**NEA Implementation Review**

|  |  |
| --- | --- |
| **Questions** | **Answers** |
| Have you completed your Implementation? (Y/N) |  |
| Describe what you did over the summer / since last term or the last review. |  |
| If you have not completed the Implementation. | |  |  | | --- | --- | | **Describe what needs to be completed.** | Leaderboard  A\*  Map/grid  Inventory | | **What problems did you have?** | Map is not generate by grid  Object instanitation | | **A rough estimate of number of days required to complete it.** |  | |
| Alternative solution – if you are unable to complete your Implementation as per planned, what alternative solution have you decided upon. |  |
| Tick all complexity features you have included in your Implementation.  For each complexity feature, briefly describe where/what you used it for and if you have mentioned this in your analysis. | |  |  |  | | --- | --- | --- | | **Technical Skills** | **Where used?** | **Mentioned in Analysis? (Y/N)** | | **Data Model**   * 4 or more tables * 2-3 tables * 1 table | Leaderboard  Player login |  | | **Data structures** (   * Hash tables / hashing * Linear List * Linked List * Stacks * Queues * Graphs * Trees * Dictionaries * Records * Arrays – multidimensional * Array – single * Advanced matrix operations | List of leaderboard |  | | **Model**   * Scientific * Mathematical * Robotic * Control * Business   **Complex / Simple / Statistical Simple Mathematical** |  |  | | **File organisation**   * Direct access * Sequential access * Text files (write & read) |  |  | | **OOP**   * Classes * Inheritance * Composition * Polymorphism, * Interfaces * Server-side scripting (request & response) * Dynamic generation of objects |  |  | | **Client-server model**   * Server-side scripting (request & response objects) * Server-side extensions * Calling parameterised Web service APIs * Parsing JSON/XML to service a complex C-S model | PHP  Json.encoded |  | | **SQL**   * Selects * Deletes * Updates * Inserts * Append * Inner joins or similar * Cross-table * Aggregate SQL functions * User/CASE-generated DDL script | Select  Insert |  | | **Searching, Sorting, Algorithms**   * Merge sort * Bubble sort * Binary search * Linear search * Optimisation * Minimisation * Scheduling * Pattern matching * Recursion * Traversals (Graph / Tree) |  |  | |
| Complete the table by ticking in the applicable columns. | |  |  |  |  | | --- | --- | --- | --- | | **Sections** | **Fully completed** | **Partially Completed** | **Not Started** | | ***Analysis*** |  |  |  | | ***Design*** |  |  |  | | ***Implementation*** |  |  |  | | ***Technical Solution*** |  |  |  | | ***Testing*** |  |  |  | | ***Evaluation*** |  |  |  | |