Task 1 Activity A (ii) – The Proposal

<table>
<tr><th colspan="5">Indicative content and marker guidance</th></tr>
<tr><td colspan="5">Learners proposals may refer to:
Creation of a solution to support health and fitness including social features such as apps like ‘Runtastic’, ‘Strava’ etc
How the solution will handle data and pass it between back end and front end
‘Social issues’ – e.g. integration with APIs for social media sites, developing a website that utilises a back-end database and ‘forum’, etc
How they will address issues such as:
Providing membership levels for free and ‘paid for’ content
Storing and using data to provide and enable customizable workout and eating plans
Supporting and providing Digital content on the chosen platform
Learners rationale should make reference to wider issues which are likely to include:
General computer related risks such as:
Security
Privacy
User support
Potential context-specific risks such as:
handling of personal data
incorrect instructions that lead to injuries
users selecting levels of training not at the appropriate level
Relevant regulations and guidelines as well as legal requirements relating to software development and the industrial context.
Learners rationale should make reference to current practice and emerging tech in the health and fitness industry such as:
Apps and mobile devices
Use of wearables
Video/remote instruction</td></tr>
<tr><th>Assessment focus</th><th>Band 0</th><th>Band 1</th><th>Band 2</th><th>Band 3</th></tr>
<tr><td></td><td>0</td><td>1-3</td><td>4-6</td><td>7-9</td></tr>
<tr><td>Decomposing the problem</td><td rowspan="5">No rewardable material</td><td>The proposal:
identifies some of the problems to be solved
Effectively decomposes some of the problems identified
The proposed solution would effectively:
meet some of the needs of the client and users
mitigating some of the potential risks
addressing some of the relevant regulatory guidelines and legal requirements, in relation to software development and the industry.</td><td>The proposal:
Identifies most of the problems to be solved
Effectively decomposes most of the problems identified
The proposed solution would effectively:
meet most of the needs of the client and users
mitigate most of the potential risks
address most of the relevant regulatory guidelines and legal requirements, in relation to software development and the industry.</td><td>The proposal:
Fully identifies the problems to be solved
Effectively decomposes the problems identified
The proposed solution would effectively:
meet the full needs of the client and users
mitigate the potential risks
address relevant regulatory guidelines and legal requirements, in relation to software development and the industry.</td></tr>
<tr><td></td><td>1-3</td><td>4-6</td><td>7-9</td></tr>
<tr><td>Appreciation of wider issues in the context</td><td>The proposal provides limited lines of reasoning that partially justify how:
the recommended solution meets the needs of the client and users
potential risks will be mitigated
the proposed solution will address relevant regulatory guidelines and legal requirements, in relation to software development and the industry.</td><td>The proposal provides good lines of reasoning that mostly justify how:
the recommended solution meets the needs of the client and users
potential risks will be mitigated
the proposed solution will address relevant regulatory guidelines and legal requirements, in relation to software development and the industry.</td><td>The proposal provides comprehensive lines of reasoning that fully justify how the:
the recommended solution meets the needs of the client and users
potential risks will be mitigated
the proposed solution will address relevant regulatory guidelines and legal requirements, in relation to software development and the industry.</td></tr>
<tr><td></td><td>1-2</td><td>3-4</td><td>5-6</td></tr>
<tr><td>Appreciation of the business context</td><td>The proposal provides basic definitions of:
functional and non-functional requirements
key performance indicators
user acceptance criteria</td><td>The proposal provides good definitions of:
functional and non-functional requirements
key performance indicators
user acceptance criteria</td><td>The proposal provides comprehensive and perceptive definitions of:
functional and non-functional requirements
key performance indicators
user acceptance criteria</td></tr>
</table>

Task 1 Activity B – The Design – The Visual/Interface Design

Indicative content and marker guidance

The design should aim to incorporate the following characteristics:
 	
Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part to another should be clearly shown in the design
 	
The design should always show clients what the users are informed about, what is going on, through appropriate feedback messages or snippets
 	
The design should not contain information which is irrelevant or rarely needed. Every piece of extra information competes with the relevant information and diminishes their relative visibility. So, the use of images which have little or no relevance to the product or brand should be avoided.
 	
The design should also consider the following factors:
 	
• Consistency and standards
• Aesthetic and minimalist design
• Help and documentation
• Consistent fonts and colours and imagery which help users know that all pieces of the solution are working together and are pieces of a consistent whole.
• Use of space
  ○ Space between lines, called leading, is important for readability.
  ○ Ensure the space is not too small to avoid making it harder for readers to follow the line properly.
  ○ However, too much space, readers can wander away from the text altogether, check for lots of white space; however, depending on the design does not necessarily mean it wasted space.
• Look for imagery used in the websites or app as much as possible because this further helps users skim the content and digest material quickly and easily.

