Empirical Software Engineering, 5, 245–267 (2000)

Replicating the CREWS Use Case Authoring Guidelines Experiment

KARL COX coxk@bournemouth.ac.uk
Empirical Software Engineering Research Group, School of Design, Engineering & Computing, Bournemouth University, Dorset, UK

KEITH PHALP kphalp@bournemouth.ac.uk
Empirical Software Engineering Research Group, School of Design, Engineering & Computing, Bournemouth University, Dorset, UK

Abstract. Use cases have become an important tool in software engineering. There has been much focus on the diagram notation but relatively little on use-case descriptions. As part of a welcome and important research project into the use of scenarios in requirements engineering, the CREWS (Co-operative Requirements Engineering With Scenarios, an EU funded ESPRIT project 21903) team has proposed a set of guidelines for writing use-case descriptions. This paper describes the replication of a CREWS project experiment that suggests CREWS use-case authoring guidelines improve the completeness of use-case descriptions. Our results show that the CREWS guidelines do not necessarily improve the use-case descriptions, only that the subjects implemented varying numbers of guidelines in their use-case descriptions. Subjects in the control group implemented a significant percentage of the guidelines by 'chance.' To further justify our results, we also apply a different marking scheme to compare with the CREWS approach. The results from the alternative marking approach show that there was no significant difference between the qualities of the use-case descriptions across the various groups.

Keywords: use cases, use-case descriptions, experiment replication

1. Introduction

Originally introduced by Ivar Jacobson at the OOPSLA conference in 1987, use cases only truly became popular in software engineering with the publication of Jacobson's book in 1992. Since then software engineers have tried to adapt use cases to their requirements engineering processes and methods. In simple terms, the use case is intended to aid the software engineer in discovering and describing all the functional requirements for the system under development. The use-case diagram notation is simple, thus making it easy for clients and users to understand, though some have suggested semantic weaknesses to the notation (Cox and Phalp, 2000), not to mention the confusion over the meaning of includes (uses) and extends (Rosenberg, 1999). The use-case description has traditionally been written in natural language text, usually a paragraph or two of prose. Bonnie Nardi, for example, described a good scenario script as one that consisted of "one or two sides of well-crafted prose" (1992). With the arrival of the UML (Unified Modeling Language), use cases have become even more popular (Rational, 1997). The UML, however, has made little attempt to guide software engineers in how to write use-case descriptions. The UML User Guide (Booch, 1999) barely touches the issue, showing example use-case descriptions as paragraphs of natural text. Ericksson and Penker suggest some guidelines for what should be in the use case but there is no description of the actual use-case structure except for a

simple numbered example (1998). Pooley and Stevens describe use cases as paragraphs of text (1999), as does Rosenberg (1999).

Some software engineers have attempted to redress this issue. Martin Fowler provides a simple structured textual approach to use-case descriptions (2000), but he fails to give any semantic or structural underpinning to the description. Alistair Cockburn describes a semi-formal structure to use-case description. Each event (action description) should take the following format:

"⟨*time or sequence*⟩. . . ⟨*actor*⟩. . . ⟨*action*⟩. . . ⟨*constraints*⟩

Thus, typical sentences might look like:

At any time after the clerk gets the quote, he may cancel the sale" (Cockburn, 1997).

Cockburn does not go on to describe any further grammar than this but does recommend, like Fowler, a use case with each event numbered and on a separate line. Harwood suggests an alternative approach by using a tabular format for use-case descriptions. This offers generic structure to the overall use case but there is no reference to the structure of individual events (Harwood, 1997). Cockburn (2000) is soon to publish a book on writing use cases and it appears that he recommends employment of a tabular format for use-case descriptions. Wiegers also recommends this approach (Wiegers, 1999). However, Rosenberg argues against tables and templates because for him they are too time consuming and confusing though he wisely recommends that you use what suits you (Rosenberg, 1999). The CREWS approach does not concern itself with a tabular format but offers guidance in structuring the contents and style of use-case descriptions.

The CREWS team has perhaps made the most significant contribution to research into scenario-based design in requirements engineering over the last few years. They show that use cases and scenarios are widely used in industry but that there are concerns about how to structure textual use-case descriptions (Jarke et al., 1997; Weidenhaupt et al., 1998). To tackle this lack of structure, CREWS has built on to the work of Cockburn in proposing a set of Style and Content Guidelines to enable better scenario authoring (Ben Achour, 1998). To test how effective the guidelines are, CREWS conducted an experiment where four groups of subjects wrote a use-case description, three of the four groups applying some part of the guidelines. CREWS report that application of the guidelines significantly improved the use-case descriptions. The results are described in Ben Achour et al. (1998) and some of the structure of the experiment is found at Ben Achour and Maiden (1999).

The structure of the paper is as follows. Section 2 discusses the issue of replication. Section 3 lists the CREWS guidelines. Section 4 describes the experimental hypotheses. Section 5 describes the experimental design. Section 6 describes the results. Section 7 describes the alternative marking approach and results. Section 8 draws some conclusions and discusses future work.

2. Replication of Experiments

When attempting to replicate an experiment, it would be ideal to perform an 'exact replication' of that experiment. The goal would be to get the same results as the original experiment

from similar conditions or, indeed, to prove the null hypothesis. This is not very easy to do in most software engineering experiments because different people will be involved who have different degrees of knowledge and interpretation and apply them in different ways. Although "no replication is ever exact" (Robson, 1993), as close a replication as possible is the next best alternative.

