T-LEVELS | IfATE Institute for Apprenticeships and Technical Education

T Level Technical Qualification in Digital Software Development

Mark Scheme

Specimen Assessment Materials

Occupational Specialism

General Marking Guidance

- All students must receive the same treatment. Examiners must mark the first student in exactly the same way as they mark the last.
- Mark schemes should be applied positively. Students must be rewarded for what they have shown they can do rather than penalised for omissions.
- Examiners should mark according to the mark scheme not according to their perception of where the grade boundaries may lie.
- All marks on the mark scheme should be used appropriately.
- All the marks on the mark scheme are designed to be awarded. Examiners should always award full marks if deserved. Examiners should also be prepared to award zero marks if the student's response is not rewardable according to the mark scheme.
- Where judgement is required, a mark scheme will provide the principles by which marks will be awarded.
- When examiners are in doubt regarding the application of the mark scheme to a student's response, a senior examiner should be consulted.
- Crossed out work should be marked unless the student has replaced it with an alternative response.
- Accept incorrect/phonetic spelling (as long as the term is recognisable) unless instructed otherwise.

Levels-Based Mark Scheme Guidance

Levels-based mark schemes (LBMS) have been designed to assess students' work holistically. They consist of two parts:

1. **Indicative content**
   Indicative content reflects content-related points that a student might make but is not an exhaustive list. Nor is it a model answer. Students may make some or none of the points included in the indicative content as its purpose is as a guide for the relevance and expectation of the responses. Students must be credited for any appropriate response.
2. **Levels-based descriptors**
   Each level is made up of a number of traits which when combined together articulate the quality of response that a student needs to demonstrate. The traits progress across the levels to demonstrate the different expectations of each level. When using a levels-based mark scheme, the 'best fit' approach should be used.

Applying the levels-based descriptors

Examiners should take a ‘best fit’ approach to determining the mark.

- Examiners should first make a holistic judgement on which level most closely matches the student’s response. Students will be placed in the level that best describes their answer. Answers can display characteristics from more than one level, and where this happens markers must use any additional guidance (e.g. weighting of traits) and their professional judgement to decide which level is most appropriate.
- The mark awarded within the level will be decided based on the quality of the answer and will be modified according to how securely all traits are displayed at that level:
  - Marks will be awarded at the top of that level if the student has evidenced each of the descriptor traits securely.
  - Where the response does not securely meet all traits, the marks should be awarded based on how closely the descriptor has been met.

Task 1: Activity A – Proposal

Indicative content and marker guidance

Students' proposals should be relevant to key requirements of the brief:

- interactive resources to support healthy living, fitness and exercise
- encouraging users to make use of ToKa Fitness's services which could include:
    - – personal training sessions
    - – facilities in the gym
  - making a distinction between paid-for and free content
  - providing access to customisable workout and eating plans
  - social features.

Students' proposals may refer to:

- how the solution will handle data and pass it between back end and front end including:
  - client- and server-side storage requirements (e.g. all stored on ToKa Fitness's system versus potentially storing customised plans etc on the user's computer)
  - APIs to integrate ToKa Fitness materials with other systems (e.g. YouTube videos, social media integration)
- how they will address issues/needs, such as:
  - interactive resources and functionality to support healthy living (e.g. BMI calculators, exercise/activity logs)
  - booking/managing personal training sessions
  - booking/managing gym classes
  - managing user accounts
  - storing and tracking users' personal data (e.g. fitness goals, weight and performance metrics)
  - storing and using data to provide and enable customisable training and eating plans
  - supporting and providing digital content across different platforms/devices (e.g. use of videos of exercises posted by trainers, clients posting videos for a remote trainer to analyse and provide feedback)
  - collaborative systems, such as trainer and client editing training and meal plans, setting goals

- accessibility features (e.g. alt text, resizable fonts, selectable colour schemes) – possible reference to W3C
- 'social' features such as:
  - integration with APIs for social media sites or fitness-specific platforms such as Strava
  - accessing and using data from fitness trackers/smart watches to utilise personal training data.

Students' proposals should refer to relevant risks and how they will mitigate these, such as:

- malicious/inappropriate use – data input validation
- intercepting data/man-in-the-middle attacks – use of SSL/encryption/HTTPS
- cross-site scripting – content security policy/data input validation and sanitisation
- SQL injection – sanitising data input.

