DrillBit

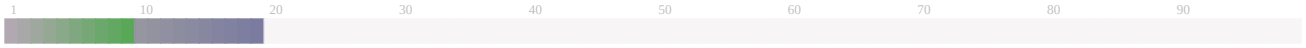
The Report is Generated by DrillBit Plagiarism Detection Software

Submission Information

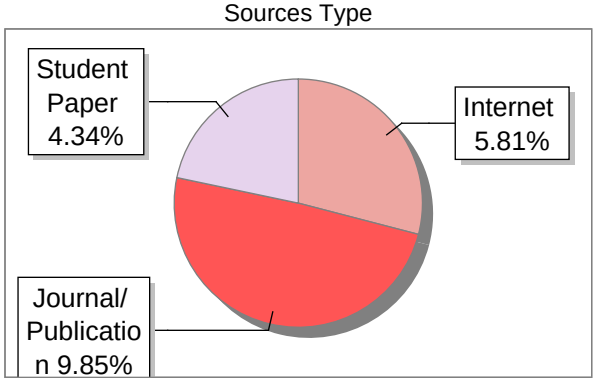
Author Name	Avani Tholpady
Title	NNM24IS048_task1
Paper/Submission ID	5265061
Submitted by	nnm24is040@nmamit.in
Submission Date	2026-02-12 22:57:05
Total Pages, Total Words	16, 2236
Document type	Assignment

Result Information

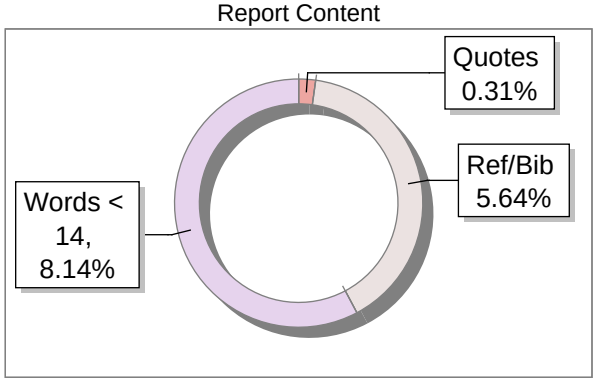
Similarity **20 %**



Sources Type



Report Content



Exclude Information

Quotes	Not Excluded
References/Bibliography	Not Excluded
Source: Excluded < 14 Words	Not Excluded
Excluded Source	**0 %**
Excluded Phrases	Not Excluded

Database Selection

Language	English
Student Papers	Yes
Journals & publishers	Yes
Internet or Web	Yes
Institution Repository	Yes

A Unique QR Code use to View/Download/Share Pdf File

DrillBit

DrillBit Similarity Report

20	28	B
SIMILARITY %	MATCHED SOURCES	GRADE

A-Satisfactory (0-10%)
B-Upgrade (11-40%)
C-Poor (41-60%)
D-Unacceptable (61-100%)

LOCATION	MATCHED DOMAIN	%	SOURCE TYPE
1	ijarcce.com	3	Publication
2	REPOSITORY - Submitted to NMAM INSTITUTE OF TECHNOLOGY on 2026-02-12 22-00 5264888	3	Student Paper
3	www.scribd.com	1	Internet Data
4	REPOSITORY - Submitted to Exam section VTU on 2026-02-10 12-57 4378558	1	Student Paper
5	Regression supervised model techniques THz MIMO antenna for 6G wireless communi, by Haque, Md. Ashraful, Yr-2024	1	Publication
6	Developing and leveraging digital twins in engineering design, by Anwer, N., Yr-2025	1	Publication
7	moam.info	1	Internet Data
8	www.cuny.edu	1	Publication
9	ieeexplore.ieee.org	1	Publication
10	www.diva-portal.org	1	Publication
11	www.readbag.com	1	Internet Data
12	coek.info	<1	Internet Data

