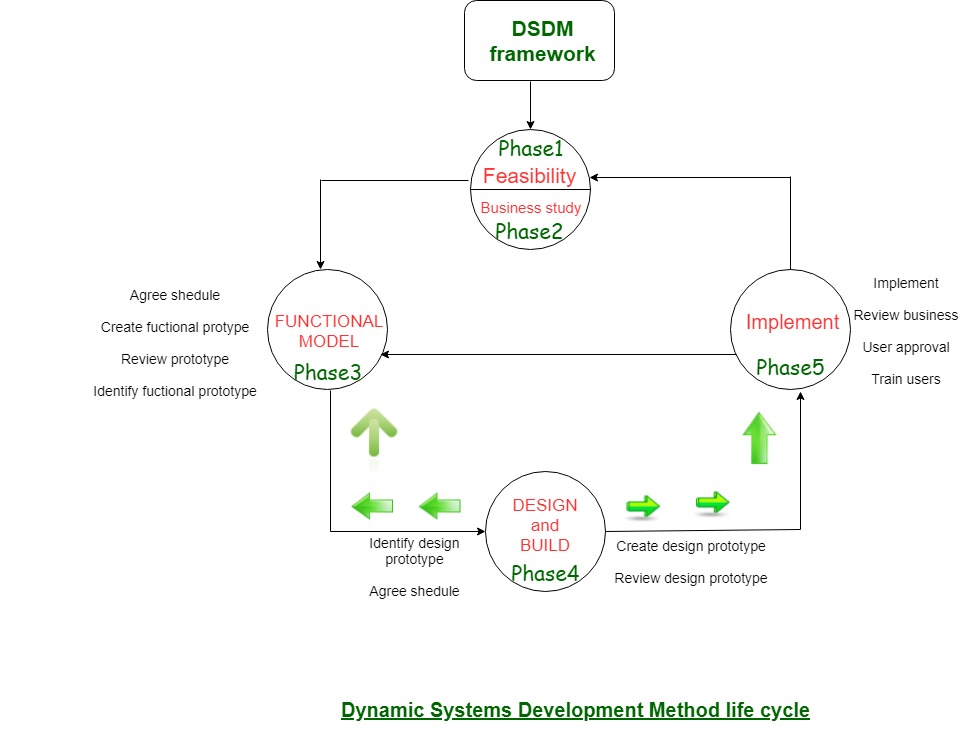
The **Dynamic Systems Development technique (DSDM)** is an associate degree agile code development approach that provides a framework for building and maintaining systems. The DSDM philosophy is borrowed from a modified version of the sociologist principle—80 % of An application is often delivered in twenty percent of the time it’d desire deliver the entire (100 percent) application.

DSDM is An iterative code method within which every iteration follows the 80% rule that simply enough work is needed for every increment to facilitate movement to the following increment. The remaining detail is often completed later once a lot of business necessities are noted or changes are requested and accommodated.

The DSDM tool (www.dsdm.org) could be a worldwide cluster of member companies that put together tackle the role of “keeper” of the strategy. The pool has outlined AN [Agile Development Model](https://www.geeksforgeeks.org/software-engineering-agile-development-models/), known as the DSDM life cycle that defines 3 different unvarying cycles, preceded by 2 further life cycle activities:

1. **Feasibility Study:**  
   It establishes the essential business necessities and constraints related to the applying to be designed then assesses whether or not the application could be a viable candidate for the DSDM method.
2. **Business Study:**  
   It establishes the use and knowledge necessities that may permit the applying to supply business value; additionally, it is the essential application design and identifies the maintainability necessities for the applying.
3. **Functional Model Iteration:**  
   It produces a collection of progressive prototypes that demonstrate practicality for the client.  
   (Note: All DSDM prototypes are supposed to evolve into the deliverable application.) The intent throughout this unvarying cycle is to collect further necessities by eliciting feedback from users as they exercise the paradigm.
4. **Design and Build Iteration:**  
   It revisits prototypes designed throughout useful model iteration to make sure that everyone has been designed during a manner that may alter it to supply operational business price for finish users. In some cases, useful model iteration and style and build iteration occur at the same time.
5. **Implementation:**  
   It places the newest code increment (an “operationalized” prototype) into the operational surroundings. It ought to be noted that:
   * **(a)** the increment might not 100% complete or,
   * **(b)** changes are also requested because the increment is placed into place. In either case, DSDM development work continues by returning to the useful model iteration activity.

