

ID

Author:

Type: {Value, Principle, Process,  
Purpose, Outcome, Practice,  
Work Product, Process Attribute,  
Level

Article:

Content:

ID

Author:

ID

Author:

Type: {Value, Principle,  
Process, Purpose,  
Outcome, Practice, Work  
Product, Process Attribute,  
Level

Article:

Type: {Value, Principle,  
Process, Purpose,  
Outcome, Practice, Work  
Product, Process Attribute,  
Level

Article:

Content:

Content:

ID

Author:

ID

Author:

Type: {Value, Principle,  
Process, Purpose,  
Outcome, Practice, Work  
Product, Process Attribute,  
Level

Article:

Type: {Value, Principle,  
Process, Purpose,  
Outcome, Practice, Work  
Product, Process Attribute,  
Level

Article:

Content:

Content:

ID: 001

Author: Aiello

ID: 002

Author: Aiello

Type: Value, Level

Article: Agile Process  
Maturity - Introducing the  
Framework

Type: Value, Principle,  
Outcome Process Attribute,  
Level

Article: : Agile Process  
Maturity - Introducing the  
Framework

Content: Adherence to Agile Principles (Purity)

Content: Repeatable process across the  
organization

ID: 003

Author: Aiello

ID: 004

Author: Aiello

Type: Value, Principle,  
Outcome, Process  
Attribute, Level

Article: Agile Process  
Maturity - Introducing the  
Framework

Type: {Value, Principle,

Article: Agile Process  
Maturity - Introducing the  
Framework

Content: Scalability – SCRUM of SCRUMS

Content: Coexistence with non-Agile

ID: 005

Author: Aiello

ID: 006

Author: Aiello

Type: Value, Principle,

Article: Agile Process  
Maturity - Introducing the  
Framework

Type: Value, Principle,  
Outcome,

Article: : Agile Process  
Maturity - Introducing the  
Framework

Content: Harmonization with industry  
standards and frameworks

Content: Support for IT governance and  
compliance

ID: 007

Author: Aiello

ID: 008

Author: Aiello

Type: Principle,

Article: Agile Process  
Maturity - Introducing the  
Framework

Type: {Value, Principle,  
Process, Purpose,  
Outcome, Practice, Work  
Product, Process Attribute,  
Level

Article: Agile Process  
Maturity - Introducing the  
Framework

Content: ensure that you  
can deliver each and every time.

Content:

ID: 009

Author: Baskarada

ID: 010

Author: Baskarada

Type: Level

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Type: Level

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Content: Chaotic

Content: INITIAL

ID: 011

Author: Baskarada

ID: 012

Author: Baskarada

Type: Process

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Type: Process,

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Content: Asset register information management

Content: Asset hierarchy information management

ID: 013

Author: Baskarada

ID: 014

Author: Baskarada

Type: Process,

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Type: Process

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Content: Asset accounting information management

Content: Asset maintenance information management

ID: 015

Author: Baskarada

ID: 016

Author: Baskarada

Type: Process,

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Type: Process,

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Content: Contract management information management

Content: Resource management information management

ID: 017

Author: Baskarada

ID: 018

Author: Baskarada

Type: Process,

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Type: {Value, Principle, Process, Purpose, Outcome, Practice, Work Product, Process Attribute, Level

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Content: Inventory control information management

Content: Condition monitoring information management

ID: 019

Author: Baskarada

ID: 020

Author: Baskarada

Type: Process,

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Type: Level

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Content: Performance monitoring information management

Content: REACTIVE

ID: 021

Author: Baskarada

ID: 022

Author: Baskarada

Type: Process

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Type: Process,

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Content: Predictive modeling information management

Content: Risk management information management

ID: 023

Author: Baskarada

ID: 024

Author: Baskarada

Type: Process,

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Type: {Value, Principle, Process, Purpose, Outcome, Practice, Work Product, Process Attribute, Level

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Content: Asset lifecycle costing information management

Content: Optimizes decision making (ODM) information management

ID: 025

Author: Baskarada

Type: Level

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Content: MANAGED

ID: 026

Author: Baskarada

Type: Process,

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Content: Asset information profiling

ID: 027

Author: Baskarada

Type: Process,

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Content: Asset information matching

ID: 028

Author: Baskarada

Type: {Value, Principle, Process, Purpose, Outcome, Practice, Work Product, Process Attribute, Level

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Content: Asset information enhancement

ID: 029

Author: Baskarada

ID: 030

Author: Baskarada

Type: Process,

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Type: Process

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Content: Asset information monitoring

Content: Asset information consolidation

ID: 031

Author: Baskarada

ID: 032

Author: Baskarada

Type: Level

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Type: Process,

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Content: Optimising

Content: Asset Information Product Management (TDQM)

ID: 033

Author: Baskarada

ID: 034

Author: Baskarada

Type: {Value, Principle, Process, Purpose, Outcome, Practice, Work Product, Process Attribute, Level

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Type: {Value, Principle, Process, Purpose, Outcome, Practice, Work Product, Process Attribute, Level

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Content:

Content:

ID:035

Author: Baskarada

ID: 036

Author: Baskarada

Type: {Value, Principle, Process, Purpose, Outcome, Practice, Work Product, Process Attribute, Level

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Type: {Value, Principle, Process, Purpose, Outcome, Practice, Work Product, Process Attribute, Level

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Content:

Content:

ID: 037

Author: Baskarada

ID: 038

Author: Baskarada

Type: {Value, Principle, Process, Purpose, Outcome, Practice, Work Product, Process Attribute, Level

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Type: {Value, Principle, Process, Purpose, Outcome, Practice, Work Product, Process Attribute, Level

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Content:

Content:

ID:039

Author: Baskarada

ID: 040

Author: Baskarada

Type: {Value, Principle, Process, Purpose, Outcome, Practice, Work Product, Process Attribute, Level

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Type: {Value, Principle, Process, Purpose, Outcome, Practice, Work Product, Process Attribute, Level

Article: Agile Maturity Model Approach to Assessing and Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems

Content:

Content:

ID: 041

Author: Benefield

ID: 042

Author: Benefield

Type: Outcome,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Type: Process, Practice,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content: cycle time improved

Content: Automated Regression Testing

ID:043

Author: Benefield

ID: 044

Author: Benefield

Type: Process, Practice,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Type: Practice,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content: establishing code standards

Content: building a culture of design and code reviews

ID: 045

Author: Benefield

ID: 046

Author: Benefield

Type: Outcome, Work Product,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Type:, Outcome,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content: Code Quality Metrics

Content: Automated Deployment and Backout

ID:047

Author: Benefield

ID: 048

Author: Benefield

Type, Practice,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Type, Principle, Outcome,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content: ensure uniformity of packaging and deployment

Content: consistent behavior between development, test and production environments

ID: 049

Author: Benefield

ID: 050

Author: Benefield

Type: Principle, Outcome,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Type: Principle, Outcome, Work Product,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content: Automated Builds

Content: effective configuration management strategy, including a standard

ID:051

Author: Benefield

ID: 052

Author: Benefield

Type: Principle, Outcome, Work Product,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Type: Principle,, Outcome,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content: known source code version control system

Content: effective code management

ID: 053

Author: Benefield

ID: 054

Author: Benefield

Type: Principle, Practice, Process

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Type: Outcome

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content: Interlocked Delivery and Interface Integration testing

Content: cross component delivery transparency

ID:055

Author: Benefield

ID: 056

Author: Benefield

Type: Principle, Process, Practice,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Type: Principle, Outcome,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content: Test Driven Development (TDD)

Content: code is tested effectively to ensure it behaves as the customer would expect it to behave

ID: 057

Author: Benefield

ID: 058

Author: Benefield

Type: Principle, Process, Practice,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Type: Process Attribute, Level

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content: Performance and Scalability Testing

Content: Level 1: Emergent Engineering Best Practices

ID:059

Author: Benefield

ID: 060

Author: Benefield

Type: {Value, Principle, Process, Purpose, Outcome, Practice, Work Product, Process Attribute, Level

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Type: {Value, Principle, Process, Purpose, Outcome, Practice, Work Product, Process Attribute, Level

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content:

Content:

ID: 061

Author: Benefield

ID: 062

Author: Benefield

Type:, Outcome,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Type: Outcome,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content: Concepts such as unit testing, code reviews, repeatable builds, configuration management, code quality, and test driven development are practiced by the team

Content: The value of concepts as unit testing, code reviews, repeatable builds, configuration management, code quality, and test driven development is understood by the team

ID:063

Author: Benefield

ID: 064

Author: Benefield

Type: Practice,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Type: Principle, Practice,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content: build special interest groups

Content: share knowledge of more mature and knowledgeable teams

ID: 065

Author: Benefield

ID: 066

Author: Benefield

Type: Outcome,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Type: Principle, Outcome, Practice,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content: Interface testing was industrialized

Content: data are collected and flow between the source tools and dashboards and story/requirements management tools

ID:067

Author: Benefield

ID: 068

Author: Benefield

Type: Benefit

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Type:, Process Attribute, Level

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content: results became much more tangible and real to senior management and the business

Content: Level 2: Continuous Practices at Component Level

ID: 069

Author: Benefield

Type: Process Attribute,  
Level

Article: Seven Dimensions  
of Agile Maturity in the  
Global Enterprise: A Case  
Study

ID: 070

Author: Benefield

Type: Principle,, Practice,

Article: Seven Dimensions  
of Agile Maturity in the  
Global Enterprise: A Case  
Study

Content: Level 3: Cross Component  
Continuous Integration

Content: test and strengthen the  
robustness between components within a flow  
through regular synchronized build/test cycles  
across interface boundaries

ID:071

Author: Benefield

Type: Principle, Outcome,

Article: Seven Dimensions  
of Agile Maturity in the  
Global Enterprise: A Case  
Study

ID: 072

Author: Benefield

Type: Principle, Outcome,

Article: Seven Dimensions  
of Agile Maturity in the  
Global Enterprise: A Case  
Study

Content: Builds, installation and regression  
tests must be heavily automated

Content: mechanistic alignment between  
teams

ID: 073

Author: Benefield

ID: 074

Author: Benefield

Type: Outcome,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Type: Benefit

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content: discovered bottlenecks

Content: allowing for potentially more smaller releases containing available feature functionality to be potentially released and used sooner

ID:075

Author: Benefield

ID: 076

Author: Benefield

Type: Benefit

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Type Process Attribute, Level

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content: the ability of the business to respond to the market

Content: Level 4: Cross Journey Continuous Integration

ID: 077

Author: Benefield

ID: 078

Author: Benefield

Type: Principle, Outcome, Practice,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Type: Outcome,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content: engineering practices of component teams are going to increasingly mirror those exhibited within XP teams

Content: refactoring becomes routine

ID:079

Author: Benefield

ID: 080

Author: Benefield

Type: Outcome

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Type: Outcome,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content: sophisticated injection test harnesses and instrumentation begin to proliferate

Content: performance and scalability extends from component to product and journey

ID: 081

Author: Benefield

Type: Outcome,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content: Integration points between product lines, parallel release streams, and partner products become more industrialized, with closer and more organized coordination between product line teams

ID:083

Author: Benefield

Type: Principle, Outcome,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study:

Content: Teams are able to quickly understand and communicate dependencies and coupling between components and products

ID: 082

Author: Benefield

Type: Process Attribute, Level

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content: Level 5: On Demand Just in Time Releases

ID: 084

Author: Benefield

Type: Principle, Outcome,

Article:Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content: Code refactoring has become so routine as teams consistently and demonstrably leave code touched in a better state than they found it to improve quality, supportability and reuse

ID: 085

Author: Benefield

ID: 086

Author: Benefield

Type: Outcome,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Type: Outcome,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content: A SOA-based model matures within the organization, with effective risk assessment built into the planning and daily build cycle

Content: Allow for decoupled releases delivered on demand.

