**Software Development Lifecycles (Advocate: Thiago Viana)**

**P1 Describe two iterative and two sequential software lifecycle models.**

|  |
| --- |
| <https://github.com/LBruni98/Project-Lifecycles> |
| The link above takes the user to a repo regarding project lifecycles. In this repo contains a report of many software lifecycles. This is sufficient evidence because at least two of both iterative and sequential lifecycles are described in this document. |

**P2 Explain how risk is managed in the Spiral lifecycle model.**

|  |
| --- |
| <https://github.com/LBruni98/Project-Lifecycles/blob/master/README.md#how-risk-is-managed-in-spiral> |
| The link above directs the user to a lone repo, containing a simple report. This report talks about the spiral model, in depth. In this description, it is also explained how risk is managed in this lifecycle, how it helps with identifying risk earlier and what sort of project fits this development lifecycle |

**P3 Explain the purpose of a feasibility report.**

|  |
| --- |
| <https://github.com/LBruni98/Feasibility-Reports#purpose> |
| Above is the link to a repo, containing a report on feasibility reports. This document describes what and explains their purpose. The report goes in depth with feasibility reports, explaining the purpose, how it is used and also goes a step further by detailing each of the components that go into a feasibility report. |

**P4 Describe how technical solutions can be compared.**

|  |
| --- |
| <https://github.com/LBruni98/Comparison-of-Technical-Solutions> |
| The link above leads to a separate repo on comparing technical solutions. Here there is a description on how technical solutions are compared, The description goes in depth about the effectiveness analysis method used for finding an effective solution along with the factors and qualities used when finding an effective solution. |

**P5 Undertake a software investigation to meet a business need.**

|  |
| --- |
| <https://github.com/LBruni98/ZSL-The-Climate-Menace/blob/master/README.md#2-management>  <https://github.com/LBruni98/ZSL-The-Climate-Menace/blob/master/README.md#25-aims-and-objectives>  <https://github.com/LBruni98/ZSL-The-Climate-Menace/blob/master/README.md#261-gantt-chart>  <https://github.com/LBruni98/ZSL-The-Climate-Menace/blob/master/README.md#262-burndown-chart>  <https://github.com/LBruni98/ZSL-The-Climate-Menace#33-concept>  <https://github.com/LBruni98/ZSL-The-Climate-Menace#29-zsl-project-backlog> |
| The link above leads to the ZSL project repo, specifically the README document. The information is located underneath the “Concept” section and the project backlogs. These parts of the document outline the client’s requirements, what the client is exactly looking for and the backlogs detailing the development and working towards the requirements in regards to the software used and experience of the team. |

**P6 Use appropriate software analysis tools/techniques to carry out a software investigation and create supporting documentation.**

|  |
| --- |
| <https://github.com/LBruni98/ZSL-The-Climate-Menace#42-problem-solving-techniques-used-in-the-design-and-delivery>  <https://github.com/LBruni98/ZSL-The-Climate-Menace/blob/master/README.md#421-tools-used-in-the-design-and-delivery>  <https://github.com/LBruni98/ZSL-The-Climate-Menace#36-application-structure>  <https://github.com/LBruni98/ZSL-The-Climate-Menace#292-project-backlog> |
| In the first link, I have listed down techniques used in the design and delivery of the project, which are Breaking down a problem/decomposition, Trial and Error and teamwork and communication.  The second link shows the structure of the application, which was a prototype created using Photoshop. We have used prototypes throughout the whole project.  The third link is the project’s schedule, another technique that we have used is interviews. We have met with the clients in the following days (List days) |

**P7 Explain how user and software requirements have been addressed.**

|  |
| --- |
| <https://github.com/LBruni98/ZSL-The-Climate-Menace/blob/master/README.md#294-monday-17th-october-sprint-1>  <https://github.com/LBruni98/ZSL-The-Climate-Menace/blob/master/README.md#41-how-user-and-software-requirements-have-been-addressed> |
| The links above leads to the ZSL project README. Here, the first link describes the process of breaking down the project into epics, user stores, etc. creating tasks based on the requirements of the customer and build it up into the version to be released to the customer. The second link takes the user to the backlog, detailing and showing how we have addressed the user and software requirements. |

**M1 Describe, with an example, why a particular lifecycle model is selected for a development environment.**

|  |
| --- |
| <https://github.com/LBruni98/Project-Lifecycles> |
| Above is the link to a repo on Project Lifecycles, covering past criteria mentioned earlier. It contains several methodologies and details about them. Also, worth noting is that it also covers the benefits and drawbacks from them as well as examples; which company or project would best suit this type of methodology. |

**M2 Discuss the components of a feasibility report.**

|  |
| --- |
| <https://github.com/LBruni98/Feasibility-Reports#components> |
| Above is the link to a repo containing a simple report on Feasibility reports. The report is created using a README document and within contain what feasibility reports are and their impact. Listed inside also are the main components of a feasibility report, with explanation on what they are to make up the report. |

**M3 Analyse how software requirements can be traced throughout the software lifecycle.**

|  |
| --- |
| To be completed |
|  |

**M4 Discuss two approaches to improving software quality.**

|  |
| --- |
| <https://github.com/LBruni98/Improving-Software-Quality#approaches-towards-improving-software-quality> |
| Above is the link to a repo on the subject of improving software quality. It explains what CMM is and refers to two methods on how to improve upon software quality, with how they can help upon the quality of the software being developed. |

**M5 Suggest two software behavioural specification methods and illustrate their use with an example.**

|  |
| --- |
| To be completed |
|  |

**M6 Differentiate between a finite state machine (FSM) and an extended- FSM, providing an application for both.**

|  |
| --- |
| To be completed |
|  |

**D1 Assess the merits of applying the Waterfall lifecycle model to a large software development project.**

|  |
| --- |
| <https://github.com/LBruni98/Project-Lifecycles/blob/master/README.md#advantages-of-waterfall> |
| Above is the link to a repo on Project Lifecycles, covering past criteria mentioned earlier. It contains several methodologies and details about them. The waterfall methodology is listed and under the subheading ‘Advantages of Waterfall’, it explains the advantages to using the methodology in the workplace, along with the benefits of applying the model to a large development project. |

**D2 Assess the impact of different feasibility criteria on a software investigation.**

|  |
| --- |
| <https://github.com/LBruni98/Feasibility-Reports#impact-of-feasibility> |
| The link above leads to a repo containing a simple report on Feasibility reports. The importance and components are listed in the README document upon opening the repo. Different feasibility criteria are covered, including details on the impact of each. These being technical, economic and organisational feasibility. |

**D3 Critically evaluate how the use of the function design paradigm in the software development lifecycle can improve software quality.**

|  |
| --- |
| To be completed |
|  |

**D4 Present justifications of how data driven software can improve the reliability and effectiveness of software.**

|  |
| --- |
| To be completed |
|  |