There are at least two approaches to documenting the experiments. The scientific approach would be to detail in sufficient depth all that is required for someone to replicate that experiment. Another approach is to work with 'softer' or ethnographic traditions and thus not consider it necessary to take such a scientistic view (Robson, 1993). It is unclear which approach the CREWS experiment took but insufficient information has been given to allow close replication. A number of points interested us about the CREWS experiment. First, we were not convinced by all the CREWS guidelines. Second, we were not convinced about the validity of their hypotheses. Third, we were not convinced that application of the guidelines could produce better use-case descriptions than application of common sense because many of the guidelines appear to be constructs in the English language that are used every day.

3. The CREWS Use-Case Authoring Guidelines

The CREWS team propose two sets of guidelines, Style and Content.

3.1. Style Guidelines

The purpose of the Style Guidelines is to give the use-case author a list of do's and don'ts for writing use-case descriptions. The aims are to provide a standard structure for descriptions and also to make the descriptions easier to read.

- "SG1: write the Use Case normal course as a list of discrete actions in the form: ⟨action#⟩ ⟨action description⟩. Each action description should start on a new line. Since each action is atomic, avoid sentences with more than two clauses;

- SG2: use the sequential ordering of action descriptions (and hence their unique number identifiers) to indicate strict sequence between actions. CREWS imposes a precise meaning on the ordering of actions in this list. Variations should be written in a separate section;

- SG3: iterations and concurrent actions can be expressed in the same section of the Use Case, whereas alternative actions should be written in a different section;

- SG4: use consistent agent, object and action names in all action descriptions in a Use Case. Avoid use of synonyms and homonyms, and anaphoric references such as he, she, them and it. Be consistent in your use of terminology;

- SG5: use present tense and active voice when describing actions;

- SG6: avoid use of negations, adverbs and modal verbs in the description of an action." (Ben Achour et al., 1998).

3.2. Content Guidelines

The aim of the Content Guidelines is to guide a use-case author in constructing each event in the description. The guidelines act as a template so authors can substitute their vocabulary into the template and know that it is constructed to a standard format.

- "CG1: ⟨agent⟩ ⟨'move' action⟩ ⟨object⟩ from ⟨source⟩ to ⟨destination⟩;
- CG2: ⟨source agent⟩ ⟨'put' action⟩ ⟨object⟩ to ⟨destination agent⟩;
- CG3: ⟨destination agent⟩ ⟨'takes' action⟩ ⟨object⟩ from ⟨source agent⟩;
- CG4: ⟨agent⟩ ⟨action⟩ ⟨agent⟩;
- CG5: ⟨agent⟩ ⟨action⟩ ⟨object⟩;
- CG6: 'If' ⟨alternative assumption⟩ 'then' ⟨action⟩;
- CG7: 'Loop' ⟨repetition condition⟩ 'do' ⟨action⟩;
- CG8: ⟨action 1⟩ 'meanwhile' ⟨action 2⟩." (Ben Achour et al., 1998).

The CREWS team suggest that the guidelines be applied as much as possible. However, we believe there will certainly be times when they are not applied. On these occasions, it is unclear whether the lack of application means that the use-case descriptions are poorly written. The Style Guidelines can generally be applied throughout. The Content Guidelines, however, are very specific and one could suggest alternative guidelines to the CREWS selection that are equally valid.

4. CREWS Experimental Hypotheses

The hypotheses set by the CREWS team are:

The following three hypotheses are Content Guidelines (CG) related:

"**C1:** the use of CGs will lead to use-case descriptions which are more correct in terms of the number of complete action descriptions;

C2: the use of CGs will lead to less inappropriate action descriptions;

C3: the use of CGs will lead to use-case descriptions which contain more correct and unambiguous descriptions of the flow of the use case." (Ben Achour et al., 1998).

The following four hypotheses are Style Guidelines (SG) related:

"**S1:** the use of SGs will lead to more complete action descriptions;

S2: the use of SGs will lead to less unnecessary or inappropriate action descriptions;

S3: the use of SGs will lead to use-case descriptions which contain more correct and unambiguous descriptions of the flow structure of the use case;

S4: the use of SGs will lead to use-case descriptions with more consistent use of terminology." (Ben Achour et al., 1998).

The CREWS team argues that many of the hypotheses must be judged in unison. For example, hypothesis C1 and hypothesis S1 can be measured together because presumably C1 is dependent on S1 and vice versa. CREWS should have stated a null hypothesis for each of their hypotheses, such as:

$C1_0$ = the application of the guideline C1 will not significantly improve the number of complete action despcriptions.

We found that many of the CREWS hypotheses were vague (especially C1, C2, S1 and S2). This is a reason why we replicated their experiment. It is not clearly defined what a "complete action description" is and thus the hypotheses are open to interpretation.

5. Experimental Design

14 experimental subjects wrote a use-case description of an interaction between a supermarket checkout operator and the checkout machine. The task description can be found in Appendix A. The task description is identical to that given to the subjects of the CREWS experiment.

5.1. Experimental Subjects

The 14 subjects are full-time students attending the Masters degree in Software Engineering at Bournemouth University. The subjects received a half-day seminar on use cases a week before the experiment. This was their first exposure to use cases. The seminar included a discussion of use-case structure and contents, such as levels of abstraction. The subjects also completed a form about their experience of systems analysis and programming. This form (figure A1) can be seen in Appendix A. The form is identical to the one used in the CREW experiment. The information gathered from the forms helped establish an equal spread of experience and age amongst the experimental groups. The subjects were divided into the four groups shown in Table 1.

The subjects' experience ranged from zero to 6 years in systems analysis and from zero to 15 years in programming. Ages ranged from 23 to 48. Groups A and B contained four subjects each and groups C and D three subjects each.

Table 1. The four experimental groups.