ID:087

Author: Benefield

ID: 088

Author: Benefield

Type: Benefit

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study:

Type: Outcome,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content: New products and offerings can be quickly assembled and trialed on the fly

Content: Ability to understand the implications of removing or replacing components or products

ID: 089

Author: Benefield

ID: 090

Author: Benefield

Type: {Value, Principle, Outcome, Benefit

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Type: Principle,

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content: cost effective and timely response to technical or business challenges

Content: Assessments

ID:091

Author: Benefield

ID: 092

Author: Benefield

Type: Principle, Outcome, Benefit

Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study:

Type: Principle, Outcome, Practice,

Article:Article: Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study

Content: opportunity to model and tune a service on the fly

Content: Initial knowledge sessions, reading material and a road show were set up to help bolster team understanding of the dimensions and maturity levels

ID: 093

Author: Benefield

ID: 094

Author: Benefield

Type: Principle,, Outcome,  
Practice,

Article: Seven Dimensions  
of Agile Maturity in the  
Global Enterprise: A Case  
Study

Type: Practice,

Article: Seven Dimensions  
of Agile Maturity in the  
Global Enterprise: A Case  
Study

Content: Training programs, intense deep dive sessions and hothouses were set up with teams when deeper problems were encountered.

Content: identify potential trouble areas or opportunities

ID:095

Author: Benefield

ID: 096

Author: Benefield

Type: Principle, Practice,

Article: Seven Dimensions  
of Agile Maturity in the  
Global Enterprise: A Case  
Study:

Type: Work Product,

Article:Article: Seven  
Dimensions of Agile  
Maturity in the Global  
Enterprise: A Case Study

Content: Stream reviews were held at the end of each release cycle between the central team and delivery stream lead

Content: Product flow map

ID: 097

Author: Patel

Type:, Process Attribute,  
Level

Article: Agile Maturity  
Model (AMM): A Software  
Process Improvement  
framework for Agile  
Software Development  
Practices

Content: Level 2: Explored

ID: 098

Author: Patel

Type: Benefit

Article: Agile Maturity  
Model (AMM): A Software  
Process Improvement  
framework for Agile  
Software Development  
Practices

Content: fewer problems  
with their software development process than  
their level 1 counterparts.

ID:099

Author: Patel

Type: Outcome,

Article: Agile Maturity  
Model (AMM): A Software  
Process Improvement  
framework for Agile  
Software Development  
Practices

Content: Planned Projects

ID: 100

Author: Patel

Type: Outcome,

Article: Agile Maturity  
Model (AMM): A Software  
Process Improvement  
framework for Agile  
Software Development  
Practices

Content: Improved agile requirements  
engineering

ID: 101

Author: Patel

ID: 102

Author: Patel

Type: Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Type: Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Customer and stakeholders' orientation practices implemented

Content: Enhance Value, Collaboration and Planning Practices

ID:103

Author: Patel

ID: 104

Author: Patel

Type: Practice

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Type: Work Product,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Planning game

Content: project plans

ID: 105

Author: Patel

Type: Practice,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: release planning

ID: 106

Author: Patel

TypeWork Product,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: schedules

ID:107

Author: Patel

Type: Practice

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Estimation

ID: 108

Author: Patel

Type: Work Product,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Story cards

ID: 109

Author: Patel

Type: Principle, Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: story cards are written by the on-site customer

ID:111

Author: Patel

Type Principle, Practice,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: track project schedules, plan, requirements (Story cards), cost and functionality

ID: 110

Author: Patel

Type: Principle, Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: an organization has devoted resources to the planning and story cards (requirements engineering) practices as a whole.

ID: 112

Author: Patel

Type: Principle, Practice,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: identify and improve problems

ID: 113

Author: Patel

Type: Practice

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: assessment of current process

ID: 114

Author: Patel

Type: Practice

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Identify weaknesses

ID:115

Author: Patel

Type: Practice, Benefit

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Gain a general overview

ID: 116

Author: Patel

Type: Principle, Practice,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: address any planning or requirements issues associated with individual projects.

ID: 117

Author: Patel

TypeLevel

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Level 3: Defined Level

ID: 118

Author: Patel

Type: Principle, Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Customer satisfaction

ID:119

Author: Patel

Type: Outcome

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Communication improvement

ID: 120

Author: Patel

Type: Value, Principle, Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Software quality

ID: 121

Author: Patel

Type: Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Enhancement on coding practices and coding standards

ID: 122

Author: Patel

Type: Outcome, Benefit

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: The customer relationship is maintained very well

ID:123

Author: Patel

Type: Principle, Practice,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: test driven development

ID: 124

Author: Patel

Type: Principle,, Practice,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: pair programming

ID: 125

Author: Patel

Type: Principle,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Deliver software frequently

ID: 126

Author: Patel

Type: Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Control over the technical practices

ID:127

Author: Patel

Type: Value, Principle, Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Sustainable pace for development team and project management

ID: 128

Author: Patel

Type: Value, Principle, Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Self organising team

ID: 129

Author: Patel

Type: Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Empowered team and rewards

ID: 130

Author: Patel

Type: Outcome, Practice, Process

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Project management

ID:131

Author: Patel

Type: Outcome, Practice,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Risk assessment

ID: 132

Author: Patel

Type: Principle, Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: No over time

ID: 133

Author: Patel

Type: Value, Principle,  
Outcome,

Article: Agile Maturity  
Model (AMM): A Software  
Process Improvement  
framework for Agile  
Software Development  
Practices

Content: Simplicity

ID: 134

Author: Patel

Type: Value, Principle,  
Outcome,

Article: Agile Maturity  
Model (AMM): A Software  
Process Improvement  
framework for Agile  
Software Development  
Practices

Content: respect  
for the co-workers or people involved in the  
project

ID:135

Author: Patel

Type: Principle, Outcome,  
Practice,

Article: Agile Maturity  
Model (AMM): A Software  
Process Improvement  
framework for Agile  
Software Development  
Practices

Content: Context improvement

ID: 136

Author: Patel

Type: Process, Practice,

Article: Agile Maturity  
Model (AMM): A Software  
Process Improvement  
framework for Agile  
Software Development  
Practices

Content: Uncertainty management

ID: 137

Author: Patel

Type: Process, Practice,  
Process Attribute,

Article: Agile Maturity  
Model (AMM): A Software  
Process Improvement  
framework for Agile  
Software Development  
Practices

Content: Tuning project performance

ID: 138

Author: Patel

Type: Principle, Purpose,

Article: Agile Maturity  
Model (AMM): A Software  
Process Improvement  
framework for Agile  
Software Development  
Practices

Content: Defect Prevention

ID:139

Author: Patel

Type: Process, Practice,

Article: Agile Maturity  
Model (AMM): A Software  
Process Improvement  
framework for Agile  
Software Development  
Practices

Content: Adaptability and suitability  
assessment

ID: 140

Author: Patel

Type: Purpose,

Article: Agile Maturity  
Model (AMM): A Software  
Process Improvement  
framework for Agile  
Software Development  
Practices

Content: ensure or to identify the organization  
is an agile software development  
organization or not.

ID: 141

Author: Patel

Type: Value, Principle,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: be an agile software development organization

ID:143

Author: Patel

Type: Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: The plan for the improvement is identified

ID: 142

Author: Patel

Type: Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: business objectives or business goal are defined by the agile team

ID: 144

Author: Patel

Type: Practice,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: capture the best practices

ID: 145

Author: Patel

Type: Work Product,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Adaptability framework

ID: 146

Author: Patel

Type: Work Product,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: working patterns

ID:147

Author: Patel

Type: Work Product,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: adaptability questionnaire

ID: 148

Author: Patel

Type: Process,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: adaptability framework process

ID: 149

Author: Patel

Type: Practice,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Map the identified areas of improvement to the knowledge based best agile practices

ID: 150

Author: Patel

Type: Work Product,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Knowledge based Best Agile Practices Store

ID:151

Author: Patel

Type: Practice,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Select/Define Knowledge Based Best Agile Practices

ID: 152

Author: Patel

Type: Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Identified Areas of Improvement from Adaptability Assessment

ID: 153

Author: Patel

Type: Process Attribute,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: 3.1 Customer Relationship Management

ID: 154

Author: Patel

Type: Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: System metaphor is defined, which allows the customer representative to understand the system (Choose system metaphor)

ID:155

Author: Patel

Type: Principle, Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Customer and business representative present or at least invited to all team estimation sessions

ID: 156

Author: Patel

Type:, Principle, Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Refactoring is encouraged via a little-and-often approach and larger refactoring reprioritised with the customer

ID: 157

Author: Patel

Type: Principle,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Make small frequent release

ID: 158

Author: Patel

Type: Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: User stories are written

ID:159

Author: Patel

Type: Work Product,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: User Story

ID: 160

Author: Patel

Type: Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Project plans are created to create project plan

ID: 161

Author: Patel

Type: Principle,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Effectively collaborate customer

ID: 162

Author: Patel

Type: Principle, Process Attribute,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: 3.2 Delivering Working Products/  
SW frequently

ID:163

Author: Patel

Type: Principle, Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Made small frequent releases which will create feedback loop