Students' rationale should refer to wider issues which are likely to include:

- general issues, such as:
  - privacy
  - user support
  - feedback on user errors (e.g. incorrect completion of forms, completion of transactions)
  - handling payment details
- potential context-specific issues such as:
  - handling of personal data (e.g. training data, health metrics) demonstrating an understating of the sensitive nature of this data
  - personal privacy and safety concerns, such as the location and route data that is captured by GPS tracked activities such as running and cycling)
  - legal and ethical issues such as:
    - dangers associated with reducing/gaining weight and calorie intake and the need to provide support and warnings for users
    - potential for injury if attempting a workout plan/exercise without appropriate supervision/training
- copyright/intellectual property issues relating to videos and plans
- ensuring materials are at the appropriate level for each user
- ensuring only authorised users access paid-for content

- relevant regulations and guidelines (e.g. W3C, professional bodies) as well as legal requirements relating to software development and the industrial context.

Students' rationale should be relevant to current practice and/or emerging tech in the leisure (health and fitness) sector, such as:

- apps and mobile devices
- fitness trackers
- at-home training and online classes (e.g. Peloton®)
- video/remote instruction (e.g. YouTube videos)
- use of AR/VR to enhance training.

Students' rationale should include details of the functional and non-functional requirements, key performance indicators and user acceptance criteria that could be used to judge the success of a solution that they will implement. They must be appropriate for a solution that will meet the needs of ToKa Fitness.

Students' proposals should, where appropriate, provide justification for why a particular feature of design consideration works. They could refer to considerations they discovered during research, which may include (but may not be limited to):

- established conventions (e.g. layouts, icons)
- common functionality which users would expect
- improving poor functionality/common complaints about other apps/platforms
- providing unique functionality.

Assessment focus	Band 0	Band 1	Band 2	Band 3
	0	1–3	4–6	7–9
Decomposing the problem	No rewardable material	The proposal: • identifies some of the problems to be solved • basic decomposition of the problems identified. The proposed solution would partially: • meet the needs of the client and users • mitigate the potential risks • address the relevant regulatory guidelines and legal requirements in relation to software development and the industry.	The proposal: • identifies most of the problems to be solved • adequate decomposition of the problems identified. The proposed solution would mostly: • meet the needs of the client and users • mitigate the potential risks • address the relevant regulatory guidelines and legal requirements in relation to software development and the industry.	The proposal: • fully identifies the problems to be solved • effective decomposition of the problems identified. The proposed solution would fully: • meet the 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.

Assessment focus	Band 0	Band 1	Band 2	Band 3
	0	**1–3**	**4–6**	**7–9**
Appreciation of wider issues in the context	No rewardable material	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.	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.	The proposal provides comprehensive lines of reasoning that fully 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.
	0	**1–2**	**3–4**	**5–6**
Appreciation of the business context	No rewardable material	The proposal provides basic definitions of: • functional requirements • non-functional requirements • user acceptance criteria.	The proposal provides good definitions of: • functional requirements • non-functional requirements • user acceptance criteria.	The proposal provides comprehensive and perceptive definitions of: • functional requirements • non-functional requirements • user acceptance criteria.

Task 1: Activity B – Visual Interface and Data Designs

Indicative content and marker guidance

The designs should provide details of the solution that the student will implement.

The designs should be usable by a third party to implement the intended solution:

- layout and white space may include:
  - clear ‘zones’ which allow users to focus on specific content
  - space between lines to aid readability
  - related information which is close together but does not encroach on other information
  - sensible breaks/separation of information to avoid overload
  - details of reactive layouts to suit different screens/devices (e.g. mobile and desktop versions)
- visual hierarchies may include:
  - sensible and appropriate use of sizing of information/items to help optimise user information
  - order of design elements (e.g. orders in navbars and menus) to signify significance or ‘route’ through information
- common conventions including common design features which may include (but not be limited to):
  - common/recognisable icons (e.g. house = home)
  - placement of navigation item (top or left)
  - use of ‘hamburger’ button/collapsible menus for mobile layout.

