Developing a Quality of Experience (QoE) Model for Web Applications

Abstract

Web-based service providers have long been required to deliver high quality services in accordance with standards and customer requirements. Increasingly, however, providers are required to think beyond service quality and develop a deeper understanding of their customers' Quality of Experience (QoE). Though models exist that assess the QoE of Web Application, significant challenges remain in: (a) Defining QoE factors from a Web engineering perspective; (b) quantifying the relationship between so called 'objective' and 'subjective' factors of relevance; and (c) dealing with limited data available in relation to subjective factors. In response, the work here presents a novel model (and associated software instantiation) that integrates factors through Key Performance Indicators (KPI) and Key Quality Indicators (KQI). The mapping is incorporated into a correlation model that assesses the QoE of Web Applications, with a consideration of defining the factors in term of quality requirements derived from web architecture. The data resulting from the mapping is used as input of the proposed model to develop artefacts that quantify and predict QoE using Machine Learning (ML). The development of proposed model is framed and guided by Design Science Research DSR approach with the purpose of enabling providers to more informed decisions regarding QoE and/or to optimise resources accordingly. Though the work is oriented at developing an artefact that has clear utility for practice, the nascent design theory underpinning the work is developed and discussed.

Keywords: Design Science Research, Design Theory, Machine Learning, Quality of Web-based Services, Quality of Experience.

1 Introduction

In this paper, we focus on the production of a DSR artefact that has utility clearly associated with potential practice. Though there has been much discussion on DSR in the literature to-date (e.g.,Hevner & Gregor 2013; Myers & Venable 2014; Lee et al. 2015), there is little in the way of work that evidences the interaction between problem and solution spaces and, thus, shows how design theory evolves and mediates between practice and its conceptual, theoretical and/or philosophical grounds. This is an omission if one accepts that design is what links creativity and innovation and may be seen as creativity deployed to a specific end (Cox, 2005). In addressing this interaction, we

describe the development of a novel model (and associated software instantiation) that assesses the Quality of Experience (QoE) of Web Applications. Doing this highlights the key points that frame our use and subsequent discussion of DSR – design theory, the importance of iteration and the creativity inherent in the process.

In the context of the Internet and electronic networks, QoE has emerged as a multidisciplinary construct that measures the overall service quality perceived by customers (Baraković & Skorin-Kapov, 2013). The measurement of QoE allows a service provider to make an informed decision regarding service delivery and customer satisfaction and to optimise hardware or software resources accordingly (Laghari & Connelly, 2012; Menkovski, Liotta, Sánchez, & Vargas, 2009). The measurement of QoE is usually performed by a combination of what are termed as 'objective' and 'subjective' factors (Mitra, Zaslavsky, & Ahlund, 2011). Objective factors are typically measured by Quality of Service (QoS) parameters (Brooks & Hestnes, 2010), while subjective factors are typically measured by Mean Opinion Score (MOS) tests, which assess how service quality is perceived by customers (Khan, Sun, & Ifeachor, 2012) – typically via an ordinal scale (e.g., representing bad to excellent). A number of issues underlie the measurement of QoE however (Alreshoodi & Woods, 2013; Aroussi & Mellouk, 2014). First, existing QoE factors are defined from a multimedia and network perspective, rather than a Web engineering perspective – arguably leading to naive and inappropriate metrics for web and software quality requirements. Second, facilitating and quantifying the relationship between so called 'objective' and 'subjective' factors of QoE is a noted challenge (Laghari & Connelly, 2012; Schatz, Hoßfeld, Janowski, & Egger, 2013). Third, the MOS process is time-consuming and expensive, thus it is not possible to conduct measurement in real-time (Wang & Wang, 1998). Combined, these issues lead to somewhat of a mismatch between objective and subjective factors both in space and time. Factors are spatially mismatched (to a degree), in that multimedia and network factors operate at lower-levels than MOS factors, and the collection of MOS data is removed in time from actual service use. It was the resolution of these issues in the context of our partners need to improve the quality of service experienced by users that provides the motivation for our work.

With the above challenges in mind, we propose a novel model appropriate for assessing the QoE of Web Applications (called QoEWA from this point) that integrates Key Performance Indicators (KPI) and Key Quality Indicators (KQI). Following a Design Science Research (DSR) approach, the QoEWA is iteratively developed to: (a) Design, build and evaluate an initial artefact that quantifies QoE; and, then, (b) advance the

functionality of the artefact such that it can intelligently measure and predict QoE. The work is undertaken in the context of Web Applications employed within a UK University that wanted to address ongoing issues around service quality. In describing the development of the QoEWA, the paper is structured as follows. Section 2 exposes the primary elements that frame our use of DSR across the paper. Section 3 discusses the awareness of the problem. Section 4 presents an overview of the suggested solution. Section 5 presents the two iterations of the QoEWA, covering the initial solution and its subsequent extension using machine learning techniques. Section 6 examines the outcomes in relation to the developing understanding of DSR. Lastly, Section 7 summarises the work and presents the conclusions.

2   Design Science Research Approach

Broadly speaking, theory in DSR has been discussed in terms of informing the design, as a means of expressing design knowledge and as an outcome of the design instantiation. Design may be informed by kernel theory, generally taken as the underlying knowledge or theory imported from other fields of interest that provide a basis and/or explanation of (aspects) of the design (Gregor & Jones, 2007; Kuechler & Vaishnavi, 2008; Walls, Widmeyer, & El Sawy, 1992). Importantly, however, some have argued that focusing on kernel theory is a potential distraction to artefact design itself (Orlikowski & Baroudi, 1991). Theory has also been considered as a means by which design knowledge is captured, formalised and communicated – in this sense, theory may take a different form from other disciplines (S Gregor & Jones, 2007; Walls et al., 1992). As an outcome, theory can contribute to research and practice bi-dimensionally through originality and utility (Gay & Weaver, 2011). In squaring the circle here, one perspective is that kernel theories can be refined and developed by DSR as an outcome of design (Kuechler & Vaishnavi, 2008) – contributing to a theory's explanatory power or incrementally adding to the lexicon of facts for example. More pragmatically perhaps, Venable (2006) proposes utility theory as a (generalisable) mapping between problem and solution space. He suggests prototypical forms:

- (New) Technology X (when applied properly) will help effectively solve problems of type Y.
- (New) Technology X (when applied properly) will efficiently provide improvements of type Y.
- (New) Technology X (when applied properly to problems of type Y) is more effective than technology Z.

The points above follow the generally accepted view that DSR addresses unsolved problems in unique or innovative ways or solved problems in more effective or efficient ways (A. Hevner, March, Park, & Ram, 2004). In doing this, however, the naïve view of design as a rational and linear process that moves from problem to solution via a set of fixed moves (representing theories, methods, heuristics etc.) should not form the basis of presentation. Problems are 'wicked', designers construct the world(s) that set the dimensions of the problem space and invent their moves (Schön, 1992) and solutions do not optimise some hypothesised utility function - they satisfice (Simon, 1996).

Positively, DSR conceptualised as means of 'learning via the act of building' is one area where there is consensus in the literature (Kuechler & Vaishnavi, 2008). Our observation, however, is that this type of learning is not well-evidenced in published work to-date – design decisions often remain opaque as do iterative/incremental steps in the design process (even though software development methods have evolved to explicitly address them). If design theory is taken in the sense of Gregor and Jones(2007), then more explicit consideration is warranted. This is of particular salience if one accepts the position that the creative aspect in design is not a sudden 'leap' but emerges as a (temporary) bridge from the co-evolution of problem and solution spaces during the design process (Dorst & Cross, 2001). The design process is not linear and work in the solution space often reframes the problem space. Consequently: (a) Design theory is more 'grounded' in practice in a way that we should acknowledge; and (b) iterative and/or incremental learning forms an important part of that theory.

As it is the points above that we focus on, the practical work is described in two design-build-evaluate iterations. We remain mindful of popular process models and guidelines for DSR (A. Hevner et al., 2004; Kuechler & Vaishnavi, 2008; Peffers, Tuunanen, Rothenberger, & Chatterjee, 2008), but use the more generic form proposed by Kuechler & Vaishnavi (2008) for discursive ease re the iterative aspects of the work. This approach also allows for a simplified mapping of the work with the skeleton of a design theory (Gregor & Jones, 2007) covering: (a) The purpose and scope of the theory; (b) constructs; (c) the principles of form and function; (c) artefact mutability; (d) testable propositions; (e) justificatory knowledge (kernel theory); (f) principles of implementation; and (g) each expository instantiation. This approach in the context of the work here is illustrated at Figure 1. On the left-hand side, shows how the proposed model is framed by the DSR process (Vaishnavi, V. and Kuechler, 2004) and guided by the design theory proposed by

Gregor and Jones (2007). On the right-hand side, shows the output of each iteration, addressing four aspects the design, instantiation, application and testing, and evaluation
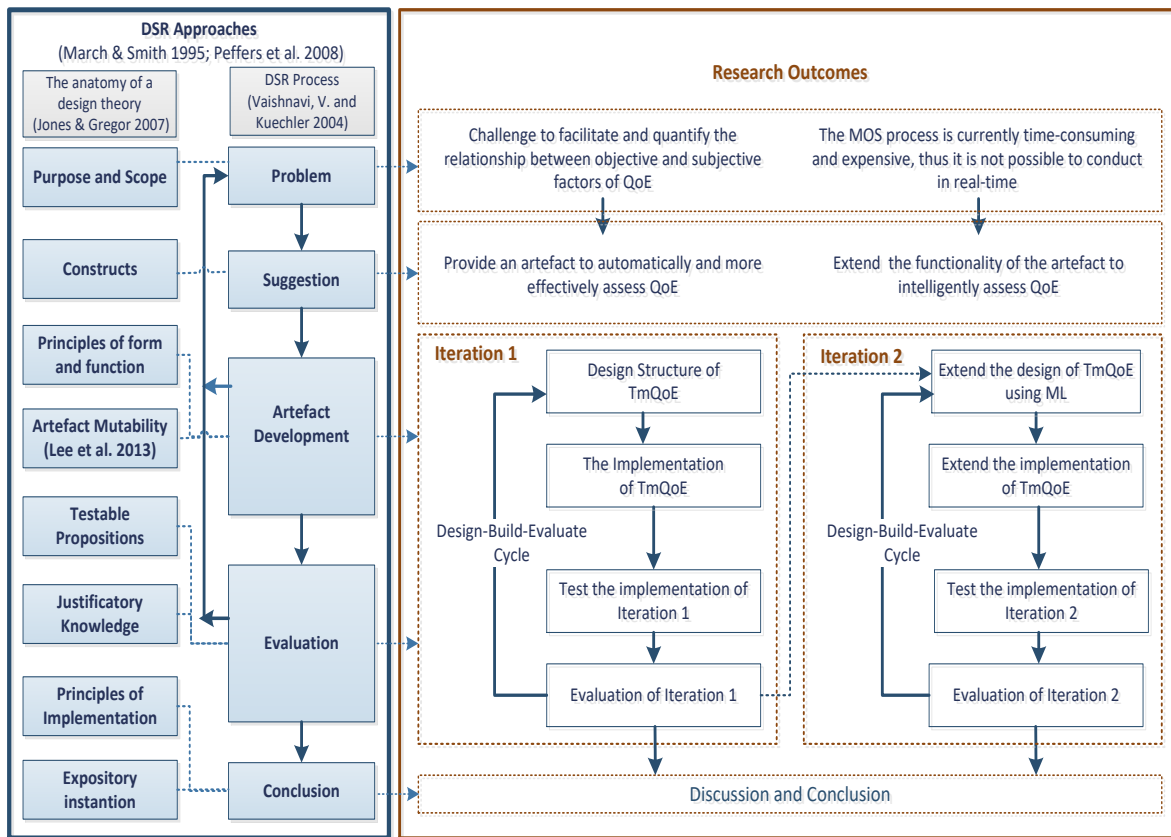

Figure 1: Research approach adopted

3   Awareness of the Problem

The QoE approach was originally introduced for multimedia and network services (Geerts, Moor, & Ketyko, 2010; Laghari & Connelly, 2012), but has subsequently been extended for Web services (ITU-T, 2014; Nguyen, Harris, & Punchihewa, 2013; Skorin-kapov & Barakovic, 2015; Yamauchi, Ito, & Tahara, 2015) and Cloud applications (Cecchet, Sims, He, & Shenoy, 2013; Hobfeld, Schatz, Varela, & Timmerer, 2012). Scholars have also, more generally, drawn attention to loosely-coupled and interoperable services – e.g., Zieliński et al. (2012) who propose an adaptive solution that integrates Service-Oriented Architecture (SOA) with QoS and QoE. General challenges remain however (Baraković & Skorin-Kapov, 2013; Hobfeld et al., 2012; Skorin-kapov, 2012):

- *The quantification of QoE*. A general problem remains in quantifying QoE in the traditional models (Alreshoodi & Woods, 2013; Aroussi & Mellouk, 2014). This problem lies in a lack of explicit mechanisms for defining QoE factors and their

relationships (Fiedler, Hossfeld, & Tran-Gia, 2010; Laghari & Connelly, 2012; Schatz et al., 2013), alongside those necessary for scaling, measuring, prioritising and weighting the QoE factors (Van, Vrije, Pierson, & Lievens, 2008; Zinner, Hohlfeld, Abboud, & Hossfeld, 2010). In addition, it remains the case that most existing QoE models are based on International Telecommunication Union (ITU) factors (ITU-T, 2006, 2014), which are extracted from network and multimedia domains, rather than Web Application ones. Consequently, they fail to define QoE factors pertinent to Web quality requirements and Web architecture design, which are generally derived from alternate standards (ISO/IEC TR 9126-3, 2002; ISO 9241-11, 1998; OASIS, 2012).

