

10.5: Summary

Developing an information system is a complex process requiring close management of costs, timelines, and development teams. Following a structured systems development lifecycle (SDLC) process helps manage risk and deliver high quality solutions. Programming languages and tools continue to rapidly evolve, requiring developers to continuously learn new skills.

Software development involves much more than just coding. Cross-functional teams must collaborate to turn requirements into working software. Agile methods encourage constant user feedback and embrace changing requirements. Involving users throughout the development process improves software quality and user adoption.

Organizations must weigh factors like resources, timelines, and customization needs when deciding whether to build, buy, or subscribe to software. Security and compliance should be priorities early in development. Maintenance keeps systems aligned with evolving business needs after deployment.

End user computing outside of IT departments can improve agility but requires oversight. Effective change management and training helps drive user adoption of new systems. Global collaboration tools now enable software teams to work together effectively across locations and time zones.

By following best practices in software development, maintenance, and implementation, organizations can deliver solutions that are on time, on budget, and tailored to their business needs.

This page titled [10.5: Summary](#) is shared under a [CC BY 4.0](#) license and was authored, remixed, and/or curated by [Ly-Huong T. Pham and Tejal Desai-Naik](#) ([Evergreen Valley College](#)) .

- [10.5: Summary](#) by Ly-Huong T. Pham, Tejal Desai-Naik, Laurie Hammond, & Wael Abdeljabbar is licensed [CC BY 3.0](#).