ID: 164

Author: Patel

Type: Principle,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Only one pair integrates code at a time

ID: 165

Author: Patel

Type: Principle,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Integrate often

ID: 166

Author: Patel

Type: Principle, Process Attribute

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: 3.3 Pair Programming

ID:167

Author: Patel

Type: Principle,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: move people around

ID: 168

Author: Patel

Type: Principle, Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: the customer pay frequent visit to the development team

ID: 169

Author: Patel

Type: Principle, Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: all production code is pair programmed

ID: 170

Author: Patel

Type: Principle,, Practice,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: use collective code ownership

ID:171

Author: Patel

Type: Principle, Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: overtime data are collected and published

ID: 172

Author: Patel

Type: Process Attribute

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: 3.4 Mutual Interaction

ID: 173

Author: Patel

Type: Principle, Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: All code is pair programmed

ID: 174

Author: Patel

Type: *Principle*, Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Story cards are written by the collaboration of on-site customer and developers

ID:175

Author: Patel

Type: Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Communicates result through acceptance testing

ID: 176

Author: Patel

Type: Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Refactoring is encouraged via a little-and-often approach and larger refactoring reprioritised with the customer

ID: 177

Author: Patel

Type: Principle, Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Team members have an open work environment that supports collaboration and conversation

ID: 178

Author: Patel

Type: Principle, Process Attribute

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: 3.5 Test Driven Development

ID:179

Author: Patel

Type: Principle,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Code the unit test first

ID: 180

Author: Patel

Type: Principle,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Refactor whenever and wherever possible

ID: 181

Author: Patel

Type: Principle,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: All code must have a unit tests

ID: 182

Author: Patel

Type: Principle,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: All code must pass the unit test and score must be published before it can be released

ID:183

Author: Patel

Type: Principle, Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: When a bug is found tests are created

ID: 184

Author: Patel

Type: Principle, Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: The best practices for automated testing are encouraged, rewarded and in place on the project

ID: 185

Author: Patel

Type: Principle, Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Perform Peer-reviews

ID: 186

Author: Patel

Type: Principle, Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Analyse results and identify corrective action

ID:187

Author: Patel

Type: Principle, Process Attribute

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: 3.6 Implementation and Interaction

ID: 188

Author: Patel

Type: Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: No functionality is added early

ID: 189

Author: Patel

Type: Principle, Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Automated testing is used to support frequent integration test

ID: 190

Author: Patel

Type: Principle,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: No up front design

ID:191

Author: Patel

Type: Principle, Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: team delivers useful content for business review every 1-4 weeks

ID: 192

Author: Patel

Type: Practice,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Business Review

ID: 193

Author: Patel

Type: Principle, Outcome,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: the list of user story is reprioritised based on an updated evaluation of the project at each iteration boundary

ID: 194

Author: Patel

Type: Outcome

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: the best practices for continuous integration are encouraged, rewarded and in place on the project

ID:195

Author: Patel

Type: Practice,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Prepare for product integration

ID: 196

Author: Patel

Type: Practice,

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Determine integration sequence

ID: 197

Author: Patel

Type: Work Product,  
Process Attribute

Article: Agile Maturity  
Model (AMM): A Software  
Process Improvement  
framework for Agile  
Software Development  
Practices

Content: 3.7 Coding Standards

ID: 198

Author: Patel

Type: Principle,

Article: Agile Maturity  
Model (AMM): A Software  
Process Improvement  
framework for Agile  
Software Development  
Practices

Content: Codes must be written to  
agreed standards

ID:199

Author: Patel

Type: Principle,

Article: Agile Maturity  
Model (AMM): A Software  
Process Improvement  
framework for Agile  
Software Development  
Practices

Content: Code the unit test first

ID: 200

Author: Patel

Type: {Value, Principle,  
Process, Purpose,  
Outcome, Practice, Work  
Product, Process Attribute,  
Level

Article: Agile Maturity  
Model (AMM): A Software  
Process Improvement  
framework for Agile  
Software Development  
Practices

Content:

ID: 201

Author: Esther Derby

ID: 202

Author: Esther Derby

Type: Work Product,

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Type: Practice,

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Content: product road map

Content: Gather information investigating  
customer support requests, field studies and  
interviews

ID:203

Author: Esther Derby

ID: 204

Author: Esther Derby

Type: Practice,

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Type: Practice,

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Content: Reveal how your customers currently  
use your product, which features need  
improvement, and give clues about unmet  
needs

Content: re-examine the factors that influence  
revenue

ID: 205

Author: Esther Derby

Type: Process,

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Content: Value Creation, the process of  
turning a product idea into something you can  
sell

ID:207

Author: Esther Derby

Type: Principle, Outcome,

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Content: attract and retain people who have  
the desire and ability

ID: 206

Author: Esther Derby

Type: Practice,

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Content: Look at missed Potential Profit, how  
much money and effort is dedicated to self-  
inflicted coordination overhead, rework,  
downtime, or support for difficult-to-use  
features

ID: 208

Author: Esther Derby

Type: Principle,

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Content: close collaboration with a product  
owner or customer

ID: 209

Author: Esther Derby

ID: 210

Author: Esther Derby

Type: {Value, Principle, Process, Purpose, Outcome, Practice, Work Product, Process Attribute, Level

Article: Achieving Agility: Means to an End, or End in Itself

Type: Principle, Outcome,

Article: Achieving Agility: Means to an End, or End in Itself

Content: teams understand the customer and his context, so they can make better decisions about feature design

Content: Short iterations and frequent demonstrations of working features keeps development close to customer needs and wants.

ID:211

Author: Esther Derby

ID: 212

Author: Esther Derby

Type: Practice,

Article: Achieving Agility: Means to an End, or End in Itself

Type: Practice,

Article: Achieving Agility: Means to an End, or End in Itself

Content: Iteration planning

Content: tracking story points

ID: 213

Author: Esther Derby

Type: Outcome,

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Content: teams and management understand  
capacity

ID: 214

Author: Esther Derby

Type: Principle, Practice,

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Content: prevent cost overruns

ID:215

Author: Esther Derby

Type: Principle, Practice,

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Content: make decisions to continue funding  
feature development—or not

ID: 216

Author: Esther Derby

Type: Principle, Practice

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Content: review progress and viability at the  
end of every iteration

ID: 217

Author: Esther Derby

ID: 218

Author: Esther Derby

Type: {Value, Principle, Process, Purpose, Outcome, Practice, Work Product, Process Attribute, Level

Article: Achieving Agility: Means to an End, or End in Itself

Type: Principle, Practice,

Article: Achieving Agility: Means to an End, or End in Itself

Content: produce completed slices of feature

Content: finish small chunks of features each iteration

ID:219

Author: Esther Derby

ID: 220

Author: Esther Derby

Type: Principle, Outcome,

Article: Achieving Agility: Means to an End, or End in Itself

Type: Practice,

Article: Achieving Agility: Means to an End, or End in Itself

Content: Every feature slice is tested and ready for customer acceptance at the end of the iteration

Content: automated unit and customer tests

ID: 221

Author: Esther Derby

Type: Principle, Practice,

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Content: frequent integration

ID: 222

Author: Esther Derby

Type: Principle, Practice,

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Content: find errors early

ID:223

Author: Esther Derby

Type: Outcome,

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Content: Reduced number of defects released  
to the customer

ID: 224

Author: Esther Derby

Type: Benefit

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Content: support costs, find and fix costs and  
bad press generated by buggy products

ID: 225

Author: Esther Derby

Type: Principle, Outcome,  
Practice,

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Content: simple design

ID:227

Author: Esther Derby

Type: Practice,

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Content: refactoring

ID: 226

Author: Esther Derby

Type: Practice,

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Content: Test Driven Development

ID: 228

Author: Esther Derby

Type: *Principle*, Practice,

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Content: build strong cross-functional teams

ID: 229

Author: Esther Derby

Type: Outcome, Benefit

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Content: satisfaction with work-life and  
work/life balance is higher

ID: 230

Author: Esther Derby

Type: Principle, Benefit

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Content: Working at a sustainable pace and  
re-establishing pride in work

ID:231

Author: Esther Derby

Type: Purpose, Outcome,

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Content: intrinsic motivation.

ID: 232

Author: Esther Derby

Type: Practice,

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Content: obtain feedback

ID: 233

Author: Esther Derby

ID: 234

Author: Esther Derby

Type: Practice,

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Type: Practice,

Article: Achieving Agility:  
Means to an End, or End in  
Itself

Content: investigate how cross-functional teams working in iteratively in time boxes can help

Content: examine your existing structures, policies and procedures

ID:235

Author: Colin Doyle

ID: 236

Author: Colin Doyle

Type: Principle,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Type: Principle,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: close (ideally continuous) involvement of the customer

Content: emphasis on empowered, self-organizing teams

ID: 237

Author: Colin Doyle

ID: 238

Author: Colin Doyle

Type: Principle, Outcome,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Type: Principle, Practice,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: Agile methodologies are geared to producing working software with minimal overhead.

Content: reduce waste (non-value adding activities),

ID:239

Author: Colin Doyle

ID: 240

Author: Colin Doyle

Type: Principle,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Type: {Value, Principle,  
Process, Purpose,  
Outcome, Practice, Work  
Product, Process Attribute,  
Level

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: continuously improving

Content: adapting to change

ID: 241

Author: Colin Doyle

ID: 242

Author: Colin Doyle

Type: Principle, Process, Practice,

Article: Agile Development Doesn't Have to Mean Fragile Enterprise Processes

Type: Principle, Outcome,

Article: Agile Development Doesn't Have to Mean Fragile Enterprise Processes

Content: replace a centralized "command and control" approach to project management with an emphasis on facilitation and mentoring

Content: people who will do the actual work perform the detailed estimation

ID:243

Author: Colin Doyle

ID: 244

Author: Colin Doyle

Type: Principle, Outcome,

Article: Agile Development Doesn't Have to Mean Fragile Enterprise Processes

Type: Principle,

Article: Agile Development Doesn't Have to Mean Fragile Enterprise Processes

Content: work done in an iteration is defined by what the team has committed to

Content: close, ongoing interaction with the customer

ID: 245

Author: Colin Doyle

Type: Enabler, Principle,  
Practice

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: use of an iterative and incremental  
development framework

ID: 246

Author: Colin Doyle

Type: Principle, Outcome,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: the schedule and team resources are  
fixed for the iteration

ID:247

Author: Colin Doyle

Type: Principle, Practice,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: Collect metrics and compare across  
iterations.