- *The prediction of QoE*. The way in which service quality is perceived by customers is generally assessed via Mean Opinion Scores (MOS) (Khan et al., 2012). This is a static survey style approach, which can be time-consuming and expensive – mandating point feedback from end-users regarding their satisfaction with the provided service (Elkotob, Grandlund, Andersson, & Ahlund, 2010). MOS is typically detached from general QoE assessment in both time and space, making it difficult to conduct holistic measurement in real-time (Wang & Wang, 1998). This detachment makes it difficult to reconcile QoE with MOS and, by dint, foreseeing the effect of the former on the latter (Menkovski et al., 2009). Consequently, there is a challenge in understanding the correlation between so-called 'objective' (QoE) and 'subjective' (MOS) factors and their mutual influence so as to improve the prediction of QoE (Aroussi & Mellouk, 2014; Mushtaq, Augustin, & Mellouk, 2012).

For reference, *Figure 2* illustrates how QoE are traditionally extracted from network services. Following the ITU standard, QoE metrics are associated to QoS metrics and collected within the network and/or at the edge nodes of the network (e.g. client/server terminals).

QoE Assessment Process

Network and Multimedia Architecture
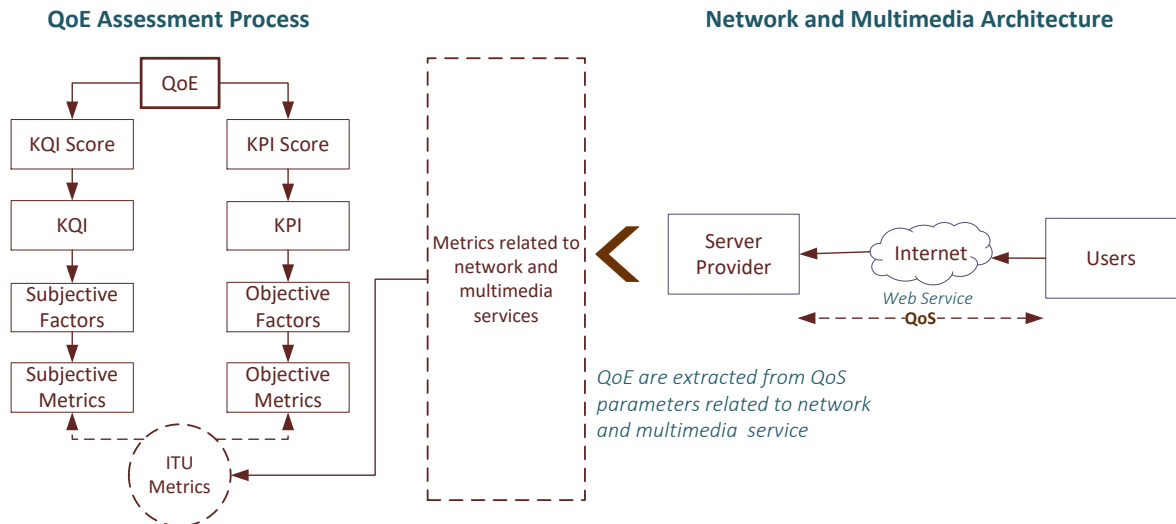
Figure 2: The traditional process of extracting QoE factors

4 The Suggested Solution

In addressing the challenges above, we have used Design Science to develop a novel model called QoEWA (Quality of Experience of Web Application). The model is presented in a high-level form here as orientation for the description of the iterations in the sections that follow. Figure 3 illustrates the architecture of QoEWA and shows how the Web QoE factors are extracted from the main components of web architecture in accordance to the ISO and OASIS quality models (ISO/IEC TR 9126-3, 2002; ISO 9241-11, 1998; OASIS, 2012). Comparing with the traditional assessment process presented in **Error! Reference source not found.**, the proposed one considers quality aspects that particularly assess quality of Web Application, addressing the core modern web architecture. Consequently, QoE are extracted from three layers: (1) Presentation, which includes usability metrics; (2) business logic, which includes functionality and operability metrics; and (3) data, which includes performance, reliability, and availability metrics. The ISO and OASIS quality models are adopted to formulate and process the metrics, as well as measure QoE by computing the values of the KPI and KQI (Baraković & Skorin-Kapov, 2013)

- **KPIs:** are internal indicators derived from the measurements of network resources (e.g. objective measurements of performance, availability, reliability, usability).

- **KQIs:** are external indicators derived from KPIs and associated with different quality aspects that reflect user experience (e.g. subjective measurements of performance, availability, reliability, usability).
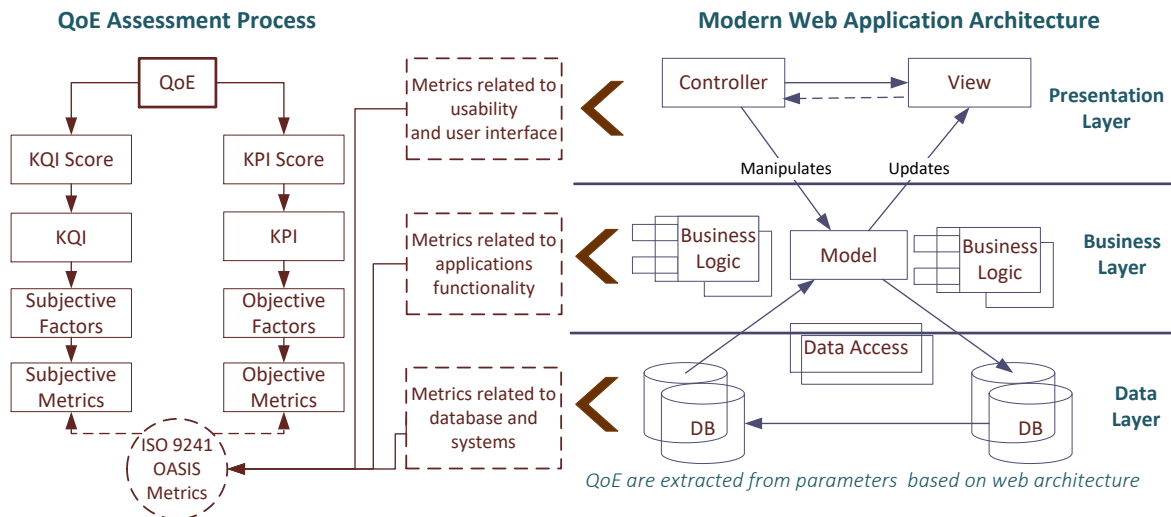
Figure 3: The proposed process of extracting QoE factors

Broadly speaking, the challenges noted above have been approached as follows:

- *The quantification of QoE*. Here we have adopted an 'actual versus target' approach based on the correlation between Key Performance Indicators (KPI - representing so-called objective factors) and Key Quality Indicators (KQI – representing so-called subjective factors) (Kan, Parrish, & Manlove, 2001). The intention here is to measure the gap between the actual measurements and the measurements defined by Service Level Agreement (SLA). The ratio between assessed and target measurements allows the model to compute the actual versus target area, which is ultimately used for quantifying QoE.

- *The prediction of QoE*. Enabling the co-evolution of problem and solution spaces, we employ Machine Learning (ML) techniques to measure the factors typically obtained via Mean Opinion Score (MOS) tests to assess how service quality is perceived by customers (Khan et al., 2012) –

i.e., the correlation between KPI and KQI enables the model to predict the unknown KQI from the known KPI (Alreshoodi & Woods, 2013) (reference withheld for review,2017).

5 Setting the Scene of the Development of QoEWA

In moving from problem space to solution space, key design decisions were required in relation to the constructs that define and map the so called 'objective' and 'subjective' factors. Consequently, we defined a set of Key Performance Indicators (KPIs) comprising traditional quality factors (F1,F2,F3,F4, and F5) derived from standard models such as ISO 9241-11 (1998), ISO/IEC TR 9126-3 (2002), and OASIS (2012), alongside a set of usability factors (F6,F7,F8,F9) derived from existing models (Mifsud, 2015; Seffah, Donyaee, Kline, & Padda, 2006). In addition we included additional non-technical factors (F10 and F11) derived from QoE ecosystem models (Laghari & Connelly, 2012; Skorin-kapov, 2012). The factors are listed in Table 1 and each factor has objective and subjective formulas. The input of the objective formulas is extracted from operational data sources (e.g., applications, middleware and database), whereas the source input of the subjective formula is extracted from MOS assessment data, which is stored in the Customer Relation Management (CRM) system.

Table 1 : List of the objective and subjective factors

Ref	KPI/KQI	Objective metrics (ISO/IEC TR 9126-3, 2002; ISO 9241-11, 1998; OASIS, 2012)	Objective formula	Subjective metrics (formulated subjectively based on the objective metrics defined by ISO and OASIS quality models)
F1	Performance	m1: Max. Completed requests m2: Unit Time	$F1 = (m1/m2)$ (Max Throughput)	User satisfaction with the time taken to send a request and receive a response from their terminals or web page.
F2	Reliability	m3: Number of correct implemented Items m4: Total number of compliance items	$F2 = (m3/m4)$ (compliance)	User satisfaction with the number of successful performed tasks over a period of time.
F3	Availability	m5: down-time m6: Unit-time	$F3 = 1 - (m5/m6)$	User satisfaction with the availability of application and the operational uptime.
F4	Accessibility	m7: Number of acknowledgement messages m8: Number of request messages	$F4 = (m7/m8)$	User satisfaction with the ratio of the successful returned acknowledgements after requesting tasks.
F5	Success-ability	m9: Number of responses m10: Number of requests	$F5 = (m9/m10)$	User satisfaction with the ratio of requests (sent by user) to responses (performed by server provider).
F6	Learnability	m11= Number of functions described m12= Total number of functions provided	$F6 = (m11/m12)$	User satisfaction with simplicity and the functions implemented with help facility and/or documentation.
F7	Operability	m13: Number of instances of operations with inconsistent behaviour m14: Total number of operations	$F7 = 1 - (m13/m14)$ (Operational consistency)	User satisfaction with the number of operations (e.g. forms layout) with consistent behaviour.
F8	Usability (Effectiveness)	m15: Number of tasks completed successfully m16: Total number of tasks	$F8 = (m15/m16)$ (Completion Rate)	User satisfaction with number of tasks completed successfully in a given time.

F9	Usability (Efficiency)	m17= Number of correctly implemented items related to efficiency compliance confirmed in evaluation m18= Total number of compliance items	$F9 = \left(\frac{m17}{m18}\right)$ (Compliance rate)	User satisfaction with the time taken to complete a number of tasks from their terminals in accordance to the compliance requirements.
F10	Responding to users	m19: Time taken to respond to user m20: Max time to respond as in SLA	$F10 = \left(\frac{m19}{m20}\right)$	User satisfaction with time taken to receive a response from customers support.
F11	Professionalism	m21: Time taken to fix issue for app. m22: Max time to fix issues as in SLA	$F11 = \left(\frac{m21}{m22}\right)$	User satisfaction with the quality of the technical support received from customer services.

