**Student: Anna Fortenberry**

|  |  |  |
| --- | --- | --- |
|  | **Functional Requirements (FR) Questions** | **Functional Requirements (FR) Answers** |
| **Individual** | **Q1:** To **refactor the hintHandler functionality** how many files you needed to update? | Two files were updated to refactor the hintHandler functionality. The main function itself was updated, and the file that of the function call was updated. |
| **Individual** | **Q2:** If several files were changed did the **maintainability** aspect of the code **increased** or **decreased**? | Refactoring allowed for easier maintenance of the code. Creating a separate function simplified the code, isolating repeated logic for clarity. |
| **Group** | **Q3:** What were the **keywords** that you [**used/thought of/searched for/discussed/]** with your team members when you decided to add the new feature(s). **Please list them all.** | **A3 - feature\_name():** <keyword 1>, <keyword 2>….<keyword\_n>  **A3 - feature\_name():** <keyword 1>, <keyword 2>….<keyword\_n>  **A3 - feature\_name():** <keyword 1>, <keyword 2>….<keyword\_n>  … |
| **Individual** | **Q4:** What were the **keywords** that you [**used/thought of/searched for]** when you **refactored the hintHandler functionality**? **Please list them all.** | **hintHandler():** clarity, efficiency, reliability  **shuffleCoords():** simplify, maintainability, reorganize  … |
| **Individual** | **Q5:** Please rank the **keywords** from **Q4** by providing a ranking score between [1-5] 5 being the **most informative** **keyword**, and **1** being the **least informative** **keywords** | **hintHandler():** clarity **(5)**, efficiency **(5)**, reliability **(3)**  **shuffleCoords():** simplify **(5)**, maintainability **(4)**,  reorganize **(5)** |
| **Individual** | **Q6:** Explain the reason why certain **keywords** were more **useful** to help yourself better **refactor the hintHandler functionality** in the code vs. the **previous implementation** that was not useful or that you needed to **change** or **discard**. | Regarding the hintHandler function, the motive for refactoring by implementing shuffleCoords was to **simplify** and **better organize** the code. It was not difficult to maintain before; this new structure just made it even easier. These key words communicated the goal behind this instance of refactoring.  “Removing the middleman” by discarding the hintCount parameter helped make the code more **efficient**. Rather than passing the count to hintHandler, only to be updated once, it is updated at the site of the call, in a successful call of the function. There is no need to pass the variable between files, so the listed key words represent this new update. |
| **Group** | **Q7**: What were the **phrases** that you **[used/thought of/searched for/discussed]** with your team members when you decided to add the new feature(s). **Please list them all.** | **A7 - feature\_name():** <phrase 1 >, <phrase 2>….<phrase\_n>  **A7 - feature\_name():** <phrase 1 >, <phrase 2>….<phrase\_n>  **A7 - feature\_name():** <phrase 1 >, <phrase 2>….<phrase\_n>  … |
| **Individual** | **Q8:** What were the **phrases** that you [**used/thought of/searched for]** when you **refactored the hintHandler functionality**? **Please list them all.** | **hintHandler():** “reorganize for simplicity,” “reduce imports”  **shuffleCoords():** “simplify like code,” “clarify functionality”  … |
| **Individual** | **Q9:** Please rank the **phrases** from **Q8** by providing a ranking score between [**1-5]** **5** being the **most informative** **phrases**, and **1** being the **least informative** **phrases.** | **hintHandler():** “reorganize for simplicity,” **(4)** “reduce imports” **(5)**  **shuffleCoords():** “simplify like code,” **(4)** “clarify functionality” **(5)** |
| **Individual** | **Q10:** Explain the reason why certain **phrases** were more **useful** to help yourself better **refactor the hintHandler functionality** in the code vs. the **original implementation** that you needed to **change** or **discard**. | These phrases identified the changes that would simplify the code on our end, while maintaining the purpose it serves on the user’s end. Each of the four listed are specific to the design of the code. Javascript is a new language to ¾ of our members, so reducing imports, in this stage of knowledge, allows for better maintenance. |

**General Question:**

Please, in a few words describe the **strategy** you used to log/list your keywords **[before/while/after]** coding functional/Non-functional requirements (NFRs/FRs).

In this stage of our code, the workload has been more separated between us. This meant a lot of the keywords and phrases are listed post refactoring, looking back on our design. For the group keywords and phrases, we discussed and recalled these during our weekly meetings on functionalities that affected both the frontend and backend code.