ID: 248

Author: Colin Doyle

Type: Principle, Practice,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: continually analyze and refactor  
the overall software product

ID: 249

Author: Colin Doyle

Type: Outcome, Benefit

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: the implementation remains clean  
and maintainable

ID:251

Author: Colin Doyle

Type: Principle,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: emphasis on maintaining cleanliness  
and quality of the software product

ID: 250

Author: Colin Doyle

Type: Outcome,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: Software design patterns are also  
used to maintain the quality of the code

ID: 252

Author: Colin Doyle

Type: Principle,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: "just enough" and "just in time"  
approach to planning and requirements  
analysis

ID: 253

Author: Colin Doyle

Type: Principle, Outcome,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: User stories are defined just enough that a rough order of magnitude scoping and risk analysis can be performed

ID:255

Author: Colin Doyle

Type: Work Product,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: product backlog

ID: 254

Author: Colin Doyle

Type: Practice,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: The customer or product owner (customer representative in Scrum terminology) prioritizes the set of user stories

ID: 256

Author: Colin Doyle

Type: Outcome, Practice,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: For a given project, referred to as a Release in Scrum, a subset of the product backlog is identified as candidates for that Release

ID: 257

Author: Colin Doyle

Type: Practice,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: daily feedback across the team

ID: 258

Author: Colin Doyle

Type: Outcome,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: opportunities to improve are  
identified

ID:259

Author: Colin Doyle

Type: Principle

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: to have "potentially shippable"  
software at the end of each release

ID: 260

Author: Colin Doyle

Type: Principle,, Practice,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: Continuous Integration, where every  
committed change triggers a build and test  
cycle to validate that change as soon as  
possible.

ID: 261

Author: Colin Doyle

Type: Principle, Process,  
Practice,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: At the end of the Release there is usually an additional review, referred to as a retrospective in Scrum, which is an overall post-mortem of the project

ID: 262

Author: Colin Doyle

Type: Outcome,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: The successes and problems are both identified, and action items raised to resolve problems or implement suggested improvements.

ID:263

Author: Colin Doyle

Type: {Principle, Process,  
Purpose, Outcome,  
Practice, Work Product,  
Process Attribute, Level

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: regular feedback between the customer or product owner and the development team is encouraged

ID: 264

Author: Colin Doyle

Type: Outcome,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: stable build of working software

ID: 265

Author: Colin Doyle

ID: 266

Author: Colin Doyle

Type: Principle, Enabler

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Type: Principle, Enabler,  
Benefit

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: teams of no more than ten people  
who are physically co-located

Content: maximize communication and  
minimize the overhead associated with  
planning and management

ID:267

Author: Colin Doyle

ID: 268

Author: Colin Doyle

Type: Principle, Practice,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Type: Outcome,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: pilot Agile methodologies

Content: the organization has proven the  
worth of the new approach and gained  
experience in its implementation

ID: 269

Author: Colin Doyle

Type: Principle, Practice,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: use flexible application lifecycle management (ALM) solutions which can support both traditional and Agile methodologies on different projects within the same organization

ID:271

Author: Colin Doyle

Type: Practice,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: Implement a common repository which enables real-time access to pertinent data regarding task and defect status as well as real-time visibility into what code changes are under development

ID: 270

Author: Colin Doyle

Type: Principle, Practice,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: scale agile approaches to handle enterprise organizations

ID: 272

Author: Colin Doyle

Type: Work Product,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: common repository

ID: 265

Author: Colin Doyle

ID: 266

Author: Colin Doyle

Type: Enabler

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Type: Principle, Process,  
Practice,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: Software change and configuration  
management (SCCM) support for parallel  
development

Content: Automating processes where  
possible, particularly the collection of data and  
generation of status metrics

ID:267

Author: Colin Doyle

ID: 268

Author: Colin Doyle

Type: Principle, Outcome,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Type: Principle, Outcome,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: Regulatory compliance

Content: senior management oversight

ID: 269

Author: Colin Doyle

Type: Principle,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: New functionality must integrate with existing functionality, and the dependencies and impacts of changes need to be clearly analyzed

ID:271

Author: Colin Doyle

Type: Principle, Practice,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: implement a Software Product Line (SPL) strategy

ID: 270

Author: Colin Doyle

Type: Principle, Practice,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: support families of related products

ID: 272

Author: Colin Doyle

Type: Principle, Process,,  
Practice,

Article: Agile Development  
Doesn't Have to Mean  
Fragile Enterprise  
Processes

Content: maximize the identification and reuse of common components

ID: 273

Author: Druckman

Type: {Value, Principle,  
Process, Purpose,  
Outcome, Practice, Work  
Product, Process Attribute,  
Level

Article: Agile  
Transformation  
Strategy

Content: Non Agile Enterprise

ID:275

Author: Druckman

Type: Practice,

Article: Agile  
Transformation  
Strategy

Content: gain knowledge,

ID: 274

Author: Druckman

Type: Level

Article: Agile  
Transformation  
Strategy

Content: Exploration

ID: 276

Author: Druckman

Type: Practice,

Article: Agile  
Transformation  
Strategy

Content: build skills

ID: 277

Author: Druckman

ID: 278

Author: Druckman

Type: Practice,

Article: Agile  
Transformation  
Strategy

Type: Practice,

Article: Agile  
Transformation  
Strategy

Content: identify organizational impediments

Content: look for examples and case studies  
of and information about how to adopt Agile

ID:279

Author: Druckman

ID: 280

Author: Druckman

Type: Practice,

Article: Agile  
Transformation  
Strategy

Type: {Value, Principle,  
Process, Purpose,  
Outcome, Practice, Work  
Product, Process Attribute,  
Level

Article: Agile  
Transformation  
Strategy

Content: train a subset of the entire  
organization

Content: evaluate about how a workgroup and  
workflow will need to change to support the  
move to Agile

ID: 281

Author: Druckman

Type: Principle, Practice,

Article: Agile  
Transformation  
Strategy

Content: Choose people who  
express a desire to learn more about Agile

ID: 282

Author: Druckman

Type: Enabler

Article: Agile  
Transformation  
Strategy

Content: have  
people involved who are committed to making  
agile work

ID:283

Author: Druckman

Type: Enabler, Practice,

Article: Agile  
Transformation  
Strategy

Content: include people from the  
organization who have influence (whose  
opinions and ideas are respected people  
whom others will follow)

ID: 284

Author: Druckman

Type: Practice

Article: Agile  
Transformation  
Strategy

Content: Identify the „Connectors“, who  
spread information about new ideas, the  
„Mavens“, who discover new ideas in the first  
place and the „Salesmen“, who help persuade  
doubters of the value of the new way of  
thinking

ID: 285

Author: Druckman

ID: 286

Author: Druckman

Type: Practice,

Article: Agile  
Transformation  
Strategy

Type: Purpose,

Article: Agile  
Transformation  
Strategy

Content: identify one or more pilot projects

Content: it is to gain the skill level necessary to do Agile well, so that the organization can later gain benefits, which they can then evaluate on a cost/benefit basis.

ID:287

Author: Druckman

ID: 288

Author: Druckman

Type: Work Product,  
Definition, Explanation

Article: Agile  
Transformation  
Strategy

Type: Practice,

Article: Agile  
Transformation  
Strategy

Content: An impediment is a process or condition that exists within the organization that is not in its best long-term interests

Content: identify organizational impediments

ID: 265

Author: Druckman

Type: {Value, Principle,  
Process, Purpose,  
Outcome, Practice, Work  
Product, Process Attribute,  
Level

Article: Agile  
Transformation  
Strategy

Content:

ID: 266

Author: Druckman

Type: {Value, Principle,  
Process, Purpose,  
Outcome, Practice, Work  
Product, Process Attribute,  
Level

Article: Agile  
Transformation  
Strategy

Content:

ID:267

Author: Druckman

Type: {Value, Principle,  
Process, Purpose,  
Outcome, Practice, Work  
Product, Process Attribute,  
Level

Article: Agile  
Transformation  
Strategy

Content:

ID: 268

Author: Druckman

Type: {Value, Principle,  
Process, Purpose,  
Outcome, Practice, Work  
Product, Process Attribute,  
Level

Article: Agile  
Transformation  
Strategy

Content:

ID: 289

Author: Druckman

ID: 290

Author: Druckman

Type: Process, Practice,

Article: Agile  
Transformation  
Strategy

Type: Practice,

Article: Agile  
Transformation  
Strategy

Content: remove organizational impediments

Content: prioritize organizational impediments

ID:291

Author: Druckman

ID: 292

Author: Druckman

Type: Outcome,

Article: Agile  
Transformation  
Strategy

Type: Level

Article: Agile  
Transformation  
Strategy

Content: Individuals from other groups see the successes of the agile teams and want to adopt the agile approach

Content: Coordination

ID: 293

Author: Druckman

ID: 294

Author: Druckman

Type: Outcome,

Article: Agile  
Transformation  
Strategy

Type: Outcome,

Article: Agile  
Transformation  
Strategy

Content: organized effort to use Agile practices throughout the enterprise

Content: one or more teams using Agile techniques on projects are achieving success with them

ID:295

Author: Druckman

ID: 296

Author: Druckman

Type: Principle, Practice

Article: Agile  
Transformation  
Strategy

Type: Practice,

Article: Agile  
Transformation  
Strategy

Content: share experiences

Content: perform lunch time „brown bag“ sessions

ID: 297

Author: Druckman

ID: 298

Author: Druckman

Type: Practice,

Article: Agile  
Transformation  
Strategy

Type: Principle, Enabler

Article: Agile  
Transformation  
Strategy

Content: plan onsite classes

Content: have a wide mix of roles attending  
classes together: developers, testers,  
architects, user interface designers, business  
analysts and members of the product groups

ID:299

Author: Druckman

ID: 300

Author: Druckman

Type: Principle, Practice,

Article: Agile  
Transformation  
Strategy

Type: Outcome,

Article: Agile  
Transformation  
Strategy

Content: Break down silos

Content: „Agile Advocats“ arise

ID: 301

Author: Druckman

Type: Enabler

Article: Agile  
Transformation  
Strategy

Content: psychological distance to problem solving

ID:303

Author: Druckman

Type: Principle, Practice,

Article: Agile  
Transformation  
Strategy

Content: Use of the inspect-and-adapt cycle

ID: 302

Author: Druckman

Type: Process, Practice,

Article: Agile  
Transformation  
Strategy

Content: implement improved technical practices.