Assessment focus	**Band 0**	**Band 1**	**Band 2**	**Band 3**
	0	**1-2**	**3-4**	**5-6**
Effectiveness of the design interface	No rewardable material	The proposed design interface is **adequate** as a result of **reasonably effective** use of: ● layout and white space ● visual hierarchies ● common conventions.	The proposed design interface is **good** as a result of **the effective** use of: ● layout and white space ● visual hierarchies ● common conventions.	The proposed design interface is **excellent** as a result of the **sophisticated** and **highly effective** use of: ● layout and white space ● visual hierarchies ● common conventions.

Task 1 Activity B – The Design –Algorithm Design

Indicative content and marker guidance

Decomposition coverage

Learners should select some key processes such as:

- Collection and processing of data
- Communication/data exchange between different platforms, front-end and backend
- Key calculations such as calorie intake tracking, BMI, BMR etc.
- Data filtering and visualisation

Algorithms

Maybe in the form of flowcharts, pseudocode, data flow diagrams, static and dynamic model diagrams or a combination of all three

- the steps are clearly defined
- each step are uniquely defined they should depend on the input and the result of the preceding steps
- the algorithm stops after a finite number of instructions are executed, two key constructs iterate and decide
- receives input, how much and what data is required
- produces output type of output, e.g. print out, return value or return list, what results are required, what happens if no results can be computed may be an error
- Links to API, CSV or Database file
- Sensible names
- Use of keywords
- Indentation
- No programming syntax
- Flow charts with the correct use of symbols

Assessment focus	Band 0	Band 1	Band 2	Band 3
	0	1	2-3	4
Decomposition of problem	No rewardable material	Basic decomposition of the identified problems that superficially cover the required: • inputs • processes • outputs	Good decomposition of the identified problems that sufficiently cover the required: • inputs • processes • outputs	Highly effective decomposition of the identified problems that comprehensively cover the required: • inputs • processes • outputs
		1 - 2	3 -4	5 - 6
Application of logical thinking and conventions		Algorithms would produce some correct outcomes as a result of: • some precise logic • some appropriate structure and sequence which is likely to be inefficient. Some effective use of accepted conventions although inconsistencies still exist.	Algorithms would produce mostly correct outcomes as a result of: • mostly precise logic • appropriate structure and sequence but which may lack efficiency. Mostly effective use of accepted conventions though some minor inconsistencies may still exist.	Algorithms would produce consistently correct outcomes as a result of: • precise logic • efficient structure and sequence. Effective and consistent use of accepted conventions

Task 1 Activity B – The Design – The Data Requirements

Indicative content and marker guidance				
Data design maybe in the form of: • data dictionaries, • entity-relationship diagrams, • data flow diagrams, • static and dynamic model diagrams or a combination as appropriate to describe the planned solution Note – data normalisation may not be required depending on the identified/proposed solution.				
Assessment Focus	**Band 0**	**Band 1**	**Band 2**	**Band 3**
	0	**1-2**	**3-4**	**5-6**
The design of the data requirements	No rewardable material	Data requirements for the proposed solution are somewhat appropriate, including (as required): • variables • data structures • data types Naming conventions used are mostly appropriate but are inconsistent. Effective error handling procedures are identified for some inputs/processes that require them.	Data requirements for the proposed solution are mostly appropriate, including (as required): • variables • data structures • data types Naming conventions used are appropriate and mostly consistent. Effective error handling procedures are identified for most inputs/processes that require them.	Data requirements for the proposed solution are fully appropriate, including (as required): • variables • data structures • data types Thoroughly appropriate and consistent naming conventions are used throughout. Thoroughly effective error handling procedures are identified for the inputs/processes that require them.

