

Reflective Report

TECHNIQUE-1

Article selection

Articles selected are [1], [2]

-- T. Gorschek and C. Wohlin, "requirements abstraction model," *Required. Eng.*, vol. 11, no. 1, pp. 79–101, 2006.

-- T. Gorschek, P. Garre, S. B. M. Larsson, and C. Wohlin, "Industry evaluation of the Requirements Abstraction Model," *required. Eng.*, vol. 12, no. 3, pp. 163–190, May 2007.

These two articles were selected in order to understand the work of Requirements Abstraction model and its application in industry. Paper 1 discusses the working of requirements abstraction model.

About Requirement Abstraction Model (RAM)

RAM is a model supporting requirement engineering process which helps in making the requirements comparable to one other thereby helping the prioritization process. This model is developed keeping in mind the market-driven development (MDD) where the requirements are more product oriented. Market driven development usually follows an incremental model whose primary goal is to select an optimal set of requirements for different releases (increments). Things like what requirements that a release should contain, what is the cost (resources) needed for implementing these requirements, decision of what features should be available depending on different customers, the point at which quality should be taken into consideration determine the success of the product.

Implementation plan

In order to implement RAM, a guide given in paper 1 is followed. Coming to the requirements, to work on a set of requirements are taken from hospital management system.

RQ1	Get information about the place of the incident for ambulance dispatch
RQ2	Able to locate nearest possible ambulance
RQ3	Allotting the ambulance to respective incident and notify the information
RQ4	Guide the ambulance personal
RQ5	Finding the route to the incident
RQ6	Logging and Reporting of incidents
RQ7	Displaying timing information and error reporting
RQ8	Tracking and monitoring of ambulance
RQ9	Manage Users
RQ10	Registration
RQ11	Patient checkouts
RQ12	Report Generation
RQ13	Database
RQ14	Security
RQ15	Performance Requirements
RQ16	Maintainability
RQ17	Reliability

Execution

Step 1: Specify the requirement

In order to plan the release of the product, requirement prioritization should be done. This process can be simplified when all the requirements are from the same level. Specifying the requirements involves providing Description, reason from including the requirement. The benefit to him or her from the requirement. Risks that are to be considered for the inclusion of requirement.

For each requirement respective description, reason, risks are included in this Step.

RQ1: Get information about the place of the incident for ambulance dispatch	
Description	When the request for the ambulance is made. Hospital management should be able to identify the location based on information given.
Reason/Benefit	This requirement helps to identify the place of the incident and dispatch an ambulance.
Restriction/risks	Information should be accurate enough to locate the place and place should be within the reach of the specific branch of the hospital.

RQ2: Able to locate nearest possible ambulance	
Description	Based on the information given, the system should be able to locate the 3-4 nearest ambulances.
Reason/Benefit	This requirement helps the patient to have least possible waiting time for the ambulance
Restriction/risks	System routing mechanisms should be up to date to eliminate wrong estimates

RQ3: Allotting the ambulance to the respective incident and notify the information.	
Description	The system should be able to allocate ambulances available, in the case of multiple ambulances requests.
Reason/Benefit	Help the hospital management to automatically allocate ambulances based with less human effort
Restriction/risks	Effective cost function should be designed satisfying all stakeholders.

RQ4: Guide the ambulance personal.	
Description	Information about the nearest hospital based on the condition, requirements of the patient should be shared with the ambulance driver.
Reason/Benefit	This requirement helps the driver navigate to the specified hospital without any delay.
Restriction/risks	The navigating mechanism should be reliable.

RQ5: Finding the route to the incident.	
Description	Ambulance driver should be able to navigate to the incident.
Reason/Benefit	Helps the ambulance reach the incident on time
Restriction/risks	The navigating mechanism should be reliable.

RQ6:	
-------------	--

Description	When the request for the ambulance is made. Hospital management should be able to identify the location based on information given.
Reason/Benefit	This requirement helps to identify the place of the incident and dispatch an ambulance.
Restriction/risks	Information should be accurate enough to locate the place and place should be within the reach of the specific branch of the hospital.

RQ7: Displaying timing information and error reporting.	
Description	Hospital management should be able to give the expected time that ambulance can be reached to the incident. Also, if any exception is raised it notify the hospital management staff so that a human attend to issue.
Reason/Benefit	As the situation involves life relying on the automatic process is not so reliable therefore human involvement is necessary.
Restriction/risks	Availability of human personal.

RQ8: Tracking and monitoring of ambulance	
Description	When the request for the ambulance is made. Hospital management should be able to identify the location based on information given.
Reason/Benefit	This requirement helps to identify the place of the incident and dispatch an ambulance.
Restriction/risks	Information should be accurate enough to locate the place and place should be within the reach of the specific branch of the hospital.