Figure 4 describes the process of computing the KPIs and illustrate how they are correlated and mapped to quantify and predict QoE.

- *KPI Assessment.* Performance indicators vary from business-to-business and from technology-to-technology, depending on the scope and purpose of the service (Al-Moayed & Hollunder, 2010). In this paper, this process is based on the objective formulas as follows:

  KPI Score = {F1obj, F2obj, F3obj, F4obj, F5obj, F6obj, F7obj, F8obj, F9obj, F10obj, F11obj}, where 'obj' indicates objective

    (1)

- *KQI Assessment.* Performance indicators here assess subjective factors most commonly performed by a MOS test (ITU-T, 2006), which is generally based on an ordinal scale of five-points: (1) bad; (2) poor; (3) fair; (4) good; (5) excellent. KQI is described as:

  KQI Score = {F1sub, F2sub, F3sub, F4sub, F5sub, F6sub, F7sub, F8sub, F9sub, F10sub, F11sub}, where 'sub' indicates subjective   (2)

- *Mapping process*: The mapping between aggregate KPIs and KQIs is an essential and important task in the life-cycle of QoE assessment (Hobfeld et al., 2012) as it defines the relationship between the objective and subjective metrics (Fiedler et al., 2010). A design decision was taken here to adopt the complementary approaches of correlation analysis and two-dimensional gap analysis. Mapping thus becomes a task that essentially combines a set of KPIs with corresponding KQIs with a specific KQI, expressed as:

  QoE = {KPI, KQI}, where KPI and KQI are represented as nominal values (3)

As a result of the mapping, the KPI and KQIs are correlated to determine the degree of association between the objective and subjective aspects of QoE. This facilitates the development of the artefacts, which are iteratively constructed through three iterations framed by the DSR methodology.
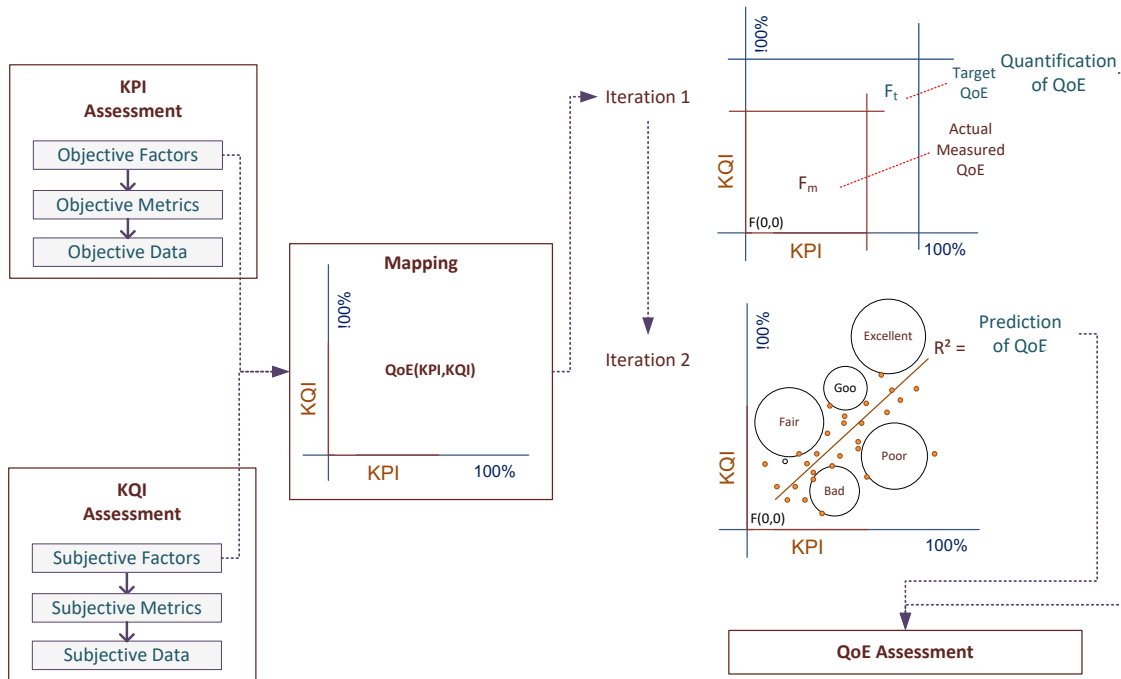

Figure 4: The conceptual design structure of QoEWA

5.1 Iteration 1: Quantification of QoE

5.1.1 Design of Iteration 1

To quantify the relationship between the objective and subjective factors of QoE, a design decision was taken to systematically explore the correlation between KPI and KQI measurements, corresponding to points on a positive coordinate axis as shown in Figure 5 – where the *x*-axis represents the measurement of the objective factors and *y*-axis represents the measurement of the subjective factors. The coordinates of the origin (0, 0) indicate the initial points of (KPI, KQI) indicators. Each increment on the *x* and *y* axes represents the actual measured values of KPIs and KQIs. The default maximum values on both the *x* and *y* axes are considered as target values that are variables and based on business-oriented parameters defined within a Service Level agreement (SLA) – i.e., they can be set to different service provider's requirement and standards. The correlation between the measured values of KPI and KQI forms a square that is expressed by *Fa (actual)*; the correlation between the targets forms a square that is expressed by *Ft*

(target). The gap between *Fa* and *Ft* is then measured by Actual-Versus-Target approach (Kan et al., 2001), which has the ability to determine the relative strength and weakness of a particular observation and make a comparison judgment between what is actually measured (*Fa*) and what it is targeted (*Ft*). The ratio of *Fa* and *Ft* expresses the QoE value, which is translated into a quantifiable form as shown in the formulas that follow.
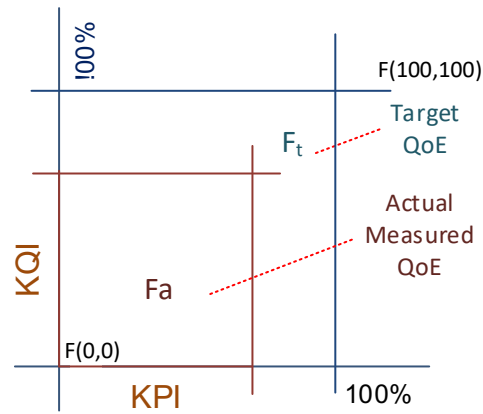
Figure 5: Actual-Versus-Target approach for quantifying QoE

$$F_t = F_{t(KPI)} * F_{t(KQI)} \qquad (1)$$

$$F_a = F_{a(KPI)} * F_{a(KQI)} \qquad (2)$$

$$QoE = \sum_{k=1}^{n} \frac{F_a}{F_t} \qquad \% \qquad (3)$$

The comparison between actual and target assesses the compliance of service quality with user needs. Importantly, within this design, each KPI/KQI factor (e.g. performance, reliability, availability, etc.) can be evaluated separately and the sum provides the overall QoE value. This enables a service provider to determine the factor that may influence QoE and, as a result, prioritise their importance (Schumacher et al., 2010).

In terms of consistency between the KPI and KQI values, the design decision taken was to systematically evaluate the consistency between them (as conceptualised by Yi et al. (2012) and Martinez (2014). The effects of this decision are that the consistency level is high when: (1) The correlation between KPI and KQI is positively strong; and (2) the gap between the measured values of KPI and KQI is close. For example, two measures with identical values of KPIs and KQIs will be correlated the most and have the same interval consistency.

5.1.2   Instantiation of Iteration 1

The implementation of the model adopted an agile development process, breaking the development activities into step-by-step increments with minimal advance planning. This is aligned with DSR methodology (Vidgen, Donnellan, Matook, & Conboy, 2011) as a practical combined approach to developing an efficient software system (Aaen, 2008). The functionality of QoEWA, which maintains the measurements of QoE was implemented by a Model-View-Controller (MVC) based Web application combined with 3-tier architecture as follows:

- *Data Layer***:** It is developed to manage and maintain data that is extracted from sessions. Data was extracted from a middleware server and CRM system through via a relational database that contains tables and views created in an Oracle. KPI data extracted from the middleware server was processed and written into a database table for objective data called (obj_table). KQI data extracted from the CRM system was processed and written into a database table for subjective data called (sub_table).

- *Business Logic Layer.* The logic of the QoEWA model (as described above) was implemented as a set of business rules implemented on Oracle PL/SQL and Java. In balancing brevity while aspiring to the ideals of DSR communication (e.g., Hevner & Gregor, 2013; Pfeffers et al., 2008) we provide a logical presentation of the static and dynamic aspects of those rules via a simplified UML Class model at Figure 6 and UML Sequence model at Figure 7.

- *Presentation layer***.** A Graphical User Interface (GUI) was implemented using an Oracle Application Development Framework (Oracle ADF) to provide a means for systems administration staff to interact with the QoEWA model.
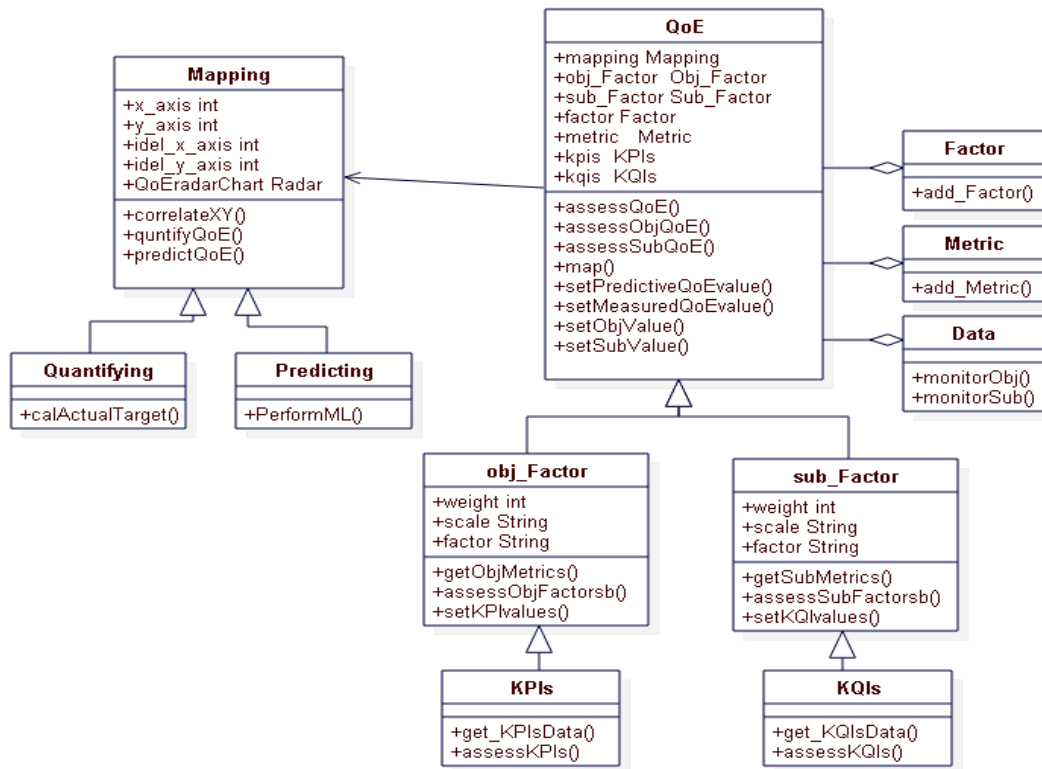
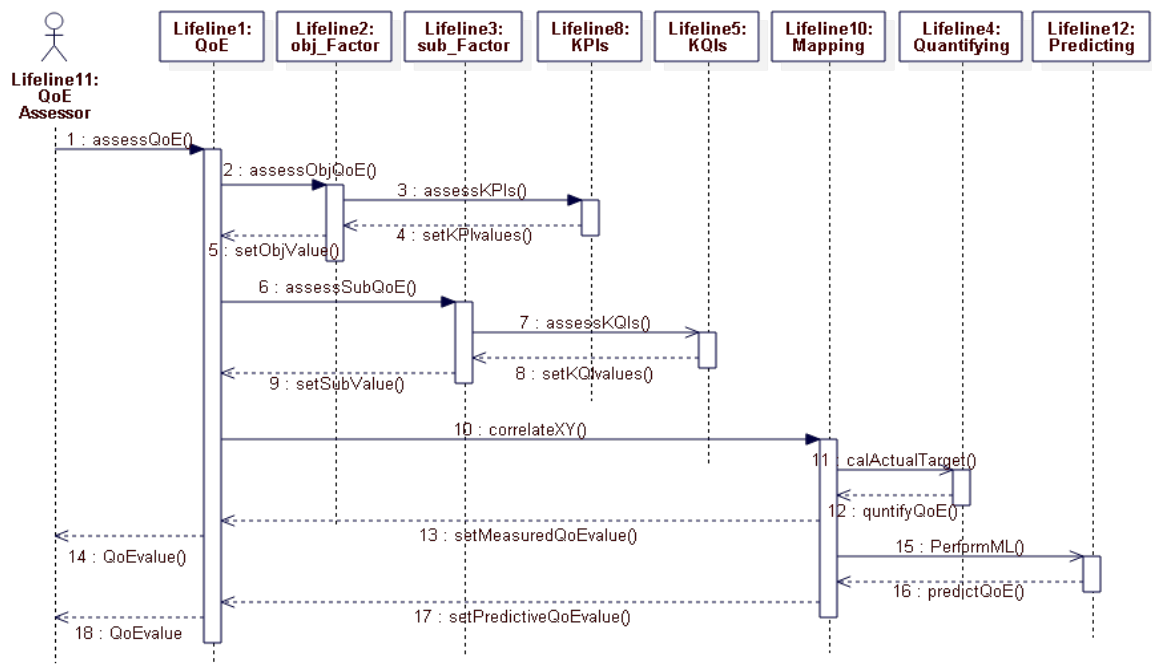**Figure 6: UML Class model for QoEWA instantiation**