Group	Definition
A	A control group in which subjects were given the problem statement only
B	An experimental group in which the subjects were given the problem statement and CREWS Style Guidelines
C	An experimental group in which the subjects were given the problem statement and CREWS Content Guidelines
D	An experimental group in which the subjects were given the problem statement and CREWS Style and Content Guidelines

The subjects were given one hour to write the use cases. Before the experiment, a set of instructions was read out by the experimenter and clarifications cleared up. The instructions are identical to those issued by the CREWS team and can be found in Appendix A. As with the CREWS experiment, subjects could ask any questions they wished but only to the experimenter.

5.2. Evaluation Scheme

The subjects' use cases were evaluated against different criteria. A problem with the CREWS experiment is that it does not fully describe how it evaluates use cases. As such, we had to decide our own evaluation criteria where necessary. Our evaluation criteria came directly from the guidelines. For example, for Style Guideline 1 (SG1), the variables were: *atomic event* and *composite events* (composite lines in a use case consist of more than one event). Counts were taken of the number of atomic events and the number of composite events. Each use-case description was of varying length so to account for this each line in the description counted as either an atomic event or composite event, dependent upon the number of events it contained. The total of atomic and composite events should match the number of lines in the use-case description. For the Content Guidelines a count was taken of the number of times a Content Guideline was used to structure each sentence. So if a use-case description used Content Guideline 5 (CG5: ⟨agent⟩ ⟨action⟩ ⟨object⟩) and it was also an atomic event then this was counted.

Appendix A lists a table (Table A1) of all the variables that were counted for the experiment. It shows that every variable described in the guidelines was counted.

The CREWS team states that many of the hypotheses must be considered in unison because it is possible to write a good style description but its contents could be poor. Application of the Content Guidelines does not necessarily mean that descriptions are written in good style, either, although because of the actual structure of the Contents Guidelines, those employing solely the Contents Guidelines did not score poorly in the Style stakes (see Section 6). To enable marking of the use-case descriptions, CREWS have an "expert" answer for the use case, found in Appendix A. Obviously there is scope for many varied answers, of which all could be considered viable solutions. The problem we found with the CREWS "expert" answer is that it violates the rules for writing use-case descriptions. A

use-case description describes an interaction between an actor and a system. This implies that there is no description of what happens inside the system. Yet the CREWS answer states:

"2. The system reads customer information from the club card.

. . .

4.4. The system records purchase details from the product bar code

. . .

10. The system records the entire transaction." (Ben Achour and Maiden, 1999)

These details are system internal and as such should not be included in a use-case description. It they were to be included then they would need to be reworded so that the actor had some interaction with the system or perhaps the actor was being informed by the system that recording was taking place. For example, "The system notifies the operator that the system is now recording the entire transaction." So judging the experimental subjects' answers against the CREWS "expert" answer was not always completely valid.

6. Experimental Results

Unrelated t-tests were performed on the data from control group (A) with the data from the experimental groups (B, C and D) to examine the significance of difference between the groups. The statistical results from the experiments are found in Appendix B (Table B1). These show there is no significant difference between the groups. The t-test formula applied was taken from (Greene, 1982).

6.1. Content Guidelines Results

Table 2 shows the application of Content Guidelines amongst the four groups.

Group A (Control) used complete Content Guidelines in 40% of their use-case descriptions without knowledge of the guidelines. Group B (Style) also used the Content Guidelines correctly in 40% of their descriptions without knowledge of guidelines. Group C (Content) used the Content Guidelines correctly in 87% of their descriptions. Group D (Style and Content) used the Content Guidelines correctly in 77% of their descriptions. It would be expected that Groups C and D score a higher percentage of correctly used Content Guidelines than groups A and B. The guideline that was used most often in all the groups was CG5:

CG5: ⟨agent⟩ ⟨action⟩ ⟨object⟩;

An example of application by 'chance' of CG5 from group A is: "The system displays the total amount payable." Another application of 'chance' of CG5 from group B is: "The checkout operator scans the barcode of the new item." Figure 1 gives the exact percentage usage of CG5 across the four groups.

Table 2. Application of content guidelines.

Groups:		A: Control			B: SG			C: CG			D: SG&CG		
Guideline	**Description**	**Cnt**	**%**	**M**	**Cnt**	**%**	**M**	**Cnt**	**%**	**M**	**Cnt**	**%**	**M**
CG1	Complete	0	0	0	0	0	0	0	0	0	0	0	0
CG2	Complete	1	0.01	0.25	1	0.02	0.25	0	0	0	0	0	0
CG3	Complete	1	0.01	0.25	0	0	0	0	0	0	0	0	0
CG4	Complete	0	0	0	0	0	0	1	0.02	0.33	1	0.01	0.33
CG5	Complete	21	0.29	5.25	22	0.32	5.5	32	0.74	10.67	40	0.58	13.33
CG6	Complete	7	0.09	1.75	5	0.07	1.25	4	0.06	1.33	11	0.16	3.67
CG7	Complete	0	0	0	0	0	0	3	0.06	1	1	0.02	0.33
CG8	Complete	0	0	0	0	0	0	0	0	0	1	0.01	0.33
CG1-8	Complete actions	30	0.4	7.5	28	0.4	7	40	0.87	13.33	54	0.77	18
CG1-8	Incomplete actions	42	0.6	10.5	45	0.6	11.25	8	0.13	2.66	16	0.23	5.33
Number of lines in UC Main Flow		72	—	18	73	—	18.25	48	—	16	70	—	23.33

Key: Cnt = total count; % = percentage usage of complete (and incomplete) CGs in group; M = Mean

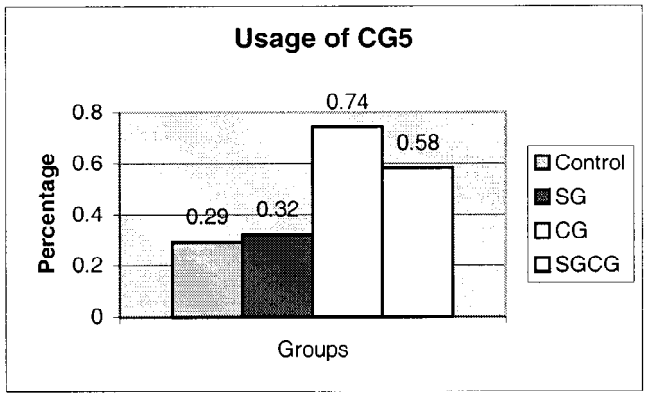


Figure 1. Usage of content guideline 5.

