AGILE VS WATERFALL COMPARISON CHART AND SITUATIONS WHERE I WOULD CHOOSE ONE OVER THE OTHER

MODELS	**WATERFALL**	**AGILE**
Project Structure	Sequential, step-by-step	Iterative, flexible
Requirements	Defined upfront and unlikely to change	Evolve throughout the project
Customer Involvement	Minimal after initial requirements	Continuous involvement through sprints
Flexibility to Change	Low — expensive and difficult to change	High — easy to adapt to feedback
Testing	After build is complete	Continuous testing throughout
Delivery	One final delivery at project end	Frequent deliveries (increments, sprints)
Best for	Construction, manufacturing, infrastructure	Software development, product design, startups
Risk Management	Risk is high if wrong at the start	Risk is spread and corrected early
Cost of Changes	High after project begins	Lower because of early and continuous feedback
Documentation	Heavy and detailed	Light and adaptive

Over the course of my career, these are situations where I would have chosen Agile over Waterfall if such situations had arisen:

- **Revamping the customer service workflow (introducing new ticketing processes):**
  Reason: One might discover better ways to serve customers as changes are rolled out, so frequent adjustments are useful.
- **Improving an internal sales support chatbot:**
  Reason: The chatbot needs regular feedback from users to improve responses and functionality. Small updates can be made often.

I would have chosen Waterfall over Agile in the following situations:

- **Setting up a call centre in a new location:**
  Reason: Setting up infrastructure (desks, phone lines, schedules) requires clear steps that must happen in sequence.
- **Migrating customer data from an old system to a new one:**
  Reason: Data migration projects require a fixed plan, detailed requirements, strict deadlines, and minimal room for changes.