ID: 304

Author: Druckman

Type: Practice,

Article: Agile  
Transformation  
Strategy

Content: identify ways to improve the processes

ID: 305

Author: Druckman

ID: 306

Author: Druckman

Type: Practice,

Article: Agile  
Transformation  
Strategy

Type: Practice,

Article: Agile  
Transformation  
Strategy

Content: code reviews

Content: automated testing

ID:307

Author: Druckman

ID: 308

Author: Druckman

Type: Practice,

Article: Agile  
Transformation  
Strategy

Type: Practice,

Article: Agile  
Transformation  
Strategy

Content: make plans to reduce technical debt

Content: identify existing technical debt

ID: 309

Author: Druckman

ID: 310

Author: Druckman

Type: Practice,

Article: Agile  
Transformation  
Strategy

Type: Practice,

Article: Agile  
Transformation  
Strategy

Content: understand the progress within a  
given iteration

Content: explore tools

ID:311

Author: Druckman

ID: 312

Author: Druckman

Type: Practice,

Article: Agile  
Transformation  
Strategy

Type: Level

Article: Agile  
Transformation  
Strategy

Content: Change technical environment

Content: Process Definition

ID: 313

Author: Druckman

Type: Outcome,

Article: Agile  
Transformation  
Strategy

Content: exhibit a more structured  
and unified approach to product development

ID:315

Author: Druckman

Type: Work Product,

Article: Agile  
Transformation  
Strategy

Content: organization-wide impediment  
backlog

ID: 314

Author: Druckman

Type: Practice,

Article: Agile  
Transformation  
Strategy

Content: develop of an organization-wide  
approach to evaluating and removing  
impediments

ID: 316

Author: Druckman

Type: Practice,

Article: Agile  
Transformation  
Strategy

Content: creating an organization-wide  
impediment backlog

ID: 317

Author: Druckman

ID: 318

Author: Druckman

Type: Principle, Outcome,

Article: Agile  
Transformation  
Strategy

Type: Principle, Outcome,  
Practice,

Article: Agile  
Transformation  
Strategy

Content: key responsibility for managers in an Agile organization is the removal of impediments

Content: Product Owners function together as a group to ensure that goals are cohesive across a program level

ID:319

Author: Druckman

ID: 320

Author: Druckman

Type: Practice,

Article: Agile  
Transformation  
Strategy

Type: Purpose, Outcome,

Article: Agile  
Transformation  
Strategy

Content: settle on a toolset that supports the Agile approach

Content: The enterprise has tools in place, that allow multiple teams working on the same project to coordinate and integrate work easily.

ID: 321

Author: Druckman

ID: 322

Author: Druckman

Type: Outcome,

Article: Agile  
Transformation  
Strategy

Type: Practice

Article: Agile  
Transformation  
Strategy

Content: Reporting is uniform and meaningful

Content: develop a set of metrics that helps everyone in the enterprise better understand the progress in projects

ID:323

Author: Druckman

ID: 324

Author: Druckman

Type: Outcome,

Article: Agile  
Transformation  
Strategy

Type: Level

Article: Agile  
Transformation  
Strategy

Content: the Product Organization is able to develop a uniform way of determining business value, so that projects may be judged against one another for funding

Content: Strategic Alignment

ID: 325

Author: Druckman

Type: Practice,

Article: Agile  
Transformation  
Strategy

Content: push Agile practices and values out to the entire organization

ID:327

Author: Druckman

Type: Practice,

Article: Agile  
Transformation  
Strategy

Content: act as a catalyst to unite the planning process

ID: 326

Author: Druckman

Type: Outcome,

Article: Agile  
Transformation  
Strategy

Content: the organization has and exercises the skill and knowledge to link the tactical and strategic planning efforts

ID: 328

Author: Druckman

Type: Outcome,

Article: Agile  
Transformation  
Strategy

Content: New job descriptions, for roles such as ScrumMaster and Product Owner, are created

ID: 329

Author: Druckman

ID: 330

Author: Druckman

Type: Level

Article: Agile  
Transformation  
Strategy

Type: Value, Principle,  
Outcome,

Article: Agile  
Transformation  
Strategy

Content: Transformation

Content: All employees value and promote  
transparency honesty, and making and  
meeting commitments

ID: 331

Author: Druckman

ID: 332

Author: Druckman

Type: Outcome, Benefit

Article: Agile  
Transformation  
Strategy

Type: Principle, Outcome,

Article: Agile  
Transformation  
Strategy

Content: organization will see problems and  
opportunities much sooner

Content: Individuals in an Agile organization  
believe the right path exists, so the question  
then becomes not if something can be done  
but rather how. And that is a much more  
empowering approach to problem solving

ID: 333

Author: Druckman

ID: 334

Author: Druckman

Type: Principle, Practice,

Article: Agile  
Transformation  
Strategy

Type: Anti Pattern

Article: Agile  
Transformation  
Strategy

Content: understand how fundamentally different the roles of scrum master and project manager are

Content: Underestimating the amount of change that must occur

ID:335

Author: Druckman

ID: 336

Author: Druckman

Type: Anti Pattern

Article: Agile  
Transformation  
Strategy

Type: Anti Pattern

Article: Agile  
Transformation  
Strategy

Content: Ignoring the Product Organization

Content: Neglecting distributed teams

ID: 337

Author: Druckman

ID: 338

Author: Druckman

Type: Anti Pattern

Article: Agile  
Transformation  
Strategy

Type: Practice,

Article: Agile  
Transformation  
Strategy

Content: Failing to make full use of the  
empirical process

Content: Provide agile coaching

ID:339

Author: Druckman

ID: 340

Author: Druckman

Type: Anti pattern

Article: Agile  
Transformation  
Strategy

Type: Practice,

Article: Agile  
Transformation  
Strategy

Content: Forgetting to celebrate successes

Content: Celbrate success

ID: 341

Author: Hibbs

ID: 342

Author: Hibbs

Type: Frame Condition

Article: Towards an Agile  
Process Maturity Model

Type: Frame Condition

Article: Towards an Agile  
Process Maturity Model

Content: The system to be created is not mission-critical (does not have, for example, safety-critical features);

Content: Requirements are volatile, that is, subject to frequent change;

ID:343

Author: Hibbs

ID: 344

Author: Hibbs

Type: Frame Condition

Article: Towards an Agile  
Process Maturity Model

Type: Frame Condition

Article: Towards an Agile  
Process Maturity Model

Content: The software development team is small (less than 50 members), efficient, and co-located;

Content: Customers, users, and problem domain experts are easily accessible, when needed.

ID: 345

Author: Hibbs

Type: Model Requirement,  
Value

Article: Towards an Agile  
Process Maturity Model

Content: An agile process maturity model should make possible a balance between discipline and agility, in such a way that development and evolution become neither anarchic nor discoordinated, but keep the characteristic of rapid response to change.

ID: 346

Author: Hibbs

Type: Value, Model  
Requirement

Article: Towards an Agile  
Process Maturity Model

Content: Agile Maturity should be related to greater professionalism of the development/evolution team, to improvements on process visibility and adaptivity, and to better quality of the software products

ID:347

Author: Hibbs

Type: Value, Model  
Requirement

Article: Towards an Agile  
Process Maturity Model

Content: The maturity model should encourage, on each process improvement stage, changing the focus from the software product to the system in which the software is placed upon.

ID: 348

Author: Hibbs

Type: {Value, Principle,  
Process, Purpose,  
Outcome, Practice, Work  
Product, Process Attribute,  
Level

Article: Towards an Agile  
Process Maturity Model

Content: improve the efficiency of the coordination and collaboration process

ID: 349

Author: Hibbs

Type: Practice,

Article: Towards an Agile  
Process Maturity Model

Content: improve the efficiency of  
the coordination and collaboration process

ID: 350

Author: Hibbs

Type: Value, Principle,  
Model Requirement, Anti  
Pattern

Article: Towards an Agile  
Process Maturity Model

Content: a maturity model should  
avoid to impose requirements that cause  
excessive bureaucratization or rigidity.

ID:351

Author: Hibbs

Type: Value, Principle,  
Model Requirement, Anti  
Pattern

Article: Towards an Agile  
Process Maturity Model

Content: mandatory documentation should be  
restricted to the minimum necessary for not to  
hamper agility.

ID: 352

Author: Hibbs

Type: Model Requirement

Article: Towards an Agile  
Process Maturity Model

Content: The maturity model should guide  
the specification of the necessary and  
sufficient artifacts for each kind of project.

ID: 353

Author: Hibbs

Type: Model Requirement,  
Anti Pattern

Article: Towards an Agile  
Process Maturity Model

Content: The practices recommended by the maturity model also should neither seriously constrain developers' creativity, nor reduce process speed

ID:355

Author: Hibbs

Type: Level

Article: Towards an Agile  
Process Maturity Model

Content: 2. Minimum

ID: 354

Author: Hibbs

Type: Level

Article: Towards an Agile  
Process Maturity Model

Content: 1. Non-Agile:

ID: 356

Author: Hibbs

Type: Level

Article: Towards an Agile  
Process Maturity Model

Content: 3. Consolidated

ID: 357

Author: Hibbs

ID: 358

Author: Hibbs

Type: Practice

Article: Towards an Agile  
Process Maturity Model

Type: Practice,

Article: Towards an Agile  
Process Maturity Model

Content: Requirements gathering

Content: Unit testing

ID:359

Author: Hibbs

ID: 360

Author: Hibbs

Type: Practice

Article: Towards an Agile  
Process Maturity Model

Type: Practice,

Article: Towards an Agile  
Process Maturity Model

Content: Version control

Content: Automated compilation

ID: 361

Author: Hibbs

ID: 362

Author: Hibbs

Type: Principle,

Article: Towards an Agile  
Process Maturity Model

Type: Principle, Practice,

Article: Towards an Agile  
Process Maturity Model

Content: Short iterative cycles

Content: Planning for the current iteration

ID:363

Author: Hibbs

ID: 364

Author: Hibbs

Type: Principle, Outcome

Article: Towards an Agile  
Process Maturity Model

Type: Principle, Outcome,

Article: Towards an Agile  
Process Maturity Model

Content: Direct contact with experts

Content: Collocated team

ID: 365

Author: Hibbs

ID: 366

Author: Hibbs

Type: Practice,

Article: Towards an Agile  
Process Maturity Model

Type: Value,

Article: Towards an Agile  
Process Maturity Model

Content: Acceptation testing

Content: rapid response to change

ID:367

Author: Hibbs

ID: 368

Author: Hibbs

Type: Principle

Article: Towards an Agile  
Process Maturity Model

Type: Enabler

Article: Towards an Agile  
Process Maturity Model

Content: requirements documentation  
and source code evolve together.