The designs should also consider the following factors:

- consistency and standards
- aesthetic and minimalist design
- help and documentation
- consistent fonts, 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
  - the space should not be too small to avoid making it harder for readers to follow the line properly
  - if there is too much space, readers can wander away from the text. Check for lots of white space – depending on the design, this is not necessarily wasted space
- use of imagery on the websites or apps to help users skim 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 simplistic. There is partially effective use of: • layout and white space • visual hierarchies • common conventions.	The proposed design interface is good. There is mostly effective use of: • layout and white space • visual hierarchies • common conventions.	The proposed design interface is excellent. There is fully effective use of: • layout and white space • visual hierarchies • common conventions.

Task 1: Activity B – The Data Requirements

Indicative content and marker guidance

Data design in the form of, for example:

- data dictionaries
- entity-relationship diagrams
- data flow diagrams
- static and dynamic model diagrams
- a combination as appropriate to describe the planned solution.

Note: data normalisation may not be required depending on the identified/proposed solution.

Data considered should be appropriate for the needs of ToKa Fitness, which may include:

- client and trainer accounts
- schedules and calendars for classes and personal training sessions
- sport/fitness metrics, e.g. number of activities/sessions completed in a week, level of demand, duration of exercise, type of exercise
- personal and health data, e.g. age/DOB, weight (current and target), heart rate, other medical issues
- progress data – identification of filtered data for public and private use, e.g. comparing times and progress with friends, such as improvements in max weight, distance covered, vs personal goals and info
- usage stats, click-through/conversion from free to paid-for, uptake of personal training sessions, classes etc.

The design should show an understanding of error-handling procedures which may include:

- data validation rules
- input masks
- type casting
- feedback to user/error messages
- limiting field/variable length.

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 partially appropriate, including (as required): • variables • data structures • data types. Limited error-handling procedures are identified. Naming conventions used are partially appropriate but are inconsistent.	Data requirements for the proposed solution are mostly appropriate, including (as required): • variables • data structures • data types. Adequate error-handling procedures are identified. Naming conventions used are mostly appropriate and consistent.	Data requirements for the proposed solution are fully appropriate, including (as required): • variables • data structures • data types. Effective error-handling procedures are identified. Fully appropriate and consistent naming conventions are used throughout.

Task 1: Activity C – Use of Generative AI

Indicative content and marker guidance

Problems that could be addressed – students should use AI to generate some code for some key problems to be solved, such as:

- validating a telephone number/email address against specific requirements
- communication/data exchange between different platforms, front end and back end
- key domain-specific functionality and calculations, such as calorie intake tracking, BMI calculator
- data filtering and visualisation
- providing access to paid-for and free content, such as:
  - APIs for user login (e.g. join with Facebook/Google)
  - APIs for payment services
  - storing and protecting data in the back end (e.g. generating and writing to tables, password hashing)
- providing access to customisable workout and eating plans
- social features, such as:
  - internal message boards/forums.
  - linking to other platforms such as social media and/or fitness-specific platforms such as Strava.

Recording the process – students should record the process they followed to generate code that can be used in their solution, which should include:

- record of the prompts entered to generate the output (including a clear indication of how these have been altered to refine the appropriateness of the AI models' output)
- review of the code the AI has output, including:
  - strengths and weaknesses of the code produced, such as:
    - how well it implements the required functionality (e.g. does it fulfil a specific required functionality such as BMI calculation or activity tracking?)
    - robustness of code (e.g. divide by zero errors, validation of a range of reasonable exceptions, SQL injections/input sanitation for back-end processes)

- the extent to which it meets students' needs, including details of any further refinement required, such as:
  - use of adapted prompts within the AI model to further refine the code
  - manual adjustments and additions to the code (e.g. altering variable names, adding additional functionality that the AI model did not include).

Problem solving

Level of coverage of the problem should be judged by the range and complexity of the identified problems to be solved.

Simplistic coverage is likely to consider only basic front-end processes, with little consideration of how these may relate to other components within the solution. Students are likely to limit their exploration to a single programming language.

At the higher end, the coverage is likely to consider a range of processes related to other components within the solution. Students will address individual problems and areas of functionality that address different types of problem appropriate to the specific demands of the context of the set task brief.

Use of prompts

Quality of the prompts should be judged based on the conciseness and accuracy of the language used.

Lower-attaining students are likely to use more generic terms, with little consideration of the specific task/problem they are trying to solve.