The results in Table 2 show that the Content Guidelines, and especially CG5, were used approximately twice as much by groups C and D, who had access to them. The results do not show anything more than that. It is shown that the Content Guidelines were not used apart from CG5 and occasionally CG6. The other guidelines were rarely used or not at all. Perhaps only CG5 and CG6 need to be considered when writing a use case.

6.1.1. Validation of Hypothesis C1

Hypothesis C1 appears to be validated. Groups C and D, who had access to the Content Guidelines (CGs), wrote more complete action descriptions than groups A and B. Yet this hypothesis appears to be a self-fulfilling prophecy. If we accept that a "complete action description" means a single event that is written in the mode of a CREWS Content Guideline then hypothesis C1 is valid. The hypothesis might have been better written, "Those given the CGs will write more complete action descriptions by applying the structures described in the CGs." Figure 1 shows this to be the case.

6.1.2. Validation of Hypothesis C2

This hypothesis states that the use of Content Guidelines will lead to less inappropriate action descriptions. The two groups that did not apply the CGs (groups A and B) managed to write 40% of their use-case descriptions in the manner of the CGs. The groups that did apply the CGs (groups C and D) wrote 87% and 77% respectively of their use-case descriptions in the manner of the CGs. The hypothesis is validated. However, are we just validating the fact that the CGs were applied rather than the actual hypothesis?

For example, here is a possible Content Guideline. It is similar to CG5 but has an extension of ⟨destination⟩. The case grammar would look like this:

⟨agent⟩ ⟨action⟩ ⟨object⟩ ⟨destination⟩

An example action description is taken from a group A (control group) use-case description:

The operator puts the money in the till.

⟨agent⟩ ⟨action⟩ ⟨object⟩ ⟨destination⟩

This seems an acceptable description yet it does not exactly match the CREWS guidelines so we are unclear if it can be counted as valid.

Indeed, the CREWS expert answer uses this case grammar:

> "7. The operator encashes the payment in the system till." (Ben Achour and Maiden, 1999).

6.1.3. Validation of Hypothesis C3

This hypothesis states that use of CGs will lead to use-case descriptions that contain more correct and unambiguous descriptions of the flow structure of the use case. It is certainly the case that there are more incomplete action descriptions in groups A and B. But some of the CGs are poorly structured. For example, CG7:

CG7: 'Loop' ⟨repetition condition⟩ 'do' ⟨action⟩;

One use-case description from group C (Content) applied CG7:

3. 'Loop' item(s) remain to be checked out

    3.1 'Do'

    3.2 Operator enters next item

    3.3 System displays item name and price

The whole structure seems a bit ambiguous. It is not clear to what event the loop returns to after its first iteration. It is only by assumption that after 3.3, we return to 3.2. Event 3.1 'Do' does not *do* anything! This is not the experimental subject's fault but the actual guideline itself. There must be better ways to structure a loop or iteration. We might place all events on a single line to make the Content Guideline read more like the guideline itself:

3. 'Loop' item(s) remain to be checked out; 'Do': Operator enters next item and system displays item name and price.

However, this violates Style Guideline 1 (SG1: write the Use Case normal course as a list of discrete actions in the form: ⟨action#⟩ ⟨action description⟩. Each action description should start on a new line. Since each action is atomic, avoid sentences with more than two clauses) as well as being messy to read. Structure is important in use-case descriptions and so is readability. Content Guideline 7 is difficult to read because its structure demands that the reader make assumptions in interpreting its meaning. Hypothesis C3 is difficult to accept.

6.2. Style Guidelines Results

Groups B and D applied the Style Guidelines. Curiously, group B, which applied only the Style Guidelines, performed worse than all the other groups, as can be seen in Table 3.

The CREWS results showed usage of anaphoric references. Anaphoric reference is the usage of articles or pronouns to 'point backwards' to a previous usage (Swan, 1980). The CREWS team has interpreted an anaphor as simply a pronoun, as stated in Style Guideline 4:

SG4: use consistent agent, object and action names in all action descriptions in a Use Case. Avoid use of synonyms and homonyms, and anaphoric references such as he, she, them and it. Be consistent in your use of terminology;

The CREWS team does not mention the fact that articles can be anaphors. In any case, in the English language, the certain usage of articles in anaphoric reference is significantly better than others. For example, *I put the money back in my pocket*. The anaphor is *the* in the sentence. Replace *the money* with *it* and the sentence reads: *I put it back in my pocket* (Swan, 1980). There is a question of avoidability in English. Although it may be a good idea to avoid anaphors, there are times when usage makes more sense than avoidance. The anaphoric reference in the first example sentence conforms to Content Guideline 5 at least at the start:

⟨agent⟩	⟨action⟩	⟨object⟩
I	put	the money. . .

Now add the ⟨destination⟩:

⟨agent⟩	⟨action⟩	⟨object⟩	⟨destination⟩
I	put	the money	back in my pocket

This structure does not conform to any Content Guideline yet is a perfectly valid structure. We did not find a single pronoun anaphoric reference in any use-case description from any of the groups. As CREWS do not tell us what anaphoric references are fine and what are not, we take it that the example usage described above is valid.

