**Iterative Advantages:**

1. Iterative development can reduce the risk that the project cannot be completed on time.
2. Get feedback from early customers. Some customers have no clear requirements for the project at the beginning of the project, so quickly make an incomplete project for customers to give feedback.
3. Continuous improvement and modification, as iterative development will complete an imperfect result very quickly, and then improve through the customer's request until the project is completed and meets the customers’ needs.
4. Improve developer productivity, iterative development allows developers to clearly understand the current work tasks, developers only need to concentrate on the current tasks.

**Iterative Disadvantages:**

1. Highly competent project managers and high-tech development teams are needed.
2. Since iterative development is gradually incorporated into functionality, later optimization is an important and arduous task.
3. It is important to coordinate the cooperation between members during iterative development. If one of the members does not complete the task, it may drag down the progress of the entire project.

**Compare SCRUM to Waterfall:**

Waterfall is a sequentially progress methodology. The minimum five stages contain: Planning, Designing, Performing, Testing, and Deploying. With it being sequentially contained, it is impossible to return to the previous stage, and none of them can begin without the previous stage being finished. This can be quite disadvantageous because if a fail occurred at an early stage of the process and it was identified at the testing stage, the whole project would be made from scratch even after all of the work had been finished. One more specific feature of the Waterfall Method is the constant use of documentation. This is absolutely required because the customer is not involved in the software development process. This means that the development team should obtain all necessary information from about the customer’s needs before the work on a project starts. Hence why all of their further work in based on those documents.

Finally, the customer only sees the product once the project has been finalized. Which is why there is a greater risk that the final product will not meet the customer’s needs/ requirements.

SCRUM on the other hand is not as strict with its structure. The process is made up of several teams which is coordinated by a Scrum Master, who is more of a coach than a leader. The work of a SCRUM team is divided into small charts that are called sprint. After each sprint, the intermediate product is shown to the “Product Owner” would estimate two things; percentage of work completed in Sprint, and how long left until completion of the project. Another thing that is different from the Waterfall Method is that the projects priorities may change after each individual sprint.

As we can see, the difference between the two are significantly large. The Waterfall Methodology is based on the stringent structure pertaining to the workflow, while SCRUM provides a lot of flexibility with a wide array of possible solutions.

**Compare SCRUM to DSDM**

Dynamic Systems Development Method (DSDM) terminology differs with each project iteration is called “Emerging Solution” with DSDM and “Potentially releasable increment” in SCRUM. Another terminology would be SCRUMs “Product Backlog” compared to DSDMs “Prioritized Requirement List”. So even if the terminology is completely different they mean the same thing and produce the same result. DSDM is very scalable between small straightforward solutions or large complex projects. SCRUM is used mainly for the development of software, whereas DSDM has been used for more non-IT solutions. SCRUM is good for reinforcing the strength of the team, while DSDM is excellent for project variables (time, cost, features and quality for example). SCRUM is informative and instructional, it is a structured framework but it can be adapted to fit the needs of the customer.

**Compare SCRUM to Spiral**

Compared to spiral opening, iterative incremental development does not rely too much on risk analysis, reducing the cost of development.