirements development, Validation and Verification, Organizational Processes, Risk Management, ...

## Level 4: Measured

Organizational Process Performance, Quantitative Project Management

## Level 5: Optimizing

Organizational Innovation and Deployment, Causal Analysis and Resolution

## TEST PROCESS IMPROVEMENT STANDARD MODELS

### Staged model

TMMi (based on CMMi)

### Continuous models

TPI Next (Test Maturity Matrix)

CTP (Critical Testing Processes)

Project Driven Improvement

STEP (Systematic Test and Evaluation Process)

Very agile

Staged model – provides roadmap for improvement – defines how to get to the next level

Continuous – driven by priorities – much more flexible

# TMMI MATURITY LEVELS

## Level 1: Initial

- Chaos
- Ad-hoc methods

## Level 2: Managed

- Test Policy and Strategy
- Test Planning
- Test Monitoring and Control
- Test Design and Execution
- Test Environment

## Level 3: Defined

- Test Organization
- Test Training Program
- Test Lifecycle and Integration
- Non-functional Testing
- Peer Reviews

## Level 4: Measured

- Test Measurement
- SW Quality Evaluation
- Advanced Peer Reviews

## Level 5: Optimized

- Defect Prevention
- Test Process Optimization
- Quality Control

## **TPI NEXT**

16 Key areas

4 Maturity levels

157 Checkpoints

13 Clusters

# AGILE ADOPTION

THE ULTIMATE TEST OF AGILITY IS WHETHER YOU CAN KEEP ALL YOUR STAKEHOLDERS HAPPY.

## WHOLE-TEAM APPROACH

Enhancing communication and collaboration within the team

Enabling the various skill sets within the team to be leveraged to the benefit of the project

Making quality everyone's responsibility

Early and Frequent Feedback

## **ROLE OF TESTERS** IN AN AGILE TEAM

Combination is the science

Reviews

Exploratory testing

Risk Based testing

Test Automation

Measuring quality

Team Role



## CHALLENGES

ADOPTION vs ADAPTION

CULTURE

Punishment vs Taking risks

MATURITY

Responsibility

INTERACTIONS

RESISTANCE TO CHANGE

MANAGEMENT

LEADERSHIP IS ACTION, NOT POSITION

"Boss" is a job; "Leader" is a career.

PEOPLE QUIT THEIR BOSS, NOT THEIR JOB

## QUOTES

### NEUGI

Testing is like sex. If it's not fun, then you're doing it wrong.

Tools don't control processes, but processes control tools.

Non-functional requirements define design of application, not functional requirements.

# TESTING MYTHODOLOGY

Myth 1: Testing is a boring job

FACT: Testing is NOT boring: It's been said that "Testing is like sex. If it's not fun, then you're doing it wrong."

Myth 2: Testing and debugging improves quality

FACT: Testing is a measure of quality. The number of defects you find indicates the quality of the product. "Testing to improve quality is like standing on a scale to lose weight".

Myth 3: Automated testing eliminates the need for manual testing

FACT: 100% test automation cannot be achieved. Manual Testing, to some level, is always necessary. Automation is a useful tool that should be taken into consideration, but it should not be the first thing to be considered when testing software. It is much useful while designing a method for testing, as the design outcome helps to decide whether automation is actually required or not. Moreover, Test Automation can never be used if requirements keep changing.

Myth 4: When a defect slips, it is the fault of the Testers

FACT: Quality is the responsibility of all members/stakeholders, including developers, of a project.

Myth 5: If the software is tested then it must be bug free

FACT: No one can say with absolute certainty that a software application is 100% bug free even if a tester with superb testing skills has tested the application

# THANKS!

Petr NEUGEBAUER

→ [www.linkedin.com/in/petrneugebauer](http://www.linkedin.com/in/petrneugebauer)



QA Evangelist | ISTQB® Agile Extension Certified Professional



PV260 - SOFTWARE QUALITY