Content: Refactoring techniques

ID: 369

Author: Hibbs

ID: 370

Author: Hibbs

Type: Principle, Outcome

Article: Towards an Agile  
Process Maturity Model

Type: Principle, Outcome,  
Enabler, Work Product

Article: Towards an Agile  
Process Maturity Model

Content: Systematic reuse

Content: Coding standards and guidelines

ID:371

Author: Hibbs

ID: 372

Author: Hibbs

Type: Principle, Practice,

Article: Towards an Agile  
Process Maturity Model

Type: Principle, Process,  
Purpose, Practice

Article: Towards an Agile  
Process Maturity Model

Content: Independent unit, integration and  
system testing

Content: Documentation to help evolution

ID: 373

Author: Hibbs

ID: 374

Author: Hibbs

Type: Process, Practice,

Article: Towards an Agile  
Process Maturity Model

Type: Process, Practice,

Article: Towards an Agile  
Process Maturity Model

Content: Human resource management

Content: Risk management

ID:375

Author: Hibbs

ID: 376

Author: Hibbs

Type: Process, Practice,

Article: Towards an Agile  
Process Maturity Model

Type: Process, Practice,

Article: Towards an Agile  
Process Maturity Model

Content: Infra-structure management

Content: Release/deployment management

ID: 377

Author: Hibbs

ID: 378

Author: Hibbs

Type: Process, Practice,

Article: Towards an Agile  
Process Maturity Model

Type: Process, Practice,

Article: Towards an Agile  
Process Maturity Model

Content: Technical support management

Content: Expectation management

ID:379

Author: Hibbs

ID: 380

Author: Hibbs

Type: Level

Article: Towards an Agile  
Process Maturity Model

Type: Level

Article: Towards an Agile  
Process Maturity Model

Content: 2: Organized

Content: 3 - Disciplined

ID: 381

Author: Hibbs

ID: 382

Author: Hibbs

Type: Practice,

Article: Towards an Agile  
Process Maturity Model

Type: Practice,

Article: Towards an Agile  
Process Maturity Model

Content: plan developers' training

Content: plan contracting and adaptation of  
recently allocated developers to a project.

ID:383

Author: Hibbs

ID: 384

Author: Hibbs

Type: Practice,

Article: Towards an Agile  
Process Maturity Model

Type: Practice,

Article: Towards an Agile  
Process Maturity Model

Content: Pair programming

Content: pair rotation

ID: 385

Author: Hibbs

ID: 386

Author: Hibbs

Type: Process, Practice,

Article: Towards an Agile  
Process Maturity Model

Type: Process, Outcome,  
Practice,

Article: Towards an Agile  
Process Maturity Model

Content: Product quality assessment

Content: Application of process metrics

ID:387

Author: Hibbs

ID: 388

Author: Hibbs

Type: Process, Practice,

Article: Towards an Agile  
Process Maturity Model

Type: Process, Practice,

Article: Towards an Agile  
Process Maturity Model

Content: Formal project closure

Content: Knowledge management

ID: 389

Author: Humble

Type: Principle,

Article: The Agile Maturity Model

Content: Ideally, each change made to your application, its environment, or its configuration should go through an automated process.

ID:391

Author: Humble

Type: Process, Practice,

Article: The Agile Maturity Model

Content: create and maintain an automated process that builds your application and runs tests on every change and then provides feedback to the whole team on the process.

ID: 390

Author: Humble

Type: Outcome, Work Product,

Article: The Agile Maturity Model

Content: deployment pipeline

ID: 392

Author: Humble

Type: Principle, Outcome,

Article: The Agile Maturity Model

Content: Environments consist of the entire stack your application requires to work: hardware, infrastructure, networking, application stacks, external services, and their configuration

ID: 393

Author: Humble

Type: Process, Practice,

Article: The Agile  
Maturity Model

Content: data management

ID:395

Author: Humble

Type: Outcome, Benefit

Article: The Agile  
Maturity Model

Content: Reduced defects, so you can  
improve your reputation and spend less on  
support

ID: 394

Author: Humble

Type: Outcome, Benefit

Article: The Agile  
Maturity Model

Content: Reduced cycle time, so you can  
respond faster to changing business  
requirements and increase revenue

ID: 396

Author: Humble

Type: Outcome, Benefit

Article: The Agile  
Maturity Model

Content: Increased predictability of your  
software delivery lifecycle to make planning  
more effective

ID: 397

Author: Humble

Type: Outcome, Benefit

Article: The Agile  
Maturity Model

Content: The ability to adopt and maintain an attitude of compliance to any regulatory regime you are subject to

ID: 398

Author: Humble

Type: Outcome, Benefit

Article: The Agile  
Maturity Model

Content: The ability to determine and manage software delivery risks effectively

ID:399

Author: Humble

Type: Outcome, Benefit

Article: The Agile  
Maturity Model

Content: Reduced costs due to better risk management and fewer issues delivering software

ID: 400

Author: Humble

Type: Practice,

Article: The Agile  
Maturity Model

Content: Identify where your organization lies on the model. You may find that different parts of your organization achieve different levels on each of the different categories.

ID: 401

Author: Humble

ID: 402

Author: Humble

Type: Practice,

Article: The Agile  
Maturity Model

Type: Practice,

Article: The Agile  
Maturity Model

Content: Choose what to focus on. You should work out what the possible improvements you could implement are, how much they will cost, and what benefit they will deliver

Content: choose a few improvements you could make and decide how to implement those changes

ID: 403

Author: Humble

ID: 404

Author: Humble

Type: Practice,

Article: The Agile  
Maturity Model

Type: Practice,

Article: The Agile  
Maturity Model

Content: set acceptance criteria to define the results you are expecting to see

Content: decide if the changes were successful

ID: 405

Author: Humble

ID: 406

Author: Humble

Type: Practice,

Article: The Agile  
Maturity Model

Type: Practice,

Article: The Agile  
Maturity Model

Content: Implement the changes

Content: plan how to implement the changes

ID: 407

Author: Humble

ID: 408

Author: Humble

Type: Principle, Practice,

Article: The Agile  
Maturity Model

Type: Practice, Principle

Article: The Agile  
Maturity Model

Content: start with a proof of  
concept

Content: choose a part of your organization  
that is really suffering

ID: 409

Author: Humble

Type: {Value, Principle, Process, Purpose, Outcome, Practice, Work Product, Process Attribute, Level

Article: The Agile Maturity Model

Content: execute your (Improvement) plan

ID: 410

Author: Humble

Type: Practice,

Article: The Agile Maturity Model

Content: Check if the changes you implemented had the desired effect, using predefined acceptance criteria

ID: 411

Author: Humble

Type: Work Product,

Article: The Agile Maturity Model

Content: Change/Improvement Plan

ID: 412

Author: Humble

Type: {Value, Principle, Process, Purpose, Outcome, Practice, Work Product, Process Attribute, Level

Article: The Agile Maturity Model

Content: Change/improvement acceptance criteria

ID: 413

Author: Humble

ID: 414

Author: Humble

Type: Practice,

Article: The Agile  
Maturity Model

Type: Level

Article: The Agile  
Maturity Model

Content: Hold a retrospective to find out how well the changes were executed.

Content: 0: Repeatable

ID: 415

Author: Humble

ID: 416

Author: Humble

Type: Level

Article: The Agile  
Maturity Model

Type: Level

Article: The Agile  
Maturity Model

Content: 1: Consistent

Content: 4 Quantitatively Managed

ID: 417

Author: Humble

ID: 418

Author: Humble

Type: Level

Article: The Agile  
Maturity Model

Type: Principle,

Article: The Agile  
Maturity Model

Content: 3 Optimizing

Content: Teams regularly meet to discuss  
integration problems and resolve them

ID: 419

Author: Humble

ID: 420

Author: Humble

Type: Principle, Process,  
Outcome, Practice,

Article: The Agile  
Maturity Model

Type: Work Product,

Article: The Agile  
Maturity Model

Content: Build metrics gathered made visible  
and acted on

Content: Build Metric

ID: 421

Author: Humble

ID: 422

Author: Humble

Type: Principle,

Article: The Agile  
Maturity Model

Type: Outcome,

Article: The Agile  
Maturity Model

Content: Automated build and test cycle every  
time a change is committed

Content: Dependencies managed

ID: 423

Author: Humble

ID: 424

Author: Humble

Type: Principle, Process,  
Outcome, Practice,

Article: The Agile  
Maturity Model

Type: Work Product,

Article: The Agile  
Maturity Model

Content: Re use of scripts and tools

Content: Script

ID: 425

Author: Humble

Type: Principle, Purpose,  
Outcome,

Article: The Agile  
Maturity Model

Content: All environments managed effectively

ID: 427

Author: Humble

Type: Outcome,

Article: The Agile  
Maturity Model

Content: Orchestrated deployments managed

ID: 426

Author: Humble

Type: Outcome,

Article: The Agile  
Maturity Model

Content: Provisioning fully automated

ID: 428

Author: Humble

Type: Outcome

Article: The Agile  
Maturity Model

Content: Release and rollback processes  
tested

ID: 429

Author: Humble

ID: 430

Author: Humble

Type: Outcome,

Article: The Agile  
Maturity Model

Type: Principle,, Outcome,  
Practice,

Article: The Agile  
Maturity Model

Content: Fully automated, self service push  
button process for deploying software

Content: Operations and delivery teams  
regularly collaborate to manage risks and  
reduce cycle time.