At the higher end, the prompts will be concise and will concentrate on specific problems with a narrower scope. Students will refine their prompts after each iteration of the code in order to produce a more appropriate outcome.

Output review

Lower-attaining students are likely to review the code in more general terms, such as the overall functionality of the code produced. They are likely to show a much more limited technical understanding of the code and/or focus on just positives or negatives of the code. Their review will likely be very narrow in scope.

At the higher end, students will consider strengths and weaknesses in relation to their prompts and the task at hand. They will review the output in a more granular fashion, identifying specific refinements that will need to be made.

Assessment focus	Band 0	Band 1	Band 2	Band 3
	0	**1**	**2**	**3**
Problem solving	No rewardable material	Superficial coverage of the task requirements.	Good coverage of the task requirements.	Effective coverage of the task requirements.
	0	**1**	**2**	**3**
Use of prompts	No rewardable material	Prompts would allow a model to produce some correct outcomes as a result of: • some precise language • limited consideration of the requirements of the task.	Prompts would allow a model to produce mostly correct outcomes as a result of: • mostly precise language • mostly appropriate consideration of the requirements of the task.	Prompts would allow a model to produce consistently correct outcomes as a result of: • highly precise language • effective consideration of the requirements of the task.
	0	**1–2**	**3–4**	**5–6**
Output review	No rewardable material	A basic review of how well the generated code meets their needs as a result of superficial consideration of: • positives and negatives of the code generated • the refinements that may need to be made.	A good review of how well the generated code meets their needs as a result of appropriate consideration of: • positives and negatives of the code generated • the refinements that may need to be made.	An effective review of how well the generated code meets their needs as a result of thorough consideration of: • positives and negatives of the code generated • the refinements that may need to be made.

Task 1: Quality of Communication

Indicative content and marker guidance

The quality of communication should be assessed across Task 1 Activity A, Activity B and Activity C. This includes the following:

- appropriateness of communication:
  - uses appropriate technical vocabulary
  - is suitable for the target audience
  - ensures the different elements of the documentation are clear, allowing a third party to easily interpret them
  - use of common vocabulary:
    - there is good use of specific data-related technical vocabulary
    - there is good use of terms commonly used within the digital industry.
- specific examples may include:
  - proposal:
    - clear presentation of functional/non-functional requirements, e.g. use of tables and/or bullet points to present the specific measurable requirements, with expansion and explanations given in a separate part/paragraph
    - specific and measurable requirements
    - use of technical language, including an appropriate balance between technical language for some areas of the designs and technical specifications, with less technical language to explain and justify decisions made to the client
  - visual designs:
    - clear outline/wireframe drawing
    - inclusion of specific style and design decisions – higher-attaining students may provide a detailed style guide covering aspects of the design that will be applied across the design, providing greater clarity than littered/excessive annotation on the layout design itself
  - data designs:
    - combination of visual and written (e.g. ERD and data dictionary)
    - use of accepted conventions, such as notation for relationships on ERD, combined with suitable descriptions on the data dictionary for less technical audiences.

Assessment focus	**Band 0**	**Band 1**	**Band 2**	**Band 3**
	0	**1–2**	**3–4**	**5–6**
Quality of communication	No rewardable material	Superficial communication that uses partially appropriate: • techniques, methods and formats • technical language for the intended audience.	Good communication that uses mostly appropriate: • techniques, methods and formats • technical language for the intended audience.	Effective communication that uses consistently appropriate: • techniques, methods and formats • technical language for the intended audience.

Task 2: Activity A – Developing the Solution

Indicative content and marker guidance

Functionality

- Must demonstrate the use of two different languages, JS and PHP or SQL, or Python and SQL.
- The techniques used are appropriate and demonstrate a level of technical skill.
- May be complex data model in database interlinking more than one table.
- Recursive algorithms – reminder that if used in PHP there 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 logical and with an efficient approach. The use of function, procedures and classes.
- The solution will be well structured and modular in nature.

