1. a.i. 

ii. REASONS FOR CONSIDERING FORMAL METHODS

* Systems are becoming increasingly dependent on software components;
* Complexity of Systems with embedded software has increased - A modern mid-sized sedan car is equipped with at least 150 processors;
* Maintaining reliability in software-intensive systems is very difficult. Quality problems with software may cause minor irritations or major damage to a customer’s business including loss of life.

b. A functional requirement defines a system or its component. It describes the functions a software must perform. A function is nothing but inputs, its behavior, and outputs. It can be a calculation, data manipulation, business process, user interaction, or any other specific functionality which defines what function a system is likely to perform. Functional software requirements help you to capture the intended behavior of the system. While, a non-functional requirement defines the quality attribute of a software system. They represent a set of standards used to judge the specific operation of a system. A non-functional requirement is essential to ensure the usability and effectiveness of the entire software system. Failing to meet non-functional requirements can result in systems that fail to satisfy user needs. Non-functional Requirements allows you to impose constraints or restrictions on the design of the system across the various agile backlogs.

1. a.
	* + Agile development: Agile software development" refers to a group of software development methodologies based on iterative development, where requirements and solutions evolve via collaboration between self-organizing cross-functional teams.
		+ Waterfall development: The waterfall model is a sequential development approach, in which development is seen as flowing steadily downwards (like a waterfall) through several phases
		+ Spiral development: it combines some key aspect of the [waterfall model](https://en.wikipedia.org/wiki/Waterfall_model) and [rapid prototyping](https://en.wikipedia.org/wiki/Rapid_application_development) methodologies, in an effort to combine advantages of [top-down and bottom-up](https://en.wikipedia.org/wiki/Top-down_and_bottom-up_design) concepts. It provided emphasis in a key area many felt had been neglected by other methodologies.

b. formal methods refers to various mathematical techniques used for the formal specification and development of software. They consist of a formal specification language and employ a collection of tools to support the syntax checking of the specification, as well as the proof of properties of the specification. They allow questions to be asked about what the system does independently of the implementation. The use of mathematical notation avoids speculation about the meaning of phrases in an imprecisely worded natural language description of a system. Natural language is inherently ambiguous, whereas mathematics employs a precise rigorous notation.

At the heart of formal methods one finds the notion of specification. A specification is a model of a system that contains a description of its desired behaviour—what is to be implemented, by opposition to how. This specification may be totally abstract (in which case the model is the description of the behaviour), or it may be more operational, in which case the description is somehow contained, or implied, by the model.

a.i. Propositional logic, is the branch of logic that studies ways of joining and/or modifying entire propositions, statements or sentences to form more complicated propositions, statements or sentences, as well as the logical relationships and properties that are derived from these methods of combining or altering statements. In propositional logic, the simplest statements are considered as indivisible units, and hence, propositional logic does not study those logical properties and relations that depend upon parts of statements that are not themselves statements on their own, such as the subject and predicate of a statement. While, first order predicate logic is symbolized reasoning in which each sentence, or statement, is broken down into a subject and a predicate. The predicate modifies or defines the properties of the subject. In first-order logic, a predicate can only refer to a single subject.



b.

* Model-Oriented Approach:The model-oriented approach to specification is based on mathematical models, where a model is a simplification or abstraction of the real world that contains only the essential details. For example, the model of an aircraft will not include the colour of the aircraft, and the objective may be to model the aerodynamics of the aircraft. There are many models employed in the physical world, such as meteorological models that allow weather forecasts to be made.The importance of models is that they serve to explain the behaviour of a particular entity and may also be used to predict future behaviour.
* Axiomatic Approach :The axiomatic approach focuses on the properties that the proposed system is to satisfy, and there is no intention to produce an abstract model of the system. The required properties and behaviour of the system are stated in mathematical notation. The difference between the axiomatic specification and a model-based approach may be seen in the example of a stack.
1. a.
2. a well formed formula is a syntactically correct formula created according to the syntactic rules of the underlying calculus
3. a quantification is performed on formulas of predicate logic (called wff ).There are two types of quantifiers: universal quantifier and existential quantifier. The universal quantifier turns, for example, the statement *x > 1* to "for every object *x* in the universe, *x > 1*", which is expressed as "*x* *x > 1*". This new statement is true or false in the universe of discourse. Hence it is a proposition once the universe is specified.

Similarly the existential quantifier turns, for example, the statement *x > 1* to "for some object *x* in the universe, *x > 1*", which is expressed as " *x x > 1*." Again, it is true or false in the universe of discourse, and hence it is a proposition once the universe is specified.

1. A predicate is an expression of one or more variables defined on some specific domain.

b.

1. Please I did not see anything on 6 and 7.