**Presenter questions:**

How do you determine requirements?

*Watson*: Uses User Story as a tool to gather requirements (As a <user/who> I want <action/what> So that <purpose/why>). The “epics” (big requirements) are divided into small pieces, like in the Agile model. Customer development approach – create the product that the clients need (don’t have a ready standard solution).

*Gullerud*: The development team help gathering the requirements and guide the customer to a solution that will be best for them (opposite approach compared to Watson’s).

How do you communicate within and across groups?

*Watson*: Uses a RACI matrix as a tool to set the responsibilities on the team between Responsible, Accountable, Consulted or just Informed. With roles and responsibilities clear to all, communication within and across groups are easier. He also mentioned Jeff Bezo’s two pizzas approach, for meetings productivity.

*Gullerud*: Maintain small teams. communication will flow within and across teams using Jira, email and calls as the main tools.

How do you move from stage to stage in a project? (What stages do you have?)

*Watson*: Development lifecycles placed on a Kanban board. Stages go from To Do (pool of ideas, similar to a backlog in SCRUM) to Done (Feature accepted), passing through planning, developing and testing stages. They also included a Deploy stage. Verifications are performed in checkpoints between the stages, according to stablished polices. They used Jira as the tool to design the Kanban.

*Gullerud*: Staged gate approach, with checkpoints on every phase / by the end of each flow. They perform verifications (similar to a sprint review in Agile model), with the presence of one or two people who have the customer view. They also use Jira as a tool to follow the flow.

How do you ensure quality as you are developing a project?

*Watson*: They perform reviews by the end of each cycle (similar to the sprint reviews and sprint retrospectives of the Agile model). The reviews are conducted based on a quality assurance spreadsheet (checklist). They also do constant peer review along all development stages.

*Gullerud*: Reviews and retrospectives based on the Agile-SCRUM rituals to guarantee product quality. They also perform audits on the process to guarantee process quality.

How do you test your projects? How do you know it is ready for release?

*Watson*: They build test cases. They structure includes input – expected output – actual output – evaluation. When questioned which criteria they use to make sure that the feature is enough tested (how do you know that the test end?), Watson gave an evasive answer related the efforts in tests to the amount of time they have before the deadline. He mentioned however the importance of the final user participation at the validation of the features.

*Gullerud*: They invest a lot of efforts in unitary tests (it was mentioned hundreds of tests). Each requirement is exhaustive tested. They understand that the test cases should be written before coding but admitted that sometimes they do it after the implementation is done due workforce limitations. It was not clear if they have any automation, or how do they know it’s time to stop.

How is a product maintained after release?

*Watson*: They use Jira as the bug tracking tool. Track the bugs according to the developer who wrote the code. He also mentioned a solution called “Applause”, where developers all over the world can contribute finding and solving bugs.

*Gullerud*: They have a control that triggers when bugs are raised and inform the support team.

What management structure do you have?

*Watson*: Functional structure, with the organization divided in groups with specific tasks or roles. Small teams (6-7 members), highly participative and focused.

*Gullerud*: Matrix structure, where sometimes they have to report to managers that are not their direct superior office.

How do you improve your development process?

*Watson*: They have a model similar to PDCA for continuous improvement, based on Observe, Orient (identify experience improvements), Decide (focus on customer value) and Act. They also have a Sprint Retrospective feedback ritual to evaluate what went wrong and what can be improved.

*Gullerud*: They perform a lessons learned meeting by the end of each project.