The program should ensure correct outcomes to meet the needs of ToKa Fitness. For example:

- handling user access, e.g. setting up an account, logging in, changing password
- collection and processing of data, e.g. starting weight/physical measurements, target weight and/or target performance goal
- communication/data exchange between different platforms or between front-end and back-end systems, e.g. analysing exercise data provided manually or from a fitness tracker, measurement of progress from starting point to make recommendations and provide encouragement, pulling in data to personalise meals and/or training
- key calculations, e.g. performance improvements, rate of weight gain/loss to ensure healthy progress, calculate BMI or other measurement benchmarks
- data filtering and visualisation, e.g. performance progress, usage stats.

Code organisation

- Students 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 wherever 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

- Inputs from the user are handled appropriately, such as:
- validation and sanitisation
- ease of user input, e.g. GUI features, input masks
- well-designed interface including UI/UX features:
    - 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 and find the required data/info/functionality?
- user guidance and error messages ensure that the product is easy to use and helpful, which may include:
    - short and meaning full error messages (avoid jargon)
    - provides information about solution/corrective action that the user can take
    - help/instructions and/or meaningful feature labels to reduce errors

- a robust solution which may include:
    - good exception handling that can deal with unexpected or incorrect outputs
    - ‘fall back’ code to deal with different systems and platforms without crashing
    - appropriate use of validation.

Legal and regulatory guidelines and standards

- Accessibility has been considered and there should be evidence that the design takes different users into account, such as:
    - selectable accessibility features (e.g. adjustable fonts, colours)
    - sensible font and colour base colour selection.
- Compatibility with different platforms has been considered (where appropriate), which may include:
    - creating a standalone/self-contained program
    - implementation of web standards/program is accessible through a web browser
    - ‘fall back’ code to deal with different systems and platforms.
- Legal and ethical aspects have been considered as appropriate to their solution, which may include:
    - issues around providing advice relating to health and wellbeing, concerns of negative impacts of training plans and advice on weight loss etc.
    - cookie consent notice
    - GDPR considerations
    - privacy and data policy/informing the user
    - intellectual property considerations (see asset log)
    - terms and conditions.

Assessment focus	Band 0	Band 1	Band 2	Band 3	Band 4
	0	**1–2**	**3–4**	**5–6**	**7–8**
Functionality	No rewardable material	The prototype implements code in at least one language with basic functionality. The code lacks efficiency and some major errors persist. Uses basic precise logic and programming structures which would result in some correct outcomes.	The prototype implements code with partial functionality in at least two different languages. 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. Some minor errors persist. Uses mostly precise logic and programming structures which would result in mostly correct outcomes.	The prototype implements highly efficient functional code in at least two different languages. Any errors that exist are minimal. Uses fully precise logic and programming structures throughout which would result in consistently correct outcomes.
	0	**1–2**	**3–4**	**5–6**	**7-8**
Code organisation	No rewardable material	Code would present significant difficulties for maintainability through the use of inconsistent and limited: • naming conventions • logical organisation • commenting.	Code is maintainable by a third party, but would present some difficulties through the use of somewhat appropriate: • naming conventions • logical organisation • commenting.	Code is maintainable by a third party and would present only a few minor difficulties through the use of mostly appropriate: • naming conventions • logical organisation • commenting.	Code is easily maintainable by a third party through the use of consistently appropriate: • naming conventions • logical organisation • informative commenting.

Assessment focus	**Band 0**	**Band 1**	**Band 2**	**Band 3**	**Band 4**
	0	**1–2**	**3–4**	**5–6**	**7–8**
User experience	No rewardable material	Basic user experience is provided through limited effective use of: • input handling • user guidance and error messages • outputs. The solution is basic and effectively handles some common errors.	Adequate user experience is provided through some effective use of: • input handling • user guidance and error messages • outputs. The solution is adequately robust and effectively handles some 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 mostly 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 most common and unexpected errors.

Assessment focus	Band 0	Band 1	Band 2	Band 3	Band 4
	0	1–2	3–4	5–6	
Legal and regulatory guidelines and standards	No rewardable material	Superficial application of standards and guidelines in relation to: • accessibility • compatibility • legal and ethical considerations. Partially effective application of procedures and security controls to ensure confidentiality, integrity and availability.	Adequate application of standards and guidelines in relation to: • accessibility • compatibility • legal and ethical considerations. Mostly effective application of procedures and security controls to ensure confidentiality, integrity and availability.	Effective application of standards and guidelines in relation to: • accessibility • compatibility • legal and ethical considerations. Fully effective application of procedures and security controls to ensure confidentiality, integrity and availability.	

