



2007 Information Systems

Advanced Higher

Finalised Marking Instructions

© Scottish Qualifications Authority 2007

The information in this publication may be reproduced to support SQA qualifications only on a non-commercial basis. If it is to be used for any other purposes written permission must be obtained from the Assessment Materials Team, Dalkeith.

Where the publication includes materials from sources other than SQA (secondary copyright), this material should only be reproduced for the purposes of examination or assessment. If it needs to be reproduced for any other purpose it is the centre's responsibility to obtain the necessary copyright clearance. SQA's Assessment Materials Team at Dalkeith may be able to direct you to the secondary sources.

These Marking Instructions have been prepared by Examination Teams for use by SQA Appointed Markers when marking External Course Assessments. This publication must not be reproduced for commercial or trade purposes.

Question 1

Type	Part	Marking Instructions
KU	(a)	<p>For example:</p> <ul style="list-style-type: none"> • Background information about the company • Departmental objectives • Description of the components in the existing system • Description of procedures within the organisation • Existing input/output formats • Existing human computer interface <p>Award 1 mark each for any two. Max 2 marks.</p>
PS	(b)	<p>For example:</p> <p>Questionnaires could be used to gather information from customers or suppliers; they could also be used to gather info from employees since employees would be able to complete them in their own time and therefore not be taken away from doing their job.</p> <p>Staff operating the current system for the company could be interviewed by the analyst. This would be appropriate because the company is a small, local company that is likely to have few staff. The analyst would be able to speak face-to-face with the users of the system and gain first-hand knowledge of how the system works and what difficulties there are with the system.</p> <p>Company documentation including invoices and orders could be examined. This would be appropriate because it would allow the analyst to see the types of data and data items that need to be stored within the system. The analyst will be able to determine data types, data lengths and calculations needed in the stock and sales system without having to ask users of the system.</p> <p>The analyst could observe the tracking of stock/sales and the flow of information between personnel/departments involved. This would be appropriate because it would allow the analyst to see the system in use for himself. Any details that were overlooked by users of the system would be apparent to the analyst.</p> <p>Award 2 marks each for description of any two valid methods of information gathering. Each method must be supported by a full explanation of why the method is relevant for this situation. Award 2 marks for description only; award 0 marks for simply naming/identifying methods. Max 4 marks.</p>
KU	(c)	<p>For example:</p> <ul style="list-style-type: none"> • The System Specification outlines all functional requirements of the new system and will identify the scope and boundaries of the development project. It is based on the findings of the information gathering and indicates any restrictions placed on the new development – for example, the new system may have to be compatible with other systems and files already existing in the business. • The System Specification describes all processes to be carried out by the replacement system. It will state the data elements that are necessary in the system and their storage requirements. • The System Specification will describe the outputs that the system is required to produce and identify any constraints placed on the system. • The System specification is important because if it is inaccurate then the new development will be unsuccessful. The correctness of the final solution depends on the accuracy of the System Specification. <p>Award 2 marks for describing the contents of the System Specification. Award 1 mark max if description lacks detail. Award 1 mark for explaining the importance of the System Specification. Max 3 marks.</p>

Question 1 – continued

Type	Part	Marking Instructions
KU	(d) (i)	<p>For example: System testing would be carried out to make sure that the completed system <i>matches its requirements</i> and carries out all tasks that are required of it. Acceptance testing would be carried out by the client or end users of the system to make sure that the user is satisfied with the completed system and that it meets their needs.</p> <p>Award 1 mark each for correct description of the two types of testing of the system that would be carried out once the system has been fully developed. Component testing and integrative testing are not acceptable. No marks should be awarded for simply naming the types of testing. Max 2 marks.</p>
KU	(d) (ii)	<p>For example:</p> <ul style="list-style-type: none"> • Test data values to be used during testing • Expected output from test data values • Sequence of testing • Components of the system that are to be tested <p>Award 1 mark each. Max 2 marks.</p>
PS	(e)	<p>Direct changeover would be most appropriate because the current system is paper-based. By using direct changeover the benefits of the new system would be introduced immediately. As a result, productivity and profitability are likely to increase straight away. Also, there would be no duplication of running costs.</p> <p>Award 1 mark for each clear reason that justifies direct changeover; award max 1 mark for justification of parallel changeover. Award no marks for pilot or phased changeover. Max 2 marks.</p>
KU	(f) (i)	<p>Adaptive maintenance Award 1 mark</p>
KU	(f) (ii)	<p>Perfective maintenance Award 1 mark</p>
KU	(f) (iii)	<p>Adaptive maintenance Award 1 mark</p>
KU	(g)	<p>For example: Maintenance would start by analysing the current system and any new components that are necessary. An appropriate interface for new components would need to be designed along with any new code that is required. All new aspects of the system would need to be implemented by creating any necessary database structures. These would then be tested in isolation before full testing of the system could be carried out. Documentation for the upgraded system would need to be written. Finally, the upgraded system would be evaluated in order to determine the success of the maintenance work.</p> <p>Award 1 mark for correctly describing any 4 tasks that would be carried out at any 4 separate stages of the development process. Max 4 marks.</p>

