**Module 2 Assignment – SDLC**

Anthony Embry

Ivy Tech Community College

SDEV 120

Alison Bodeker

11/02/2024

The Software Development Life Cycle (SDLC) is a process that was created for the purpose of guiding those developing software applications from the beginning planning phases until it is released to the public and maintained until it’s end. It has a structured approach that makes sure the developers tasked with working on the software create high-quality, smooth, and scalable work that will ultimately be given to the end-user. My assignment today is to explain the key parts of the SLDC, including iterative and prototyping methods, modularization, and program design. By the end we should all understand how these things work together with each other to create the framework for a successful program.

The SDLC is comprised of several phases including planning, defining, designing, building, testing, deploying, and lastly maintaining. (divyanshu\_gupta1, 2024) Each one of these phases are vital to ensure the final product is something the end-user will enjoy long term without any major mishaps. The planning phase involves defining your project objectives, the scope, and allocation of resources. Following the planning phase is the defining phase this is where the more technical details of the software are recorded, including how the software will be built, user interface, functionalities, and how the software will operate. (Clark, 2024)

The design phase is where developers will create a draft of how the software will be built, this would include system interfaces and data structures. Creating a thorough design helps ensure that implementation will be successful. After the design phase is the development phase, this is when code is created based on the design created before. Coders will follow standards to ensure the end-product is reliable and this is an extremely crucial piece of this cycle. The testing phase follows development and is also a very important part of the cycle. It would comprise of strict evaluation and testing of the product to identify and remove any bugs or defects and ensure that it meets all quality standards. Once the testing is completed it will be time to release the software to the public for use in the deployment phase. This phase is immediately followed by the maintenance phase where the product is re-evaluated for post-release issues and issuing any updates or upgrades. (divyanshu\_gupta1, 2024)

Iterative and prototyping methods improve the SDLC by creating more flexibility and collaboration. Iterative development gives developers the ability to refine and enhance the product through multiple cycles, going back to past phases based on feedback from users and from results in the testing phase. Doing it this way makes the process more adaptable to changing requirements, resulting in a product more focused on the needs of the user. Prototyping is when early versions of the software are created to see and experiment with different concepts before it is deployed as a “final product”. Doing this enables investors to give feedback early in the process and it makes for less confusion when it comes to delivering end-user needs. (Fanchi, 2023)

Modularization and effective program design are cornerstones of creating a scalable and maintainable software. Modularization involves breaking down the final product into smaller projects with their own specific function that are more manageable. Simplifying it this way also allows for changes to be made to each specific module without affecting the entire system. (Johansson, n.d.) Program design covers a wide range of tasks, including UI design and data management. If a program is designed properly, it will be easy to navigate, intuitive, and process data in an efficient manner. (Wilkerson, 2022)

In conclusion, the SDLC, alongside iterative and prototyping methods, modularization, and program design, provides the framework for effective software development. By mastering these concepts, developers can create applications that are functional, secure, maintainable, and aligned with user needs. As technology continues to evolve, understanding and applying these practices will be essential for programming professionals aiming for long-term success in the industry.

# Bibliography

Clark, H. (2024, 10 29). Retrieved from The Product Manager: https://theproductmanager.com/topics/software-development-life-cycle/

divyanshu\_gupta1. (2024, 09 30). Retrieved from GeeksforGeeks: https://www.geeksforgeeks.org/software-development-life-cycle-sdlc/

Fanchi, C. (2023, 02 07). Retrieved from Backendless: https://backendless.com/the-importance-of-iterative-prototyping-in-application-development/

Johansson, T. (n.d.). Retrieved from Modular Management: https://www.modularmanagement.com/blog/methods-for-modularization-five-key-success-factors

Wilkerson, T. D. (2022, 08 22). Retrieved from Baker Tilly: https://www.bakertilly.com/insights/why-is-effective-program-design-and-execution-important