Agile creates base for a company to make decisions leading to better software.

* ***Individuals and Interactions****over processes and tools*
* ***Working Software****over comprehensive documentation*
* ***Customer Collaboration****over contract negotiation*
* ***Responding to Change****over following a plan*
* Tools and processes are important, but it is more important to have competent people working together effectively.
* Good documentation is useful in helping people to understand how the software is built and how to use it, but the main point of development is to create software, not documentation.
* A contract is important but is no substitute for working closely with customers to discover what they need.
* A project plan is important, but it must not be too rigid to accommodate changes in technology or the environment and people's understanding of the problem and its solution.
* Highest priority is to satisfy the customers by early and continuous delivery of valuable software
* Welcome changing requirements, even in late development. Agile processes harness change for the customer’s competitive advantage.
* Delivering working software frequently, from a couple of weeks to couple of months with a preference to the shorter timescale.
* Close, daily cooperation between business people and developers
* Projects are built around motivated individuals, who should be trusted.
* Face-to-face conversation is the best form of communication (co-location)
* Working software is the primary measure of progress
* Agile processes promote sustainable development, able to maintain a constant pace among sponsors, developers and users indefinitely.
* Continuous attention to technical excellence and good design.
* Simplicity—the art of maximizing the amount of work not done—is essential
* Best architectures, requirements, and designs emerge from self-organizing teams
* At regular intervals, the team reflects on how to become more effective, and adjusts accordingly

Making each decision based on the principles and values that the team has decided to follow. It is not acceptable to follow another team’s decision, because other decisions are aimed at meeting other teams’ goals.

In the [waterfall model](https://en.wikipedia.org/wiki/Waterfall_model), there is always a separate testing phase after a build phase; however, in agile software development testing is completed in the same iteration as programming.

focus on empowering people to collaborate and make decisions together quickly and effectively.

As opposed to implementing these best practices piecemeal, agile methodologies have “packaged” various customer, management, and in some cases, engineering practices and principles together in a way that helps guide teams through the process of rapidly planning and delivering working, tested software. Each of the agile methodologies combines both old and new ideas into refinements that are certainly greater than the sums of their parts.

Your principles and values are what make you agile. The way we select our practices determines if our team is agile or not. The same practice can work poorly when it is selected out of wrong reason. A team will become agile through decision-making process.

Make a story to clarify our aims from the angle of customers. “As a … I want to…so that I can…” This way we prevent implementation that can bog the team down.

When we have two things that depend on each other, we have a deadlock. Actual usable business value is what a team should focus on, even it is only on a regular basis. In a word, heading directly into where the customers want to be is our goal.

**Scene 1**

A: As a teacher, I want the students to make a presentation fully understand the ingenious ways to manage a project, in order to improve students’ understanding about group work efficiency.

B: Hi Mr. Neal! I think we can make a drama to help students understand how to improve the efficiency of group working.

C: This is how we apply agile in real life situation. We make a “story” to push our work forward. It shows sometimes we can ignore and change the existing method but focus on the purpose.

**Scene 2**

A: We have 5 days to complete a project.

B: Sure. Chai is assigned to do the first part, and we two finish the rest.

A: Well, I need to work really hard.

first

C: It's the first day, and I think I have enough time to finish the program!

Second

C: Ha! Functions has been Finished! I think I can start to add some documentations.

Third

C: Emmm… I think I should change some details here… and here…

A: … Chai has been working on her part for 3 days…

B: Yes. She spent the first day on the code. But for the other two days she was doing the documentation…

A: Now we have very limited time then…

B: If she could sit down and talk to us about her work, we only need to spend at most a morning to understand what she was doing….

(Then we go and get 3 cups of coffee to stay up for the program.)

**Scene 3**

We feel unsatisfied about the project we finished last time, because we didn’t make a reasonable schedule for our project.

A: This time we will spend some time in planning beforehand!

B&C: Sure, boss!

C finishes the plan and hand it to B.

“ schedule:

design an icloud pet-cat

1. define a class Cat …
2. define a function of feedYourCat …
3. define a function of bathYourCat …
4. define a function of catBattle
5. define a function of killYourCat
6. Observe a cat …

Mimic a cat …

Gif

1. Digitalize a real cat in your program.

A&B: Um…Why does she want to kill the cat…

B: Because she hates cat.

A: Then why the battle function?

B: Because she like battle.

A: But the game is about raising a digital cat…

B: Um…But don’t you think we need to follow the plan? Plan is important! We cannot work without the plan!

And A is very confused.

In this case, they never think about adjusting the plan, because they know how important a plan is. But changing the plan after the discussion among all the team members does not undermine the importance at all. On the contrary, we improve the quality of the project.

A: ***Responding to Change****over following a plan* is important!

B: Chai just gave me her work…

C: What do you think about it?

B(staring at paper): Are you sure? Well…a super long plan…I know the ideas are great and she must spend plenty of time on them…

C: Yes!

B: … … Some of them are unrealistic… Maybe it is because I am stupid.