Question 2

Type	Part	Marking Instructions																																												
PS	(a)	<table border="1"> <thead> <tr> <th></th> <th>Applicant</th> <th>Offer</th> <th>Enrolment</th> </tr> </thead> <tbody> <tr> <td>Receive application form</td> <td>C</td> <td></td> <td></td> </tr> <tr> <td>Make offer</td> <td>R</td> <td>C</td> <td></td> </tr> <tr> <td>Reject student</td> <td>M</td> <td></td> <td></td> </tr> <tr> <td>Archive applicant details</td> <td>D</td> <td></td> <td></td> </tr> <tr> <td>Student withdraws application</td> <td></td> <td>D</td> <td></td> </tr> <tr> <td>Student accepts offer</td> <td></td> <td>M</td> <td></td> </tr> <tr> <td>Enroll at start of course</td> <td></td> <td>R</td> <td>C</td> </tr> <tr> <td>Enrolled student withdraws from course</td> <td></td> <td></td> <td>D</td> </tr> <tr> <td>Student suspends studies</td> <td></td> <td></td> <td>M</td> </tr> <tr> <td>Student returns to course</td> <td></td> <td></td> <td>M</td> </tr> </tbody> </table> <p>Award 5 marks for correctly identifying each event and its effect on entities within the system. Award 1 mark for every 2 events recorded correctly. Max 5 marks.</p>		Applicant	Offer	Enrolment	Receive application form	C			Make offer	R	C		Reject student	M			Archive applicant details	D			Student withdraws application		D		Student accepts offer		M		Enroll at start of course		R	C	Enrolled student withdraws from course			D	Student suspends studies			M	Student returns to course			M
			Applicant	Offer	Enrolment																																									
		Receive application form	C																																											
		Make offer	R	C																																										
		Reject student	M																																											
		Archive applicant details	D																																											
		Student withdraws application		D																																										
		Student accepts offer		M																																										
		Enroll at start of course		R	C																																									
		Enrolled student withdraws from course			D																																									
		Student suspends studies			M																																									
Student returns to course			M																																											
PS	(b)	<pre> classDiagram class Enrolment class StartOfCourse[Start of course] class SuspendStudies[Suspend studies] class ReturnToStudies[Return to studies] class Withdraw class NewEnrolment[New enrolment] class EnrolmentLife[Enrolment Life] class RemoveEnrolment[Remove enrolment] Enrolment --> StartOfCourse : New enrolment Enrolment --> SuspendStudies : Enrolment Life * Enrolment --> ReturnToStudies : Enrolment Life * Enrolment --> Withdraw : Remove enrolment </pre> <p>Award 1 mark for create branch. Award 1 mark for delete branch. Award 1 mark for structure of modify branch. Award 1 mark for optionality/repetition in modify branch. Max 4 marks. Answers should be based on entity/event matrix produced in part (a) above.</p>																																												

Question 3

Type	Part	Marking Instructions
KU	(a) (i)	<p>For example: Graphical design notation is a visual representation of the flow of data and relationships between the functions and actions that take place within a process in a diagram. For non-technical people involved in the development (client and users of the system) this is much easier to understand than the more technical structured English explanations.</p> <p>Award 1 mark for explanation of benefit of graphical design notation; award 1 mark for direct comparison with structured English. Max 2 marks.</p>
KU	(a) (ii)	<p>For example: Structured English a restricted subset of the English language to define a process in clear, unambiguous style. This style is similar to the structure of a programming language without using the exact syntax of any specific language, can be easily coded and closely resembles a programming language. This makes it easier for the developers to carry out the implementation of the system than would be the case when using a graphical design notation. Also, for a large scale project, structured English can be more precise than the large complex diagrams that would be necessary using a graphical design notation.</p> <p>Award 1 mark for description of drawback of graphical design notation; award 1 mark for direct comparison with structured English. Max 2 marks.</p>
PS	(b) (i)	<p>For example: RAD systems provide a number of tools to help build graphical user interfaces that would normally take a large development effort. RAD systems are client-driven. The interface is developed in small increments that result in increased customer satisfaction without sacrificing quality while delivering the product as quickly as possible. Using RAD tools, the processes of design and implementation are concurrent and no detailed user interface design documentation is needed. Development of a user interface using RAD relies on extensive user involvement. The involvement of users in prototyping is an important factor in the early detection of errors.</p> <p>Award 1 mark each for any two supporting arguments. Max 2 marks.</p>
PS	(b) (ii)	<p>For example: Since RAD tools rely on a number of in-built tools to build the user interface, all systems created using RAD tools will have a very similar interface and operation. This would be a problem if a company required a system with an interface that was customised to their exact requirements. The use of RAD tools can generate inefficient code due to duplication and overheads. Interface is restricted to those components provided within the RAD tool. This could limit the type of interface that can be produced.</p> <p>Award 1 mark for description of appropriate problem; award 1 mark for explanation of how this problem could arise. Max 2 marks.</p>

Question 4

Type	Part	Marking Instructions
PS		<p>For example:</p> <p>The diagram shows the following relationships and cardinalities:</p> <ul style="list-style-type: none"> look-alike (strong entity) is connected to allocated (strong relationship) with cardinality 1:1. look-alike (strong entity) is connected to issued (strong relationship) with cardinality 1:M. issued (strong relationship) is connected to contract (strong entity) with cardinality 1:M. booking (weak entity) is connected to allocated (strong relationship) with cardinality M:1. booking (weak entity) is connected to makes (strong relationship) with cardinality M:M. makes (strong relationship) is connected to customer (strong entity) with cardinality 1:1. customer (strong entity) is connected to taken out (strong relationship) with cardinality 1:M. taken out (strong relationship) is connected to policy (strong entity) with cardinality 1:M. <p><i>Accept either mandatory or optional</i></p> <p>Award max 6 marks as follows:</p> <ul style="list-style-type: none"> Weak/strong entities correctly indicated – award 1 mark. Note that weak entities must be <i>clearly</i> indicated for this mark to be awarded. Weak/strong relationships correctly indicated – award 1 mark. Note that weak relationships must be <i>clearly</i> indicated for this mark to be awarded. Mandatory/optional relationships correctly indicated – award max 2 marks for all correct; award 1 mark for 1 error; award 0 marks for more than 1 error. Cardinality of relationships correctly indicated – award max 2 marks for all correct; award 1 mark for 1 error; award 0 marks for more than 1 error.

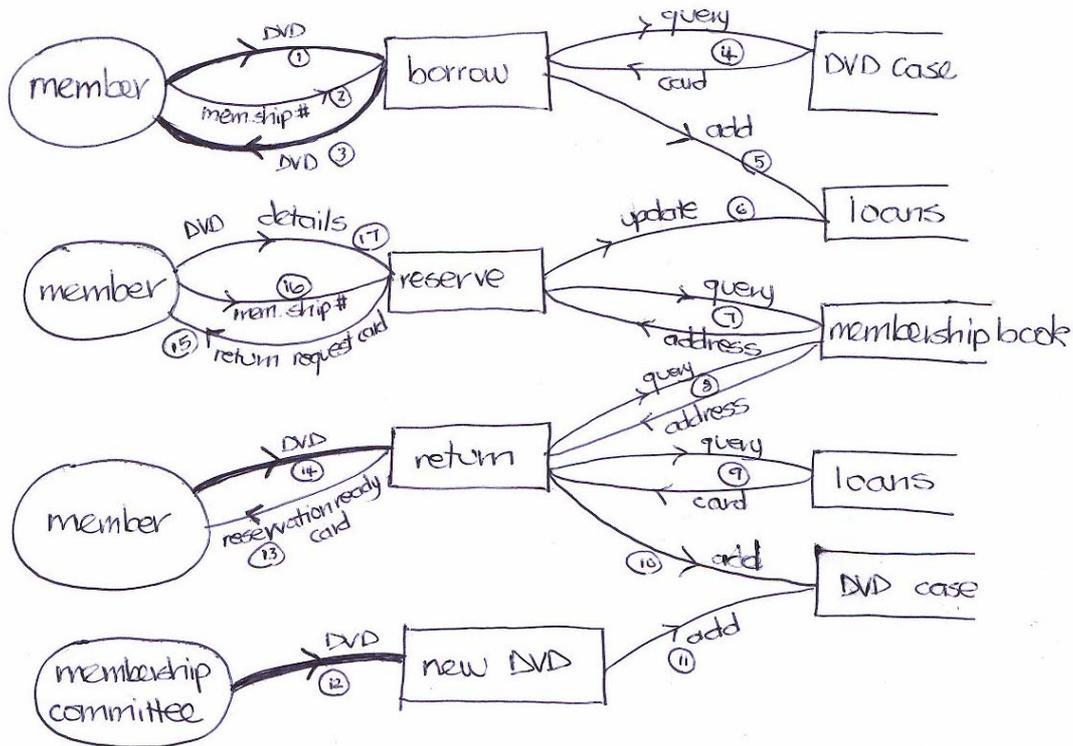
Question 5

Type	Part	Marking Instructions
PS		<p>For example:</p> <p>CustomerID → Name, Address</p> <p>CustomerID, ProductID → Date, Quantity</p> <p><i>Or</i></p> <p>CustomerID, ProductID, Date → Quantity</p> <p>ProductID → Name, Price</p> <p>Award 1 mark each. Max 3 marks</p>

Question 6

Type	Part	Marking Instructions
PS		<p>Solution should indicate 17 data flows (including 4 physical), 4 processes (borrow, reserve, return, receive new) and 3 stores (DVD case, loans file, membership book). See sample solution below.</p> <p>Award max 12 marks as follows:</p> <p>Correct processes award max 2 marks. Deduct 1 mark for each omission (max 2 deductions).</p> <p>Correct data stores award 2 marks. Deduct 1 mark for each omission (max 2 deductions).</p> <p>16/17 correct data flows – award 8 marks 14/15 correct data flows – award 7 marks 12/13 correct data flows – award 6 marks 10/11 correct data flows – award 5 marks 8/9 correct data flows – award 4 marks 6/7 correct data flows – award 3 marks 4/5 correct data flows – award 2 marks 2/3 correct data flows – award 1 marks</p>

Sample Level 1 Data Flow Diagram



Question 7

Type	Part	Marking Instructions
KU	(a) (i)	<p>For example: The purpose of the envision stage is to create a clear, shared vision of the product. At the end of the envision stage, a User Interface roadmap is produced. This document describes the design vision, manages expectations and serves as a basis for evaluating the progress throughout the project. The envision stage produces a concise description of the product, identifies the business and design objectives, identifies constraints on the project, identifies the target audience and identifies the main functionality of the product. Award 1 mark each for accurate description of any 2 objectives of the envision stage. Max 2 marks.</p>
KU	(a) (ii)	<p>Primarily, designers of the user interface, but client and end-users of the system may be involved Award 1 mark for designer (and possibly client/users)</p>
PS	(b) (i)	<p>For example: Textual mode: a form could be created to allow the receptionist to make appointments for clients and enter appointment details Graphical mode: screen could incorporate buttons and icons that are selected to active processes and operations Award 1 mark for appropriate suggestion of each type of interface mode. Max 2 marks.</p>
PS	(b) (ii)	<p>For example: Auditory interface would require speech input and/or produce speech output. This would be inappropriate since a hairdressing salon will have a lot of background noise that would make it difficult for the system to recognise speech input and difficult for receptionist to hear the speech output. Each user of the system would need to train the system to understand his/her speech patterns. This training by multiple users would be extremely difficult and time consuming. Award 1 mark for description of any two relevant difficulties presented by auditory interface. Max 2 marks.</p>
KU	(c) (i)	<p>For example: Prototyping can be used to show clients how the finished product will look. This will enable dialogue to take place between the client and the design team at an early stage. As a result, any possible difficulties can be identified before the actual implementation work is undertaken. This reduces the likelihood of problems arising when acceptance testing is carried out. Prototypes are experimental and incomplete designs which are cheaply and quickly developed. The main purpose of prototyping is to involve users in testing design ideas and get early feedback in the early stage of development. This reduces time and cost. Award 1 mark for description of purpose of prototyping; award 1 mark for description of importance; Max 2 marks.</p>
KU	(c) (ii)	<p>Low fidelity prototypes are paper-based sketches/storyboards of the user interface whereas high fidelity prototypes are close to final product with lots of detail and functionality. This means that:</p> <ul style="list-style-type: none"> • Very little time spent developing and producing the prototype • Several prototypes can be produced to represent alternatives and each of these can be tested • Non-programmers can participate in the development of the prototypes <p>Award 1 mark each for any 2 appropriate benefits of low-fidelity prototyping in comparison to high fidelity. Max 2 marks.</p>

Question 7 – continued

Type	Part	Marking Instructions					
PS	(d) (i)	Entity	Attribute	Type	Size	Valid	Index
		Appointment	Stylist	Text	4	Existing stylist	Y (PK)
		Appointment	Client ID	Text	6	Existing customer	Y (FK)
		Appointment	Treatment Code	Text	3	Existing treatment	Y (FK)
		Appointment	Date	Date			Y (PK)
		Appointment	Time	Time			Y (PK)
		<p>Award 1 mark for type/size that matches data provided; award 1 mark for correct primary key indicated; award 1 mark for correct validation indicated; award 1 mark for correct indexing of each attribute. Max 4 marks. Note: PK = Client ID + Date + Time or PK = Stylist # + Date + Time</p>					
KU	(d) (ii)	<p>A weak entity is an entity that cannot be identified from its attributes alone – it needs to have a foreign key along with its own attributes to create a primary key. In this example, appointment is a weak entity – all other entities are strong. Award 1 mark for description of weak entity that makes reference to appointment in this case</p>					
KU	(d) (iii)	<p>For example: A weak relationship exists between 2 entities when the primary key of one entity requires an attribute from its owner entity to form a unique identifier. All other relationships are strong. Award 1 mark</p>					

Question 8

Type	Part	Marking Instructions
KU	(a)	<p>For example:</p> <p>Heuristic evaluation: a technique to find usability issues in the design of an interface by having a group of “evaluators” examine the interface and rate it against recognised usability principles.</p> <p>Walkthroughs: a usability inspection method which is based on the notion that users typically prefer to learn a system by using it to accomplish tasks, rather than, for example, studying a manual. It starts with a task analysis that specifies the sequence of steps or actions required by a user to accomplish a task, and the system responses to those actions. The designers and developers of the software then walkthrough the steps as a group, asking themselves a set of questions at each step. Data is gathered during the walkthrough, and afterwards a report of potential issues is compiled. Finally the software is redesigned to address the issues identified.</p> <p>Feature set: a list of all the functions the application is capable of performing. This can help identify omissions or unnecessary features.</p> <p>Consistency inspection: a quality control technique for evaluating and improving a user interface. The interface is methodically reviewed for consistency in design, both within a screen and between screens, in graphics (colour, typography, layout, icons), text (tone, style, spelling), and interaction (consistency of task steps and command names).</p> <p>Adherence to standards: (including standard operating system design guidelines, web accessibility guidelines) – differences amongst applications increase learning times and can cause frustration, inefficiency, errors and delay.</p> <p>Award 1 mark each for detailed description of any three recognised methods. Max 3 marks.</p>
PS	(b)	<p>For example:</p> <p>Natural language querying: A natural language query is one that is expressed using normal conversational syntax. There are no syntax rules or conventions to learn. Museum visitors could use a keyboard to input direct questions about the location of departments, or where to find specific exhibits or amenities.</p> <p>Speech driven software: museum visitors could issue oral commands, instructions or queries to the kiosk. Users could respond to prompts from the kiosk with voice commands.</p> <p>Machine translation: the attempt to automate all, or part of the process of translating from one human language to another. Visitors could choose their native language and use natural language querying – the system would then have to translate these queries.</p> <p>Award 1 mark for correct description of any 2 techniques; award 1 mark for explanation of how technique relates to the scenario. Max 4 marks.</p>

Question 8 – continued

Type	Part	Marking Instructions
PS	(c)	<p>For example:</p> <p>Surveys – member of museum staff would be on-hand to ask questions; depending on user response, appropriate next question can then be posed; sections can be skipped and clarification can be sought if required; an explanation of what is required can be provided by the member of staff making it easier for members of the public to answer appropriately</p> <p>Questionnaires – these could be issued to members of the public and could provide a high volume of responses without the need to employ someone to actually carry out a survey</p> <p>User performance data logging – could be used to determine overall usage and track use made of individual functions/menus/options/features. This method is appropriate here since it would be carried out by the system itself and wouldn't interrupt or interfere with the users' experience of the system</p> <p>Award 1 mark each for recommendation of appropriate methods; award 1 mark for justification of each method. The justification must relate to scenario. Note that self-reporting logs would not be appropriate in this situation. Max 4 marks.</p>

Question 9

Type	Part	Marking Instructions
KU	(a) (i)	<p>For example: Whether or not current technology will support the requirements of the proposed system Whether or not the developers have the technical ability to create the proposed system Whether or not the proposed system will run on existing hardware/software; compatibility with existing hardware; compatibility with OS Award 1 mark each for any 2 appropriate suggestions. Max 2 marks.</p>
KU	(a) (ii)	<p>For example: When development of the proposed system would infringe copyright of an existing product When the development of the proposed system fails to take account of restrictions imposed by Data Protection Act Protection of Intellectual Properties Award 1 mark for any appropriate suggestion</p>
PS	(b)	<p>For example: Input – use of touch screen described/justified Output – use of 19” screen described/justified Many others possible but all require appropriate description of use/justification Award 1 mark each for appropriate description of use made of relevant input & output devices suggested. Devices must be suitable for age group and for the application described. Max 2 marks.</p>
PS	(c)	<p>For example: Time to learn – measuring the performance of the children in terms of the time needed to adapt to the interface provided for the graphics application. This is important because it would give an indication of how easy the graphics application was for young children to use. Speed of task performance – measuring the performance of the children in terms of the time taken to carry out a particular task. This is important because it would hopefully show that the task can be carried out more quickly with the new application than using existing packages. User error rates – this records the number of errors made by the children over a period of time. This is important because it would hopefully show a decrease in the number of errors being made over a fixed period of time. Retention of commands over time and subjective user satisfaction also acceptable but should be supported with suitable justification. Award 1 mark for selection of appropriate technique; award 1 mark for justification that relates to the scenario. Max 2 marks.</p>

KU	(d) (i)	<p>Thinking aloud</p> <p>Advantage – requires little expertise to carry out, user is encouraged to criticise the system, the evaluator can clarify points of confusion at the time they occur, thought processes are externalised, particularly successful technique with children; user can proceed at their own pace</p> <p>Disadvantage – user may provide large amount of irrelevant detail; users could become silent in which case there would be no progress</p> <p>Question asking</p> <p>Advantage – provides a direct and structured method of gathering information, levels of questions can be varied to suit the context and the users can reveal problems not anticipated by the designer</p> <p>Disadvantage – unless carefully worded, questions may be misunderstood or misinterpreted by young children, unnatural situation for young children</p> <p>Award 1 mark for each appropriate advantage/disadvantage. These must all be different for full marks to be awarded. Max 4 marks.</p>
PS	(d) (ii)	<p>(A) question asking</p> <p>(B) thinking aloud</p> <p>Award 1 mark each. Max 2 marks.</p>
PS	(e)	<p>For example:</p> <p>Factors such as age of user and reading ability will need to be taken into account. Documentation must be very simplistic. Written explanations will need to be avoided as far as possible. Consideration could be given to the inclusion of an auditory interface for the user documentation.</p> <p>Award 1 mark each for identifying any 2 factors that should be taken into account. Max 2 marks.</p>

Question 10

Type	Part	Marking Instructions
PS	(a) (i)	<p>For example: Novice user has limited/no experience of such a system and therefore could make use of –</p> <ul style="list-style-type: none"> • Wizard: step by step instructions to complete a required task • Menu: easy to identify required action from given selection • Icons: easy to identify actions • Other valid <p>Award 1 mark for level of user experience; award 1 mark for selection of an appropriate feature with justification. Max 2 marks.</p>
PS	(a) (ii)	<p>For example: Knowledgeable intermittent would be someone who was familiar with similar systems but perhaps hasn't used them for some time. They would be likely to use –</p> <ul style="list-style-type: none"> • Keyboard short cuts: use of underlined letter to access menus/options • Custom tools created by user to meet need • Other valid <p>Award 1 mark for level of user experience; award 1 mark for selection of an appropriate feature with justification. Max 2 marks.</p>
PS	(a) (iii)	<p>For example: Experienced/frequent user would be someone who was very familiar with such systems. These users are likely to make use of –</p> <ul style="list-style-type: none"> • Custom tools created by user to meet need • SQL Interface: command line interface allowing work state to be saved • Scripting editor: custom scripting to meet user needs • Other valid <p>Award 1 mark for level of user experience; award 1 mark for selection of an appropriate feature with justification. Max 2 marks.</p>
PS	(b)	<p>For example: Operation – Cut Syntax – (1) available from toolbar Syntax – (2) available as menu option</p> <p>Award 1 mark each for correct syntax of any suitable operation illustrated in the questions. Max 2 marks.</p>
PS	(c) (i)	<p>Interface should have the following:</p> <ul style="list-style-type: none"> • All fields in data dictionary extract should be shown and sizes should be appropriate • Appropriate labelling of controls • Clear indication of restricted choice for collection type • Clear/cancel button • Submit/add button <p>Award 1 mark for each of these features of the interface. Max 5 marks.</p>
PS	(c) (ii)	<p>Collection type could make use of predictive interface</p> <p>Award 1 mark</p>
PS	(d) (i)	<p>For example: 25, 56, 74, 87</p> <p>Award 1 mark for four valid test data values.</p>
PS	(d) (ii)	<p>For example: 0, 100 and (optionally) any two additional items</p> <p>Award 1 mark for 0 and 100 included in candidate response.</p>
PS	(d) (iii)	<p>For example: A, &, -12, 110</p> <p>Award 1 mark for four valid test data values.</p>

Question 11

Type	Part	Marking Instructions
PS	(a)	<p>For example, Gantt Chart:</p> <ul style="list-style-type: none"> • Alter amount of time allocated to tasks • Rearrange tasks to change dependencies • Reallocate resources from non critical to critical events • Make graphics concurrent with other tasks • Add new resources • Utilise float time <p>Award 1 mark for each issue described (max 4 issues). Issues must be relevant to the project management technique named. Max 4 marks.</p>
KU	(b)	<p>For example:</p> <ul style="list-style-type: none"> • Content submissions by authors/site members – users can contribute to articles and so site can contain user-generated content • Content rating/cross linking/references – content can be allocated a score and linked to other content • Publication/Approval of submissions – site administrator can approve and manage site content • Content editing – members can edit existing content • Stylesheets/templates – content kept separate from formatting and styling • Others valid <p>Award 2 marks each for detailed description of each relevant feature of CMS (max 3 features). Award 1 mark each for brief description of each relevant feature. Award 0 marks for naming features without providing a description. Max 6 marks.</p>
PS	(c) (i)	<p>For example: 25, 56, 74, 87</p> <p>Award 1 mark for four valid test data values.</p>
PS	(c) (ii)	<p>For example: 0, 100 and (optionally) any two additional items</p> <p>Award 1 mark for 0 and 100 included in candidate response.</p>
PS	(c) (iii)	<p>For example: A, &, -12, 110</p> <p>Award 1 mark for four valid test data values.</p>

Question 12

Type	Part	Marking Instructions
PS	(a)	<p>For example: Database server holds the tables of the underlying databases needed to support the historyofmytown web site. It understands the SQL commands and instructions that are used to query the tables. Award 1 mark each for accurate description of any two features of the database server. Note that answers must relate to the scenario. Max 2 marks.</p>
PS	(b)	<p>For example: Server-side scripting language is the go-between – it understands both HTML and SQL instructions. It processes the user request and fetches the data from the database. The server-side scripting language returns query results dynamically as the nicely-formatted HTML page that the browser expects. Award 1 mark each for accurate descriptions of any two features of server-side scripting language. Note that answers must relate to scenario. Max 2 marks.</p>

Question 13

Type	Part	Marking Instructions
PS	(a) (i)	<p>For example: Table would be needed to store details in product catalogue. Details would include description of products, product code and price. Table would be needed to store details of registered customers. Details would include customer ID, name, contact details, payment details, delivery details. Table would be needed to store details of each order. This would include customer ID, product code, quantity, delivery date and delivery time. Award 1 mark each for correctly suggesting contents of each table required. Max 3 marks.</p>
PS	(a) (ii)	<p>For example: Product table would be queried by customers looking to purchase particular items. Customer table would be updated when a new customer registers with the company. A new record would be added to the Order table every time an order was created. Award 1 mark each for appropriate suggestion for use made of each table in the system. Max 3 marks.</p>
KU	(b) (i)	<p>For example: Transaction standardisation defines the data elements required, a definition of each data item in terms of type and size, technical transaction formats that are used for transmission of data and code sets/values that can appear in the selected data items for transmission; provides a standard format that can be used with all parties concerned. This ensures the compatibility of data and allows secure transmission of data. Ensures that multiple developers can work on components of the system and share data between them; needed to allow exchange of data between trading partners since each trading partner is likely to have its own data standards/formats. Award 1 mark for description of transaction standardisation. Award 1 mark for explanation of why transaction standardisation is necessary for this application. Max 2 marks.</p>
KU	(b) (ii)	<p>For example: Data Protection Act states that personal details of customers must be held securely and must be accurate. Computer Misuse Act states that accessing details without authorisation is illegal. Award 1 mark each for accurate description of any two legal restrictions that would apply in this situation. Max 2 marks.</p>

Question 14

Type	Part	Marking Instructions
KU	(a)	<p>For example:</p> <p>Security</p> <ul style="list-style-type: none"> • As code open-source, weaknesses in coding may be quickly identified leaving the system vulnerable to attack. • Company may add own coding to provide for security appropriate to their needs. • May take less time between discovery of flaw and time to fix/patch. <p>Flexibility and adaptability</p> <ul style="list-style-type: none"> • Ability to alter the software and redistribute it amongst a community [similar businesses] without cost [except development time]. • Potential for higher quality software due to wide range of possible contributors. • Ability to alter code to particular needs/functionality. <p>On-going support</p> <ul style="list-style-type: none"> • Each user may be seen as a possible developer, providing a wide range of support. • Speedier patching of faults found in open-source software. • Internet forums with many users to provide tips. <p>Award 2 marks each for security, flexibility/adaptability, and on-going support. Award 1 mark for each clear point made. Max 6 marks.</p>
PS	(b)	<p>In PHP</p> <pre>\$dbserver = mysql_pconnect('dbserver.724hosting.com', 'dbadmin', 'hollywood247');</pre> <pre>\$dbselected= mysql_select_db(' dbocrm ', \$dbserver);</pre> <p>In ASP (direct server connection)</p> <pre>Server=dbserver.724hosting.com; Database=dbocrm; UserID=dbadmin; Password=hollywood247; Trusted_Connection=False;</pre> <p>In ASP (via IP)</p> <pre>Data Source=dbserver.724hosting.com;Network Library=DBMSSOCN;Initial Catalog=dbocrm;User ID=dbadmin;Password=hollywood247;</pre> <p>Award 4 marks for correct script. 1 error – 3 marks, 2 errors – 2 marks, 3 errors – 1 mark, 4 or more errors – 0 marks. Max 4 marks.</p>
PS	(c)	<p>For example, benefit:</p> <ul style="list-style-type: none"> • Ability to customise the response based on the user requirements, access rights or queries on databases. • Provides interactive web sites that interface with databases. • User is not able to see the source code – therefore more secure. <p>For example, drawback:</p> <ul style="list-style-type: none"> • The language interpreter must be installed on the server. • If not properly secured, scripts may be used to gain access to the server. • Reduced response time. <p>Award 1 mark for any sensible benefit; award 1 mark for any sensible disadvantage. Max 2 marks.</p>

Question 14 – continued

Type	Part	Marking Instructions
PS	(d)	<p>For example:</p> <pre><form action="complaints.php" method="get" name="complaint"> <input type="text" name="customer_name"> <input type="text" name="date_of_complaint"> <textarea name="details_of_complaint" cols="40" rows="5"> </textarea> <input type="Submit" value="Submit Complaint"> </form></pre> <p>Award max 4 marks as follows: Form tag correct – 1 mark Each input type correct – 1 mark Textarea correct – 1 mark (Note that the values for cols and rows must give 200 when multiplied) Submit button correct – 1 mark There may be other valid syntax which should be given credit.</p>

Question 15

Type	Part	Marking Instructions
KU	(a)	<p>For example: A data dictionary is a set of metadata that contains definitions and representations of the data items within the relational database model. During implementation it is used to create table structures.</p> <p>Award 1 mark each for any two different uses made during implementation. Max 2 marks.</p>
KU	(b)	<p>DML – Used to change database data. Example: delete, insert or update data DQL – Used to get data from the database and impose ordering upon it. Example: Select</p> <p>Award 1 mark each. Max 2 marks.</p>
PS	(c)	<p>INSERT INTO Games (<i>Title, Developer, Genre, Price</i>) VALUES (“City Racing” , “Gamesoft”, “Driving”, 34.99)</p> <p>Award 1 mark for INSERT INTO clause and detail (Note that details in italics are optional) Award 1 mark for VALUES clause and detail Note that sequence of clauses must be correct – deduct 1 mark if incorrect. Max 2 marks.</p>
PS	(d)	<p>UPDATE Games SET Price=“20.00” WHERE Genre=“Sports”</p> <p>Award 1 mark for $\left\{ \begin{array}{l} \text{UPDATE clause and detail} \\ \text{and} \\ \text{SET clause and detail} \end{array} \right.$ Award 1 mark for WHERE clause and detail Note that sequence of clauses must be correct – deduct 1 mark if incorrect. Max 2 marks.</p>
PS	(e)	<p>SELECT COUNT(Orders.Title) AS Total_Orders, * FROM Orders, Games WHERE Orders.Title = Games.Title ORDER BY Total_Orders</p> <p>Award 1 mark for SELECT COUNT clause Award 1 mark for FROM clause and detail Award 1 mark for WHERE clause and detail Award 1 mark for ORDER BY clause Note that sequence of clauses must be correct – deduct 1 mark if incorrect. Max 4 marks.</p>
PS	(f)	<p>SELECT MemberID, MemberName, DeliveryAddress, (MemberEmail), Title, Developer, Genre, Price FROM Members, Orders, Games WHERE Members.MemberID=Orders.MemberID AND Orders.Title = Games.Title AND OrderDate Between “20061201” AND “20061231”</p> <p>Award 1 mark for SELECT clause and detail Award 1 mark for FROM clause and detail Award 3 marks for WHERE clause as follows:</p> <ul style="list-style-type: none"> • MemberID – award 1 mark • Title – award 1 mark • Date criteria – award 1 mark <p>Note that sequence of clauses must be correct – deduct 1 mark if incorrect. Max 5 marks.</p>

[END OF MARKING INSTRUCTIONS]