

# TECHNOLOGY & SOFTWARE

Asst. Prof. Dr. Yalçın İŞLER

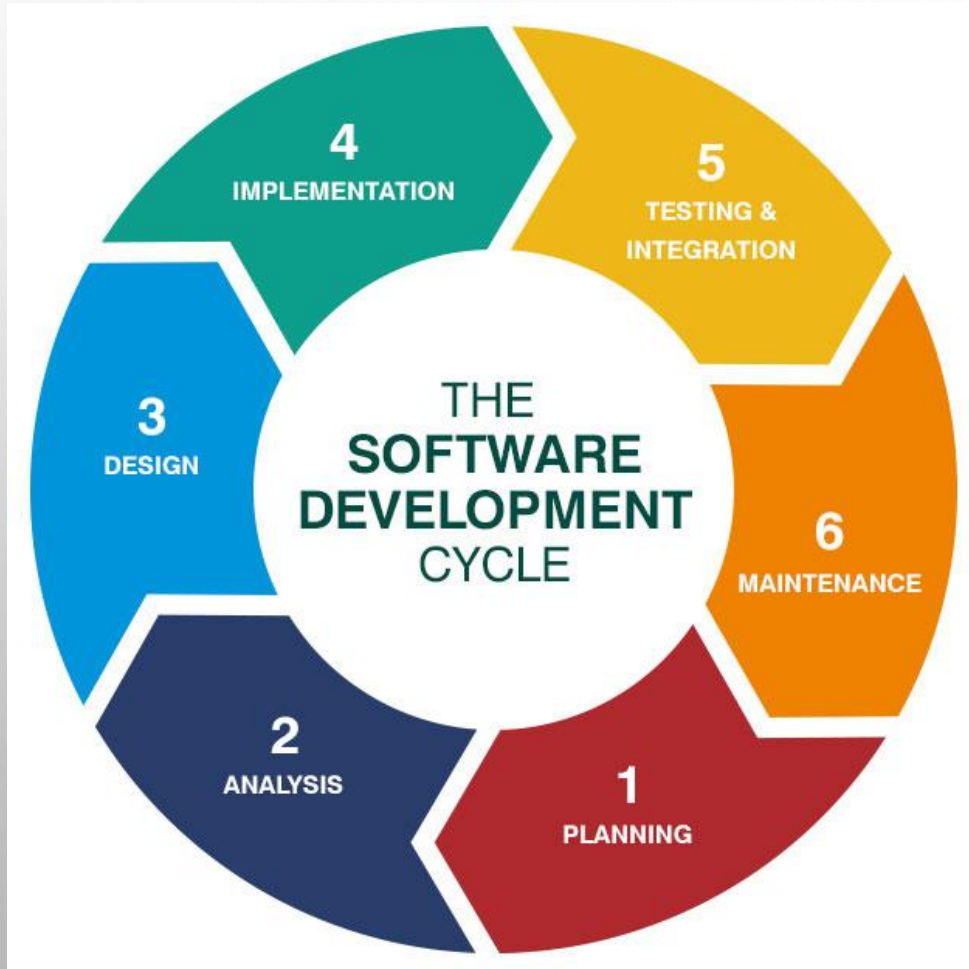
[http://me.islerya.com/tech\\_sw.php](http://me.islerya.com/tech_sw.php)

@ 2019

# SOFTWARE

- Software usually refers to instructions for computer hardware to execute. Software may also refer to:
  - IEEE Software, a magazine
  - Software (novel), part of the Ware Tetralogy, a 1982 cyber-punk novel by Rudy Rucker
  - Software (album), 1999 album by Grace Slick
  - Software (band), a German electronic duo active between 1984 and 2000
  - Software, Etc., a predecessor of the video game retailer GameStop
  - GNOME Software, a GNOME software package for managing software installation
- Software engineering is the application of engineering to the development of software in a systematic method.

# SOFTWARE DEVELOPMENT CYCLE



Web and software developers all over the world use the software development life cycle (SDLC) to implement tech projects successfully. The software industry follows the SDLC to design, develop, and test projects before deployment. When carried out effectively, the SDLC produces high-quality software that meets customer expectations and reaches completion within time and cost estimates.