Below diagram describe the DSDM life cycle:



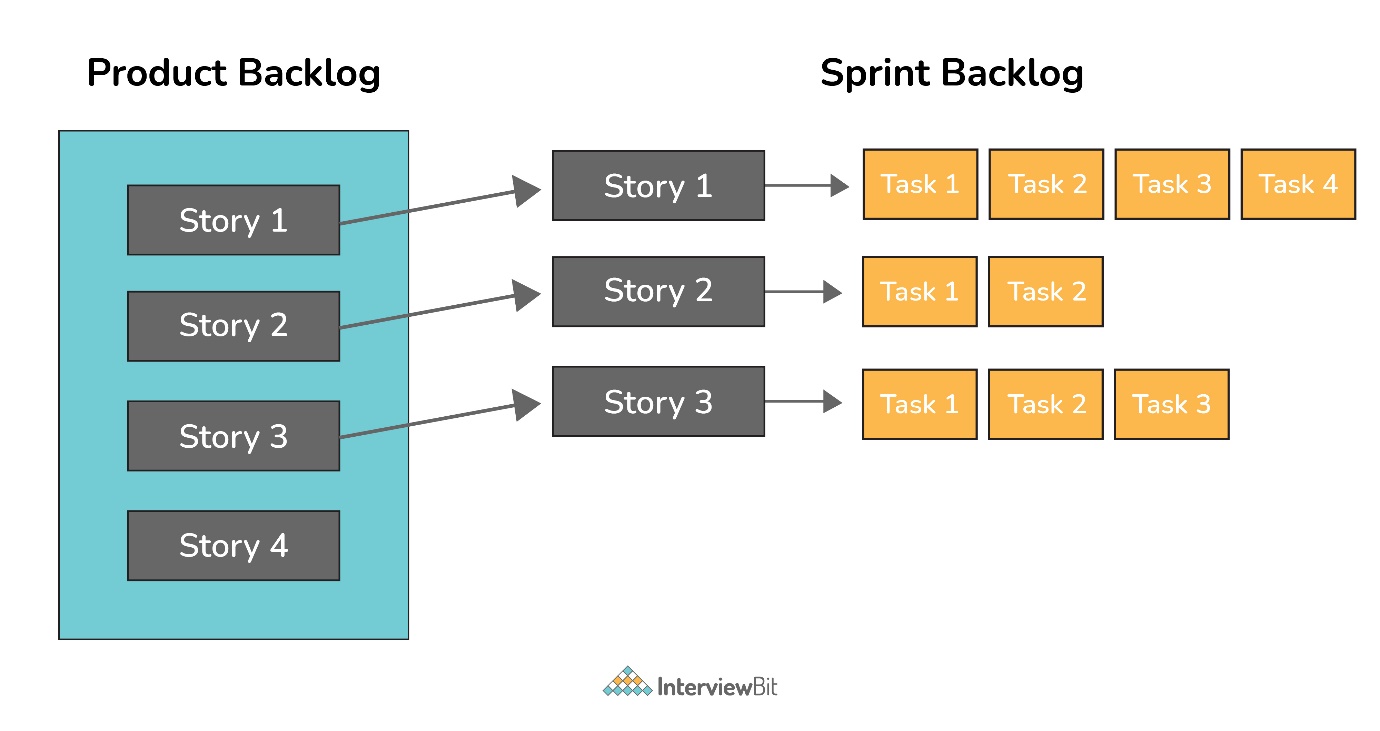
DSDM is often combined with XP to supply a mixed approach that defines a solid method model (the DSDM life cycle) with the barmy and bolt practices (XP) that are needed to create code increments. additionally, the ASD ideas of collaboration and self-organizing groups are often tailored to a combined method model.