13	dochero.tips	<1	Internet Data
14	repositorioslatinoamericanos	<1	Publication
15	Thesis Submitted to Shodhganga Repository	<1	Publication
16	Thesis Submitted to Shodhganga Repository	<1	Publication
17	www.dx.doi.org	<1	Publication
18	www.ethanetechnologies.com	<1	Internet Data
19	www.rapidinnovation.io	<1	Internet Data
20	IEEE 2015 12th International Conference on Information Technology -	<1	Publication
21	1library.co	<1	Internet Data
22	aei.pitt.edu	<1	Internet Data
23	moam.info	<1	Internet Data
24	Navigating the institutionalisation of smart technologies for urban sustainabil, by Vu, Khanh N., Yr-2025	<1	Publication
25	Platform for Plasmodium vivax vaccine discovery and development by Valencia-2011	<1	Publication
26	Symptoms, anxiety and handicap in dizzy patients Development of the V by Luc-1992	<1	Publication
27	www.freepatentsonline.com	<1	Internet Data
28	www.undp.org	<1	Publication

PROJECT REPORT

ON

Comparative Analysis of Software Process Models and Their Impact on Requirements Engineering in E-Commerce Systems

Submitted to

NMAM INSTITUTE OF TECHNOLOGY, NITTE

(Off-Campus Centre, Nitte Deemed to be University, Nitte - 574 110, Karnataka, India)

In partial fulfillment of the requirements for the award of the

Degree of Bachelor of Technology

in

INFORMATION SCIENCE AND ENGINEERING

By

Avani Tholpady NNM24IS048

Under the guidance of

Dr. JASON ELROY MARTIS

Associate Professor

NITTE (Deemed to be University) | NMAM INSTITUTE OF TECHNOLOGY

Abstract

This report examines how different software development models influence requirements engineering in an online e-commerce system. The Waterfall, Incremental, and Spiral models are compared based on flexibility, risk handling, and adaptability to change. Since e-commerce platforms operate in dynamic environments with frequent feature updates and strict security requirements, selecting a suitable SDLC model becomes critical. The study shows that iterative and risk-oriented approaches provide better support for evolving requirements compared to rigid sequential models.

GitHub Repository Link

https://github.com/avani-tholpady/avani-tholpady-sdlc-process-models-requirements-ecommerce

Introduction

Building a reliable software system requires more than technical implementation; it requires a structured development strategy. The process model adopted for a project influences communication, requirement clarity, cost, and overall system quality. In systems such as online shopping platforms, where features are continuously updated and security standards evolve, requirement management becomes especially important.

This report analyzes how three development models — Waterfall, Incremental, and Spiral — handle requirements engineering in an e-commerce environment. By comparing their strengths and limitations, the study aims to identify the most appropriate approach for dynamic and competitive digital platforms.

Problem Statement

In today's digital market, e-commerce platforms operate in fast-paced and competitive environments. Customer expectations change quickly, businesses frequently add new features, and technology is always reshaping online shopping experiences. In this dynamic context, choosing an appropriate Software Development Life Cycle (SDLC) model is vital.

An inappropriate process model can cause unclear requirements, poor communication among stakeholders, higher development costs, delays in delivery, and even system failure. E-commerce systems consist of complex components like user authentication, product management, payment gateways, and security measures. Therefore, effective requirements engineering is crucial for ensuring system success.

This study aims to analyze and compare three popular SDLC models: the Waterfall Model, Incremental Model, and Spiral Model. The goal is to assess how each impacts requirements management in the context of an e-commerce system. The objective is to find which model best supports flexibility, risk management, and ongoing improvement in a dynamic online business environment.

Tasks

- Study the Waterfall, Incremental, and Spiral models in detail
- Compare how each model handles requirements engineering
- Identify the most suitable model for developing an e-commerce system

Expected Outcomes

- A deeper understanding of how different SDLC models affect requirement management.
- Insights into choosing the right development model for dynamic systems like e-commerce platforms.

- Better strategies for managing evolving requirements in real-world projects.

Assumptions

- The system supports online product browsing, shopping cart functionality, and secure online payments
- The study is conceptual and analytical in nature, focusing on process comparison rather than implementation

System Description – Online E-Commerce Platform

System Overview

An online e-commerce platform is a web-based system that allows customers to browse products, place orders, and make secure online payments. Such systems integrate multiple modules including user management, product catalog management, order processing, and payment gateways. These platforms must be designed to handle high traffic while maintaining performance, reliability, and security.