ID: 431

Author: Humble

ID: 432

Author: Humble

Type: Principle, Process,  
Outcome, Practice,

Article: The Agile  
Maturity Model

Type: Outcome

Article: The Agile  
Maturity Model

Content: Environment and application health  
monitored and proactively managed

Content: Cycle time monitored

ID: 433

Author: Humble

ID: 434

Author: Humble

Type: Outcome,

Article: The Agile  
Maturity Model

Type:, Outcome,

Article: The Agile  
Maturity Model

Content: Change Management and approvals  
processes defined and enforced

Content: Regulatory and compliance  
conditions met

ID: 435

Author: Humble

ID: 436

Author: Humble

Type: Benefit

Article: The Agile  
Maturity Model

Type: Principle, Outcome,

Article: The Agile  
Maturity Model

Content: Production rollbacks rare

Content: Defects found and fixed immediately

ID: 437

Author: Humble

ID: 438

Author: Humble

Type: Outcome,

Article: The Agile  
Maturity Model

Type: Outcome,

Article: The Agile  
Maturity Model

Content: Quality Metrics and trends tracked

Content: Non functional requirements defined  
and measured

ID: 439

Author: Humble

ID: 440

Author: Humble

Type: Outcome,

Article: The Agile  
Maturity Model

Type: Principle, Outcome,

Article: The Agile  
Maturity Model

Content: Automated unit and acceptance tests

Content: Testing part of the development  
process

ID: 441

Author: Humble

Type: Principle, Process,  
Outcome, Practice

Article: The Agile  
Maturity Model

Content: Automated tests written as part of  
story development

ID: 443

Author: Humble

Type: Principle, Outcome,

Article: The Agile  
Maturity Model

Content: Database upgrades and rollbacks  
tested with every deployment

ID: 442

Author: Humble

Type: Principle, Outcome,

Article: The Agile  
Maturity Model

Content: Release to release feedback loop of  
database performance and deployment  
process

ID: 444

Author: Humble

Type: Outcome,

Article: The Agile  
Maturity Model

Content: Database performance monitored  
and optimized

ID: 445

Author: Humble

Type: Principle,

Article: The Agile  
Maturity Model

Content: Database changes performed  
automatically as part of deployment process

ID: 447

Author: Humble

Type: Process,

Article: The Agile  
Maturity Model

Content: Testing

ID: 446

Author: Humble

Type: Principle, Outcome,

Article: The Agile  
Maturity Model

Content: Changes to databases done with  
automated scripts versioned with application

ID: 448

Author: Humble

Type: Process,

Article: The Agile  
Maturity Model

Content: Data Management

ID: 449

Author: Humble

ID: 450

Author: Humble

Type: Process,

Article: The Agile  
Maturity Model

Type: Process,

Article: The Agile  
Maturity Model

Content: Release management & compliance

Content: Environments & deployments

ID: 451

Author: Humble

ID: 452

Author: Humble

Type: Process,

Article: The Agile  
Maturity Model

Type: Principle, Practice,

Article: The Agile  
Maturity Model

Content: Build management and continuous  
integration

Content: implement change incrementally

ID: 453

Author: Jayaraj

ID: 454

Author: Jayaraj

Type: Process,

Article: The Agile  
Maturity Model

Type: Process,

Article: The Agile  
Maturity Model

Content: Testing

Content: Source code management

ID: 455

Author: Jayaraj

ID: 456

Author: Jayaraj

Type: Value, Principle,

Article: The Agile  
Maturity Model

Type: Value, Principle,

Article: The Agile  
Maturity Model

Content: Collective code ownership

Content: Collaboration

ID: 457

Author: Jayaraj

Type: Value, Principle,

Article: The Agile  
Maturity Model

Content: Responsiveness to business

ID: 459

Author: Jayaraj

Type: Principle, Process,  
Practice,

Article: The Agile  
Maturity Model

Content: Story formation

ID: 458

Author: Jayaraj

Type: Value, Principle,

Article: The Agile  
Maturity Model

Content: Assurance and governance

ID: 460

Author: Jayaraj

Type: Value, Principle,  
Outcome,

Article: The Agile  
Maturity Model

Content: Design simplicity

ID: 461

Author: Jayaraj

ID: 462

Author: Jayaraj

Type: Process, Outcome,

Article: The Agile  
Maturity Model

Type: Level

Article: The Agile  
Maturity Model

Content: Build process

Content: Collaborative

ID: 463

Author: Jayaraj

ID: 464

Author: Jayaraj

Type: Level

Article: The Agile  
Maturity Model

Type: {Value, Principle,  
Process, Purpose,  
Outcome, Practice, Work  
Product, Process Attribute,  
Level

Article: The Agile  
Maturity Model

Content: Operating (Consistent exhibition of  
competance)

Content: Adaptive (Expertise to adapt to  
change)

ID: 465

Author: Jayaraj

Type: Level

Article: The Agile Maturity Model

Content: Innovating (Creative evolution of practice, and spread these practices throughout the organization)

ID: 466

Author: Jayaraj

Type: Principle,

Article: The Agile Maturity Model

Content: Developers write unit tests before writing functional code

ID: 467

Author: Jayaraj

Type: Value, Principle, Outcome,

Article: The Agile Maturity Model

Content: Unit testing integrated into build process, testing automated as much as reasonable given application

ID: 468

Author: Jayaraj

Type: Value, Principle, Outcome,

Article: The Agile Maturity Model

Content: Test are identified and produced as part of a story creation

ID: 469

Author: Jayaraj

Type: Principle, Outcome,

Article: The Agile  
Maturity Model

Content: Automated functional testing (e.g..  
GUI testing); stories remain in development  
until all bugs are fixed or deferred

ID: 471

Author: Jayaraj

Type: {Value, Principle,  
Process, Purpose,  
Outcome, Practice, Work  
Product, Process Attribute,  
Level

Article: The Agile  
Maturity Model

Content: All development collateral is in SCM

ID: 470

Author: Jayaraj

Type: Principle, Outcome,

Article: The Agile  
Maturity Model

Content: IDE(s) integrate with SCM

ID: 472

Author: Jayaraj

Type: Principle, Outcome,

Article: The Agile  
Maturity Model

Content: SCM is transparent to delivery team,  
behind build process

ID: 473

Author: Jayaraj

Type: Principle,

Article: The Agile  
Maturity Model

Content: Within an sprint, functionality delivery is signed up for just in time

ID: 475

Author: Jayaraj

Type: Principle, Outcome,

Article: The Agile  
Maturity Model

Content: Project collaboration tools (wiki, mailing list, IM) in place and used throughout project team; project status published and visible to all

ID: 474

Author: Jayaraj

Type: Principle,

Article: The Agile  
Maturity Model

Content: Old (bugs) and new functionality is queued, development team pops the stack in real time

ID: 476

Author: Jayaraj

Type: Principle, Outcome,

Article: The Agile  
Maturity Model

Content: Daily stand-ups and sprint meetings; problem solving is bottom-up as opposed to top-down; team sets sprint objectives and agrees on estimates.

ID: 477

Author: Jayaraj

Type: Principle, Outcome,

Article: The Agile  
Maturity Model

Content: Integrated, continuous build process,  
with build status notification to the team and  
collective responsibility for state of the build

ID: 479

Author: Jayaraj

Type: Principle, Outcome,

Article: The Agile  
Maturity Model

Content: Frequent (near-real time)  
prioritization of old and new functionality

ID: 478

Author: Jayaraj

Type: Principle, Outcome,

Article: The Agile  
Maturity Model

Content: Business is part of the team,  
stakeholders accept working software at  
reviews in lieu of other tracking or progress  
metrics

ID: 480

Author: Jayaraj

Type: Principle, Outcome,

Article: The Agile  
Maturity Model

Content: Build automatically deploys to QA  
environment available to any interested party

ID: 481

Author: Jayaraj

Type: Principle, Outcome,

Article: The Agile  
Maturity Model

Content: Iterative process with sprints of length short enough to respond to business change

ID: 482

Author: Jayaraj

Type: Principle, Outcome,

Article: The Agile  
Maturity Model

Content: Showcases per sprint; business prioritizes functionality per sprint

ID: 483

Author: Jayaraj

Type: Principle, Outcome,

Article: The Agile  
Maturity Model

Content: Continuous involvement of the business on the team: business certifies story complete out of dev; involved in (rather than approval of) story definition and acceptance tests

ID: 484

Author: Jayaraj

Type: Principle, Outcome,

Article: The Agile  
Maturity Model

Content: Business writes high-level stories for all requests; organizational story queue exists; prioritization decisions made from full story backlog

ID: 485

Author: Jayaraj

Type: Principle, Outcome,

Article: The Agile  
Maturity Model

Content: Development process is integral to business initiatives; development teams work as a part of the business unit rather than as a service to the business unit

ID: 487

Author: Jayaraj

Type: Outcome,

Article: The Agile  
Maturity Model

Content: Sprint planning is introduced

ID: 486

Author: Jayaraj

Type: Outcome,

Article: The Agile  
Maturity Model

Content: Concept of release planning is introduced

ID: 488

Author: Jayaraj

Type: Principle, Outcome,

Article: The Agile  
Maturity Model

Content: Plan becomes communication and tracking tool rather than an exception report

ID: 489

Author: Jayaraj

Type: Principle, Outcome,  
Practice,

Article: The Agile  
Maturity Model

Content: Every sprint review examines value delivered and assesses next sprint by need, priorities (and delivery)

ID: 490

Author: Jayaraj

Type: Outcome,

Article: The Agile  
Maturity Model

Content: Organizational adoption: Introduction of Portfolio Planning and global optimization of resources (use of metric and standard measures)

ID: 491

Author: Jayaraj

Type: Outcome, Practice,

Article: The Agile  
Maturity Model

Content: Business review of value and return helps teams on regular basis based upon status

ID: 492

Author: Jayaraj

Type: Principle, Outcome,  
Practice,

Article: The Agile  
Maturity Model

Content: Marshaling use cases into discreet statements of functionality that can be delivered in time boxed development sprints

ID: 493

Author: Jayaraj

Type: Principle, Outcome,  
Practice, Work Product,

Article: The Agile  
Maturity Model

Content: Stories are an expression of end-to-end functionality to be developed, including testable acceptance criteria and a statements of value

ID: 495

Author: Jayaraj

Type: Outcome, Work  
Product,

Article: The Agile  
Maturity Model

Content: Global repository of functional requirements for stories developed by business for any business requirements, including a formal measure of business value delivered

ID: 494

Author: Jayaraj

Type: Principle, Process,  
Outcome, Practice, Work  
Product,

Article: The Agile  
Maturity Model

Content: Spontaneous story development provided by business for in-flight projects; stories not derived from extant sources but are immediate expressions of customer demand/need

ID: 496

Author: Jayaraj

Type: Principle, Outcome,

Article: The Agile  
Maturity Model

Content: Application of fundamental design patterns

ID: 497

Author: Jayaraj

Type: Process,, Practice,

Article: The Agile Maturity Model

Content: Structural refactoring to decouple application architecture

ID: 498

Author: Jayaraj

Type: Principle,

Article: The Agile Maturity Model

Content: Doing only what needs to be done

ID: 499

Author: Jayaraj

Type: Principle, Process, Practice,

Article: The Agile Maturity Model

Content: Aggressive and constant refactoring to improve code quality/simplicity extant code