Task 1 Activity B – The Design – The Test Strategy

Indicative content and marker guidance
Look for the most appropriate test strategies used: • Black-box • White-box Testing should: • Provides mechanisms to trace tests to product objectives and their associated priorities. • Engage all stakeholders were possible • Makes it easy to create, view, and report linkage between requirements, test cases, test data, test scripts, test results, and defects. • be evident in some form at every stage of the development life cycle • Improves decomposition. • Ensures that components and subcomponents all come together to make a useful, functional system, by relating the lower-level requirements to the higher-level requirements. • Checks to see that some pieces have been put together and implement late-stage changes

Assessment focus	**Band 0**	**Band 1**	**Band 2**	**Band 3**
	0	**1-2**	**3-4**	**5-6**
Test strategy	No rewardable material	The test strategy demonstrates a basic understanding of: • how components interrelate • the order in which components should be tested • the types of test that are required.	The test strategy demonstrates a good understanding of: • how components interrelate • the order in which components should be tested • the types of test that are required.	The test strategy demonstrates a thorough and detailed understanding of: • how components interrelate • the order in which components should be tested • the types of test that are required.

Task 1 Activity B – The Design – The Design Documentation

Indicative content and marker guidance				
Appropriateness of communication: • suitability for the intended audience • clarity • use of technical language • choice of tools / how information is presented.				
	No rewardable material	**1-2**	**3-4**	**5-6**
Quality of communication		Some effective communication of the design as a result of: • some use of appropriate techniques, methods and formats • some use of technical language that is appropriate for the intended audience	Mostly effective communication of the design as a result of: • the use of mostly appropriate techniques, methods and formats • the use of technical language that is mostly appropriate for the intended audience	Communication of the design is consistently effective as a result of: • the use of consistently appropriate techniques, methods and formats • the use of technical language that is consistently appropriate for the intended audience

Task 2 - Developing the solution

The solution

Indicative content and marker guidance

Functionality:

- Must demonstrate the use of two different languages, JS and PHP or SQL. Python and SQL.
- The techniques used are appropriate and demonstrate a level of technical skill
- Maybe complex data model in database interlinking more than one table
- Recursive algorithms – reminder that if used in PHP here must be a mechanism (IF statement, etc.) that stops the recursion after the desired result has been found
- Server-side scripting using request and response objects and server-side extensions for a complex client-server model
- Web service APIs and parsing JSON/XML to service a complex client-server model
- Use of MySQL, the use of DML and DDL
- Various outcomes which may be basic in terms of very linear structure to more logically and efficient approach. The use of function, procedures and Classes
- The solution will be well structured and modular in nature – clearly see subsections this may be separated modules
-

Code organisation:

- For top marks should avoid multiple pages of nested if clauses and for loops with a lot of copy-pasted procedural code
- Clear and meaningful indentation
- Code should consist of pieces of logic, classes or objects, with proper structure.
- comments should be used where ever possible to help explain the logic
- good use of local variables and minimal use of global variables
- use of constants
- well-designed interface
- consistent style throughout
- defensive programming
- good exception handling

User Experience considerations:

- Useful – does it meet the needs of the user? Are outputs etc. accurate
- Usable – is it easy to use / intuitive?
- Desirable – is it aesthetically pleasing? Does it represent a brand? Consistency.
- Findable – is it easy to navigate around and find the required information/features
- Accessible – what accessibility features have been included? Are these well implemented? Are they sufficient?
- Credible – is the information credible?

Assessment focus	Band 0	Band 1	Band 2	Band 3	Band 4
	0	1-2	3-4	3-5	6-8
Functionality	No rewardable material	The prototype implements code in a single language with some functionality but the code lacks efficiency and some major errors persist. Uses some precise logic and programming structures which would result in some correct outcomes	The prototype implements code with some functionality in at least two different languages but the code lacks efficiency and some major errors persist. Uses sufficient precise logic and programming structures which would result in adequate correct outcomes.	The prototype implements mostly efficient functional code in at least two different languages, but some minor errors still persist. Uses mostly precise logic and programming structures which would result in mostly correct outcomes.	The prototype implements consistently efficient functional code in at least two different languages. Uses precise logic and programming structures throughout which would result in consistently correct outcomes.
		1-2	3-4	3-5	6-8
Code organisation		Code is maintainable by a third party, but would present significant difficulties through the use of: • inconsistent naming conventions. • limited logical organisation • limited informative commenting.	Code is maintainable by a third party, but would present some difficulties through the use of: • somewhat appropriate naming conventions. • Some logical organisation • some informative commenting.	Code is maintainable by a third party, and would present only a few minor difficulties through the use of: • mostly appropriate naming conventions. • mostly logical organisation • mostly informative commenting.	Code is easily maintainable by a third party through the use of consistently appropriate: • naming conventions. • logical organisation • informative commenting.

		1-2	**3-4**	**3-5**	**6-8**
User experience		Basic user experience is provided through limited effective use of: • input handling • user guidance and error messages • outputs The solution is partially robust and effectively handles some common errors	Adequate user experience is provided through somewhat effective use of: • input handling • user guidance and error messages • outputs The solution is adequately robust and effectively handles sufficient common and unexpected errors	Good user experience is provided through mostly effective use of: • input handling • user guidance and error messages • outputs The solution is largely robust and effectively handles most common and unexpected errors	Excellent user experience is provided through consistently effective use of: • input handling • user guidance and error messages • outputs The solution is fully robust and effectively handles common and unexpected errors
		1-2	**3-4**	**5-6**	[redacted]
Legal and regulatory guidelines and Standards		Some effective application of standards and guidelines in relation to: • accessibility • compatibility • legal and ethical considerations Some effective application of procedures and security controls to ensure confidentiality, integrity and availability.	Mostly effective application of standards and guidelines in relation to: • accessibility • compatibility • legal and ethical considerations A mostly effective application of procedures and security controls to ensure confidentiality, integrity and availability.	Consistent and effective application of standards and guidelines in relation to: • accessibility • compatibility • legal and ethical considerations Thoroughly effective application of procedures and security controls to ensure confidentiality, integrity and availability.	

Task 2 – Developing the solution

Testing

Indicative content and marker guidance

Test execution should contains numerous stages and may include different types of software testing with different data to thoroughly test the solution. Data is needed for performance, stress, and load testing and can also be achieved through the use of testing software.

There should be evidence that when errors are identified procedures to correct these are implemented and regression testing is applied to ensure no additional errors have been introduced as a result of the changes made.

Test data can be grouped according to different parameters such as;

- **test-specific data**: influence the system behaviour and reveal the case specifics under the test
- **test-reference data**: have little influence on the test performance
- **application reference data:** irrelevant to the behaviour under test, and are needed to start the application
- **valid test data**: does the system functions are in compliance with the requirements, does the system processes and stores the data as intended
- **invalid test data**: check to see if the software correctly processes invalid values, shows the relevant messages, and notifies the user that the data are improper
- **boundary test data:** help to reveal the defects connected with processing boundary values
- **wrong data:** entering the data of inappropriate format, whether it shows the correct error messages thus showing the use of validation if appropriate
- **absent data:** should check that the solution handles entering a blank field

Assessment focus	Band 0	Band 1	Band 2	Band 3
	0	1-2	3-4	5-6
Suitability of test data	No rewardable material	Tests selected show a basic understanding of how to effectively test inputs, calculations, validation and processes using test data which makes limited use of: • normal data • erroneous data • extreme data	Tests selected show a good understanding of how to effectively test inputs, calculations, validation and processes using test data which includes: • normal data • erroneous data • extreme data	Tests selected show a thorough and detailed understanding of how to effectively test inputs, calculations, validation and processes using test data which includes: • normal data • erroneous data • extreme data
		1-2	**3-4**	**5-6**
Use of testing to inform the iterative development process		Comments show a basic understanding of how errors/problems were identified and how they were rectified (as appropriate) for: • inputs • calculations • validation and processes Testing shows evidence of a basic iterative development process	Comments show a good understanding of how errors/problems were identified and how they were rectified (as appropriate) for: • inputs • calculations • validation and processes Testing shows evidence of a good iterative development process	Comments show a comprehensive understanding of how errors/problems were identified and how they were rectified (as appropriate) for: • inputs • calculations • validation and processes Testing shows evidence of an effective iterative development process

Task 2 – Developing the solution

Documentation

Indicative content and marker guidance

The learner should demonstrate an iterative approach which consists of repeating the following four phases in sequence:

- **Requirements phase**, in which the requirements for the software are gathered and analysed, eventually result in a completed and final specification of requirements.
- **Design phase** that the software solution meets the requirements is designed. This may be a new design or an extension of an earlier design.
- **Implementation and Test phase**, when the software is coded, integrated and tested. Using the correct and appropriate test strategy
- **Review phase**, in which the software is evaluated, the current requirements are reviewed, and changes and additions to requirements proposed.

For each iteration, documentation should demonstrate a decision has been made as to whether the software produced in this phase will be discarded, or kept as a starting point for the next iteration. Should be rigorous validation of requirements, and verification (including testing) of each version of the software against those requirements within each iteration.

Assessment Focus	Band 0	Band 1	Band 2	Band 3
	0	1-2	3-4	5-6
Quality of iterative development process	No rewardable material	A basic iterative development process is demonstrated, including: ● Limited and/or superficial records of changes made throughout the development stage ● A superficial or vague rationale for changes made. ● Some effective use of versioning	An adequate iterative development process is demonstrated, including: ● Adequate recording of notable changes made throughout the development stage ● a supported rationale for some notable changes made ● mostly effective use of versioning	An effective iterative development process is demonstrated, including: ● Thorough and detailed recording of notable changes made throughout the development stage ● Convincing and perceptive rationales for notable changes made ● fully effective use of versioning

Task 3 Part A – Gathering feedback to inform future development

Indicative content and marker guidance

Types of feedback ad materials used to support the gathering of feedback might include:

- screencasts to demonstrate the prototype to both technical and non-technical audiences' user
- questionnaires
- records of observation of users
- paired coding review records.

Testers/feedback users might include:

- range of ages and abilities
- gym users
- programming professionals.

Assessment focus	Band 0	Band 1	Band 2	Band 3	Band 4
	0	**1-3**	**4-6**	**7-8**	**9-12**
Effectiveness of materials to support the feedback process	No rewardable material	The materials would allow for the gathering of limited quality feedback for different aspects of the developed prototype	The materials would allow for the gathering of adequate quality feedback for different aspects of the developed prototype	The materials would allow for the gathering of good-quality feedback for different aspects of the developed prototype	The materials would allow for the gathering of high-quality feedback for different aspects of the developed prototype.
	0	**1-2**	**3-4**	**5-6**	
Use of appropriate feedback tools to support the gathering of effective feedback	No rewardable material	The use of the tools has resulted in feedback that provides some opportunity for evidence-informed further iteration	The use of the tools has resulted in feedback that mostly provides the opportunity for evidence-informed further iteration.	The use of the tools has resulted in feedback that consistently provides the opportunity for evidence-informed further iteration.	
		1-2	**3-4**	**5-6**	
Effectiveness of communication	No rewardable material	Quality of communication is only sometimes effective for both technical and non-technical audiences as a result of limited use of appropriate techniques, methods and formats Limited use of technical language that is appropriate for the intended audience	The quality of communication is mostly effective for both technical and non-technical audiences as a result of the use of mostly appropriate techniques, methods and format The use of technical language that is mostly appropriate for the intended audience	The quality of communication is effective for both technical and non-technical audiences as a result of the consistent use of appropriate techniques, methods and formats The use of technical language that is consistently appropriate for the intended audience	

Task 3 Part B– Evaluating feedback to inform future development

Indicative content and marker guidance

#updated each series by the LE to provide scope and assistance for the markers#

Legal considerations - licencing, copyright etc.

Possible content collected:

- 3rd party data (e.g. to simulate customer data)
- Multimedia content (e.g. fitness videos, images)
- Snippets of pre-written code
- Any other appropriate information (e.g. health and fitness information).

Assessment Focus	Band 0	Band 1	Band 2	Band 3
	0	1-2	3-4	5-6
Effectiveness of assets and content	No rewardable material	A limited review of the content selected, including superficial consideration of the: • appropriateness of the assets selected • validity and reliability of the sources used • legal and ethical implications of using the identified assets. The review is only sometimes supported by superficial consideration, comparison and corroboration across multiple sources.	A good review of the effectiveness of the content selected, including good consideration of the: • appropriateness of the assets selected • validity and reliability of the sources used • legal and ethical implications of using the identified assets. The review is mostly supported by good consideration, comparison and corroboration across multiple sources.	A comprehensive review of the effectiveness of the content selected, including thorough consideration of the: • appropriateness of the assets selected • validity and reliability of the sources used • legal and ethical implications of using the identified assets. The review is well supported by effective consideration, comparison and corroboration across multiple sources.
		1-3	4-6	7-9
Evaluation of project outcomes		A basic or superficial evaluation of how well the prototype meets: • functional and non-functional requirements of the system. • key performance indicators (KPIs) • user acceptance criteria for the proposed system. Basic and/or simplistic rationale for future iteration is provided. Points made are supported by limited relevant: • selection of examples • consideration of feedback.	A good evaluation of how well the prototype meets: • functional and non-functional requirements of the system. • key performance indicators (KPIs) • user acceptance criteria for the proposed system. Good rationale for future iteration is provided. Points made are supported by mostly relevant: • selection of examples • consideration of feedback.	A thorough and detailed evaluation of how well the prototype meets: • functional and non-functional requirements of the system. • key performance indicators (KPIs) • user acceptance criteria for the proposed system. Convincing and perceptive rationale for future iteration is provided. Points made are supported by an entirely relevant and perceptive: • selection of examples • consideration of feedback.