RQ10: Manage Users	
Description	The system should be able to manage different hospital management users.
Reason/Benefit	Management ease
Restriction/risks	Personal data should be secured

RQ11: Registration	
Description	Different Hospital Management users should be able to Register in order to use the product
Reason/Benefit	Only privileged users are able to use the system
Restriction/risks	Details provided should be verified

RQ12: Patient checks out	
Description	System should be able to manage the patient check out details such as billing and reports
Reason/Benefit	Removes the need for manually writing billing info, thereby reducing human error.
Restriction/risks	Formats should be specified.

RQ14:Report Generation	
Description	Different reports that are involved in hospital, system should be able to generate them.
Reason/Benefit	Reducing the human error and more accurate
Restriction/risks	Human verification should also be required to confirm the geniuses of the generated reports.

RQ15:Database	
Description	Data which involves billing information, reports should be stored in database
Reason/Benefit	Fast retrieval
Restriction/risks	Databases that is used by the previous system should be compatible with this database so that the previous data can also be migrated.

RQ16:Security	
Description	Patients personal data and his hospital record should be secure and should only be available to others when granted access.
Reason/Benefit	Quality requirement
Restriction/risks	Secure protocols should be continuously upgradeable with changing technology.

RQ17:Performance Requirements	
Description	Performance requirements are mandatory
Reason/Benefit	As the system should give the response as soon as possible.
Restriction/risks	Performance should not neglect security of the user

RQ18:Reliability	
Description	The system should be reliable.
Reason/Benefit	As the system deals with the situation involving human life it should be Reliable

Step 2: Place the requirements in respected level.

This step involves the identifying the abstraction level of the obtained requirements. RAM consists of four abstraction levels

Product Level: Can be compared directly with product strategies.

Feature Level: Features that are supported by the product, not the function

Function Level: Describe what users should be able to do,

Component Level: How the requirement should be implemented.

RQ1	Feature level
RQ2	Functional level
RQ3	Feature level
RQ4	Feature level
RQ5	Feature level
RQ6	Feature level

RQ7	Feature level
RQ8	Feature level
RQ9	Product level
RQ10	Product level
RQ11	Product level
RQ12	Product level
RQ13	Product level
RQ14	Product level
RQ15	Product level
RQ16	Product level
RQ17	Product level
RQ18	Product level

Step 3: Abstraction

Based on the abstraction level of the original requirement respective work-up requirements are generated as per the abstraction levels. Work-up requirements are the custom made requirements that specify the original requirement in different Abstraction level. In the implementation original requirements are bold are respective work-up requirements are identified and specified.

Requirement 1	
Product level	Ambulance Dispatch Support
Feature level	Get information about the place of the incident for ambulance dispatch
Functional level	Details like street name, nearby landmarks.
Component level	Auto fills for nearby landmarks.

Requirement 2	
Product level	Ambulance Dispatch Support
Feature level	Locate Ambulance
Functional level	Able to locate nearest possible ambulance
Component level	Having a map displaying the nearest possible ambulance in that area.

Requirement 3	
Product level	Ambulance Dispatch Support
Feature level	Allotting the ambulance to respective incident and notify the information
Functional level	<ul style="list-style-type: none"> Allocate the ambulance based on the cost function Send information to the ambulance to pick up
Component level	Send a notification to the ambulance driver along with the option to navigate Dispatch is taken as granted only when a driver accepts the request.

Requirement 4	
Product level	Ambulance Dispatch
Feature level	Guide the ambulance personal.
Functional level	Navigation mechanism
Component level	Providing the nearest and best possible route

Requirement 5	
Product level	Ambulance Dispatch
Feature level	Finding the route to the incident
Functional level	Get the map coordinates and plot the route
Component level	Gps device, working internet connection

Requirement 6	
Product level	Logging
Feature level	Logging and Reporting of incidents
Functional level	Log the time taken,
Component level	

Requirement 7	
Product level	Ambulance Dispatch
Feature level	Displaying timing information and error reporting
Functional level	Un clear requirement
Component level	-

Requirement 8	
Product level	Ambulance Dispatch
Feature level	Tracking and monitoring of ambulance
Functional level	Un clear requirement
Component level	-

Requirement 9	
Product level	Registration
Feature level	Manage Users
Functional level	Allow to add delete edit users info
Component level	User's information contains name, age, details, medical record.

Requirement 10	
Product level	Registration
Feature level	Feature to add new users
Functional level	New users based on the type of users all allowed to register in the system
Component level	Simple form gathering the basic necessary information.

