|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Task:** | | **2** | | |
| **Task Title:** | | **Project** | | |
| **Task Code:** | | **AT2 PRJ Task 1** | | |
|  | |  | | |
| Assessment type (): | | | | |
|  | Questioning (Oral/Written) | |  | Portfolio |
|  | Practical Demonstration | |  | Project |
|  | 3rd Party Report | |  | Other – Please Specify |

|  |
| --- |
| The base requirements this assessment task include:   * IDE or editor for developing Python programs (only PyCharm supported by the college) * Access to Office 365 & Microsoft Word * Access to the program template (.py) and the readme.md instructions * Task list and task template   Use of some of these items may not occur in this part of the assessment task. |
| Assessment Due NOTICE: While this assessment is due on the second last week of the course. You **MUST** review elements of this submission with your lecturer before submission to pass this assessment.  This assessment is due on the second last week of the term on which it was administered.  Refer to Blackboard for most accurate dates, which may alter due to unforeseen circumstances. |
| Instructions The project consists of three parts:   1. This journal and general project instructions 2. A template python program that you can use as a basis of your development work 3. A readme.md file (markdown) that contains development specific instructions as well as sections that you need to complete.   Follow the steps outlined in each of these individual components and submit all three to successfully complete this assessment.  You must follow the instructions in and complete the tasks for all three components to complete this course. You may need to submit additional files |
| Important If you are using a different IDEs or a different structure for your application, then assistance with those tools and forms may be limited. Discuss with your lecturer before straying too far off the path. |
| Scenario You are currently working as a junior software developer at words-are-us, a Perth-based company that develops content and apps to build engagement.  You have been contracted to gain a deeper understanding of the popular tool Wordle and develop a prototype app that emulates the original wordle algorithm but provides these capabilities from the command line. As a junior developer, you have access to your peers, the senior developer (played by your lecturer), and occasional access to the customer (usually also played by your lecturer). |
| Specific Instructions This file is to be submitted as a journal that demonstrates the following aspects of your competency:   * Your ability to prioritise your own work * Obtain feedback to validate and enhance your design decisions * Confirm that the application meets specifications * Evaluate and reflect on decisions you made in the process of development   Save the file as:   * XXX\_ICTPRG302\_Proj\_Journal.docx   Replacing XXX with your initials.  For example, Adrian Gould would use AG\_ICTPRG302\_Proj\_Journal.docx for his submitted filename.  Upload any code as a PyCharm project in a zip-file. Remove the virtual environment (**venv** or **.venv**) from the zip-file before uploading it to Blackboard. |
| Answering Questions When a step includes a question, you must attempt to answer it. A word count is sometimes given, but most answers require at most 1-3 paragraphs.  All answers must be in complete sentences unless indicated. You must use your own words unless otherwise specified. |
| Sources of Information In industry, it is good practice to keep track of where information was obtained. This is especially true if it is a written document, or even code.  If you answer any questions using information from web sites, please include the site name and URL (Web site address) after the answer. Likewise, include the title and author for books and magazine articles. For example:   * RS Electronics Ltd: <https://au.rs-online.com/> * Slack API Documentation, Users List Method: <https://api.slack.com/methods/users.list>  Code Storage and Issue Tracking An industry-standard tool for managing program source is git. GitHub is a free service that provides hosting for git repositories. You may choose to use this system for source control, task management, or both. However, it is not required for this project.  You may also use OneDrive within your college Office365 to store a backup of your code or keep a copy on a USB thumb drive. |
| *This space left intentionally blank.* |

| **Weekly** | **Reflections** | Words 50-100 |
| --- | --- | --- |
| 00 | Familiarisation At any stage during this assignment, you may consult the stakeholder(s) or their representative(s).  Today, you were introduced to the business requirements of the project. You would have been given an opportunity to interact with competitor products. For example, if you are performing the Wordle project, you would have got to try the commercial version of the game and discuss how the game works (system behaviour) it in class.  Bellow, note down the following:   * 1-2 lines describing the business motivation * 5-6 lines describing the system behaviour * An example of at least **two** python constructs you learned previously and one line explaining how they could be relevant to the problem. For example, for loops, Boolean expressions, file reading, etc. |  |