Figure 7: UML Sequence model for illustrating the assessment scenario

5.1.3   Application and Testing of Iteration 1

As noted in the introduction, the QoEWA was employed within a UK University that wanted to address ongoing issues around service quality. Data was drawn from 4

applications, which were developed in-house and provide services for accommodation, student centre activities, student registration, and campus security. These applications are served by two Web servers running Oracle WebLogic, which manage session information on end users who provide support for students (where the session information is shaped by the metrics defined earlier in Table 1). The session information itself is stored in a back-end CRM system (which can be drawn upon for subsequent analysis by system managers, at whom the model is targeted). The resulting dataset has nearly 100,000 sessions collected over a 12-month period from 335 users across the 4 different applications, the details of which are shown in Table 2. For transparency, we note that one of the authors here is responsible for systems management.

Table 2. Data Collected

Application	User Percentage	Session Percentage
1 Accommodation	23	11
2 Student Centre	24	19
3 Student Registration	18	21
4 Campus Security	35	49

Table 3 provides summary statistics for the KPI/KQIs, each of which is weighted according to its importance (Behkamal, Kahani, & Akbari, 2009). The scale of each factor is expressed as a percentage of the ratio between the difference between the measured value and the target value.

Table 3: Summary of the KPI and KQI values obtained from the dataset

|		F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11
	Scale	%	%	%	%	%	%	%	%	%	%	%
	No of issues logged into Remedy system (CRM)	17	16	17	13	15	16	13	13	13	14	13
	Weight	0.11	0.10	0.11	0.08	0.09	0.10	0.08	0.08	0.08	0.09	0.08
KPI	Mean	60.48	58.94	58.68	59.67	57.63	56.65	56.17	53.80	54.95	57.15	55.11
	Maximum	90.86	88.56	88.17	89.66	86.59	85.12	84.39	80.83	82.56	85.87	82.80
	Minimum	19.75	19.25	19.17	19.49	18.82	18.50	18.35	17.57	17.95	18.67	18.00
	Standard Deviation	22.61	22.03	21.94	22.31	21.54	21.18	21.00	20.11	20.54	21.36	20.60
KQI	Mean	61	59.19	55.98	56.93	52.09	56.89	56.41	51.32	52.42	57.39	52.57
	Maximum	99	98.19	92.86	94.43	86.40	94.37	93.57	85.14	86.95	95.20	87.21
	Minimum	19.75	19.25	18.21	18.52	16.94	18.50	18.35	16.69	17.05	18.67	17.10
	Standard Deviation	23	22.79	21.55	21.92	20.06	21.90	21.72	19.76	20.18	22.10	20.24

Building upon the above inputs, the instantiation of the QoEWA was tested in two ways. First, to provide a benchmark in relation to the state-of-art, the correlation between the overall score of KPIs and KQIs was examined based on the assumption that a strong positive correlation indicates an excellent relationship between the objective and subjective factors (Upadhyaya, Zou, Keivanloo, & Ng, 2014) – examining the relationship between service quality and user satisfaction. Second, the data was run in the context of

the full QoEWA model, quantify QoE by comparing the actual values against the target values. A small gap between KPI and KQI values indicates consistency between KPI and KQI, which means that the feedback obtained by user is consistent with (technical) quality of the service.

For the first test, Table 4 summarises the R squared value of each factor (F1-F11). The result shows that there is a strong positive correlation between the objective and subjective factors, ranged between $R^2= 88$ and $R^2= 97$. Figure 8 shows the overall correlation which is formulated by KPI and KQI. The overall result shows a high strong correlation with $R^2= 96$.

Table 4: Summary of R squared values for each factor

Key	Calculation	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11
KPI	Mean	60.48	58.94	58.68	59.67	57.63	56.65	56.17	53.80	54.95	57.15	55.11
KQI	Mean	61	59.19	55.98	56.93	52.09	56.89	56.41	51.32	52.42	57.39	52.57
	R2	90	96	97	88	96	95	94	97	97	96	93
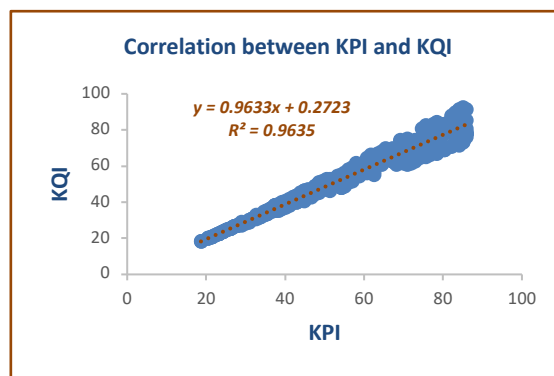
Figure 8: Correlation between the measurements of KPIs and KQIs

For the second test, Table 5 summaries the actual-target values of each factor (F1-F11). The result shows that there is small gap between KPI and KQI thus, according to the assumption above, the result indicates a high level of equality and consistency between KPI and KQI. Figure 9 shows the actual and target values. Hence, the quantitative value of QoE is computed as:

$$Fa= 57.20 * 55.60 \qquad (1)$$

$$Ft= 85.90 * 85.90 \qquad (2)$$

$$QoE = (57.2 * 55.6) / (85.9 * 85.9) \qquad (3)$$

Thus the overall QoE is 44%. Importantly, the target value of each indicator should be strategically-driven (Eckerson, 2009) by the maximum value that the indicator can reach

with respect to available resources – e.g., in this test, the maximum level of performance that can be achieved for the ratio of the completed requests and the unit time is 91%.

Table 5: Summary of the actual-target values

Value	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11
Target value of the calculation of (KPI,KQI)	91.00	88.00	88.00	89.00	87.00	86.00	85.00	81.00	82.00	86.00	82.00
Actual value of the calculation of (KPI,KQI)	60.74	59.065	57.33	58.3	54.86	56.77	56.29	52.56	53.685	57.27	53.84
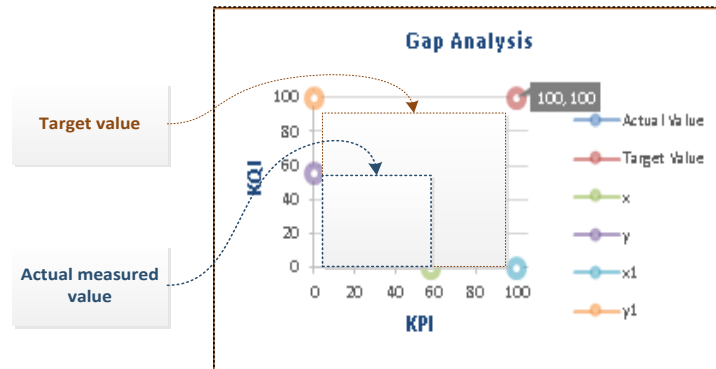
Figure 9: Gap analysis based on Actual-Versus-Target approach

After performing both tests, the result was validated by splitting the dataset into four, each subset consisting of a particular web application used by particular users. The correlation between KPI and KQI was examined for each subset and, across the four tests, a strong positive correlation between the KPI and KQI and a relatively constant QoE value (of between 44% and 46%) was observed.

5.1.4 Learning from Iteration 1

Reflecting on the outcomes of the testing process, it is argued that the actual-versus-target area obtained from the correlation enables a more holistic measurement of QoE, drawing both so-called objective and subjective indicators closer together and providing a means of analysing the difference between values set in a Service Level Agreement (SLA) and those actually perceived by people using services 'in anger'. The model is flexible in that indicators can be examined from an individual or aggregated perspective and, importantly, that they can be determined and/or contextualised to given applications/domains.

Nonetheless, evaluation of the KQIs remains time-consuming and expensive as the model stands. This is because the values are given as outcomes of the polling of users on their satisfaction, without considering factors that may have a strong influence on QoE such as context, previous experience, and scalability (Mirkovic, Vrgovic, Culibrk,

Stefanovic, & Anderla, 2014). The QoEWA still, therefore, holds the limitations of traditional MOS approaches. In developing Iteration 1, approximately 60% of users did not provide feedback (especially those who work in a busy and customer-facing environment). Further, it was observed that the majority of users who did provide their feedback did so only once, limiting the tracking dynamic tracking of user satisfaction.

This is an issue that has not gone unnoticed and research exists re dynamic models that evaluate user experience from a QoE perspective in a sequential manner – e.g., Mitra et al. (2011). In addition, researchers have started to look toward models that enable MOS to be intelligently classified and predicted (Balachandran et al., 2013; Khan et al., 2012; Menkovski, Exarchakos, & Liotta, 2010; Menkovski et al., 2009). Most QoE prediction models are based on Machine Learning (ML) and use an inductive supervised learning approach, where the predictive rules are generated from particular observation or learning - see Aroussi & Mellouk (2014) for a review. Consequently, we embarked on further work to address the limitation.

5.2 Iteration 2: Prediction of QoE

5.2.1 Design of Iteration 2

Following the state-of-the-art, and given that the data is classified and labelled in Iteration 1, a design decision was taken to adopt a supervised learning approach. Based on a review of relevant research in QoE prediction (Aroussi & Mellouk, 2014; Mushtaq et al., 2012), five supervised learning algorithms for comparison: Decision Tree J48 (DT), Naive Bayes (NB), Sequential Minimal Optimization (SMO), Instance-based learning with parameter K and (IBK) and Random Forest (RF). Our design provides the model with a classifier for predicting the values of the subjective metrics (Mushtaq et al., 2012) using the common five-point MOS scale discussed earlier. The classifier was trained with KQI data from the previous iteration, drawn from a Customer Relation Management (CRM) system (called Remedy).

