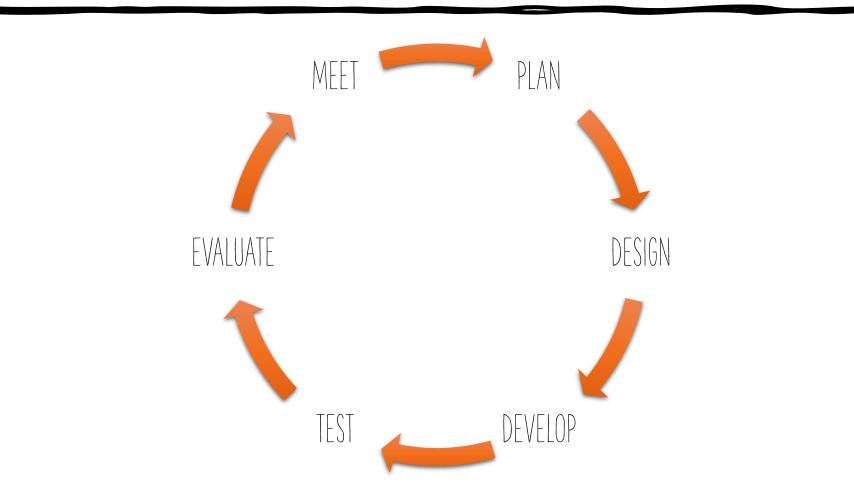# PRACTICES OF AN AGILE DEVELOPER

PRIYANKA R

AGILE SOFTWARE DEVELOPMENT CYCLE

WHAT IS AGILE SOFTWARE DEVELOPMENT?

- Agile is a term used to describe approaches to software development highlighting incremental delivery, team collaboration, continual planning, and continual learning, instead of trying to deliver it all at once near the end.

- Feedback is gathered and implemented continually and in all, it is a much more dynamic process where everyone is working together towards one goal.

BEGINNING AGILITY

- Start with the hardest

- Work for outcome: **Blame doesn't fix bugs**. If you are not making any mistakes, you're probably not trying hard enough.

- Criticize the idea, not people

- Do what's right

FEEDING AGILITY

- Keep up with the change.

- Know when to unlearn.

- Question until you understand.

- Feel the rhythm: Tackle tasks before they bunch up.

- Delivering what users want; Let customers make decision

- Keep it releasable

- Integrate early, integrate often

- Get frequent feedback using demos

AGILE FEEDBACK AND AGILE CODING;

- Use automated unit tests

- Measure real progress; Focus on where you're going.

- Listen to the users(every complaint holds a truth)

- ❖AGILE CODING

- Write code to be clear, not clever

- Communicate in code

AGILE DEBUGGING

- Keep a solutions log: maintain a log of problems and their solutions.

- Treat warnings as errors: Just because your compiler treats warnings lightly doesn't mean you should.

- Report all exceptions: Handle or propagate all exceptions

- Present useful error messages: provide an easy way to find the details of errors

AGILE COLLABORATION

- Schedule regular face time: use Stand up meetings

- Be a mentor: "You gain as you give!"

- Allow people to figure it out.

- Review all code.

- Keep others informed.

THE END?

- Apply these practices yourself and see some personal benefit or you can get your whole team on board and start developing better software faster and more easily.

- THANK YOU !