

What is essential?

- A pilot survey on views about the requirements metamodel of reqT.org

Björn Regnell

Lund University

March 14, 2016

refsq.org/2016

github.com/bjornregnell/reqT-survey

1 Objective

- Research Question
- Approach

2 Background

- About reqT

3 Methodology

- Data Collection
- Data Analysis

4 Results

- Essentiality

5 Conclusion

- Future Work

Research question

In the context of software requirements engineering education:

- How to choose a set of **essential requirements engineering concepts** that allows for **sufficient expressiveness**, without overloading the metamodel with esoteric concepts just for the sake of **completeness**?

Research question

In the context of software requirements engineering education:

- How to choose a set of **essential requirements engineering concepts** that allows for **sufficient expressiveness**, without overloading the metamodel with esoteric concepts just for the sake of **completeness**?
- Presumption: as teachers we should be method agnostic; there is no single "correct" dogma

Approach

- Make a survey among RE scholars

Approach

- Make a survey among RE scholars
 - How to quantify "essentiality"?

Approach

- Make a survey among RE scholars
 - How to quantify "essentiality"?
 - One possible quantification:
The more scholars that **agree** on a definition of a concept
and
the more scholars that **use** the concept,
the more **essential** is the concept.

Approach

- Make a survey among RE scholars
 - How to quantify "essentiality"?
 - One possible quantification:
The more scholars that **agree** on a definition of a concept
and
the more scholars that **use** the concept,
the more **essential** is the concept.
- Use the reqT metamodel as a basis for the survey

What is reqT?

- reqT is an open source requirements engineering tool developed in an **educational** context
Download from here: <http://reqt.org/>
`java -jar reqT.jar`
- reqT aims to inspire **code-loving** cs students to learn more about RE through a **Scala**-internal DSL
- reqT has a **metamodel** including entities, relations and attributes gathered from various sources such as text books, research papers, IREB, wikipedia, etc.

A small reqT model

```
val m = Model(  
  Feature("chat") has (  
    Spec("The system shall enable sending text messages among users"),  
    Prio(42)  
  )  
)
```

An larger example reqT model

```
Model(  
  Component("appearance") has (  
    VariationPoint("color") has (  
      Min(0), Max(2), Variant("blue"), Variant("red"), Variant("green")),  
    VariationPoint("shape") has (  
      Min(1), Max(1), Variant("round"), Variant("square")),  
    VariationPoint("payment") has (  
      Min(1), Max(2), Variant("cash"), Variant("credit")),  
    VariationPoint("payment") requires Variant("cash"),  
    Variant("round") excludes Variant("red"),  
    Variant("green") requires Variant("square")),  
  Component("appearance") requires VariationPoint("shape"),  
  App("free") has Component("appearance"),  
  App("free") binds (VariationPoint("shape") binds Variant("round")),  
  App("premium") has Component("appearance"),  
  App("premium") binds (  
    VariationPoint("color") binds (Variant("red"), Variant("green")),  
    VariationPoint("shape") binds (Variant("round"), Variant("square")),  
    VariationPoint("payment") binds Variant("cash")))
```

49 entities of the metamodel (see paper appendix)

Entity

Actor
App
Barrier
Breakpoint
Class
Component
Configuration
Data
Design
Domain
Epic
Event
Feature
Function
Goal
Idea
Interface
Issue
Item
Label
Member
Meta
MockUp
Module
Product
Quality
Relationship
Release
Req
Resource
Risk
Scenario
Screen
Section
Service
Stakeholder
State
Story
System
Target
Task
Term
Test
Ticket
UseCase
User
Variant
VariationPoint
WorkPackage

Definition

A human or machine that communicates with a system.
A computer program, or group of programs designed for end users, normally with a graphical user interface. Short for application.
Something that makes it difficult to achieve a goal or a higher quality level.
A point of change. An important aspect of a (non-linear) relation between quality and benefit.
An extensible template for creating objects. A set of objects with certain attributes in common. A category.
A composable part of a system. A reusable, interchangeable system unit or functionality.
A specific combination of variants.
Information stored in a system.
A specific realization or high-level implementation description (of a system part).
The application area of a product with its surrounding entities.
A large user story or a collection of stories.
Something that can happen in the domain and/or in the system.
A releasable characteristic of a product. A (high-level, coherent) bundle of requirements.
A description of how input data is mapped to output data. A capability of a system to do something specific.
An intention of a stakeholder or desired system property.
A concept or thought (potentially interesting).
A defined way to interact with a system.
Something needed to be fixed.
An article in a collection, enumeration, or series.
A descriptive name used to identify something.
An entity that is part of another entity, eg. a field in a class.
A prefix used on a concept to mean beyond or about its own concept, e.g. metadata is data about data.
A prototype with limited functionality used to demonstrate a design idea.
A collection of coherent functions and interfaces.
Something offered to a market.
A distinguishing characteristic or degree of goodness.
A specific way that entities are connected.
A specific version of a system offered at a specific time to end users.
Something needed or wanted. An abstract term denoting any type of information relevant to the (specification of) intentions behind system development. Short for requirement.
A capability of, or support for development.
Something negative that may happen.
A (vivid) description of a (possible future) system usage.
A design of (a part of) a user interface.
A part of a (requirements) document.
Actions performed by systems and/or humans to provide results to stakeholders.
Someone with a stake in the system development or usage.
A mode or condition of something in the domain and/or in the system. A configuration of data.
A short description of what a user does or needs. Short for user story.
A set of interacting software and/or hardware components.
A desired quality level or goal.
A piece of work (that users do, maybe supported by a system).
A word or group of words having a particular meaning.
A procedure to check if requirements are met.
(Development) work awaiting to be completed.
A list of steps defining interactions between actors and a system to achieve a goal.
A human interacting with a system.
An object or system property that can be chosen from a set of options.
An opportunity of choice among variants.
A collection of (development) work tasks.