**Principles of Agile Testing**

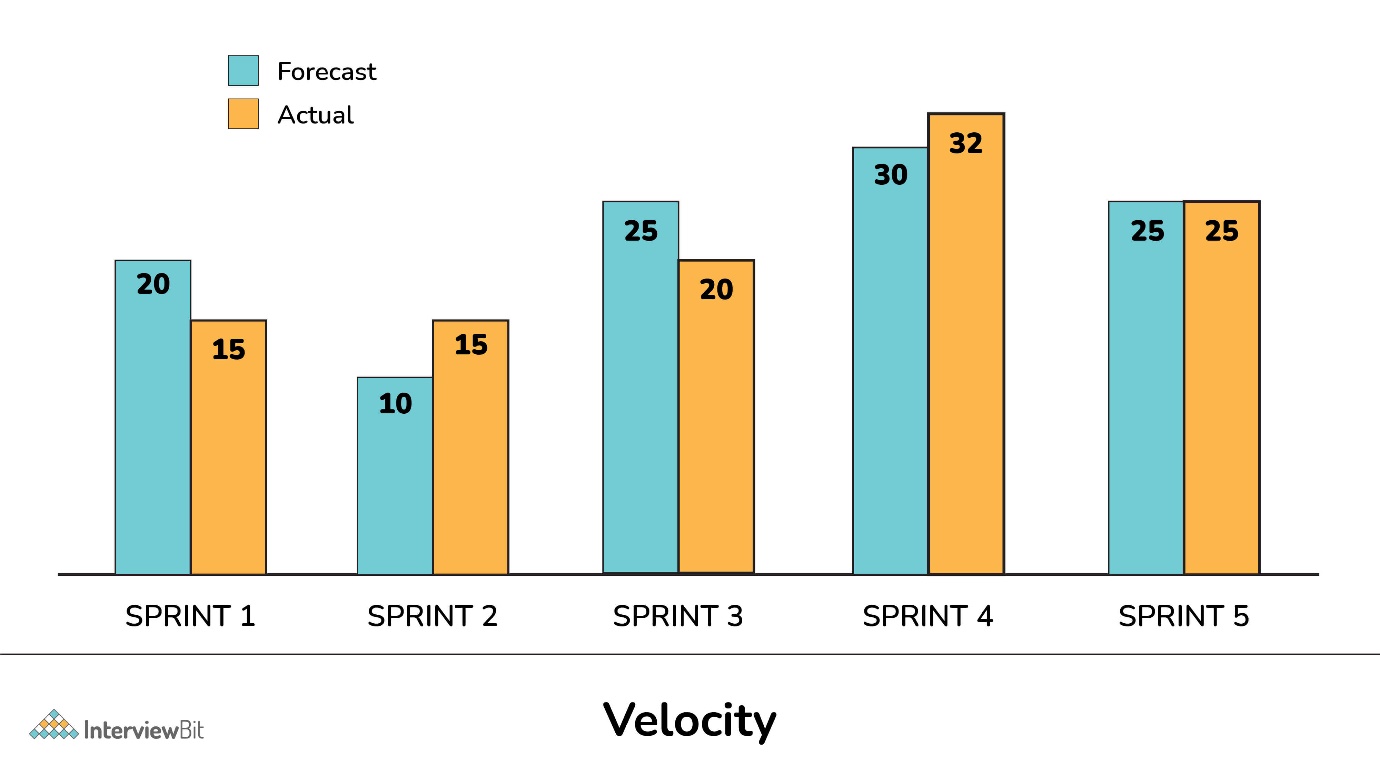
There are eight main principles of Agile Testing as given below:

* **Continuous Testing:** Testing should be conducted continuously by the Agile team to ensure continuous development progress.
* **Continuous Feedback:** This process generally encourages taking feedback from clients to make sure that the product meets the requirements of the client or customer.
* **Team Work or collective work:** Not only testers but developers, business analysts can also perform software testing or application testing.
* **Clean Code:** Quality of software is maintained as the team tests the software to ensure that the code is clean, simple, and tight. All errors and defects that are found during the testing phase are fixed quickly within the same iteration by the Agile Team.
* **Less Documentation:** This process usually involves the usage of reusable checklists instead of lengthy documentation.
* **Test-Driven:** In other conventional methods, testing is only performed after the implementation but in agile testing, testing is done during the implementation so that errors or any issues can be removed on time.
* **Customer Satisfaction:** During the agile testing process, development progress is being shown to clients or customers so that they can adapt and update their requirements. This is done to ensure customer satisfaction.

**Sprint Backlog:** It is generally owned by the development team. It only contains those features and requirements that are related to the specific sprint only. It is considered a subset of the product backlog. It is compiled of everything that must be done to complete a particular sprint. It only includes items that can be completed during each agile sprint. It is specific to the sprint goal only in a particular sprint.   
**Product Backlog:**It is generally owned and maintained by the project owner. It usually contains each and every feature of the product as well as the requirements of the product. It is compiled to everything that must be done to complete the whole process. It just breaks down every item into a series of steps. It is more specific to the end goal of the product.



A velocity is basically a measurement unit that measures or calculates how much work an agile development team can successfully complete in a single sprint and how much time will be required to finish a project. It is widely used as a calibration tool that helps development teams to create accurate and efficient timelines. It is also used to identify problems and measure the improvements that occur with time.



**What is SDLC?**

**SDLC** is a systematic process for building software that ensures the quality and correctness of the software built. SDLC process aims to produce high-quality software that meets customer expectations. The system development should be complete in the pre-defined time frame and cost. SDLC consists of a detailed plan which explains how to plan, build, and maintain specific software. Every phase of the SDLC life Cycle has its own process and deliverables that feed into the next phase. SDLC stands for **Software Development Life Cycle** and is also referred to as the Application Development life-cycle.

* [Phase 1: Requirement collection and analysis](https://www.guru99.com/software-development-life-cycle-tutorial.html#4)
* [Phase 2: Feasibility study](https://www.guru99.com/software-development-life-cycle-tutorial.html#5)
* [Phase 3: Design](https://www.guru99.com/software-development-life-cycle-tutorial.html#6)
* [Phase 4: Coding](https://www.guru99.com/software-development-life-cycle-tutorial.html#7)
* [Phase 5: Testing](https://www.guru99.com/software-development-life-cycle-tutorial.html#8)
* [Phase 6: Installation/Deployment](https://www.guru99.com/software-development-life-cycle-tutorial.html#9)
* [Phase 7: Maintenanc](https://www.guru99.com/software-development-life-cycle-tutorial.html#10)