The Control Group scored badly on the total count of occurrence of the passive tense (see Table 3). Here the Style groups faired much better in comparison to Group A. However, the

Table 3. Correct application of style guidelines.

Guideline	Groups: Description	A: Control Cnt	%	M	B: SG Cnt	%	M	C: CG Cnt	%	M	D: SG&CG Cnt	%	M
SG1	Atomic event	54	0.75	13.5	49	0.67	12.25	42	0.9	14	66	0.94	22
	Composite	18	0.25	4.5	24	0.33	6	6	0.1	2	4	0.05	1.33
SG2	Unique order	70	0.97	17.5	71	0.97	17.75	48	1	16	68	0.97	22.67
	Not unique order	2	0.03	0.5	2	0.03	0.5	0	0	0	2	0.03	0.67
	Separate variations	6	n/a	1.5	6	n/a	1.5	9	n/a	3	15	n/a	5
SG3	Iterations in main	4	n/a	1	4	n/a	1	2	n/a	0.67	4	n/a	1.33
	Alternatives in main	11	n/a	2.75	12	n/a	3	5	n/a	1.67	14	n/a	4.67
SG4	Homonyms	0	n/a	0	0	n/a	0	0	n/a	0	0	n/a	0
	Synonyms	2	n/a	0.5	5	n/a	1.25	2	n/a	0.67	3	n/a	1
	Anaphors	0	n/a	0	0	n/a	0	0	n/a	0	0	n/a	0
SG5	Not present tense	0	n/a	0	2	n/a	0.5	0	n/a	0	0	n/a	0
	Passive tense	15	n/a	3.75	8	n/a	2	1	n/a	0.33	6	n/a	2
SG6	Negatives	2	n/a	0.5	4	n/a	1	0	n/a	0	3	n/a	1
	Adverbs	0	n/a	0	0	n/a	0	0	n/a	0	0	n/a	0
	Modal verbs	1	n/a	0.25	0	n/a	0	0	n/a	0	0	n/a	0

Key: Cnt = total count; % = percentage usage of SGs in groups; M = Mean; n/a = not applicable

Content group did even better than the Style groups. This is probably due to the structure of the Content Guidelines; they show no passive tense construction.

All groups applied SG2 almost perfectly. Every use-case description from each group was uniquely ordered. There were only one or two cases of confused numbering of action descriptions.

6.2.1. Style Guideline Hypotheses

The Style Guideline hypotheses are not validated. Those groups that applied the Style Guidelines (groups B and D) scored very similar results to the two groups that did not apply the Style Guidelines (groups A and C).

6.3. Style and Content Hypotheses Combined

As stated above, the CREWS team argue that several of the hypotheses complement each other. That is, hypothesis C1 and hypothesis S1 work together. Section 6.1.1 describes a validation of hypothesis C1. But just because C1 is valid does not mean we can bunch S1 with it. On its own, S1 does not appear to be validated. The same can be argued for the other combination hypotheses—one cannot assume that S2 or S3 are valid simply because C2 and C3 might be.

7. An Alternative Marking Approach and Results

We applied an alternative marking approach because we were not happy with the CREWS results. As the experiment formed part of an assignment, we took a typical assignment marking approach to the subjects' use cases. There are four variables to consider (individual marks are show in Table 4 and group marks are shown in Table 5):

i. Plausibility—how realistic was the use case? For example, the subject who only scored 4 in Group D did so because they made no reference to payment.

ii. Readability—did the use case flow? Was each event numbered sequentially? Did the use case follow a logical path?

iii. Consistent Structure—Was terminology consistent? Was Present Simple tense used throughout?

iv. Alternative flow—marks were allocated for consideration of (viable) alternatives in the use case.

Table 4. Individual marks.

Group	Plausibility	Readability	ConstStructure	Alternative	**Total**
A: Control					
1	6	5	5	1	17
2	4	3	3	1	11
3	5	6	5	1	17
4	4	5	5	0	15
B: Style					
1	5	5	5	1	16
2	5	5	5	2	17
3	6	5	4	0	15
4	5	5	6	0	16
C: Content					
1	5	4	5	2	16
2	3	3	6	1	13
3	6	4	5	1	16
C: Style & Content					
1	5	3	5	0	13
2	5	5	4	0	14
3	4	5	6	5	20
Marks out of 25	7	7	6	5	25
Average	4.86	4.50	4.93	1.07	15.43

Table 5. Average marks.

Average Plausibility		**Average Readability**	
A: Control	4.75	A: Control	4.75
B: Style	5.25	B: Style	5.00
C: Content	4.67	C: Content	3.67
D: S&C	4.67	D: S&C	4.33
Average Consistent Structure		**Overall average marks:**	
A: Control	4.50	A: Control	15.00
B: Style	5.00	B: Style	16.00
C: Content	5.33	C: Content	15.00
D: S&C	5.00	D: S&C	15.67

7.1. Marking

The points listed above formed the marking criteria. For example, for consistent structure we looked for consistent terminology and usage of present simple tense. So, in fact, we started marking the use-case descriptions with a subset of the CREWS guidelines.

However, when we decided to simply read the descriptions without searching for these elements we realised that some the descriptions were not as good as we thought and some

were better. It was a case of not seeing the wood for the trees because we had focussed on the detail and had missed the overall description. When it became clear that we were marking with a subset of the CREWS guidelines, we decided to mark based on the overall description and took the marking scheme as an aid and nothing more. Both the authors marked the use-case descriptions and we took the average of the two scores. The groups' results (Table 5) are discussed in the next sections.