* Functional Suitability or (Accuracy) **(A)**
* Performance Efficiency **(P)**
* Compatibility **(C)**
* Usability **(U)**
* Reliability **(R)**
* Security **(S)**
* Maintainability **(M)**
* Portability **(Pb)**

**[3] Rethinking the Notion of Non-Functional Requirements.**

<http://www.ptidej.net/courses/log3410/fall11/Lectures/Article_5.pdf> (online google scholar source)

**NOTE: <\*NFR Type\*> refers to the NFR type that you implemented. It could be the followings: 1) Accuracy (A), 2) Performance Efficiency (P), 3) Compatibility (C), 4) Usability (U), 5) Reliability (R), 6) Security (S), 7) Maintainability (M), 8) Portability (Pb) or 9) Other NFR (O)**

|  |  |  |
| --- | --- | --- |
|  | **Non-Functional Requirements (NFRs) Questions** | **Non-Functional Requirements (NFRs) Answers** |
| **Individual** | **Q1:** To **refactor the hintHandler functionality a second time** how many files you needed to update? | Only the site of the hintHandler function itself was adjusted the second time it was refactored. So, only one file. |
| **Individual** | **Q2:** If several files were changed did the **maintainability** aspect of the code **increased** or **decreased**? | The maintainability increased because the design was simplified. |
| **Group** | **Q3:** What were the **keywords** that you [**used/thought of/searched for/discussed/]** with your team members when you decided to add the new feature(s). **Please list them all.** |  |
| **Individual** | **Q4:** What were the **keywords** that you [**used/thought of/searched for]** when you **refactored the hintHandler functionality a second time**? **Please list them all.** | **hintHandler(), <maintainability>:** simplify, reduce, time  **hintHandler(), <performance efficiency>:** runtime, accuracy, measurable |
| **Individual** | **Q5:** Please rank the **keywords** from **Q4** by providing a ranking score between [1-5] 5 being the **most informative** **keyword**, and **1** being the **least informative** **keywords** | **hintHandler(), <maintainability>:** simplify **(5)**, reduce **(4)**, time **(3)**  **hintHandler(), <performance efficiency>:** runtime **(5)**, accuracy **(4)**, measurable **(3)** |
| **Individual** | **Q6:** Explain the reason why certain **keywords** were more **useful** to help yourself better **refactor the hintHandler functionality** **a second time** in the code vs. the **previous implementation** that was not useful or that you needed to **change** or **discard**. | In the second refactoring, the logic was simplified to find the first instance of an error or blank cell, rather than a random one. This was largely in part to the time restraint of a summer class and difficulty with testing functionality with an unfamiliar language. Keywords such as simplify and reduce communicate our strategy to overcome this problem, given the circumstances. We needed it to be very maintainable as we finalize our product. The performance was increased as well – we shorted the runtime by only traversing until we find the first instance of the criteria. The first instance insured very visible results on the coding end for testing purposes. |
| **Group** | **Q7**: What were the **phrases** that you **[used/thought of/searched for/discussed]** with your team members when you decided to add the new feature(s). **Please list them all.** |  |
| **Individual** | **Q8:** What were the **phrases** that you [**used/thought of/searched for]** when you **refactored the hintHandler functionality a second time**? **Please list them all.** | **hintHandler(), <maintainability>:** “reduce the complexity of the code,” “remove outside calls of functions,”  **hintHandler(), <performance efficiency>:** “output visible and measurable test results,” “reduce the runtime,” |
| **Individual** | **Q9:** Please rank the **phrases** from **Q8** by providing a ranking score between [**1-5]** **5** being the **most informative** **phrases**, and **1** being the **least informative** **phrases.** | **hintHandler(), <maintainability>:** “reduce the complexity of the code,” **(5)** “remove outside calls of functions,” **(5)**  **hintHandler(), <performance efficiency>:** “output visible and measurable test results,”**(5)** “reduce the runtime,” **(4)** |
| **Individual** | **Q10:** Explain the reason why certain **phrases** were more **useful** to help yourself better **refactor the hintHandler functionality** **a second time** in the code vs. the **original implementation** that you needed to **change** or **discard**. | The phrases better communicate how and why we needed to refactor to meet the nonfunctional requirements of maintainability and performance efficiency. We specifically removed the outside call of the shuffle function and instead returned a value based on the first instance. This in turn increased performance efficiency and allowed for more expected results for testing purposes. |

**General Question:** Please, in a few words describe the **strategy** you used to log/list your keywords/phrases **[before/while/after]** coding functional requirements (FR) and nonfunctional requirements (NFRs).

As before, refactoring was more personal that group based. The keywords and phrases listed were thoughts during the plan for refactoring that occurred before and during the process as minor adjustments were made.