Requirement 11	
Product level	Patient checks out
Feature level	Features to provide patient checkout
Functional level	Billing information, reports generated and medical id

Component level	Billing information includes no of days patient resided, type of treatment also the reports of the patient.
-----------------	---

Requirement 12	
Product level	Report Generation
Feature level	Different reports for different type of tests and treatments should be generated
Functional level	Generated test report should include different details necessary for the doctors to review the test report
Component level	Predefined format for entering test results

Requirement 13	
Product level	Data storing
Feature level	Database
Functional level	All the details about the patient should be stored in database which includes reports, billing information, insurance information
Component level	Limited storage for each patient

Lessons learned

Applying this technique helps in understanding the requirement and level of requirements. it's also clearer to work with different abstraction levels which give most of what requirements actually is. Though it mandates issues being a more specific requirement and also need expert judgment in understanding the abstraction level of requirement. Especially in large scale market driven domain where there is a continuous flow of requirements, this help in the prioritization of the requirements and assist release planning process.

Usually, for a company who are performing the requirement engineering process as an ad-hoc method implementing this method requires more time and improved results are realized slowly.

Reflection

Though I have understood the technique I am not able to apply it and use it to the maximum possible extent as I the no of requirements I have considered is small when compared with large scale requirements. Some of the observations presented in [1] such as abstractness of requirements in higher level make it find to find relevant low level requirement and ending up wrong requirements results in failure of the product. Similar problem is face with requirements 7, 8. Also lack of experience in the domain makes it hard to identify other level requirements. Identifying the same product level requirements helps in eliminating the repeatable requirements.

References

- [1] T. Gorschek, M. Svahnberg, A. Borg, A. Loconsole, J. Börstler, K. Sandahl, and M. Eriksson, "A controlled empirical evaluation of a requirements abstraction model," *Inf. Softw. Technol.*, vol. 49, no. 7, pp. 790–805, 2007.

TECHNIQUE-2 QUPER model

Article selection

- [1] R: Berntsson Svensson, B. Regnell (2015) "A case study evaluation of the guideline-supported QUPER model for elicitation of quality requirements", 21st International Working Conference on Requirements Engineering: Foundation for Software Quality (REFSQ'15), Essen, Germany, pp. 230-246, 2015.

– Berntsson Svensson & Olsson "Introducing support for release planning of quality requirements –an industrial evaluation of the QUPER model"

About QUPER

Release planning plays a major role in market-driven development where the organizations compete in the open market. To survive competition organization should be able to bring new, innovative, necessary features as early as possible to market which requires better more formal release planning. Along with value of the features considering quality requirements acts as added advantage. it addresses quality and cost constraints for a requirement

Implementation plan

In order to implement the QUPER model a detailed guide from article 1 is used which involves different steps.

Quality Requirements related to an e-commerce website are taken into consideration. These are collected from various ecommerce sites.

STEP1: Identity candidate Quality requirements

Features	Identified quality requirements
Feature 1	Page generation speed
Feature 2	User experience
Feature 3	User-interface experience
Feature 4	Session time
Feature 5	Navigability
Feature 6	Maintainability
Feature 7	Graphics generation speed
Feature 8	Time to Complete Task
Feature 9	Site Traffic
Feature 10	Average Order Value
Feature 11	Conversion Rate
Feature 12	Bounce Rate

STEP2: Define scale and unit

Identified quality requirements	Scale	Unit
Availability	Page load time	Time in millisecond(minimum better)
User experience	No of products per page	Integer number(optimal number)
User-interface experience	Resolution of the image	1024X1024 format(maximum the better)

Session Time	Average Session Duration	Time in seconds
Navigability	No of links to navigate	Integral number
Maintainability (Ease to add new features)	Maintainability index	Integral number between 1 and 100
Graphics generation speed		Time in milliseconds
Time to Complete Task	Buy a product	Time in minutes
	Buy a product by searching	
Site Traffic	No of unique visitors	Integral value
Average Order Value	Order value	Integral value
Conversion Rate	effectiveness is your store at closing deals	Integral value
Bounce Rate	percentage of visitors who leave your site immediately	Percentage

STEP3: Identity reference levels

Identified quality requirements	Product	Reference levels
Availability	Own product	2000 milliseconds
	Competitor	1700 milliseconds
User experience	Competitor	30
	Own product	35
User-interface experience	Own product	240X240 - 1024X1024
	Competitor	512X512
Session time	Own product	120 seconds
	Competitor	180 seconds
Navigability	Own product	130 links
	Competitor	90 links
Maintainability	Own product	40
	Competitor	-
Graphics generation speed	Own product	850 milliseconds
	Competitor	900 milliseconds
Time to Complete Task a	Own product	3mins
	Competitor	2mins
Time to Complete Task b	Own product	1 mins
	Competitor	1 mins
Site Traffic	Own product	5000 visits/month
	Competitor	4000 visits/month
Average Order Value	Own product	20\$
	Competitor	17\$
Conversion Rate	Own product	6%
	Competitor	4%
Bounce Rate	Own product	30%
	Competitor	45%

There are no standards for the reference levels most of the values are googled based and selected from different sources. Some values are more generic that can be applied to not related to any ecommerce website .