Task 2: Activity B – Documenting the Development Process

Indicative content and marker guidance

Suitability of testing

Students should demonstrate they have tested different aspects of their solution. Testing may include:

- use of automated testing tools – expected to see evidence of the initial setup of the tools and the results produced by the tools, including commentary on how the results of the test will be actioned. Automated testing could be used for:
    - performance testing
    - stress testing
    - load testing
    - testing validations
    - unit testing
- use of AI – e.g. sections of code copied into ChatGPT/Google Gemini along with suitable prompts to identify issues, or use of other AI tools that check wider aspects of the solution
- manual testing and debugging, including:
    - visual debugging
    - functionality/black box testing
    - paired code reviews
    - compatibility testing
- wide range of tests and utilising a number of different tools to test most/all aspects of their solution.

Use of testing to inform iteration

- There should be evidence that where testing has identified potential errors, refinements and improvements to the solution, these have been considered and where appropriate actioned. Iteration may include:
  - errors – evidence that incorrect syntax or logic that would lead to runtime errors, or unreliable output, have been identified and corrected
  - refinements – identification of where code is functional but could be refined (e.g. more efficient code, better use of agreed standards, reduced data redundancy)
  - improvements – identification of ways that would improve the overall quality of the solution from a user and/or business perspective, e.g. additional or improved functionality, improved user experience (e.g. usability, accessibility, efficiency).

Use of content produced by others

- Students should provide a change and source log of where they have utilised content that has been created by others (including content generated using AI tools). The student should consider a range of different content, including:
  - code snippets
  - synthetic test data
  - information content (e.g. training plans, workout/exercise information)
  - assets and media (e.g. videos, images).

Students should provide a record of the source of the information and any relevant intellectual property considerations.

- The log should include details of how the content and code were used and any changes that were made from the original source, such as:
  - the page/screen it will be integrated with
  - how a code snippet will be integrated into a larger function (e.g. for checking an email has been generated, describing how this has been integrated into a larger format/data-collection process)
  - any other functions or data these will be dependent on (e.g. BMI calculator would need stored personal data (weight, height) passing from another function; calorie calculator would need user input/stored data of the food eaten that day/week by the user and stored/online nutritional data)
  - any legal considerations that may arise, e.g. is the code based on proprietary technology, how was written or image-based content generated (and what was used to seed the AI model that developed them), is there a potential IP infringement?

Change log

Students should provide a brief summary of notable/important changes made during the development of the solution. The change log should not be a step-by-step guide/narrative of how the solution was created.

The log should include:

- the date of the changes
- a brief description of what has changed
- a record of the file name and version number that the change applies to.

Assessment focus	Band 0	Band 1	Band 2	Band 3
	0	**1–2**	**3–4**	**5–6**
Suitability of tests	No rewardable material	Tests selected show a basic understanding of how to effectively test inputs, calculations, validation and processes.	Tests selected show a good understanding of how to effectively test inputs, calculations, validation and processes.	Tests selected show a thorough and detailed understanding of how to effectively test inputs, calculations, validation and processes.
	0	**1**	**2**	**3**
Use of testing to inform iteration	No rewardable material	Testing shows evidence of a basic iterative development process.	Testing shows evidence of a good iterative development process.	Testing shows evidence of an effective iterative development process.
	0	**1–2**	**3–4**	**5–6**
Use of content produced by others	No rewardable material	Basic description of how content created by others has been used in the solution, including superficial consideration of: • legal and ethical implications • the requirements of the task.	Adequate description of how content created by others has been used in the solution, including appropriate consideration of: • legal and ethical implications • the requirements of the task.	Comprehensive description of how content created by others has been used in the solution, including thorough and detailed consideration of: • legal and ethical implications • the requirements of the task.
	0	**1**	**2**	**3**
Change log	No rewardable material	Superficial records of changes made throughout development.	Adequate records of changes made throughout development.	Thorough and detailed records of changes made throughout development.