7.1.1. Plausibility

Group B (Style) scored on average 5.25 out of 7, slightly higher than the other groups. Group A (Control) scored 4.75, narrowly higher than groups C (Content) and D (Style and Content), who both scored 4.67. The difference between groups is negligible.

7.1.2. Readability

Group B's average score was 5 out of 7, slightly higher than Group A (4.75). Group D scored 4.33 but group C scored rather poorly (3.67). There is a significant difference in readability between groups B and C. The best score came from the Style Guidelines group. However, the Style Guidelines group (B) scored poorest with the CREWS evaluation criteria (see section 6).

7.1.3. Consistent Structure

Group C scored highest (5.33 out of 6). Groups B and D were only marginally poorer (5). Group A scored a bit lower (4.5). Although Group A (Control) has the highest use of passive voice, this is a slight anomaly because the vast majority of that passive usage came from just one use-case description.

7.2. Overall Average Marks

Group B scored highest overall (16 out of 25). Group D was close behind with 15.67. Groups A and C were equal with 15. The difference between the four groups is negligible.

The results from the alternative marking approach can be interpreted thus:

- There was little difference between the groups in terms of plausibility.
- There was a difference between the groups in terms of readability although Group B was better. Considering they applied the Style Guidelines, this is not surprising.
- There was little difference between the groups in terms of consistent structure although Group C (Content) scored marginally better. Yet the Content Guidelines do not suggest how to improve the structure of the overall use-case description, only individual events.

Table 6. Validation of CREWS hypotheses.

Improved UC feature	Do Style Guidelines help?	Do Content Guidelines help?
Action Completeness	(S1) No.	(C1) Yes, in that they were applied.
UC Completeness	(S2) No.	(C2) Yes, in that they were applied.
UC Structure	(S3) Yes for non-passive tense. No for the rest.	(C3) No help.
Terminology Correction	(S4) Did **not** help.	

The Style Guidelines do suggest how to improve overall structure and Group B scored only marginally less than Group C.

- Overall Group B was (narrowly) the best group. This contradicts the results from the CREWS marking scheme, where it is argued that Group B is the worst group.

This alternative marking approach is somewhat subjective. Nevertheless, we believe that the use cases across all the groups were comparable. It we judged by solely the CREWS evaluation criteria then we would give the wrong impression.

8. Conclusions and Future Work

Table 6 shows how valid we found the hypotheses set out by CREWS. The CREWS results show significant success for the Style Guidelines and some good success for the Content Guidelines (Ben Achour *et al*, 1998). However, our results are somewhat opposed to the CREWS findings. We found that the Style Guidelines were not particularly useful for Action completeness or Use-Case Completeness. They did score some success, however, for Use-Case Structure.

The Content Guidelines faired better. But all we could prove was that the guidelines were applied. They did nothing for the completeness of the actions or of the use case as a whole. In terms of structure, the Content Guidelines were of no discernible value, except perhaps in terms of implicit non-passive voice compliance.

As a validity test, all the hypotheses were tested against the control group (group A). Style Guidelines group (group B) results were compared to the Control group results. Content Guidelines group (group C) results were compared to the Control group results. Style and Content group (group D) scored marginally worse results to group C. This implies that number of guidelines to implement might have been too much for Group D. However, they did apply the Style Guidelines more appropriately than group B (Style). Our results are contradictory in that the Style Guidelines hypotheses are not proven yet our alternative approach shows Group B does better than any other group, even if the difference is marginal.

There is a replication issue to consider regarding this study. Since the CREWS experimental procedures are not completely documented, it is not justified to call our experiment an exact replication. However, it has proven a useful exercise nonetheless to implement the guidelines and see what difference they make to use-case descriptions.

The body of work produced by CREWS is very significant and we welcome this research. The guidelines, as part of that research, are very interesting and we recommend that the guidelines *should* be considered when authoring use cases, especially aspects of the Style Guidelines. However, our results found, especially with regards the Content Guidelines, only the number of times a guideline was correctly implemented. As such, we are unclear how some of the guidelines necessarily improve use-case descriptions. We think it important that further studies be carried out to implement the CREWS guidelines.

8.1. Future Work

We aim to conduct further experiments that implement the CREWS guidelines. We also think that writing use-case descriptions is only half the story. We consider that some guidelines are probably helpful when reading the documents that contain the elicited information used for writing use-case descriptions. We plan to develop guidelines that help analysts locate appropriate information in documents which is then used in writing use-case descriptions. We then plan to conduct experiments that combine both reading and writing guidelines.

Appendix A: Experiment Information

Name:	Male/Female: Age:	
Previous systems analysis experience:		
Application / System	Industry, Academic or individual experience:	Number of Years:
Previous programming experience:		
Application / System	Industry, Academic or individual experience:	Number of Years:

How often do you shop in supermarkets?

Figure A1. Pre test form.

Experiment Instructions

Welcome. In the exercise you will be asked to write a use to describe typical interaction between a user and a system that you will be familiar with. First, you will be given a simple problem statement and asked to write a use case for it. This will last one hour.

During the exercise, write your use case on the sheets of paper provided. If there are any parts of this interaction that you cannot recall or are not familiar with, please make assumptions and add these assumptions to a separate piece of paper that we shall collect at the end of the exercise.