STEP4: Elicit quality breakpoints

Identified quality requirements	Type of quality break points	Quality break points
Availability	Utility	3000
	Saturation	1200
	Differentiation	300
User experience	Utility	50
	Saturation	20
	Differentiation	10
User-interface experience	Utility	240X240
	Saturation	1920X1080
	Differentiation	
Session time	Utility	< 6min
	Saturation	1 min
	Differentiation	3 min
Navigability	Utility	Less than 500
	Saturation	100
	Differentiation	300
Maintainability	Utility	10
	Saturation	90
	Differentiation	50
Graphics generation speed	Utility	2000
	Saturation	500
	Differentiation	Less than 1000
Time to Complete Task a	Utility	5 mins
	Saturation	Less than 1min
	Differentiation	2 mins
Time to Complete Task b	Utility	3 mins
	Saturation	30 sec
	Differentiation	1 min
Site Traffic	Utility	1000
	Saturation	10000
	Differentiation	5000
Average Order Value	Utility	5\$
	Saturation	30\$
	Differentiation	Greater than 20\$
Conversion Rate	Utility	1%
	Saturation	10%
	Differentiation	4%
Bounce Rate	Utility	70%
	Saturation	30%
	Differentiation	50%

Note:

Session time is based on the goals of the company.

Navigability is based on the size of the website values are given for an average spaced website.

Utility, saturation, and differentiation metrics are defend based on the analytics of the various ecommerce websites. Data from different analytics such as google analytics, RJMetrics. Data based on different products are considered to understand the market and guidelines to determine these metric, which is specified in [1] is also followed.

STEP5: Estimate cost barriers

Identified quality requirements	Qref	Present quality	RATIONALE:	cost
Availability	2000	3000	Software necessary	3 weeks
User experience	30	35	Software implementation	1 week
User-interface experience	512X512	240X240	Data base/networking support	4 weeks
Session Time	2 mins	3 min	Software implementation	Less than one week
Navigability	200	300	Redesign and Software Implementation	2 weeks
Maintainability (Ease to add new features)	40	20	Software Implementation	2 weeks
Graphics generation speed	900	2000	Software Implementation	3 weeks
Time to Complete Task a	2min	3 mins	Software Implementation	1 week
Time to Complete Task b	1min	2 min	Software Implementation/hardware implantation	2 Weeks
Site Traffic	6000	3000	Software Implementation (Better AI)	3 Weeks
Average Order Value	20\$	17\$	Better AI-Software Implementation	3 weeks
Conversion Rate	4%	2%	Software Implementation	1 Week
Bounce Rate	40%	50%	-	-

Cost in weeks are given based on the experience in developing similar projects.

STEP6: Set candidate requirements

Based on requirements value and discussions decided targets.

Identified quality requirements	Target	Reference levels
Availability	Good	2000
	Stretch	1500
User experience	Good	25-30
	Stretch	20-25
User-interface experience	Good	512X512
	Stretch	640X640
Session time	Good	2

Navigability	Good	250
	Stretch	220
Maintainability	Good	40
	Stretch	60
Graphics generation speed	Good	1000
	Stretch	Less than 900
Time to Complete Task a	Good	2 mins
	Stretch	Less than 2 mins
Time to Complete Task b	Good	-
	Stretch	-
Site Traffic	Good	4500
	Stretch	5500
Average Order Value	Good	20\$
	Stretch	25\$
Conversion Rate	Good	3%
	Stretch	4%
Bounce Rate	Good	-
	Stretch	-

Lessons learned

Applying this technique helps in understanding the necessity of quality requirements which is necessary to improve a number of satisfied customers. This technique helps in evaluating the current quality of the product, expected quality. Quality that competing products have can also be known by using this process which helps in analyzing of our current level of quality of the product. This process helps in building better products. From the articles referenced it can be seen that it does provide evaluation, but expected results are not seen. The impact can either be positive where improving one quality requirement also improves the quality of other quality requirement or negative where improving one quality requirements decreases the quality of another requirement

Reflection

As the technique is only applied on the small no of requirements cannot complexity of the process for large scale environment cannot be judged. It can be said that increase in a number of requirements, this process becomes more complex if the dependencies of the candidate requirements and quality requirements are considered.

observations

- "The concepts of breakpoints, competitor analysis, and identification of own products quality level provides a greater understanding of the current market segment and why a certain quality level is needed in a particular release." [1]. This seem very valid point and also observed by me, because all the above mentioned points needs the analysis of market and evaluation of current product.

- "The main identified challenge was difficulties to identify and specify the values for the differentiation and saturation breakpoints." [1]. As there is not particular standard value available for determining the breakpoints its l a difficult task to identify them.