Task 3: Activity A – Gathering Feedback

Indicative content and marker guidance

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

- screencasts:
  - demonstrate the prototype to both technical and non-technical audiences/users – may use two different screens (technical/ non-technical) or may be in sections
  - could be used alongside some method of collecting feedback from the user, e.g. questionnaire, interview notes to be completed during/after watching
- questionnaires:
  - use of open and closed questions to ensure a range of quantitative and qualitative feedback
  - different sections or different questionnaires for different audiences
  - relevant questions that consider different aspects of the solution (appropriate for the targeted tester/user)
- records of observation of users:
  - observation of people using the product
  - identification of ‘pain points’
  - types of observation, which may include structured, unstructured, participant observation
- paired coding review records:
  - technical review of code from an experienced coder
  - identification of issues that may not show up in testing, e.g. efficiency, quality of commenting, code structures, alternative solutions.

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**	**7–9**	**10–12**
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 partial 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 fully provides the opportunity for evidence-informed further iteration.	

Assessment focus	Band 0	Band 1	Band 2	Band 3	Band 4
	0	1–2	3–4	5–6	
Effectiveness of communication	No rewardable material	Quality of communication is partially effective for both technical and non-technical audiences as a result of the use of partially appropriate techniques, methods and formats. The use of technical language is partially 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 formats. The use of technical language is mostly appropriate for the intended audience.	The quality of communication is fully effective for both technical and non-technical audiences as a result of the use of fully appropriate techniques, methods and formats. The use of technical language is fully appropriate for the intended audience.	

Task 3: Activity B – Evaluation

Indicative content and marker guidance

Effectiveness of assets and content

The student should provide an evaluation of the assets and content within their prototype:

- appropriateness of the assets selected which may include:
  - third-party data (e.g. to simulate customer data)
  - multimedia content (e.g. training videos, images)
  - snippets of pre-written code
  - any other appropriate information (e.g. trainer information, demand of different activities, e.g. the number of calories per hour burned)
- validity and reliability of the sources used, which may consider:
  - primary and secondary information and data
  - websites used, e.g. is the subject information likely to be accurate, is it up to date etc.
  - information that is realistic/suitable for the scenario, e.g. subscription or personal trainer session prices
  - potential bias
- legal and ethical implications of using the identified assets, which may consider:
  - copyright of content and IP
  - security and privacy of data and information, e.g. GDPR, encryption, privacy and data policy/informing the user
  - representation, e.g. range of gender, ethnicity, not conforming to/reinforcing stereotypes
  - accessibility
  - cookie consent notice
  - terms and conditions.

Evaluation of the digital solution

The student should provide an evaluation of the final product against measurable criteria defined in the proposal generated in Task 1.

The evaluation should include a review of:

- the extent to which functional and non-functional requirements have been met
- data to compare outcomes to key performance indicators (as appropriate)
- the extent to which user acceptance criteria have been met.

Consideration of potential developments/improvements

The student should provide a rationale supported by:

- identification of how far requirements have been met
- feedback from test users
- clear explanation of why an improvement should be made (with reference to feedback and the needs of ToKa Fitness and its stakeholders).
- examples taken from their prototype and specific extracts from feedback provided by test users.

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 basic review of the effectiveness of the content selected. There is 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 partially supported by superficial consideration, comparison and corroboration across multiple sources.	A good review of the effectiveness of the content selected. There is 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 adequate consideration, comparison and corroboration across multiple sources.	A comprehensive review of the effectiveness of the content selected. There is 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 fully supported by effective consideration, comparison and corroboration across multiple sources.
	0	**1–2**	**3–4**	**5–6**
Evaluation of the digital solution	No rewardable material	A basic evaluation of how well the prototype meets: • functional and non-functional requirements of the system • user acceptance criteria for the proposed system.	A good evaluation of how well the prototype meets: • functional and non-functional requirements of the system • user acceptance criteria for the proposed system.	A comprehensive evaluation of how well the prototype meets: • functional and non-functional requirements of the system • user acceptance criteria for the proposed system.

Assessment focus	**Band 0**	**Band 1**	**Band 2**	**Band 3**
	0	**1–2**	**3–4**	**5–6**
Further development	No rewardable material	A basic rationale for future iteration is provided. Points made are supported by partially relevant: • selection of examples • consideration of feedback.	A good rationale for future iteration is provided. Points made are supported by mostly relevant: • selection of examples • consideration of feedback.	A comprehensive rationale for future iteration is provided. Points made are supported by fully relevant and perceptive: • selection of examples • consideration of feedback.

Copyright and Acknowledgements