# SOFTWARE DEVELOPMENT CYCLE #1

- 1. Planning
- This stage, which also involves requirement analysis, is the most fundamental part of the SDLC process. It is performed primarily by the development team, along with input from customers, stakeholders, market research, sales, and others. This stage is the foundation of the requirement analysis. Sometimes, teams study their competitors to determine ways to stand out. Once everyone is clear on the specifications, the team can use the information gained from outside sources to plan the basic project approach and identify potential risks. Another aspect of the planning stage is putting together a cost-benefit analysis. This involves determining how the project helps further the organization's business objectives.

# SOFTWARE DEVELOPMENT CYCLE #2

- 2. Analysis
- The analysis stage of the SDLC process involves defining project goals as functions. Next, the development team determines what operations the intended application should have. In general, analysis requires gathering and interpreting facts, as well as diagnosing issues with the current system and recommending improvements. This can be achieved in part by studying end-user information needs and removing inconsistencies. Think of this stage as the problem-solution step. The development team determines where obstacles lie and identifies ways to fix them with the new project. The aim is to remove barriers that would hinder the new product's successful implementation. This step is when a software requirement specification (SRS) document is created that identifies the scope of the project. Once management approval is obtained, the project can begin.

# SOFTWARE DEVELOPMENT CYCLE #3

- 3. Design
- In this stage, the team uses the SRS to design the product's optimal architecture. The requirements in the SRS dictate the design approaches that are included in a design document specification (DDS). Stakeholders review this document, and a design approach is selected based on their feedback. Risk assessment, market research, design modularity, budget, and time constraints must all be taken into consideration. This will define the architectural modules along with communication and data flow representation, including external and third party modules. The internal design of the architecture should be defined in a complete, detailed manner. Components that should be included are screen layouts, business rules, process diagrams and other documentation. Developers and software engineers must be able to develop and deliver the system based on this information with minimal additions.

# SOFTWARE DEVELOPMENT CYCLE #4

- 4. Implementation
- This stage is the start of actual development. During implementation, the team builds the product and creates programming code to match the DDS. During this stage, it is important for developers to follow the coding guidelines defined by their organization. Developers use various programming tools like interpreters, debuggers, and compilers to generate the code, along with high-level programming languages like C, C++, Java and more. Developers determine the language to use based on the type of software along with customer requirements. At the end of this stage, the software is put into production.

# SOFTWARE DEVELOPMENT CYCLE #5

- 5. Testing and Integration
- This stage of the SDLC involves bringing the separate parts of the project together into a dedicated testing environment to check for errors, bugs and other issues. During the testing phase, the product is checked to ensure that defects are reported, tracked, fixed, and tested again until the product meets quality standards. Some companies don't view testing as an official stage of the process. They view it as a subset of maintenance. In these cases, it may be referred to as post-implementation review. Testing involves asking important questions such as:
  - Does the new system meet business requirements and objectives?
  - Is it reliable?
  - Are there any remaining bugs?
  - Does it function according to approved functional requirements?
- During this stage, it is also a good idea to evaluate the development process itself to determine if there are aspects that did not go according to plan. The development team should also examine any aspects of the completed product that did not meet their clients' expectations. This enables the team to correct errors and inconsistencies for the next project.



# SOFTWARE DEVELOPMENT CYCLE #6

- 6. Maintenance
- When testing is complete and the product is ready for deployment, it is time for its release into the marketplace. Sometimes this happens in stages or all at once, depending on the organization's business strategy. For example, the product may only be released to current clients. Based on feedback, changes may be made before complete deployment occurs. After the product's release, maintenance is carried out for the customer. The team makes software improvements or change requests as needed. The ultimate goal of the maintenance phase is to ensure that the product remains relevant and high quality. It involves ongoing evaluations of the system's performance.

# LTS (LEFT TO STUDENTS)

- In your field, please discover how you can apply these design principles. And is there any necessary improvements for this cycle if you want to develop a software in your area?

# REFERENCES

- [https://en.wikipedia.org/wiki/Software \(disambiguation\)](https://en.wikipedia.org/wiki/Software_(disambiguation))
- [https://en.wikipedia.org/wiki/Software\\_engineering](https://en.wikipedia.org/wiki/Software_engineering)
- <https://online.husson.edu/software-development-cycle/>