AR AESTHETIC RECORD

Lenovo Practices

Lenovo Project Management

Jira is used for management of development cycles.

Jira development statuses typically follow a flow methodology of: “To Do” -> “In Development” -> “In UAT” -> “Done” with an optional *completion* type of status labelled “Cancelled” for tracking development tickets which turned out to be unnecessary.

Software development is premised upon the Agile methodology.

Lenovo Sprint Practices

Sprint cycles last 3 weeks.

Sprints have 4 meetings associated with them: Grooming, Planning, Demonstration and Retrospective.

Grooming Meeting (typically lasts a minimum of 1 hour, maximum of 2): refine backlog “stories” to meet the defined DOR status; estimate either numerical complexity or time through group consensus of development stories (using either Fibonacci sequence for complexity, or self-estimations for time).

Planning Meeting (typically lasts a minimum of 1 hour, maximum of 2): assign stories to pertinent capable developers for the forthcoming sprint, review and balance workloads (if necessary), analyze and report project development velocity.

Demonstration Meeting (typically lasts less than 30 minutes): period for demonstration to business side representation, if so desired, of development completed during the prior sprint. Sprint Demos were rarely called at Lenovo for anything other than completed UI development.

Retrospective Meeting (typically lasts approximately 2 hours): opportunity for lead developer to gather feedback from managed developers regarding practices or behaviors performed during the sprint which the team would like to either continue doing, stop doing or start doing. An open floor discussion focused on how to work together better.

Standup: Sprint Standups are handled through Slack/Teams integrated chat applications, with standup prompts typically occurring on a timed, daily basis. Standups were optional at Lenovo, and typically consisted of only 2 questions: “What are you working on today?” and “Are you blocked by anything?”

Velocity Determinations: development projections typically required 2-3 sprint cycles to accurately estimate approximate completion timelines.

DOR (Definition of Ready): the qualifying criteria that determines whether or not a ticket / story / issue is groomed well enough to be planned into a forthcoming sprint cycle. TBD.

DOD (Definition of Done): the qualifying criteria that determines whether or not a “completed” ticket / story / issue meets *both* the AC’s declared in the source, and the communally agreed upon standards for development to be done. TBD.

AC (Acceptance Criteria): the functional, or technical, criteria outlined in a source ticket / story / issue for a ticket to meet the author’s status of “complete.”

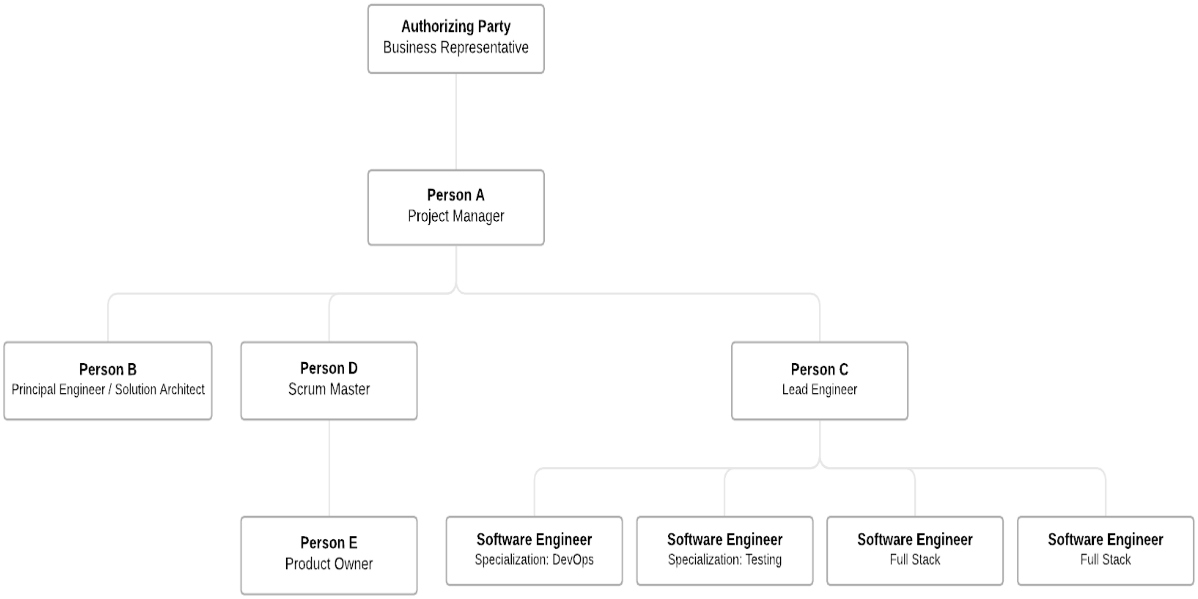
Lenovo Business Practices

Scheduling of meetings, barring emergencies, were typically restricted to 1 specific day each week, for maintaining employee-centric work-life balances.

Scheduled meetings included in the invitation: required attendees, optional attendees, meeting title, meeting time and frequency of occurrence, meeting agenda and the meeting overview.

Lenovo Project Staffing Practices

Development of projects typically followed Greenfield Staffing Practices: a single Project Manager (intermediary between Business and Project Staff), a single Product Owner (intermediary between PM and Engineering Team), a single Scrum Master trained in Agile, a single Principal Engineer (or Solution Architect), a single Lead Software Engineer, at least one Software Engineer specializing in DevOps, at least one Software Engineer specializing in Testing and Test Frameworks, and a business-needs determined amount of full-stack developers. Typical project hierarchy followed the below minimum headcount format:



Lenovo Coding Standards

Modularization: directories are structured by *entity* and not function.

Classes: should be titled primarily after the entity, or data object, the class interacts with; treated as a proper noun regarding capitalization; does not use snake casing or camel casing.

Variables: purposeful, meaningful, descriptive names representative of the function the data variable serves; uses camel casing.

Methods: purposeful, meaningful, descriptive name in a verb fashion of the function the method performs; uses camel casing.

Data Storage (SQL & NoSQL): purposeful, meaningful, descriptive name of entity for which data is retained; uses snake casing.

Default Data Type (Storage & Transmission): String, unless functionality needs of application dictate otherwise.

GitFlow & Branching Strategy: master branch locked from –force; flow methodology: branches to be merged are titled by either feature or bug descriptor, for tracking purposes; branches to be merged include environment descriptor, for tracking purposes; branches to be merged include hyphenated ticket number. Example full Git branch titling for feature development to be merged into the development environment: "feature/DEV-196" and example for a bug fix to be merged into a test environment: "bug/TEST-207" (methodology primarily employed for reporting purposes).

IDE: JetBrains products (paid) for Relational Database Management (DataGrip) and DevOps product development (IntelliJ); MongoDB's Compass (free) for Document Database Management; Visual Studio Code (free) for Node development and React development.