Stakeholders

The main stakeholders in an e-commerce system often include customers, vendors, and system administrators:

Customers: Users who browse products, add items to the shopping cart, and complete transactions.

Vendors: Sellers who manage product listings, pricing, and inventory.

System Administrators: People responsible for maintaining system integrity, monitoring operations, and enforcing security policies.

Effective requirements engineering must consider the needs of all stakeholder groups to ensure system success.[4].

Data Handled

E-commerce platforms manage large volumes of data, including user information, product details, order history, and payment records. [1]. Since financial and personal data are involved, strong security measures like encryption and authentication are essential to prevent data breaches and fraud. [5].

Key Characteristics

E-commerce systems typically exhibit the following characteristics:

- Large User Base: They must support concurrent users with minimal latency [2].
- Frequent Updates: Features are regularly enhanced to remain competitive in dynamic markets [4].
- High Security Requirements: Secure payment processing and data protection are critical for user trust [5].
- Scalability: Systems must scale efficiently as transaction volume increases [2].

Requirement Sources

The system requirements are derived from stakeholder needs, business objectives, and established best practices in software engineering [4].

Methodology

1. Waterfall Model

The Waterfall model follows a strict sequence of phases. Each stage must be finished before moving on to the next. Requirements are gathered completely at the start, documented, and used as the basis for design and implementation.

While this structured process simplifies planning and documentation, it assumes requirements will remain unchanged throughout development. In the context of e-commerce systems, this assumption is often

unrealistic. Online platforms frequently add new payment options, promotional features, and customization capabilities. Since the Waterfall model provides limited support for revisiting previous phases, implementing these changes can greatly increase costs and delay delivery. Although the model offers clarity and control, its lack of flexibility reduces its effectiveness in dynamic environments.

