Role of Testing, Business Perspective

	Concept	Pretotype	Prototype	Productize	Deploy	Support	Update	Retire
Business Drivers	More Money	ROI Potential, Risk	Market Potential, Progress, Risk	Total Cost to Release < Continue Improving		Cost to Fix < Lost Time/ Revenue	Increased Revenue Potential	Cost to Support > Revenue Potential
Business Decisions		Fund? Amount?	More Funds?	Release?		Fix?	New Version?	End Of Life?
Product State		Pitch	Beta	Release Candidate		Fix	Ver. n+1	
Project Activity		Idea, Research, Create & Deliver "Pitch" Identify Roadblocks & Deal-Breakers	Build MVP (quick & cheap) Ensure MVP Is: On Vision, Minimally Fit for Use & Worthy of Sale	Convert MVP to Production Product, (quick & cheap) Assess & Pursue Shortest Path to Revenue Generating/ Protecting Product	Keep Support & Maintenance Costs Down Predict & Prevent Revenue/Profit Loss (a.k.a. Mitigate Risk)	Resolve. Fast! (and cheap) Stop Revenue Loss w/o Causing Other Revenue Loss	Pitch → MVP → Release Candidate; Faster & Cheaper Than Before Realize New Revenue Opportunities w/o Damaging Existing Revenue Streams	Upgrade/ Upsell Alternatives Enable Smooth Transition for Customers/ Users
Expected Questions Addressed Via Testing		Is there a viable market for it? What are the business risks?	Consistent with the vision? Minimally fit for use? Most common/ important functionality sufficient for sale/use?	Is it fit for production? Does it comply with regulations? Are business & technical risks being appropriately mitigated?	Why do users choose a competitor/ return product/ not renew? What support calls cost most? Why do users replace with a competitor?	Impact of the change? How risky is it? Quicker/ cheaper/safer option? How do we get it out fast with confidence?	Same as previous, only better, faster and cheaper.	What can we do at an absolutely minimal cost to protect customer data? Loyalty? Privacy?