CSOL 560 – Module 1 Assignment Table

SDLC Models

Factors	Waterfall	V-Shaped	Evolutionary Prototyping	Spiral	Iterative and	Agile	Incremental	RAD
Unclear User Requirements	Poor – Waterfall needs fixed requirements before starting.	Poor – V-Shaped relies on stable requirements.	Excellent – Prototyping helps clarify changing needs.	Excellent – Spiral refines unclear requirements through iterations.	Good – Iterative allows gradual refinement.	Excellent – Agile adapts easily to changing requirements.	Good – Incremental adds functionality as clarity improves.	Excellent – RAD involves users continuously for clarity.
Unfamiliar Technology	Poor – Waterfall doesn't adapt well to new tech challenges.	Poor – V-Shaped lacks flexibility for experimentation.	Good – Prototyping supports early experimentation.	Excellent – Spiral manages risk with unfamiliar tech.	Good – Iterative allows exploration in cycles.	Good – Agile adjusts quickly to tech learning curves.	Good – Incremental integrates learning per release.	Good – RAD allows rapid testing of new tech.
Complex Systems	Good – Waterfall handles complexity via structure.	Good – V-Shaped provides strong testing for complex builds.	Good – Prototyping visualizes complex processes early.	Excellent – Spiral manages complexity through risk analysis.	Good – Iterative simplifies by breaking complexity down.	Excellent – Agile uses modular sprints for complexity.	Good – Incremental manages complexity with staged builds.	Good – RAD handles complexity with iterative prototypes.
Reliable	Excellent –	Excellent – V-	Good –	Good – Spiral	Good –	Good –	Good –	Poor – RAD

Systems	Waterfall emphasizes documentation and predictability.	Shaped ensures reliability via verification steps.	Prototyping validates reliability progressively.	increases reliability through repetition.	Iterative improves reliability with each version.	Agile adds reliability through testing at the sprint level.	Incremental ensures reliability by integration testing.	trades reliability for speed.
Short Time Schedule	Poor – Waterfall takes a long time to complete all phases.	Poor – V-Shaped extends development with testing stages.	Good – Prototyping delivers quick partial builds.	Good – Spiral enables faster iterations per loop.	Excellent – Iterative allows early deliverables.	Excellent – Agile releases working software fast.	Excellent – Incremental provides usable builds sooner.	Excellent – RAD is designed for speed and quick output.
Strong Project Management	Excellent – Waterfall needs strong control to stay on schedule.	Excellent – V-Shaped depends on strict alignment with management.	Good – Prototyping needs coordination to avoid scope creep.	Excellent – Spiral relies on disciplined planning and risk tracking.	Good – Iterative benefits from coordinated team leads.	Good – Agile depends on skilled facilitators.	Good – Incremental needs management for staged planning.	Good – RAD requires balancing speed with control.
Cost Limitation	Good – Waterfall manages predictable costs, but changes are expensive.	Good – V-Shaped is cost-effective when requirements are fixed.	Poor – Prototyping can increase costs from iterations.	Poor – Spiral can be costly due to risk assessments.	Good – Iterative spreads cost across phases.	Good – Agile controls cost via prioritized sprints.	Excellent – Incremental offers ROI per release.	Excellent – RAD reuses components to save cost.
Visibility of Stakeholders	Poor – Waterfall	Poor – V-Shaped has	Excellent – Prototyping	Excellent – Spiral	Good – Iterative	Excellent – Agile	Good – Incremental	Excellent – RAD keeps

	hides progress until late.	limited visibility until testing.	keeps users engaged.	involves stakeholders in each loop.	provides regular feedback cycles.	ensures transparency through reviews.	gives visibility after each phase.	users active throughout.
Skills Limitation	Good – Waterfall guides inexperienced teams via structure.	Good – V-Shaped's defined process helps less skilled teams.	Good – Prototyping promotes hands-on learning.	Poor – Spiral requires expert risk management.	Good – Iterative supports learning by doing.	Good – Agile encourages team collaboration and mentorship.	Excellent – Incremental allows skill growth across releases.	Excellent – RAD supports rapid learning with user involvement.
Documentation	Excellent – Waterfall documents every phase.	Excellent – V-Shaped ensures traceability and records.	Good – Prototyping captures iterative updates.	Good – Spiral maintains documentation across loops.	Good – Iterative documents each phase's progress.	Good – Agile maintains minimal but effective records.	Good – Incremental updates to documentation with each release.	Poor – RAD prioritizes speed over documentation.
Component Reusability	Poor – Waterfall rarely reuses components.	Poor – V-Shaped focuses on testing over reuse.	Good – Prototyping allows reuse of prototype elements.	Excellent – Spiral supports modular reuse.	Excellent – Iterative encourages reusable designs.	Excellent – Agile promotes modular, reusable code.	Excellent – Incremental fosters component reuse.	Excellent – RAD reuses templates and components to speed up development.

References

Existek. (2025). SDLC Models: Agile, Waterfall, V-Shaped, Iterative, Spiral. https://existek.com/blog/sdlc-models/

GeeksforGeeks. (2025). Top 8 Software Development Models Used in Industry. https://www.geeksforgeeks.org/software-engineering/top-8-software-development-models-used-in-industry/

Ransome, J., & Misra, A. (2013). Core Software Security: Security at the Source. Auerbach Publishers.

Vasudevan, V. (2015). Application Security in the ISO27001:2013 Environment. IT Governance Ltd.