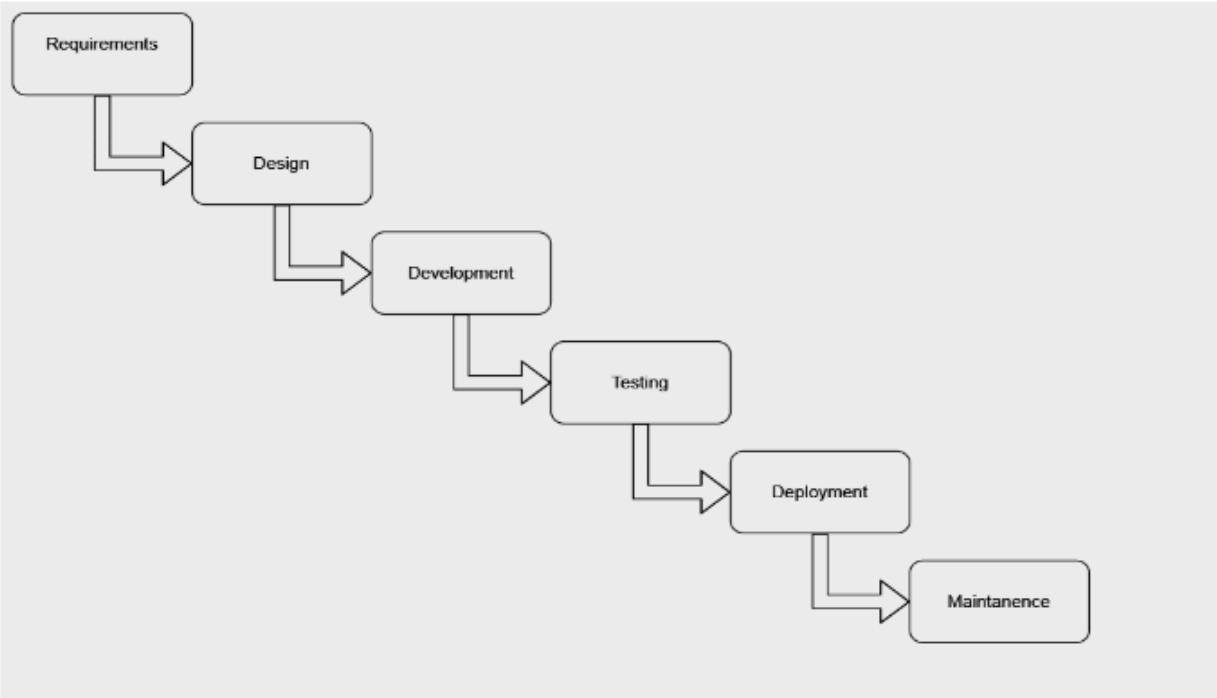


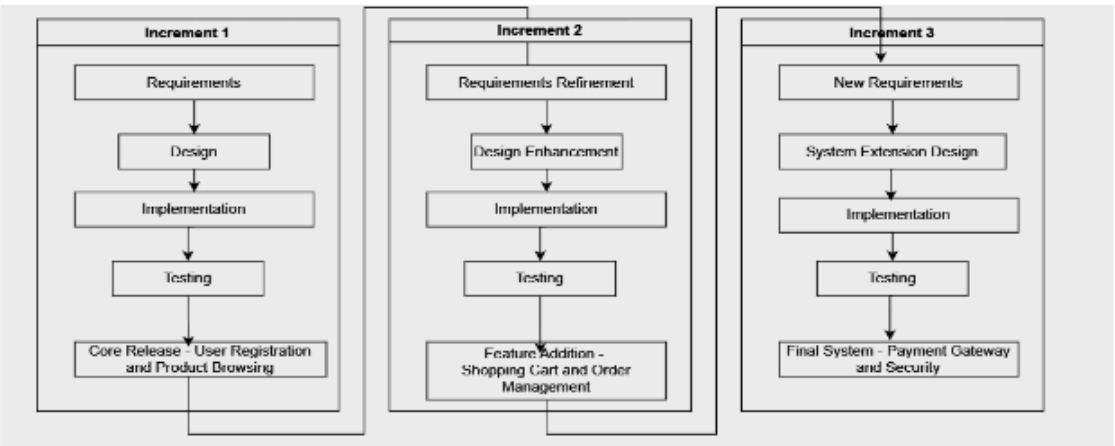
Fig (a)waterfall model

2. Incremental Development Model

The Incremental model divides development into smaller, manageable releases instead of a single complete delivery. Instead of building the entire e-commerce platform at once, core functions are implemented first, followed by additional features.

This staged approach allows early system use and encourages feedback from stakeholders after each release. For instance, user

registration and product browsing might be developed first, with cart management and payment integration added in later stages. This gradual development enhances requirement refinement and reduces the risk of large-scale failures. Compared to Waterfall, the Incremental model provides more adaptability while maintaining a structured approach.