The theory behind the design combines top-down and bottom-up approaches, using a known value to predict the unknown value of QoE (Alreshoodi & Woods, 2013). The top-down aspect draws on subjective data collected from user-side related to KQIs; the bottom-up aspect draws on objective data collected from the server-side related to KPIs. Both approaches can be applied alongside each other in complementary ways, depending on the data availability and the degree of association between QoE parameters – here, the known KPIs enable the model to predict and estimate the unknown KQIs. Figure 10

illustrates the correlation between the top-down and bottom-up, where x-axis represents the known data (KPIs), and the y-axis represents the unknown data (KQIs).
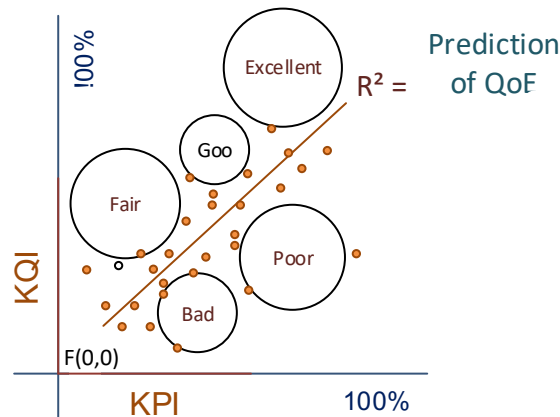
Figure 10: Machine Learning (ML) apoproach for predicting QoE

The training dataset includes KPI and KQI data, in which the KPI is expressed as the independent variable (feature) , whereas the KQI is expressed as the dependent variable (target) (Witten, Frank, & Hall, 2011). Consequently, 11 dependent variables correspond with 11 independent variables. Each independent variable is incorporated into the whole set of the dependent variables to be used as an input of the utilised algorithms. The output of each algorithm is generated as rules that predict the target values of the KQis.

5.2.2 Instantiation of Iteration 2

The Waikato Environment for Knowledge Analysis (WEKA) tool was utilised to implement the chosen ML classifiers. WEKA is a popular research tool but can also be used for commercial applications under a General Public License (GPL). The tool covers the majority of machine learning and data mining tasks such as filtering, classification, clustering, and ranking (Witten et al., 2011).

Dataset Structure

The training dataset used in Iteration 1 was structured and extracted into an Attribute Relation File Format (ARFF), which is provided by Weka. ARFF has two sections Header and Data (Witten et al., 2011). In the Header section, the objective factors are defined as features, whiles the subjective factors are defined as targets (as shown in Figure 11). The Data section contains the raw data of the training dataset, which is subsequently filtered and transformed by ARFFLoader.

Figure 11: Sample of the dataset

Training and Knowledge Flow

The dataset is transformed and processed by the Knowledge Flow interface, which provides the components required to configure the inputs and outputs of the chosen ML algorithms (e.g. DT, NB, SMO, IBK, and RF). Each target attribute is assigned with the features from {fo1, fo2... fo11} to provide a batch class as an input to the algorithms. The output of the algorithms is presented by a text viewer and model performance chart and includes the rules (a set of nested if-else statements) that are used to predict the KQI scores of users who did not provide their feedback. Figure 12 illustrates the developed Knowledge Flow.
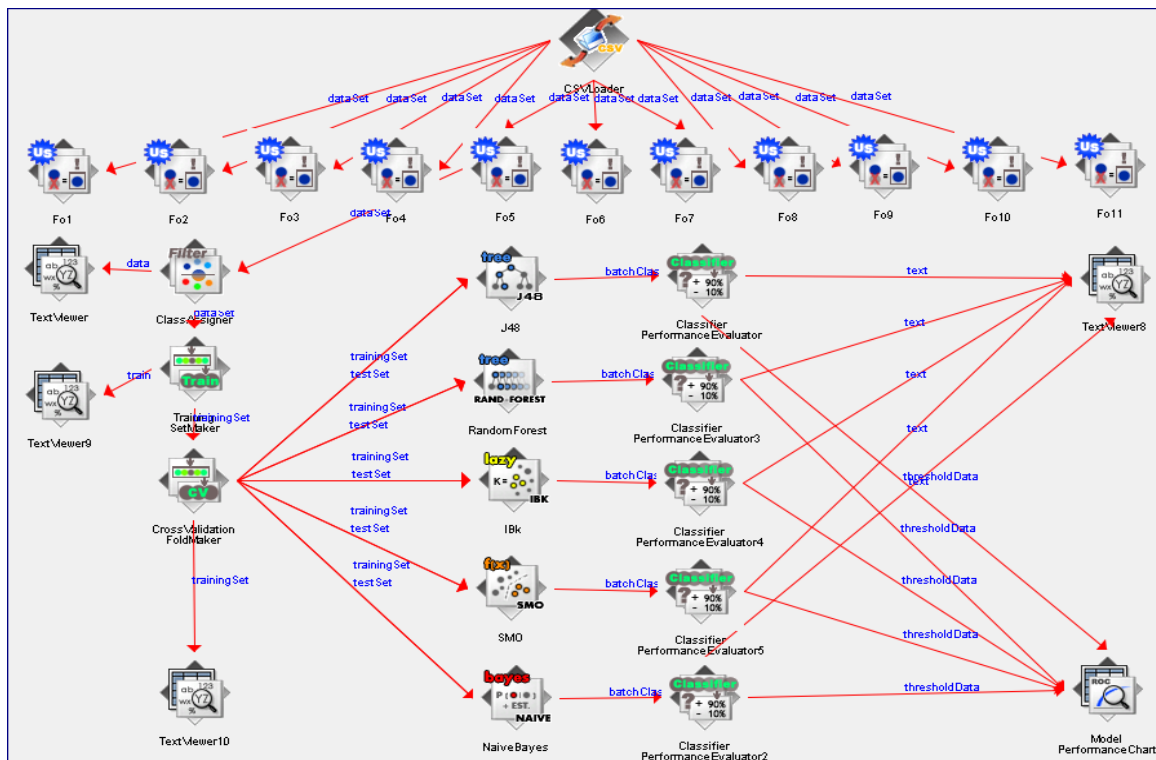

**Figure 12: Machine Learning Knowledge Flow**

5.2.3 Application and Testing of Iteration 2

The application of the extended QoEWA is illustrated by a test that trains the model on real data obtained from previous feedback on service quality – providing a comparative assessment of machine learning algorithms for predicting the poll scores. To minimise bias, a 10-fold cross-validation test was employed to evaluate the results of the applied algorithm, which is a widely adopted approach (Menkovski et al., 2009; Mushtaq et al., 2012). A measure of the Correct Classified Instances (CCI) was used to show the best performing algorithm, alongside the Mean Absolute Error rate (MAE) as a means of comparing algorithms (Menkovski et al., 2009; Mushtaq et al., 2012). Table 6 shows the classification of each of the labelled vectors in relation to the standard MOS scale of each target (from Fs1 to Fs11).

Table 6: Summary of the labelled and classified vectors

Label	Fs1	Fs2	Fs3	Fs4	Fs5	Fs6	Fs7	Fs8	Fs9	Fs10	Fs11
Excellent	93	81	52	58	23	58	52	17	27	62	27
Good	87	89	108	106	120	106	108	121	116	102	116
Fair	73	80	83	79	90	79	83	95	92	80	92
Poor	64	67	67	67	70	67	67	70	68	69	68
Bad	18	18	25	25	32	25	25	32	32	22	32
Total	335	335	335	335	335	335	335	335	335	335	335

A knowledge flow (Figure 12) was then run for each target and the results of CCI and MAE are summarised in Table 7 with the outcomes averaged and expressed in
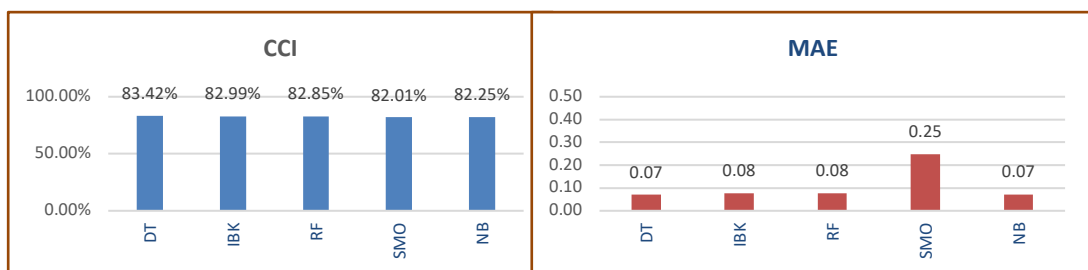
Figure 13. Outcomes show that the DT algorithm has the minimum absolute error rate (with value 0.07) and, in terms of the correctly classified instances, is the best classification algorithm amongst the set employed (with value 83.4). The difference between the five algorithms evaluated is small, however, with a standard deviation of 0.077 for MAE and 0.05% for CCI. Nonetheless, the results confirm the findings of Mushtaq et al. (2012), which observe that DT and RF have higher performance and accuracy than NB, SMO, IBK.

Table 7: Summary of CCI and MAE of each classifier

| Classifier	Test	Fs1	Fs2	Fs3	Fs4	Fs5	Fs6	Fs7	Fs8	Fs9	Fs10	Fs11
DT	CCI	0.83	0.79	0.82	0.80	0.90	0.80	0.81	0.88	0.87	0.80	0.87
	MAE	0.07	0.08	0.06	0.11	0.07	0.11	0.08	0.05	0.07	0.07	0.07
IBK	CCI	0.81	0.81	0.79	0.80	0.90	0.80	0.79	0.89	0.88	0.79	0.88
	MAE	0.09	0.09	0.09	0.09	0.06	0.09	0.09	0.06	0.06	0.09	0.06
RF	CCI	0.81	0.81	0.79	0.79	0.89	0.79	0.79	0.88	0.88	0.79	0.88
	MAE	0.09	0.09	0.09	0.09	0.05	0.09	0.09	0.06	0.06	0.09	0.06
SMO	CCI	0.82	0.81	0.80	0.75	0.89	0.75	0.80	0.88	0.87	0.79	0.87
	MAE	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25
NB	CCI	0.83	0.81	0.81	0.78	0.89	0.78	0.81	0.88	0.84	0.78	0.84
	MAE	0.07	0.08	0.08	0.08	0.05	0.08	0.08	0.05	0.06	0.08	0.06
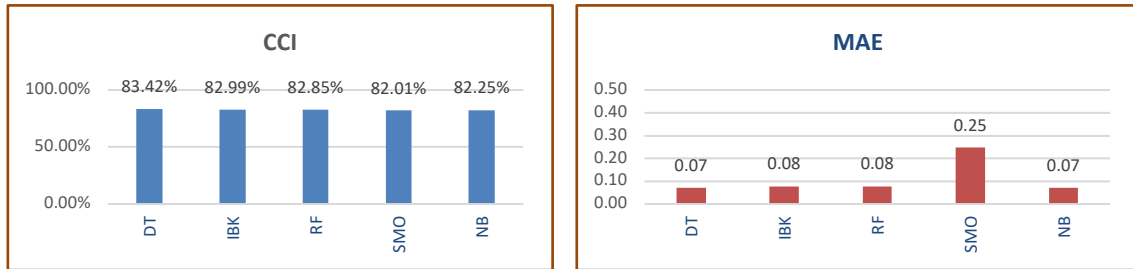
Figure 13: CCI and MAE results

Table 8 shows the efficiency of each algorithm as evaluated by the standard measures of True Positives (TP), True Negatives (TN), precision, recall, F-measure and the ROC area. The results for each algorithm are averaged and expressed in Figure 14. Outcomes show that the DT algorithm performs best. In general, however, evaluation of the five algorithm shows a large ROC area lies between 0.0931 and 0.958 - this indicates that all algorithms predict effectively on the extracted training dataset. Given prior results, the DT algorithm was taken to be the most efficient algorithm.