ID: 500

Author: Jayaraj

Type: Principle, Outcome, Practice,

Article: The Agile Maturity Model

Content: Spiking solution with each release to introduce new ideas and challenge architectural decision

ID: 501

Author: Jayaraj

Type: Principle, Outcome, Practice,

Article: The Agile Maturity Model

Content: Technical design decisions taken with each story

ID: 502

Author: Jayaraj

Type: Outcome,

Article: The Agile Maturity Model

Content: Consistent, repeatable build process with durable artifacts executed manually with each release

ID: 503

Author: Jayaraj

Type: Outcome, Practice,

Article: The Agile Maturity Model

Content: Build is automated, executed on a timed basis

ID: 504

Author: Jayaraj

Type: Principle, Outcome,

Article: The Agile Maturity Model

Content: Unit tests are integrated with the automated build; team is constantly notified of the status of the build; build triggered by SCM updates

ID: 505

Author: Jayaraj

Type: Principle, Outcome,  
Practice,

Article: The Agile  
Maturity Model

Content: Test and metrics (e.g., code quality, complexity, check style, etc.) are integrated as gatekeeper events; build data archived and reported in build portal/dashboard

ID: 507

Author: Jayaraj

Type: Outcome,

Article: The Agile  
Maturity Model

Content: Product integration tests included; build process executes automatic deployment to QA testing environment

ID: 506

Author: Jayaraj

Type: Outcome,

Article: The Agile  
Maturity Model

Content: All product components advertise dependencies; master repository established

ID: 508

Author: Jayaraj

Type: Outcome,

Article: The Agile  
Maturity Model

Content: Status reports document progress

ID: 509

Author: Kurrupath

Type: Work Product,

Article: Maturing Agile Processes to Deliver Better Value

Content: Product Backlog

ID: 510

Author: Kurrupath

Type: Practice,

Article: Maturing Agile Processes to Deliver Better Value

Content: Backlog Grooming

ID: 511

Author: Kurrupath

Type: Practice,

Article: Maturing Agile Processes to Deliver Better Value

Content: Sprint Planning

ID: 512

Author: Kurrupath

Type: Practice,

Article: Maturing Agile Processes to Deliver Better Value

Content: Task Estimation

ID: 513

Author: Kurrupath

Type: Work Product,

Article: Maturing Agile Processes to Deliver Better Value

Content: Sprint Backlog

ID: 514

Author: Kurrupath

Type: Practice,

Article: Maturing Agile Processes to Deliver Better Value

Content: Daily Scrum Meeting

ID: 515

Author: Kurrupath

Type: Work Product,

Article: Maturing Agile Processes to Deliver Better Value

Content: Burn down Chart

ID: 516

Author: Kurrupath

Type: Work Product,

Article: Maturing Agile Processes to Deliver Better Value

Content: Milestone Map

ID: 517

Author: Kurrupath

Type: Outcome,

Article: Maturing Agile Processes to Deliver Better Value

Content: Potentially Shippable Product Increment

ID: 518

Author: Kurrupath

Type: Benefit

Article: Maturing Agile Processes to Deliver Better Value

Content: increased responsiveness to changes in business requirements

ID: 519

Author: Kurrupath

Type: Benefit

Article: Maturing Agile Processes to Deliver Better Value

Content: organizations can get to market faster by reducing the overhead of analyzing, designing, and documenting the entire feature set of a product or an application

ID: 520

Author: Kurrupath

Type: Principle, Practice,

Article: Maturing Agile Processes to Deliver Better Value

Content: improve the overall efficiency by increasing the visibility into product development and by enabling quick and frequent feedback from business owners

ID: 521

Author: Martin Proulx

Type: Level

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: Team Level

ID: 523

Author: Martin Proulx

Type: Level

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: Business Level

ID: 522

Author: Martin Proulx

Type: Level

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: Department Level

ID: 524

Author: Martin Proulx

Type: Level

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: Project Management Level

ID: 525

Author: Martin Proulx

Type: Level

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: Management Level

ID: 526

Author: Martin Proulx

Type: Indicator, Outcome,

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: A Scrum Master is in place

ID: 527

Author: Martin Proulx

Type: Outcome, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: The 3 Scrum roles are well understood and respected

ID: 528

Author: Martin Proulx

Type: Outcome, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: If there is more than 1 Scrum team, a Scrum of Scrum has been put in place

ID: 529

Author: Martin Proulx

Type: Outcome, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: Team members are attending Agile conferences

ID: 530

Author: Martin Proulx

Type: Outcome, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: A Product Owner is clearly identified

ID: 531

Author: Martin Proulx

Type: Outcome, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: There is a clear segmentation between the role of the Scrum Master and that of the project manager

ID: 532

Author: Martin Proulx

Type: Outcome, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: The team is autonomous and the Scrum rituals and artifacts are respected and standardized

ID: 533

Author: Martin Proulx

Type: Outcome, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: The process is documented and is consistent across projects

ID: 535

Author: Martin Proulx

Type: Outcome, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: The project manager is well accepted and is part of the Product Owner team

ID: 534

Author: Martin Proulx

Type: Outcome, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: Product Owners are clearly identified and are dedicated to their project

ID: 536

Author: Martin Proulx

Type: Outcome, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: •The concept of incremental and iterative development is fully accepted from the business representatives

ID: 537

Author: Martin Proulx

Type: Outcome, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: Agile is accepted as a solid approach for large scale projects

ID: 538

Author: Martin Proulx

Type: Outcome, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: Training initiatives have begun for management and attendance is high

ID: 539

Author: Martin Proulx

Type: Outcome, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: A strong evangelist is in place at the management / executive level to promote the new approach

ID: 540

Author: Martin Proulx

Type: Outcome, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: The 3 Scrum roles are well understood and respected

ID: 541

Author: Martin Proulx

Type: Outcome, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: The concept of incremental and iterative development is fully accepted from the business representatives

ID: 542

Author: Martin Proulx

Type: {Value, Principle, Process, Purpose, Outcome, Practice, Work Product, Process Attribute, Level

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content:

ID: 543

Author: Martin Proulx

Type: Outcome, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: The traditional project management approach has been adapted to include a more Agile approach

ID: 544

Author: Martin Proulx

Type: Outcome, Practice, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: Review the best practices to adapt to changing realities

ID: 545

Author: Martin Proulx

Type: Outcome, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: Managers have fully transferred the authority and responsibility to the teams to allow them to do their job properly

ID: 546

Author: Martin Proulx

Type: Outcome, Practice, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: Managers promote collaboration and teamwork

ID: 547

Author: Martin Proulx

Type: Outcome, Practice, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: Managers support continuous learning and do not systematically penalize failures

ID: 548

Author: Martin Proulx

Type: Outcome, Practice, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: Managers adapt their management style to the context of their team

ID: 549

Author: Martin Proulx

Type: Outcome, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: Strong collaboration between all parties involved.

ID: 550

Author: Martin Proulx

Type: Outcome, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: Organization is able to quickly react to changes in its environment

ID: 551

Author: Martin Proulx

Type: Outcome, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: Management is considering implementing Agile to projects that do not require software development

ID: 552

Author: Martin Proulx

Type: Outcome, Indicator

Article: Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity

Content: Friction around the new approach has disappeared

ID: 554

Author: Patel

Type: Level

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Initial (level 1)

ID: 555

Author: Patel

Type: Level

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Explored (level 2). Project planning, customer and stakeholders' orientation, value, Collaboration

ID:556

Author: Patel

Type: Level

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Defined (level 3). Customer satisfaction, software quality and development practices

ID: 557

Author: Patel

Type: Level

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Improved (level 4). People orientation and project management.

ID: 558

Author: Patel

Type: Level

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Sustained (level 5). Project performance, defect prevention

ID: 559

Author: Patel

Type: Process (KPA)

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: 2.1 Project planning (Release planning, at level 2)

ID:560

Author: Patel

Type: Process (KPA)

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: 2.2 Story cards driven Development (requirements engineering, at level 2)

ID: 561

Author: Patel

Type: Process (KPA)

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: 2.3 On-site customer availability

ID: 553

Author: Patel

Type: Process (KPA)

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: 2.4 Introduction of Test driven development TDD

ID:555

Author: Patel

Type: Process (KPA)

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: 3.2 Delivering Working Products / SW Frequently

ID: 554

Author: Patel

Type: Process (KPA)

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: 3.1 Customer relationship management

ID: 556

Author: Patel

Type: Process (KPA)

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: 3.3 Pair programming

ID: 557

Author: Patel

Type: Process (KPA)

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: 3.4 Mutual interaction

ID: 558

Author: Patel

Type: Process (KPA)

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: 3.5 Test Driven Development

ID:559

Author: Patel

Type: Process (KPA)

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: 3.6 Implementation and Interaction

ID: 560

Author: Patel

Type: Process (KPA)

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Coding standards

ID: 561

Author: Patel

Type: Process (KPA)

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: 4.1 Project management

ID: 562

Author: Patel

Type: Process (KPA)

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Sustainable Pace

ID:563

Author: Patel

Type: Process (KPA)

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: Self-Organising Team

ID: 564

Author: Patel

Type: Process (KPA)

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: 4.4 Risk Assessment

ID: 565

Author: Patel

Type: Process (KPA)

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: 4.5 Code Optimisation planning

ID: 566

Author: Patel

Type: Process (KPA)

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: 5.1 Project planning (release planning, at level 5)

ID:567

Author: Patel

Type: Process (KPA)

Article: Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices

Content: 5.2 Story cards driven development (requirements engineering, at level 5)

ID: 401

Author: Humble

ID: 402

Author: Humble

Type: {Value, Principle,  
Process, Purpose,  
Outcome, Practice, Work  
Product, Process Attribute,  
Level

Article: The Agile  
Maturity Model

Type: {Value, Principle,  
Process, Purpose,  
Outcome, Practice, Work  
Product, Process Attribute,  
Level

Article: The Agile  
Maturity Model

Content:

Content:

ID: 403

Author: Humble

ID: 404

Author: Humble

Type: {Value, Principle,  
Process, Purpose,  
Outcome, Practice, Work  
Product, Process Attribute,  
Level

Article: The Agile  
Maturity Model

Type: {Value, Principle,  
Process, Purpose,  
Outcome, Practice, Work  
Product, Process Attribute,  
Level

Article: The Agile  
Maturity Model

Content:

Content: