1. Clearly Defined Requirements

Application’s scope, entities, actions, and workflows are well-defined from the outset. Waterfall is most effective when the requirements are fixed and not likely to change during the development process, as it assumes that all details are known upfront. Newsagent’s system has a structured flow (areas, delivery processes, dockets, invoices, etc.), which makes it suitable for a linear development approach.

2. Structured Phases for Documentation

The Waterfall model emphasizes documentation, ensuring each phase (requirement gathering, design, implementation, testing, etc.) is documented thoroughly. Given the number of entities, actions, and complex processes (order tracking, stock management, invoice generation, etc.), documentation is critical. This would help in establishing a clear understanding of the business logic and workflows for future references or updates.

3. Well-Defined End Goal

This system has a concrete end goal — to handle deliveries, orders, customer management, invoicing, and other related processes efficiently. Waterfall excels in projects where the goal is well-known and unlikely to evolve much. Once all features are implemented and tested (like stock control, customer order updates, dockets handling, and invoicing), the software will be expected to function without frequent changes.

4. Simple, Sequential Process

Process for building application can follow a clear, sequential path:

-Requirement Analysis: Understand and document all the processes involved (like handling delivery dockets, generating invoices, tracking stock).

- System Design: Architect the system (mapping areas, deliveries, customer entities).

- Implementation: Code the features such as customer creation, order management, and delivery schedules.

- Testing: Test each entity and function independently (e.g., does stock decrement when a delivery is made?).

- Deployment: Deploy the final product and integrate with any third-party systems (if required).

5. Predictable Timeline and Milestones

Since requirements seem fixed, Waterfall allows for creating a predictable timeline with clear milestones (e.g., delivery system first, invoicing second). It’s easier to manage because each stage is completed before moving to the next.

6. Low Risk of Change

Waterfall is ideal when you anticipate low risks for changes during development. Since this application is for managing deliveries, orders, and customer relations in a specific town (Athlone) with predefined areas, changes in requirements seem unlikely. New features may not be added frequently, making Waterfall a suitable choice.

7. Good Fit for a Smaller, Well-Scoped Project

While the application involves multiple areas and processes, it has a clear and relatively confined scope. This makes it easier to define deliverables and track progress, which aligns with the Waterfall model's phase-wise approach. Waterfall tends to work well for well-bounded projects with clear deliverables.

Conclusion:

Waterfall would be a good choice for this project because it is suitable for applications with stable requirements, sequential tasks, and a predictable development process. Each part of this application—customer management, order processing, delivery tracking, invoicing—can be built and tested in isolated stages before moving to the next, ensuring quality and thorough testing.