Table 8: The efficiency of the applied algorithms

Classifier	Test	Fs1	Fs2	Fs3	Fs4	Fs5	Fs6	Fs7	Fs8	Fs9	Fs10	Fs11
DT	TP Rate	0.83	0.79	0.82	0.80	0.90	0.80	0.81	0.88	0.88	0.80	0.88
	FP Rate	0.06	0.07	0.07	0.07	0.05	0.07	0.07	0.05	0.04	0.07	0.04
	Precision	0.83	0.79	0.82	0.81	0.85	0.81	0.80	0.85	0.87	0.81	0.87
	Recall	0.83	0.79	0.82	0.80	0.90	0.80	0.81	0.88	0.88	0.80	0.88
	F-Measure	0.82	0.79	0.81	0.80	0.87	0.80	0.80	0.87	0.87	0.79	0.87
	ROC Area	0.98	0.98	0.97	0.98	0.97	0.92	0.95	0.97	0.95	0.93	0.95
IBK	TP Rate	0.81	0.81	0.79	0.80	0.90	0.80	0.79	0.89	0.88	0.79	0.88
	FP Rate	0.07	0.07	0.08	0.07	0.05	0.07	0.08	0.05	0.05	0.07	0.05
	Precision	0.80	0.80	0.78	0.80	0.85	0.80	0.78	0.85	0.86	0.79	0.86
	Recall	0.81	0.81	0.79	0.80	0.90	0.80	0.79	0.89	0.88	0.79	0.88
	F-Measure	0.80	0.81	0.78	0.80	0.88	0.80	0.78	0.87	0.87	0.79	0.87
	ROC Area	0.95	0.94	0.94	0.94	0.97	0.73	0.94	0.96	0.97	0.94	0.97
RF	TP Rate	0.82	0.81	0.79	0.79	0.89	0.79	0.79	0.88	0.88	0.79	0.88
	FP Rate	0.06	0.06	0.07	0.07	0.05	0.07	0.07	0.06	0.05	0.07	0.05
	Precision	0.81	0.81	0.79	0.79	0.85	0.79	0.79	0.85	0.87	0.79	0.87
	Recall	0.82	0.81	0.79	0.79	0.89	0.79	0.79	0.88	0.88	0.79	0.88
	F-Measure	0.81	0.81	0.79	0.79	0.87	0.79	0.79	0.86	0.88	0.79	0.88
	ROC Area	0.96	0.94	0.95	0.95	0.98	0.95	0.95	0.97	0.98	0.95	0.98

SMO	TP Rate	0.82	0.81	0.80	0.75	0.89	0.75	0.80	0.88	0.87	0.79	0.87
	FP Rate	0.06	0.06	0.09	0.10	0.05	0.10	0.09	0.05	0.06	0.07	0.06
	Precision	0.84	0.82	0.70	0.64	0.85	0.64	0.70	0.85	0.81	0.78	0.81
	Recall	0.82	0.81	0.80	0.75	0.89	0.75	0.80	0.88	0.87	0.79	0.87
	F-Measure	0.81	0.80	0.74	0.69	0.86	0.69	0.74	0.86	0.84	0.78	0.84
	ROC Area	0.93	0.93	0.92	0.91	0.95	0.91	0.92	0.95	0.95	0.92	0.95
NB	TP Rate	0.83	0.81	0.81	0.78	0.89	0.78	0.81	0.88	0.85	0.78	0.85
	FP Rate	0.06	0.06	0.81	0.08	0.05	0.08	0.07	0.05	0.05	0.07	0.05
	Precision	0.84	0.81	0.80	0.77	0.87	0.77	0.80	0.85	0.85	0.78	0.85
	Recall	0.83	0.81	0.81	0.78	0.89	0.78	0.81	0.88	0.85	0.78	0.85
	F-Measure	0.82	0.80	0.80	0.77	0.87	0.77	0.80	0.87	0.85	0.78	0.85
	ROC Area	0.96	0.95	0.95	0.95	0.97	0.95	0.95	0.97	0.97	0.95	0.97
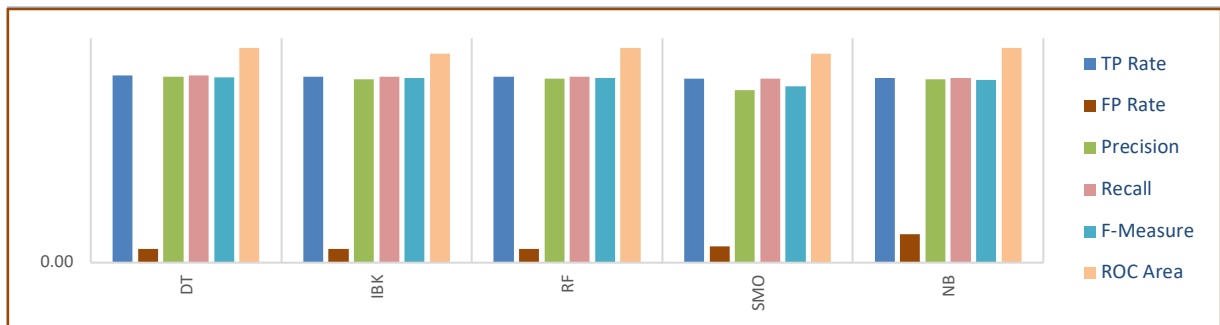
Figure 14: ML tests of the applied algorithms

The output of the DT algorithm is expressed as rules that can be programmatically developed as nested if-else statements. The control structure of the if-else rules specifies the inputs, which are based on the KPIs, while the decision structure of the if-else generates the outputs, which are interpreted as MOS for those who do not provide their feedback on the quality of the provided services. The MOS values are then translated to predicted KPI values.

5.2.4   Learning from Iteration 2

Incorporating machine learning approach into QoEWA is a valuable addition in that enables the model to dynamically predict and evaluate KQIs (via MOS) based on limited user data via a set of decision rules for classification. By dint, the outcomes allow the QoEWA to better facilitate the relationship between KPIs and KQIs. Reflecting this back into the problem space, the enhanced QoEWA provides a better understanding of the links and requirements that bridge service quality and user experience by predicting their converging or diverging directions. Importantly, however, that does not provide a prescription for controlling and optimising QoE – that is, it does not prescribe how to refine and adjust KPIs in accordance with user satisfaction balanced against the hardware, software or staff resources available (Martinez, 2014; Yi et al., 2012). Arguably, the enhanced QoEWA still falls a step short if the ultimate goal of QoE assessment is to

ensure that users are satisfied and resources are well-controlled and efficiently managed (Baraković & Skorin-Kapov, 2013; Elkotob et al., 2010). Consequently, there is a need for further work that extends QoEWA. Such work is in line with the Multi-Objective Optimization (MOO) approach, which is widely used for the optimization problems (Ivesic, Matijasevic, & Skorin-kapov, 2011). While, from a conceptual perspective, MOO can be utilised to adjust the balance between user experience and network resources (Baraković & Skorin-Kapov, 2013; Ivesic et al., 2011) it is outside the scope of this paper and represents ongoing work.

6 Discussion and Overall DSR Evaluation

6.1 Technical Contributions of the Work

In the QoEWA, we have developed a model (and associated instantiation) that connects QoE measurement theories (e.g. Alreshoodi & Woods 2013; Aroussi & Mellouk 2014) with a gap analysis technique (Kan et al., 2001), providing (technical) contribution in the following ways:

- Quantifying QoE. Iteration 1 presents a model that quantifies QoE by utilising an Actual-Versus-Target approach, enabling measurement that: (a) Is more holistic in its nature; (b) correlates objective factors defined in Service Level Agreements (SLAs) with user perception of those factors in operational use; and (c) exposes any gap between actual and target measurements. Doing this allows enhanced monitoring of SLAs and allows the QoE assessor (e.g., systems managers) to address issues more effectively – in good part, this is because issues can be better prioritised and hardware, software and staff resources targeted in a more refined manner that current approaches allow. In developing the model, we have exposed a set of factors that we propose as more appropriate for Web-based systems (see *Table 1*) that are now used in operation. Importantly these factors are mutable (as we stress below) within the model and can be explored by the community in future work. For completeness, an indicative screenshot of the management dashboard is presented at Figure 15.

- Predicting QoE. At the outset of the work, we noted that the relationship between objective and subjective factors represented a challenge. This challenge is rooted in the fact that the latter are typically collected via MOS, which is removed in time from actual service use. We have addressed this challenge by introducing machine learning as a means for the QoEWA model to predict and evaluate subjective data dynamically

(via MOS data feeds), based on limited user data, in a manner that builds a training dataset intelligently from a few samples. In testing several machine learning algorithms, our finding were in accordance with Mushtaq et al. (2012), who observe that Decision Trees and Random Forest have higher performance and accuracy than other approaches (marginally in some cases however). By dint, the outcomes allow the QoEWA to better facilitate the relationship between KPIs and KQIs, though we note that it does not prescribe for controlling and optimising QoE.
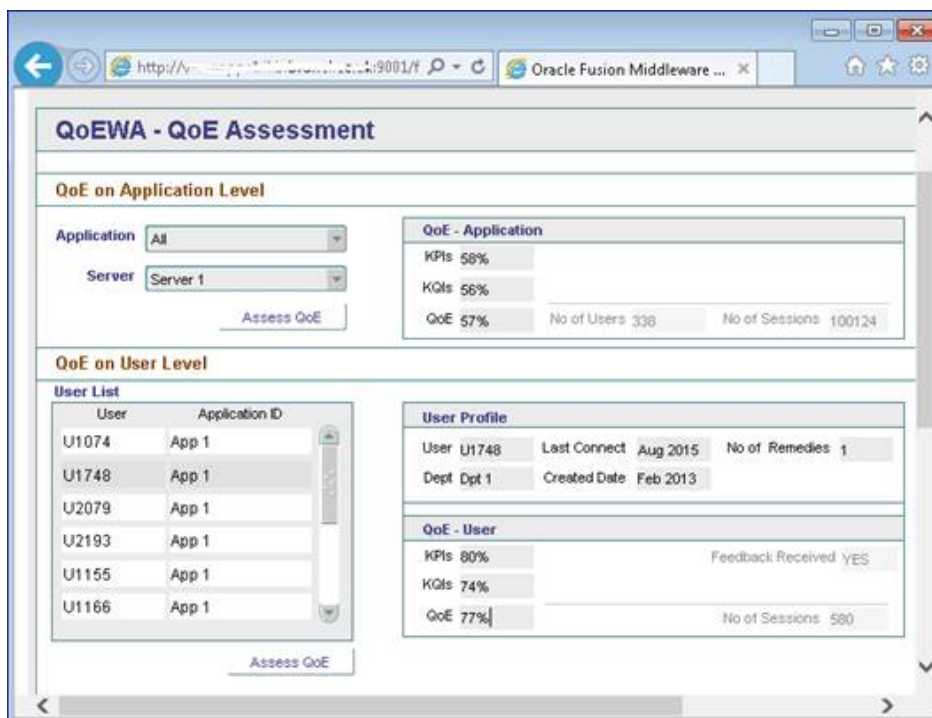

Figure 15: indicative Management Dashboard

6.2 Evaluating the Outcomes

In line with DSR methodology, the practice is to consider the key guidelines of DSR when the developed artefacts are evaluated. In this research, the primary artefacts developed are QoEWA (conceptual) model and its (technological) instantiation – both forms are considered as legitimate in DSR terms. They are carried out with mindful awareness of the debate on theory within the DSR literature – particularly the anatomy of a design theory (S Gregor & Jones, 2007; Walls et al., 1992). For brevity, the work here is presented according to the previously published tenets of a design theory in Table 9.

Table 9: TmQoE as a design theory

Component (Gregor & Jones, 2007)	QoEWA Response
Purpose and scope	To address the challenge to facilitate and quantify the relationship between the KPI and KQI of QoE. Pragmatically, the purpose is to enable service providers to make more informed decisions regarding service delivery and customer satisfaction and/or to optimise resources accordingly.
Constructs	Represented in the core QoEWA model, which initially computes the so-called 'objective' and 'subjective' for determining KPI and KQI. QoEWA has four main constructs that quantify, predict, optimise and perceive QoE.
Principle of form and function	Represented in the measures underlying the QoEWA model, the means by which they are aggregated per construct and the means by which constructs are combined to evidence the QoE assessment. Broadly illustrated in Figure 4.
Artefact mutability	The notion of mutability is addressed in part in the purposeful exposition of the iterations. Mutability is addressed more general terms via the separation of measures from constructs: As a design principle, the QoEWA can be specialised to different contexts of use via the specialisation of measures and/or that addition (or removal) of constructs. See discussion below.
Statements made are testable propositions	Testable propositions are presented at the micro-level in the testing, results and evaluations of both iterations 1 and 2. At the macro-level, the proposition is that the QoEWA will enable service providers to make more informed decisions regarding service delivery and customer satisfaction.
Justificatory knowledge is provided	Specifically, the underlying knowledge has informed the design here is in-and-around existing work related to QoE (e.g., Mirkovic et al., 2014) and machine learning techniques associated with improving understanding of aspects of that (e.g., Mushtaq et al., 2012). It is accepted that this is a more technical that social scientific conception of kernel theory, but it is one that arguably aligns well with theory being considered as a means by which design knowledge is captured, formalised and communicated. Kernel theory in this sense is the input that provides a basis for aspects of the design
Principles of implementation	The principles of the implementation are shown primarily in the form of the equations for QoE calculation, standard software development communication techniques – e.g., use cases, class diagrams and machine learning algorithms and outcomes.
Expository	Instantiations exist both in the form of the QoEWA model and its computational

instantiation	implementation. The model has both generic and specific forms – the latter populating the metrics that can be drawn from the system that are employed with the University systems used in the case.