|  | The business motivation would be the motivation of any business – profit – given the success of the Wordle game, it is quite clear that there is a market for this kind of games.  The game is an action driven application. The user’s input is the determinant action, as depending on the guess entered, different outcomes can occur. No clues are given, apart from telling whether the letters are in the right/wrong place or not in the word.  Using **dictionaries** to get a letter count is a good way to determine how many times a letter would be able to score. **For** and **While loops** are crucial tools to loop through the words(strings) checking for matching indexes/positions which means an exact match. |  |
| 01 | Creating an algorithm Complete at least two versions of the algorithm (lecturer determines the algorithm). Present your initial version to one of your peers and the senior developer. Document how your thoughts changed as a result of the interaction.  Your algorithm is **not** documented here (it should be in the readme.md) |  |
|  | *Space for your answers* |  |
| 02 | Edge cases What are some edge cases that you have considered that could impact the design of your algorithm? Discuss with your lecturer and peers and write your reflection here |  |
|  | The main issue to be overturned from the start are the cases where there are more occurrences of a letter in the guess than in the target word. In those cases, the letters that would score would be only as many as there were in the target word.  The issue on that would be that if a ‘MISPLACED’ score were given to a letter, then the next one was an exact match, the result would be incorrect. |  |
| 03 | Project planning What tool have you used to manage your tasks and why? Include a screen shot of your initial task list. |  |
|  | The tool used was the provided ‘to do list’ with influence of the roadmap (included in readme file), due to the ease of use. Initial and final checklist are almost the same. |  |
| 04 | Simple testing and inspection How do you intend on testing your code? Give an example of an existing test case and a test case that you will generate. |  |
|  | Apart from the doctest examples provided on the project files, new words would be tried (including various combinations) to catch any flaws in the code.  Using the test provided (melee, erect) was a milestone in the project as it showed the issues on my first take on the scoring code.  Variety of tests conducted below:  Graphical user interface, text  Description automatically generated  Text  Description automatically generatedGraphical user interface  Description automatically generatedGraphical user interface, text, application, chat or text message  Description automatically generatedText  Description automatically generatedText  Description automatically generatedText  Description automatically generatedText  Description automatically generated |  |
| 05 | Dealing with uncertainty What are some issues that you are still uncertain about? How do you intend to get clarity? Ensure you discuss with your lecturer |  |
|  | Most of the uncertainties that came up were dealt with the assistance of forums and tutorials and at times consulting with colleagues that had a better understanding of the code than me. |  |
| 06 | Catch up This section is not assessed but is strongly recommended  Write any challenges and achievements you are having |  |
|  | The challenges were always huge, mainly to personal reasons. However, when developing the code, it took a while to see the big picture, as while I could understand the techniques learnt throughout the semester, I was still not connecting all the bits.  Being able to develop the whole project in less than a week, whether I pass or not, will certainly be a great achievement. |  |
| 07 | Code review Review your code with the senior developer and write down any changes you need to make because of the review. Add them to your task list and include a screenshot. |  |
|  | Final (and only) review occurred when I had a fully functioning code.  There were however previous versions of the code that I had developed until reaching the final version. Below are the first concluded version and the final respectively.  Text  Description automatically generated  Text  Description automatically generated |  |
| 08 | Application review Review the working of the application with the client (likely, your lecturer). Write down one thing you got right and one thing you got wrong and explain why you think what worked and what didn’t.  Were there any non-functional issues? Crashes, unexpected output? |  |
|  | No review occurred since only final version was presented. However, while at it, I have realised some bits of my code were deemed unnecessary when the code was through the last reviews.  There was, in my opinion, no need of a module just to check the score, as I believe it would be more practical including it as part of the score module.  Throughout the progress of the code there were issues, but at the end, no issues occurred. |  |
| 09 | You DID IT!!! Well done on completing your first software development project!  You’ve come a long way. Don’t forget to submit all elements |  |