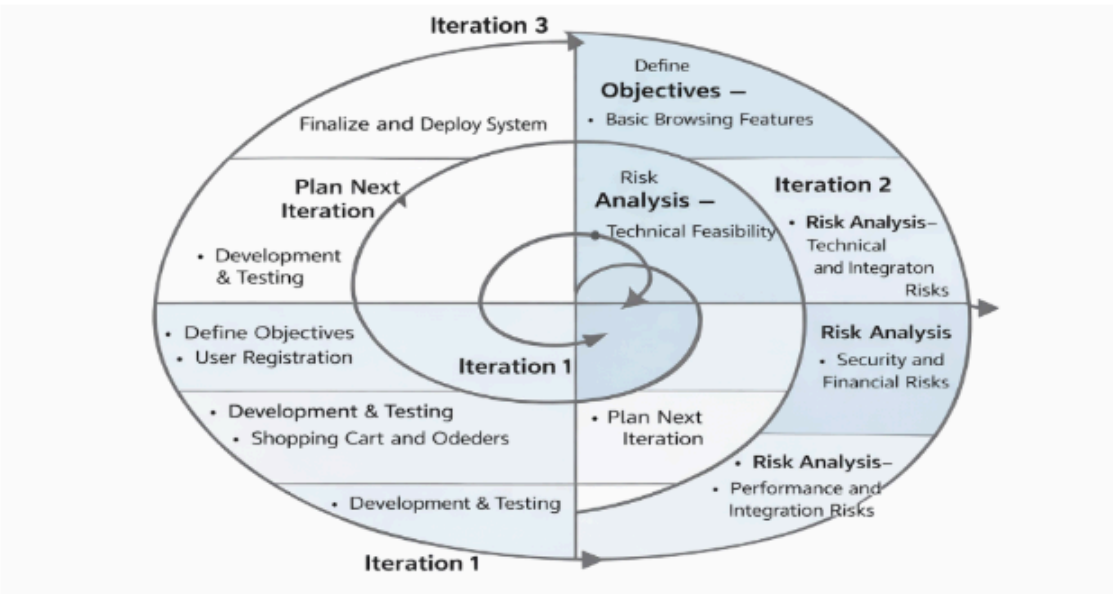


Fig(b) Incremental Development Model

3. Spiral Model

The Spiral model merges iterative development with systematic risk evaluation. Each development cycle includes identifying objectives, assessing potential risks, implementing solutions, and planning the next iteration. Unlike purely sequential models, this method continuously revisits and refines requirements.

For e-commerce systems dealing with financial transactions and sensitive user data, early risk identification is vital. Security vulnerabilities, integration challenges, and scalability issues can be evaluated before complete implementation. Although this model may require more planning and may incur higher development costs, it improves reliability and decreases long-term failures. Its focus on risk-driven refinement makes it ideal for complex and high-stakes platforms.



Fig(c)spiral model

Requirements Engineering

Requirements Engineering involves identifying, analyzing, documenting, and managing user needs and expectations for a software system. In an e-commerce system, requirements are generally categorized into Functional Requirements and Non-Functional Requirements.

Functional Requirements

Functional requirements describe what the system should do. They define the core features and operations of the online shopping system.

- User Registration and Authentication: The system must allow users to create an account, log in securely, and manage their profile information.
- Product Search and Filtering: Users should be able to search for products using keywords and apply filters such as price, category, brand, and ratings.
- Shopping Cart Management: The system should allow users to add, update, or remove items from the shopping cart before placing an order.
- Online Payment Processing: The system must support secure online payments through multiple payment methods such as credit/debit cards, UPI, or net banking.
- Order Tracking: After placing an order, users should be able to track the status of their order until delivery.

Non-Functional Requirements

Non-functional requirements focus on how the system performs, rather than what it does. They emphasize quality attributes.

- Security of Transactions: The system must ensure secure handling of user data and payment information using encryption and secure authentication mechanisms.
- High System Availability: The platform should be available to users at all times with minimal downtime.
- Fast Response Time: The system should respond quickly to user requests, even during peak traffic periods.
- Scalability to Handle High Traffic: The system must be capable of handling a large number of users simultaneously, especially during sales or promotional events.

Comparative Analysis of SDLC Models for an E-Commerce System

To determine the most appropriate development approach for an online e-commerce platform, the Waterfall, Incremental, and Spiral models are compared based on requirement flexibility, risk handling, cost implications, and overall suitability for dynamic business environments.

Comparison Table

Aspect	Waterfall Model	Incremental Model	Spiral Model
Requirement Flexibility	Low – requirements must be fixed at the beginning	Moderate – allows changes between increments	High – supports continuous refinement of requirements
Risk Management	Weak – risks identified late in testing phase	Moderate – risks reduced through staged releases	Strong – explicit risk analysis in every iteration
Customer Involvement	Minimal after requirement phase	Regular feedback after each increment	Continuous stakeholder evaluation
Development Cost	Low initially but costly if changes occur	Moderate and distributed over increments	Higher due to repeated risk analysis and prototyping
Suitability for E-Commerce	Low	High	Very High

Analytical Discussion

A structured comparison of the three models reveals clear differences in their ability to manage evolving requirements. The Waterfall model performs adequately in projects where requirements remain fixed; however, it struggles when changes occur mid-development. In competitive online markets, where business strategies and customer expectations shift rapidly, such rigidity can become a limitation.

The Incremental approach improves adaptability by allowing gradual expansion of functionality. Stakeholder feedback after each release supports continuous refinement. Nevertheless, it does not explicitly prioritize risk evaluation in the same way as the Spiral model.

