



Software Development Life Cycle

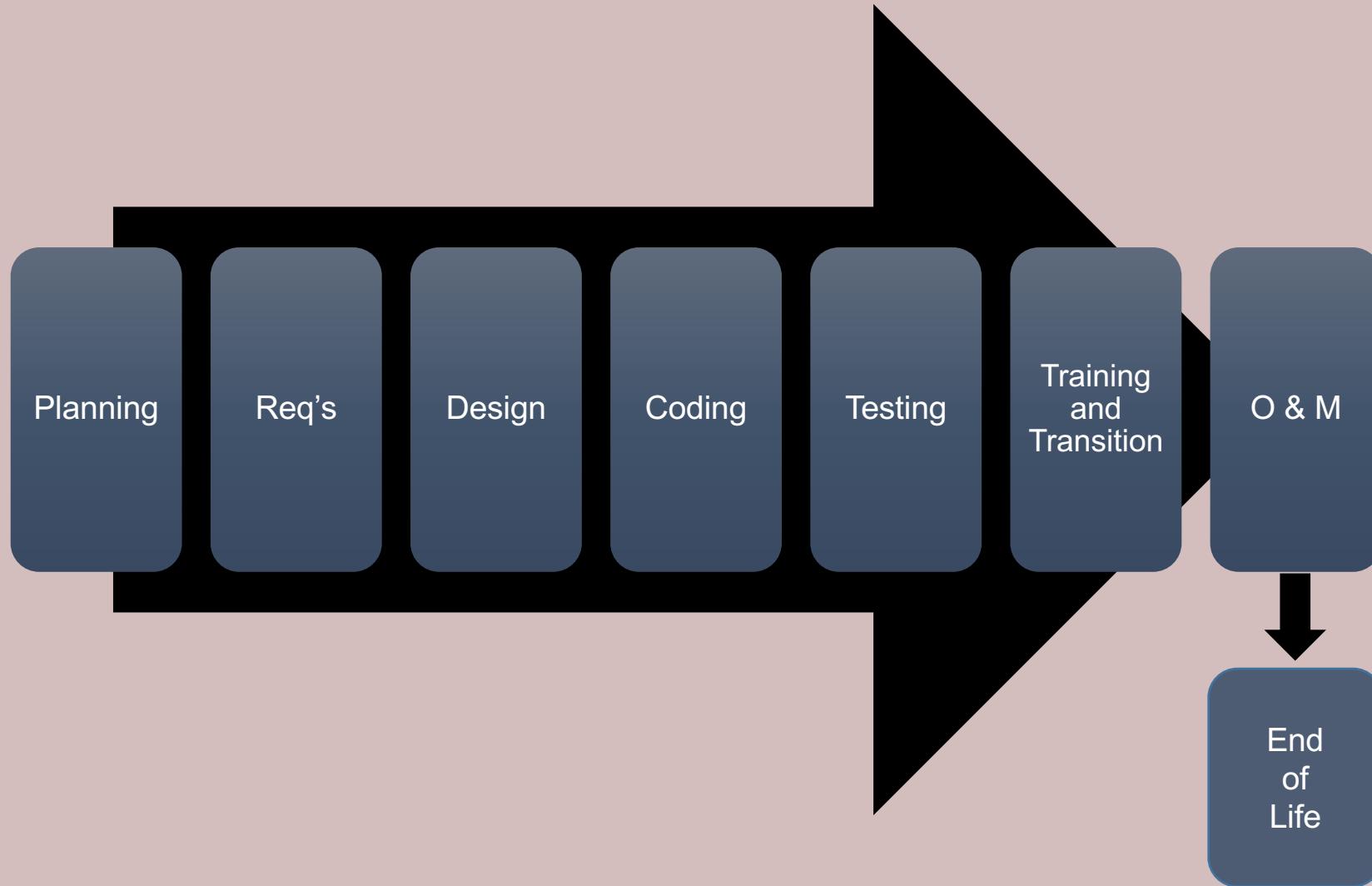
Security Architecture & Tool Sets

Software Development Life Cycle (SDLC)

- Software development doesn't always follow formal models
- Many different forms of SDLC... but all share 8 basic functions/phases
- SDLC can also be used for applications, services, systems, or other desired outputs
- Planning for security early in the process will provide better security at a cheaper price...



Software Development Life Cycle (SDLC)



Planning

- Initial investigations into the effort conducted
- Determines feasibility of designing the desired software, costs, and any alternate solutions
- End result: decision to move forward or not



Requirements

- Gain customer or stakeholder input to determine required functionality
- What should the program do?
- What does your current program not do?



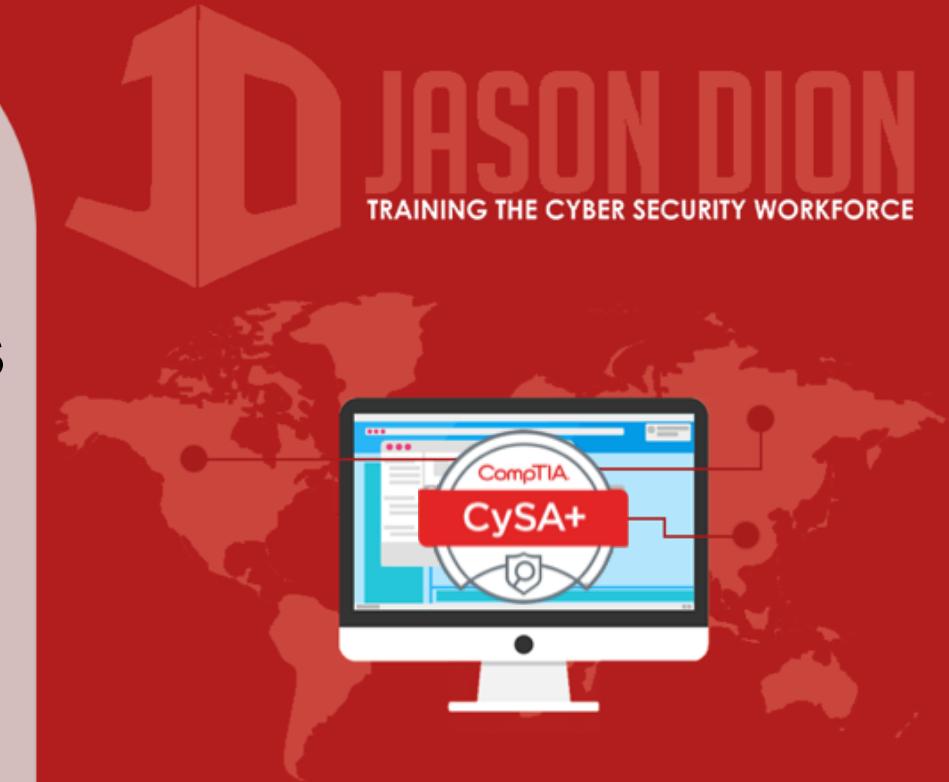
Design

- Creates designs for functionality, architecture, integration points, techniques, data flows, processes, and other elements



Coding

- Programmers start writing the code for the software and conduct testing of individual units of code and through code analysis



Testing

- Formal testing with outside development team (stakeholders, customers, beta group, etc)
- User Acceptance Testing ensure users are satisfied with the functionality



Training and Transition

- Ensures the end users are trained on software
- Consists of acceptance, installation, and deployment of software into live environment



Operations and Maintenance

- Longest phase of the SDLC
- Patching, updating, modification, and daily support for the new software occurs



End of Life

- Disposition and retirement of the software
- How will you stop supporting the software?
- Will you migrate users to a new version?



Software Development Life Cycle (SDLC)