There are three points that we raise in connection with the table above. First, it has been argued that, constructs, models and methods are one type of thing and can be equated to the components of a theory, while instantiations are a different type (Gregor & Jones, 2007). This is a more pragmatic view of design theory and one we accord with here: The constructs of the QoEWA and their relations produce the model (which represents the design theory); the software instantiation is the material artefact that makes said design theory 'concrete' within its domain of application. Second, it is important that we specify the degree of mutability of the QoEWA as both a model and artefact. Adaptation and/or evolution of the model is allowed in terms of the: (a) Constructs, that assess QoE (i.e., quantify and predict QoE) may be appended depending on the nature of the context/domain; (b) measures which, practically, are constrained by the availability of the data via system interfaces; (c) formulaic method(s) by which the QoE is constructed; and (d) the machine learning methods by which the MOS is achieved.

Third, we consider the testable proposition and notions of generalisation. The proposition at the outset was that the QoEWA will enable service providers to make more informed decisions regarding service delivery and customer satisfaction and/or to optimise resources accordingly. It is this proposition that defines the utility of the artefact(s) which, in prototypical form is most akin to 'new technology X (when applied properly) will provide improvements of Type Y' (Venable, 2006). To that extent we provide improvement – existing artefacts are clearly limited in the ways we have identified (Gregor and Hevner, 2013). Problem awareness provides a start point in that respect but, via the learning from each iteration of work, we have sought to illustrate that the move from problem to solution space is not linear but circular in nature. With each iteration of work, we learned more about the problem space: Initially, that a way of dealing with lack of data in relation to KQIs was required; later that, though machine learning was of benefit, it did not deal with optimising the relationship between user satisfaction and the management of resources in service delivery.

6.3 Reflecting on Problem and Solution Space in DSR

We make two observations in relation to the circular relationship between problem and solution space. First, that it is our belief that we could not have arrived at the learning of Iteration 2 at the outset, and that our design theory results from co-evolution of problem and solution spaces (Dorst & Cross, 2001) and is indeed 'grounded' in practice. Nunamaker et al. (2017) note that, for any given instance of a technology that can be used to improve a situation, a different instance can be built that will not – and, further, that subtle differences in instances may give rise to differences in efficacy. Our second observation relates to this point in that, once 'moves are invented' (Schön, 1992) and a course of action is underway, a path dependency is created between problem and solution spaces, which narrows the solution space with the learning from each iteration of work. Consequently, we strongly concur with the view that the means by which design knowledge is captured, formalised and communicated is important (Gregor & Jones, 2007).

Nunamaker and Briggs (2011 p. 20:2) tentatively propose that the enduring purpose of the IS discipline is to 'understand and improve the ways people create value with information' and that our goal should be to 'help organizations design, build and use information systems in ways that create value'. We argue that design theory as presented here is key in achieving value, as it provides a map across the territory of the design space. If adequately exposed, design theory explicitly links problem with solution (and, ergo, *potential* value) – importantly, exposing the path dependency created in circulating between problem and solution spaces. That path dependency captures and exposes both knowledge related to the design and knowledge related to the context (of the problem). From one perspective, this knowledge is important in understanding the degree to which design knowledge developed and applied in a specific situation can be followed in a similar situation (Prat et al., 2014; Venable, 2006; Gregor, 2009; Gregor and Hevner, 2013). In this regard, we respect the ideographic nature of our design context and go no further than to assert that, in detailed terms, the QoEWA is only an approximation to what might work in other contexts. Our design theory is nascent (Gregor and Hevner, 2013) and, thus, provides only potential for impact (Nunamaker et al. 2017).

From another perspective, however, we observe a strong juxtaposition between generalisation and mutability of the artefact – it is the latter, via the adaptation/evolution mechanisms noted in Table 9 that allows for the artefacts to be appropriated across other domains. Though generalisation remains future work, the importance of design theory

and its presentation endures in relation. In addressing the impact of IS research, Nunamaker et al. (2017) note that, as researchers, we face wicked problems and ongoing conflict between cognitive, economic, emotional, political, physical, psychological, social and technological concerns. Their point is that, as we cannot be masters of all these domains, we work on singular aspects leaving us the intractable task of synthesising disjointed contributions into a cohesive whole – an issue that limits the impact of the discipline. Though the map is not the territory (Korzybski 1931), we propose that exposing design theory in a structured and cohesive manner provides a means by which, for example: (a) Other DSR researchers can understand the path dependency, build on that to generalise outcomes and/or explore other (competing) paths through the design space; (b) behavioural IS researchers can explore antecedents, outcomes and aspects of the design process and/or context and/or value; and (c) the input of researchers from other domains can be specifically harnessed. In that sense, DSR has the potential to provide one means of synthesising and building on prior contribution, allowing bodies of more explicitly related work to build. We have thus sought to make models, iterations and design decisions as clear as we can within the confines of a paper and believe that doing so should be a core requirement of the DSR communication process as a part of the development of the IS discipline.

7   Conclusion and Future Work

The work presented here develops a DSR artefact that has utility clearly associated with potential practice. The scope of that artefact relates to the assessment of the Quality of Experience re the use of Web-based services and, at a meta-level, the purpose is to enable service providers to make more informed decisions regarding service delivery and customer satisfaction and/or to optimise resources accordingly. In meeting the purpose and scope, the core challenge of the work was that of better facilitating and quantifying the relationship between so-called 'objective' and 'subjective' factors related to QoE. This was addressed in the first iteration of the research, via the development of a novel model for QoE (called QoEWA) and its subsequent instantiation. That model developed and integrated constructs to produce a 'balanced scorecard' (of sorts) of the overall Quality of Experience. A second iteration addressed an issue arising with the static and/or retrospective nature of subjective factors – which are generally surfaced via a MOS. User feedback on systems is generally limited and, as the state-of-the-art stands, addressing MOS is generally a time-consuming and (thus) expensive process that distances the opinion of the service from its use in both time and space. Our work in this regard used

machine learning techniques to predict feedback in line with a standard MOS model. This enhances the QoEWA, greatly reducing the temporal and spatial distance between opinion and action.

In achieving the above, we have sought to expose the DSR approach taken in as clear a manner as space and format allows. We have also sought to frame and examine the DSR practice here with the skeleton of a design theory – in this sense we have majored with the view that design theory is the means by which design knowledge is captured, formalised and communicated. We have addressed this notion in two ways. First, by attempting to make our key design decisions transparent – both in relation to the development of the model (via the discussion of constructs, their measures and the formulaic manner in which they are combined) and in showing in abridged form how the model was translated into software design. Indeed, techniques (e.g., UML) are well developed in the software domain and DSR researchers should not be afraid to use them. Second, we have framed and examined the work here in the context of Gregor and Jones's (2007) framework for examining the anatomy of a design theory.

In examining the anatomy, we respect the ideographic nature of the design context and go no further than to assert that, in general terms, the QoEWA is only an approximation to what might work in other contexts. Importantly, however, we observe a strong juxtaposition between generalisation and mutability of the artefact – good design (often enforced by accepted software design principles) can provide adaptation/evolution mechanisms that allow for the artefacts to be appropriated across other domains. In this case, mutability was provided in the constructs, measures and the formulaic means for relating them. As an important part of this anatomy, we were also clear at the outset that: (a) Design theory is more 'grounded' in practice in a way that we should acknowledge; and (b) iterative and/or incremental learning forms an important part of that theory. In demonstrating Point (a) through Point (b), we have sought to show that our evaluation at the end-of-Iteration 1 're-framed' the problem space: Having met the challenge of improving the relationship between so-called 'objective' and 'subjective' dimensions of QoE, the retrospective/static nature of current approaches to the perception of service quality (i.e., as approached via MOS) became more apparent as an issue (and limited the efficacy of the QoEWA). In this way at least, we believe that we have positively demonstrated how the act of building provides a means of learning.

Bibliography

Aaen, I. (2008). Essence: facilitating software innovation. *European Journal of Information Systems*, *17*(5), 543–553. http://doi.org/10.1057/ejis.2008.43

Al-Moayed, A., & Hollunder, B. (2010). Quality of Service Attributes in Web Services. In *2010 Fifth International Conference on Software Engineering Advances* (pp. 367–372). Nice, France: IEEE. http://doi.org/10.1109/ICSEA.2010.62

Alreshoodi, M., & Woods, J. (2013). Survey on QoE\ QoS Correlation Models For Multimedia Services. *International Journal of Distributed and Parallel Systems (IJDPS) Vol.4, No.3, May 2013*, *4*(3), 53–72. Retrieved from http://arxiv.org/abs/1306.0221

Aroussi, S., & Mellouk, A. (2014). Survey on machine learning-based QoE-QoS correlation models. In *The 2nd International Conference on Computing, Management and Telecommunications, ComManTel 2014* (pp. 200–204). Da Nang, Vietnam. http://doi.org/10.1109/ComManTel.2014.6825604

Balachandran, A., Sekar, V., Akella, A., Seshan, S., Stoica, I., Zhang, H., … Zhang, H. (2013). Developing a predictive model of quality of experience for internet video. In *ACM SIGCOMM Computer Communication Review* (Vol. 43, pp. 339–350). Hong Kong, China: ACM. http://doi.org/10.1145/2486001.2486025

Baraković, S., & Skorin-Kapov, L. (2013). Survey and Challenges of QoE Management Issues in Wireless Networks. *Journal of Computer Networks and Communications*, *2013*, 1–28. http://doi.org/10.1155/2013/165146

Behkamal, B., Kahani, M., & Akbari, M. K. (2009). Customizing ISO 9126 quality model for evaluation of B2B applications. *Information and Software Technology*, *51*(3), 599–609. http://doi.org/10.1016/j.infsof.2008.08.001

Brooks, P., & Hestnes, B. (2010). User measures of quality of experience: Why being objective and quantitative is important. *IEEE Network*, *24*(2), 8–13. http://doi.org/10.1109/MNET.2010.5430138

Cecchet, E., Sims, R., He, X., & Shenoy, P. (2013). mBenchLab: Measuring QoE of Web applications using mobile devices. In *IEEE/ACM International Symposium on QoS (IWQoS2013)* (pp. 1–10). Montreal, Canada. http://doi.org/10.1109/IWQoS.2013.6550259

Cox, G. (2005). Cox Review of Creativity In Business: building on the UK's strenghts. *HM Treasury, London, U.K., Executive Summary*. Retrieved from http://www.hm-treasury.gov.uk/d/Cox_review-foreword-definition-terms-exec-summary.pdf

Dorst, K., & Cross, N. (2001). Creativity in the design process, convolution of problem-solution. *Design Studies*, *22*(5), 425–437.

Eckerson, B. W. W. (2009). Performance Management Strategies: How to Create and Deploy Effective Metrics. *The Data Warehousing Institute TDWI*, 1–32.