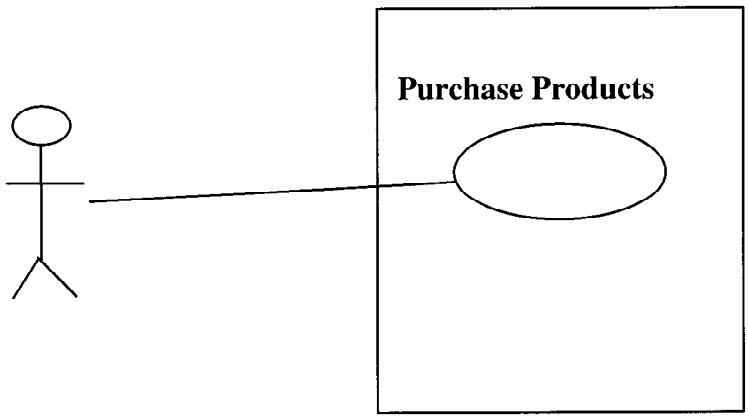
We would like you to try and write a use case which is as correct, complete and precise as possible. You have one hour to write the use case. This should be sufficient time to write a correct, complete and precise use case. Please use all the time available to complete the use case. Imagine that the use case will be later used to acquire important requirements for the system. Therefore, please write the use case in text form. Furthermore, most of you have been provided with guidelines for how to write use cases, above and beyond advice given as part of the use-case method. Please use these guidelines as you see fit to ensure the use case is correct, complete and precise.

Please ask the instructor any questions if you are unsure about how to proceed. Do not communicate with others during the exercise. If you are unclear about the problem domain under analysis, make assumptions and write them on a separate sheet of paper. The instructor will collect all use cases and assumptions at the end of the exercise. Does anyone have any questions before we start?

OK. Good luck. You have one hour to write the use case. Please turn over the page and read the problem statement and guidelines for writing the use case.

Use Case Question

A supplier of supermarket checkout machines is to produce a new checkout machine. It wants to use UML notation to develop the software system in the machine. A requirements analysis has revealed a number of important functions. A checkout operator must use each checkout machine to record purchases and receive payments from customers (including the use of club cards, payment using credit cards and the printing of receipts). An operator must be able to use the checkout to call the store manager when problems arise. The use case model for the system including the all actors and use cases is shown:



What you have to do

1. Please produce a use case description for the use case handling customer purchases. Show the basic course and all relevant variations and alternative courses. Please remember there is no right answer so make all relevant assumptions clear.

Counted Variables

Table A1 lists all the variables that were counted in replicating the CREWS experiment.

CREWS Expert Use Case

"1. The operator swipes the customer's club card.

2. The system reads customer information from the club card.

3. If the customer's card is valid then

    4. Repeat until no more products to be purchased

    4.1 The operator swipes the product label using the bar code reader.

        4.2 If the product information is retrieved

            4.3 The system displays the item name and price

            4.4 The system records purchase details from the product bar code.

5. The operator presses the 'transaction total' button.

6. The system displays the total amount due for the purchase.

7. The operator encashes the payment in the system till.

8. The operator takes the customer change from the system till.

9. The operator presses the 'transaction paid' button.

10. The system records the entire transaction.

11. The system updates the customer's club card details with the purchase information.

12. The system prints a receipt with all purchase details." (Ben Achour and Maiden, 1999).

CREWS also provide a selection of alternative courses as variations to the events in the main flow (see Ben Achour and Maiden, 1999).

Table A1. Counted variables.

Guideline	description
CG1	complete
CG2	complete
CG3	complete
CG4	complete
CG5	complete
CG6	complete
CG7	complete
CG8	complete
CG1–8	Complete actions
CG1–8	Incomplete actions
SG1	Atomic event
	Composite
SG2	Unique order
	Not unique order
	Separate variations
SG3	Iterations in main
	Alternatives in main
SG4	Homonyms
	Synonyms
	Anaphors
SG5	Not present tense
	Passive tense
	Negatives
SG6	Adverbs
	Modal verbs

No. Events <=> Expert UC

No. of lines in use cases:

Appendix B: Results

Table B1. Shows the group mean results and t-test results.

Groups:		**A: Control**		**B: SG**		**C: CG**		D: SGCG	
Guideline	**Description**	**M**	**M^2**	**M**	**M^2**	**M**	**M^2**	**M**	**M^2**
CG1	Complete	0	0	0	0	0	0	0	0
CG2	Complete	0.25	0.06	0.25	0.06	0	0	0	0
CG3	Complete	0.25	0.06	0	0	0	0	0	0
CG4	Complete	0	0	0	0	0.33	0.11	0.33	0.11
CG5	Complete	5.25	27.56	5.5	30.25	10.67	113.85	13.33	177.69
CG6	Complete	1.75	3.06	1.25	1.56	1.33	1.77	3.67	13.47
CG7	Complete	0	0	0	0	1	1	0.33	0.11
CG8	Complete	0	0	0	0	0	0	0.33	0.11
CG1–8	Complete actions	7.5	56.25	7	49	13.33	177.69	18	324
CG1–8	Incomplete actions	10.5	110.25	11.25	126.56	2.66	7.08	5.33	28.41
SG1	Atomic event	13.5	182.25	12.25	150.06	14	196	22	484
	Composite	4.5	20.25	6	36	2	4	1.33	1.77
SG2	Unique order	17.5	306.25	17.75	315.06	16	256	22.67	513.93
	Not unique order	0.5	0.25	0.5	0.25	0	0	0.67	0.45
	Separate variations	1.5	2.25	1.5	2.25	3	9	5	25
SG3	Iterations in main	1	1	1	1	0.67	0.45	1.33	1.77
	Alternatives in main	2.75	7.56	3	9	1.67	2.79	4.67	21.81
SG4	Homonyms	0	0	0	0	0	0	0	0
	Synonyms	0.5	0.25	1.25	1.56	0.67	0.45	1	1
	Anaphors	0	0	0	0	0	0	0	0
SG5	Not present tense	0	0	0.5	0.25	0	0	0	0
	Passive tense	3.75	14.06	2	4	0.33	0.11	2	4
SG6	Negatives	0.5	0.25	1	1	0	0	1	1
	Adverbs	0	0	0	0	0	0	0	0
	Modal verbs	0.25	0.06	0	0	0	0	0	0

(continued)

Table B1. (continued).

