Q) In your view which software development methodology focuses heavily on maximising stakeholders’ Return on Investment (ROI) and minimizing the cost of change? Justify your answer. **[6 marks]**

Agile methodology

**Justification:**

**Superior Prioritization and Risk Management for Business Success:**

Agile is one of the only methodologies that allow iterative evaluation of project progress and viability enabling risk assessment between sprints. The risk is either minimized greatly or feedback loops allow for better risk management. The Agile approach allows cash flow early in the project lifecycle, enabling low upfront costs due to the self-funding nature of projects.

**High-Value Revenue even with Low Upfront and Ongoing Costs:**

High communication within team members, cohesive development approach, quality focus and incremental development process allows for superior, marketable products. Enabling scope and feature set flexibility through a cost containment approach restrict frequent outflow of cash. Overall, the ability of Agile to generate a steady revenue stream overlapping project delivery cycles ensures business success in the long run.

Q) Software experts seem to agree that the easiest way to maximize stakeholder ROI, and to minimize the cost of change, is to implement the highest priority requirements first. Because requirements change frequently, explain how your chosen software development methodology ensures a streamlined, flexible approach to requirements change management. **[13 marks]**

With Agile, three versions of this practice to manage requirements changes are:

* Product backlog (Scrum)
* Work item list (Disciplined Agile)
* Option pool (Lean)

Any one of them could be elaborated as part of answering this part of this question.

Scrum approach:

Used to manage requirements where your software development team has a stack of prioritized and estimated requirements which need to be addressed (Scrum calls this prioritized stack a "product backlog"). There are several important points to understand:

* New requirements are prioritized by your project stakeholders and added to the stack in the appropriate place.
* Fundamentally a single person needs to be the final authority when it comes to requirement prioritization. In Scrum, and Disciplined Agile Delivery (DAD) which adopts this role from Scrum, this person is called the product owner.
* Although there is often many project stakeholders, including end users, managers, architects, operations staff, to name a few and the product owner is responsible for representing them all.
* The stack/backlog is initially filled as the result of your requirements envisioning efforts at the beginning of the project (in Scrum they call this "populating the backlog").
* Each iteration (a "sprint" in Scrum terminology) your team pulls an iteration's worth of work off the top of the stack and commits to implementing it by the end of the iteration.
* Your project stakeholders have the right to define new requirements, change their minds about existing requirements, and even reprioritize requirements as they see fit.
* Stakeholders are responsible for making decisions and providing information in a timely manner. On some projects a business analyst, often in the role of product owner, may take on this responsibility. Whoever is in this role will need to work together with the other stakeholders to ensure everyone is represented fairly, often a difficult task.
* The priorities of non-requirement work items are either negotiated by the team with stakeholders or are addressed as part of slack time within the schedule. Many Scrum teams are now putting more than just requirements, such as defects, on their backlogs.