Elkotob, M., Grandlund, D., Andersson, K., & Ahlund, C. (2010). Multimedia QoE optimized management using prediction and statistical learning. In *35th Annual IEEE Conference on Local Computer Networks* (pp. 324–327). Denver, Colorado, USA: Ieee. http://doi.org/10.1109/LCN.2010.5735733

Fiedler, M., Hossfeld, T., & Tran-Gia, P. (2010). A generic quantitative relationship between quality of experience and quality of service. *IEEE Network*, *24*(2), 36–41. http://doi.org/10.1109/MNET.2010.5430142

Gay, B., & Weaver, S. (2011). Theory building and paradigms: A primer on the nuances of theory construction. *American International Journal of Contemporary Research*, *1*(2), 24–32. Retrieved from http://www.aijcrnet.com/

Geerts, D., Moor, K. De, & Ketyko, I. (2010). Linking an integrated framework with appropriate methods for measuring QoE. In *Quality of Multimedia Experience (QoMEX), 2010 Second International Workshop* (pp. 158–163). Trondheim, Norway: IEEE. Retrieved from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5516292

Gregor, S. (2009). Building theory in the sciences of the artificial. In *09 Proceedings of the 4th International Conference on Design Science Research in Information Systems and Technology* (p. 10). Philadelphia, Pennsylvania: ACM. http://doi.org/10.1145/1555619.1555625

Gregor, S., & Jones, D. (2007). The anatomy of a design theory. *Journal of the Association for Information Systems*, *8*(5), 312–335. http://doi.org/Article

Hevner, A., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly*, *28*(1), 75–105. Retrieved from http://www.springerlink.com/index/pdf/10.1007/s11576-006-0028-8

Hevner, A. R., & Gregor, S. (2013). Positioning and Presenting Design Science Research for Maximum Impact. *MIS Quarterly*, *37*(2), 1–6.

Hobfeld, T., Schatz, R., Varela, M., & Timmerer, C. (2012). Challenges of QoE Management for Cloud Applications. *IEEE Communications Magazine*, *50*(4), 58–65.

ISO/IEC TR 9126-3. (2002). Software engineering – Product quality – Part 3: Internal metrics. *Software Engineering Secretariat: CANADA (SCC)*. ISO. Retrieved from ISO

ISO 9241-11. (1998). ISO 9241-11:1998, Ergonomic requirements for office work with visual display terminals (VDTs) - Part 11: Guidance on usability. Retrieved from https://www.iso.org/obp/ui/#iso:std:iso:9241:-11:ed-1:v1:en

ITU-T. (2006). Recommendation P. 800.1: Mean opinion score (MOS) terminology. *ITU-T P-Series, The International Telecommunication Union (ITU)*.

ITU-T. (2014). Recommendation G.1031: QoE factors in web-browsing. *ITU-T G-Series, The International Telecommunication Union (ITU)*.

Ivesic, K., Matijasevic, M., & Skorin-kapov, L. (2011). Simulation Based Evaluation of Dynamic Resource Allocation for Adaptive Multimedia Services. In *The 7th International Conference on Network and Services Management. (CNSM)* (pp. 432–435). Paris, France.

Kan, S. H., Parrish, J., & Manlove, D. (2001). In-process Metrics for Software Testing. *IBM Syst. J.*, *40*(1), 220–241. http://doi.org/10.1147/sj.401.0220

Khan, A., Sun, L., & Ifeachor, E. (2012). QoE Prediction Model and its Application in Video Quality Adaptation Over UMTS Networks. *IEEE Transactions on Multimedia*, *14*(2), 431–442. http://doi.org/10.1109/TMM.2011.2176324

Kuechler, B., & Vaishnavi, V. (2008). On theory development in design science research: anatomy of a research project. *European Journal of Information Systems*, *17*(5), 489–504. http://doi.org/10.1057/ejis.2008.40

Laghari, K., & Connelly, K. (2012). Toward Total Quality of Experience : A QoE Model in a Communication Ecosystem. *Communications Magazine, IEEE*, *50*(4), 58–65. Retrieved from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6178834

Lee, A. S., Thomas, M. a., & Baskerville, R. L. (2015). Going Back to Basics in Design: From the IT Artifact to the IS Artifact. *Americas Conference on Information Systems*, *25*(1), 5–21. http://doi.org/10.1111/isj.12054

Martinez, L. (2014). QoE : A market perspective analysis. In *25th European Regional Conference of the International Telecommunications Society (ITS)* (pp. 22–25). Brussels, Belgium.

Menkovski, V., Exarchakos, G., & Liotta, A. (2010). Online QoE prediction. In *2010 Second International Workshop on Quality of Multimedia Experience (QoMEX)* (pp. 118–123).

Trondheim, Norway: IEEE. http://doi.org/10.1109/QOMEX.2010.5517692

Menkovski, V., Liotta, A., Sánchez, A. C., & Vargas, E. (2009). Predicting Quality of Experience in Multimedia Streaming. In *International Conference on Advances in Mobile Computing and Multimedia* (pp. 52–59). Kuala Lumpur, Malaysia. http://doi.org/http://dx.doi.org/10.1145/1821748.1821766

Mifsud, J. (2015). Usability Metrics - A Guide To Quantify The Usability Of Any System. Retrieved from http://usabilitygeek.com/usability-metrics-a-guide-to-quantify-system-usability/

Mirkovic, M., Vrgovic, P., Culibrk, D., Stefanovic, D., & Anderla, A. (2014). Evaluating the role of content in subjective video quality assessment. *The Scientific World Journal*, *2014*, 1–9. http://doi.org/10.1155/2014/625219

Mitra, K., Zaslavsky, A., & Ahlund, C. (2011). Dynamic Bayesian networks for sequential quality of experience modelling and measurement. In *Smart Spaces and Next Generation Wired/Wireless Networking* (pp. 135–146). Berlin Heidelberg: Springer Berlin. http://doi.org/10.1007/978-3-642-22875-9_12

Mushtaq, M. S., Augustin, B., & Mellouk, A. (2012). Empirical study based on machine learning approach to assess the QoS/QoE correlation. In *17th European Conference on Networks and Optical Communications* (pp. 1–7). Vilanova i la Geltrú, Spain: IEEE. http://doi.org/10.1109/NOC.2012.6249939

Myers, M. D., & Venable, J. R. (2014). A set of ethical principles for design science research in information systems. *Information Management*, *51*(6), 801–809. http://doi.org/10.1016/j.im.2014.01.002

Nguyen, L. T., Harris, R., & Punchihewa, A. (2013). Assessment of quality of experience for web browsing-as function of quality of service and content factors. In IEEE (Ed.), *International Conference on Ubiquitous and Future Networks, ICUFN* (pp. 764–769). Danang, Vietnam. http://doi.org/10.1109/ICUFN.2013.6614923

Nunamaker, Jr., J. F., & Briggs, R. O. (2011). Toward a broader vision for Information Systems. *ACM Transactions on Management Information Systems*, *2*(4), 1–12. http://doi.org/10.1145/2070710.2070711

Nunamaker, J. F., Twyman, N. W., Giboney, J. S., & Briggs, R. O. (2017). Creating high-value real-world impact through systematic programs of research. *MIS Quarterly*, *41*(2), 335–351.

OASIS. (2012). Web Services Quality Factors Version 1 . 0 Candidate OASIS Standard 01. *OASIS Open 2012*, 1–29.

Orlikowski, W. J., & Baroudi, J. J. (1991). Studying Information Technology in Organizations: Research Approaches and Assumptions. *Information Systems Research*, *2*(1), 1–28. http://doi.org/10.1287/isre.2.1.1

Peffers, K., Tuunanen, T., Rothenberger, M. a., & Chatterjee, S. (2008). A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems*, *24*(3), 45–77. http://doi.org/10.2753/MIS0742-1222240302

Prat, N., Comyn-Wattiau, I., & Akoka, J. (2014). Artifact Evaluation in Information Systems Design Science Research-A Holistic View. In *18th Pacific Asia Conference* (pp. 1–17). Chengdu, China. Retrieved from http://aisel.aisnet.org/cgi/viewcontent.cgi?article=1368&context=pacis2014

Schatz, R., Hoßfeld, T., Janowski, L., & Egger, S. (2013). From Packets to People: Quality of Experience as a New Measurement Challenge. In *Data Traffic Monitoring and Analysis, Springer* (Vol. LNCS 7754, pp. 219–263). Verlag Berlin Heidelberg: Springer-Verlag. Retrieved from http://link.springer.com/chapter/10.1007/978-3-642-36784-7_10

Schön, D. (1992). The Theory of Inquiry : Dewey 's Legacy to Education. *Curriculum Inquiry*, *22*(2), 119–139.

Schumacher, J., Dobler, M., Dillon, E., Power, G., Fiedler, M., Erman, D., ... Argente, J. R. (2010). Providing an User Centric Always Best Connection. In *2nd International Conference on Evolving Internet* (pp. 80–85). Valencia, Spain. http://doi.org/10.1109/INTERNET.2010.23

Seffah, A., Donyaee, M., Kline, R. B., & Padda, H. K. (2006). Usability measurement and metrics: A consolidated model. *Software Quality Journal*, *14*(2), 159–178. http://doi.org/10.1007/s11219-006-7600-8

Simon, H. A. (1996). *The sciences of the artificial, Third Edition*. *Computers & Mathematics with Applications* (Vol. 33). London, England: The MIT Press. http://doi.org/10.1016/S0898-1221(97)82941-0

Skorin-kapov, L. (2012). A Multi-Dimensional View of QoE : The ARCU Model. In *The 35th Jubilee International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO '12)* (pp. 662–666). Opatija, Croatia. Retrieved from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6240728

Skorin-kapov, L., & Barakovic, S. (2015). Multidimensional modelling of quality of experience for mobile Web browsing. *Computers in Human Behavior*, *50*, 314–332. http://doi.org/10.1016/j.chb.2015.03.071

Upadhyaya, B., Zou, Y., Keivanloo, I., & Ng, J. (2014). Quality of experience: What end-users say about web services? In *2014 IEEE International Conference on Web Services, ICWS 2014* (pp. 57–64). Alaska, USA: IEEE. http://doi.org/10.1109/ICWS.2014.21

Vaishnavi, V. and Kuechler, W. (2004). Design science Research in Information Systems. Retrieved 28 November 2014, from http://desrist.org/desrist/article.aspx

Van, W., Vrije, B., Pierson, J., & Lievens, B. (2008). Confronting video-on-demand with television viewing practices. In *Innovating for and by Users* (pp. 1–222). COST and IBBT.

Venable, J. R. (2006). The role of theory and theorising in design science research. In *Proceedings of the 1st International Conference on Design Science in Information Systems and Technology* (pp. 1–18). Claremont, USA. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.110.2475&rep=rep1&type=pdf

Vidgen, R., Donnellan, B., Matook, S., & Conboy, K. (2011). Design Science Approach to Measure Productivity in Agile Software Development. In *European Design Science Symposium* (pp. 171–177). Verlag Berlin Heidelberg. http://doi.org/10.1007/978-3-642-33681-2_15

Walls, J. G., Widmeyer, G. R., & El Sawy, O. a. (1992). Building an Infomiarion System Design Theory for Vigilant EIS. *Information Systems Research*, *3*(1), 36–58.

Wang, Z., & Wang, Z. (1998). Objective image/video quality measurement - a literature survey. *Literature Survey for EE381K: Multidimensional Signal Processing Project*, *Project EE*. Retrieved from c:%5CLibrary%5CWang1998.pdf

Witten, I., Frank, E., & Hall, M. (2011). *Data Mining: Practical Machine Learning Tools and Techniques 3rd* (Third). Morgan Kaufmann Publishers Inc. San Francisco, CA, USA ©2011. http://doi.org/10.1002/1521-3773(20010316)40:6<9823::AID-ANIE9823>3.3.CO;2-C

Yamauchi, D., Ito, Y., & Tahara, T. (2015). A Method of QoE Management in Online Shopping Web Services with TCP Variables. *INTERNETWORKING INDONESIA*, *7*(1), 3–7.

Yi, L., Shou, G., Yihong, H., & Zhigang, G. (2012). A model for evaluating QoE of mobile Internet services. In *Proceedings of 2012 15th International Symposium on Wireless Personal Multimedia Communications (WPMC)* (pp. 438–442). Taipei, Taiwan: IEEE.

Zieliński, K., Szydło, T., Szymacha, R., Kosiński, J., Kosińska, J., & Jarzab, M. (2012). Adaptive SOA solution stack. *IEEE Transactions on Services Computing*, *5*(2), 149–163. http://doi.org/10.1109/TSC.2011.8

Zinner, T., Hohlfeld, O., Abboud, O., & Hossfeld, T. (2010). Summary to Impact of Frame Rate and Resolution on Objective QoE Metrics. In *In Quality of Multimedia Experience (QoMEX), 2010 Second International Workshop, IEEE* (pp. 29–34). Trondheim, Norway: IEEE.