No. Events <=. Expert UC	3.25	10.56	2.25	5.06	5.33	28.41	5	25
No. of lines in use cases:	18	324	18.25	333.06	16	256	23.33	544.29
Guideline Description	**M**	**M^2**	**M**	**M^2**	**M**	**M^2**	**M**	**M^2**
Groups:	**A: Control**		**B: SG**		**C: CG**		**D: SGCG**	
Sum	93	1066.3	92.5	1066	88.99	1054.7	131.32	2167.91
Squared totals of individual scores	8649		8556.25		7919.2		17244.9	
Mean of Mean for each group	3.44		3.43		3.3		4.86	
t-test calculations:			**-0.0275**		**0.0548**		**-0.82**	
Df			**52**		**52**		**52**	
Lowest df in table $= 40$			**There is no significance in *t***					

(Key: M = Mean; M^2 = Mean squared)

References

Ben Achour, C. 1998. Guiding scenario authoring. *Proc. 8th European–Japanese Conference on Information Modeling and Knowledge Bases*. Finland, 181–200.

Ben Achour, C., Rolland, C., Maiden, N., and Souveyet, C. 1998. Guiding use case authoring: Results of an empirical study. *Proc. Fourth IEEE Int. Symposium on Requirements Engineering* University of Limerick, Ireland. Version taken from the CREWS web site: http://sunsite.informatik.rwth-aachen.de/CREWS/reports.htm. Report Number: 93-31.

Ben Achour, C., and Maiden, N. 1999. Empirical study of use case authoring: Experimental material. Internal Report. Centre de Recherche en Informatique, Universite de Paris 1–Sorbonne. http://panoramix.univ-paris1.fr/CRINFO/users/benachour/ESEM/.

Booch, G., Rumbaugh, J., and Jacobson, I. 1999. *The Unified Modelling Language User Guide*. Reading, MA: Addison-Wesley.

Cockburn, A. 1997. Using goal-based use cases. *Journal of Object-Oriented Programming* Nov–Dec.

Cockburn, A. 2000. *Writing Effective Use Cases: Part One*. In Print. Addison-Wesley.

Cox, K., and Phalp, K. T. 2000. A case study implementing the unified modeling language use-case notation version 1.3. *Proc. Sixth Int. Workshop on Requirements Engineering: Foundation for Software Quality*. Stockholm, Sweden.

Ericksson, H., and Penker, M. 1993. *UML Toolkit*. Wiley.

Fowler, M., and Scott, K. 2000. *UML Distilled Second Edition: A Brief Guide to the Standard Object Modelling Language*. Reading, MA: Addison-Wesley.

Greene, J., and D'Oliveira, M. 1982. *Learning to Use Statistical Tests in Psychology: A Student's Guide*. Bury St Edmunds, Suffolk: Open University Press.

Harwood, R. 1997. Use case formats: Requirements, analysis and design. *Journal of Object-Oriented Programming* January: 54–66.

Jacobson, I. 1987. Object-oriented development in an industrial environment. *Proc. OOPSLA'87*. Special Issue of *SIGPPLAN NOTICES* 22(12): 183–197.

Jacobson, I., Christerson, M., Jonsson, P., and Oevergaard, G. 1992. *Object-Oriented Software Engineering: A Use Case Driven Approach*. Addison Wesley.

Jarke, M., Pohl, K., Haumer, P., Weidenhaupt, K., Dubois, E., Heymans, P., Rolland, C., Ben Achour, C., Cauvet, C., Ralyte, J., Sutcliffe, A., Maiden, N., and Minocha, S. 1997. Scenario use in European software organisations—Results from the site visits and questionnaires. Taken from the CREWS web site: http://sunsite.informatik.rwth-aachen.de/CREWS/reports.htm. Report Number: 97-10.

Nardi, B. 1992. The use of scenarios in design. *SIGCHI Bulletin* 24(4): 13–14.
Pooley, R., and Stevens, P. 1999. *Using UML: Software Engineering with Objects and Components*. Reading, MA: Addison-Wesley.
Rational Software Corporation. 1997. UML notation version 1.1. http://www.rational.com/uml
Robson, C. 1993. *Real World Research.* Blackwell.
Rosenberg, G., with Scott, K. 1999 *Use Case Driven Object Modeling with UML: A Practical Approach*. Reading, MA: Addison Wesley.
Swan, M. 1980. *Practical English Usage*. Oxford University Press.
Weidenhaupt, K., Pohl, K., Jarke, M., and Haumer, P. 1998. Scenarios in System Development: Current practice. *IEEE Software* March/April: 34–45.

Karl Cox is a doctoral student at Bournemouth University, UK, where he is a member of the Empirical Software Engineering Research Group. He graduated with a History degree from the University of Wales Swansea in 1990. He then obtained a Masters with Distinction in Software Engineering from Bournemouth University in 1998. His research interests are in the uses of scenarios and use cases in requirements engineering.

Keith Phalp graduated from the University of Kent with a BSc in Mathematics in 1986. Having taught mathematics, he returned to further study, gaining a Masters in Software Engineering (1992), and a PhD in Computer Science (1995), both from Bournemouth University. He is a Chartered Mathematician, and a Member of the Institute of Mathematics and its Applications (IMA). Dr Phalp's research concentrates on the Empirical Study of the Software Development process, and specifically upon the efficacy of requirements techniques. He currently teaches software implementation, software design and process modelling at Bournemouth University, where he is Course Leader for the Masters in Computing.