The Spiral model stands out due to its deliberate focus on identifying and mitigating risks throughout development. By integrating iterative

refinement with structured analysis, it provides stronger alignment with modern software engineering recommendations, particularly in high-risk systems like e-commerce platforms.

Most Suitable Model

Based on the comparison, the Spiral model is most suitable for large-scale e-commerce platforms due to its strong risk management and flexibility. For small to medium-scale systems, the Incremental model offers a balanced approach between adaptability and cost efficiency.

Discussion

The comparative evaluation clearly indicates that rigid and sequential models such as Waterfall face significant limitations in environments where requirements evolve frequently. In the context of an e-commerce platform, customer preferences, payment technologies, and security standards change rapidly, making adaptability a critical success factor.

Incremental development improves flexibility by enabling staged delivery and early stakeholder feedback.

However, the Spiral model provides the strongest support for requirements engineering due to its structured emphasis on risk assessment and continuous validation. According to IEEE guidelines on software requirements engineering [3], ongoing validation and stakeholder involvement are essential to ensure system reliability and quality. Iterative models such as Spiral align more closely with these recommendations, making them more suitable for complex and high-risk systems like e-commerce platforms.

Requirements Validation Strategy

Effective requirements validation is essential to ensure that the developed e-commerce system aligns with stakeholder expectations and business goals. Since e-commerce platforms operate in dynamic environments, continuous verification and validation of requirements become crucial.

The following validation strategies were considered:

• **Requirement Reviews with Stakeholders:** Formal review meetings are conducted with customers, vendors, and system administrators to verify that requirements are accurate, complete, and feasible. This helps in identifying inconsistencies at an early stage.

• **Prototyping:** Low-fidelity or functional prototypes are developed to visually demonstrate system features such as product search, cart management, and payment processing. Prototyping helps stakeholders better understand system functionality and reduces ambiguity.

• **Walkthrough Sessions:** Structured walkthroughs of requirement documents are performed to detect missing details, logical errors, and unclear specifications. These sessions improve requirement clarity and mutual understanding.

• **User Acceptance Testing (UAT):** Before final deployment, stakeholders validate the system against agreed requirements to ensure it satisfies business objectives and user expectations.

Challenges in Requirements Validation

Despite structured validation techniques, several challenges may arise:

• **Ambiguous Requirements:** Vague or poorly defined requirements may lead to misinterpretation during development.

• **Frequent Requirement Changes:** Due to evolving customer demands and market competition, requirements may change frequently, impacting project scope and timeline.

• **Stakeholder Communication Gaps:** Miscommunication between developers and stakeholders can result in incomplete or incorrect system functionality.

Conclusion

The analysis demonstrates that the effectiveness of requirements engineering is closely connected to the selected development model. In rapidly evolving domains such as e-commerce, flexibility and risk awareness are essential for long-term system success.

While the Waterfall model provides a structured process, it is less suitable for environments where requirements frequently change. The Incremental model enhances adaptability through phased delivery, and the Spiral model strengthens reliability by integrating continuous risk evaluation.

Ultimately, selecting an appropriate SDLC model contributes to improved requirement control, reduced project risk, and better alignment with business goals. For modern e-commerce platforms, iterative and risk-driven approaches provide stronger support than rigid sequential development methods.

References

[1] I. Sommerville, *Software Engineering*, 10th ed. Harlow, U.K.: Pearson, 2016.

[2] R. S. Pressman and B. R. Maxim, *Software Engineering: A Practitioner's Approach*, 8th ed. New York, NY, USA: McGraw-Hill Education, 2019.

[3] IEEE Computer Society, *IEEE Std 830-1998: IEEE Recommended Practice for Software Requirements Specifications*. New York, NY, USA: IEEE, 1998.

[4] R. Mall, *Fundamentals of Software Engineering*, 5th ed. New Delhi, India: PHI Learning, 2018.

[5] W. W. Royce, "Managing the Development of Large Software Systems," in *Proceedings of IEEE WESCON*, Los Angeles, CA, USA, 1970, pp. 1–9.

Declaration of Academic Integrity

I hereby declare that this assignment is my original work and has not been copied or plagiarized from any source. All references have been properly cited.

Student

Signature: 

Date: 13-02-2026