28 atributes of the metamodel (see paper appendix)

Attributes

Benefit

Capacity

Code

Comment

Constraints

Cost

Damage

Deprecated

Example

Expectation

FileName

Frequency

Gist

Image

Input

Max

Min

Order

Output

Prio

Probability

Profit

Spec

Status

Text

Title

Value

Why

Definition

A characterisation of a good or helpful result or effect (e.g. of a feature).

The largest amount that can be held or contained (e.g. by a resource).

A collection of (textual) computer instructions in some programming language, e.g. Scala. Short for source code.

A note that explains or discusses some entity.

A collection of propositions that restrict the possible values of a set of variables.

The expenditure of something, such as time or effort, necessary for the implementation of an entity.

A characterisation of the negative consequences if some entity (e.g. a risk) occurs.

A description of why an entity should be avoided, often because it is superseded by another entity, as indicated by the deprecated attribute.

A note that illustrates some entity by a typical instance.

The required output of a test in order to be counted as passed.

The name of a storage of serialized, persistent data.

The rate of occurrence of some entity.

A short and simple description of an entity, e.g. a function or a test.

(The name of) a picture of an entity.

Data consumed by an entity,

The maximum estimated or assigned (relative) value.

The minimum estimated or assigned (relative) value.

The ordinal number of an entity (1st, 2nd, ...).

Data produced by an entity, e.g. a function or a test.

The level of importance of an entity. Short for priority.

The likelihood that something (e.g. a risk) occurs.

The gain or return of some entity, e.g. in monetary terms.

A (detailed) definition of an entity. Short for specification.

A level of refinement of an entity (e.g. a feature) in the development process.

A sequence of words (in natural language).

A general or descriptive heading.

An amount. An estimate of worth.

A description of intention. Rationale.

15 realtions of the metamodel (see paper appendix)

Relation

binds
 deprecates
 excludes
 has
 helps
 hurts
 impacts
 implements
 interactsWith
 is
 precedes
 relatesTo
 requires
 superOf
 verifies

Definition

Ties a value to an option. A configuration binds a variation point.
 Makes outdated. An entity deprecates (supersedes) another entity.
 Prevents a combination. An entity excludes another entity.
 Expresses containment, substructure. An entity contains another entity.
 Positive influence. A goal helps to fulfil another goal.
 Negative influence. A goal hinders another goal.
 Some influence. A new feature impacts an existing component.
 Realisation of. A module implements a feature.
 Communication. A user interacts with an interface.
 Sub-typing, specialization, part of another, more general entity.
 Temporal ordering. A feature precedes (is implemented before) another feature.
 General relation. An entity is related to another entity.
 Requested combination. An entity is required (or wished) by another entity.
 Super-typing, generalization, includes another, more specific entity.
 Gives evidence of correctness. A test verifies the implementation of a feature.

What is essential?

└ Methodology

└ Data Collection

Data Collection

reqT-survey.xls - LibreOffice Calc

File Edit View Insert Format Tools Data Window Help

Q1-USAGE In my software development or teaching, this concept is...

0 = never or very seldom used, or not heard of

1 = used, but almost only in an informal, non-persistent way, e.g. in oral communication, emails, chats,

2 = used also persistently to some extent, e.g. repeatedly stored in wikis, documents, reports, models,

Q2-MEANING Do you interpret the word similar as in the suggested definition?

0 = no, I am used to a significantly different meaning of the word

1 = I don't know

2 = yes, I'm used to this or a similar meaning of the word

TYPE	CONCEPT	APPROXIMATE MEANING(S) / DEFINITION	Q1-USAGE	Q2-MEANING
Entity	Actor	A human or machine that communicates with a system.		
Entity	App	A computer program, or group of programs designed for end users, normally with a graphical user interface. Short for application.		
Entity	Barrier	Something that makes it difficult to achieve a goal or a higher quality level.		
Entity	Breakpoint	A point of change. An important aspect of a (non-linear relation between quality and benefit.		
Entity	Class	An extensible template for creating objects. A set of objects with certain attributes in common. A category.		

Q1-USAGE

0 = no

1 = used, but only orally

2 = used, also in writing

Q1 use? = no, orally, also in writing

Q2 agree? = no, don't know, yes

Answered by 15 swedish RE scholars (100% response rate)

<https://github.com/bjornregnell/reqT-survey>

What is essential?

└ Methodology

└ Data Analysis

Data Analysis

Essentiality

Future Work