

Portfolio Team: Team Lannister (consists of 3 Portfolio managers, 2 SVPs, and 4 VPs)



Background: After the April Release, the Lannister's conducted a Release Review, and subsequent retrospective to identify how well the Enterprise is meeting delivery milestones. The following observations were made:

- 3 out of 12 Programs are not hitting release goals
- 22 of 52 Epics (Projects) are behind on delivery
- Sprint Team capacities seem to be full; 12 out of 50 teams are over 125% utilization (this means teams are reporting 50+ hours per team member, per week)
- Business Partners do not seem happy; last few conference calls have been contentious
- MVPs (Minimal Viable Product) are consistently delayed every release
- Lannister's are questioning whether all teams have the same understanding of the "definition of done"
- Team Net Promoter Scores have gradually trended down the last 3 releases
- Product Net Promoter Scores have trended even
- 2 of the last 3 releases have required rollbacks mid release – seems like a sprint team or release team deficiency.
- Velocity Delivery has been under 80% for at least 10 teams
- Dependencies on vendors haven't gone well

Discuss these observations, and determine which observation should be escalated to the scaled retrospective. You may ONLY select one.

Categorize the observation, and try to determine how to measure, or make the observation actionable for Executives.

Release Management Team: Team Alcatraz (consists of 5 Release Managers)



Background: After the April Release, the Alcatrazers conducted a Release Review, and subsequent retrospective to identify how well the Enterprise is executing on production releases. The following observations were made:

- There were 4 separate builds after the code freeze date;
  - 1 Build broke the QA Environment after regression had kicked off (regression had to be re-started)
- There was a config change managed by a business team that created a severe integration fail.
- Not all Features passed performance test baselines as there was not enough time to finish performance tests (business assumed risk, and requested all Features be pushed to production anyway)
- There was 1 showstopper defect identified by compliance during prod validation that resulted in an initiated rollback for remediation
- Users reported site latency after rollback requiring emergency deployment and fix.
- The Alcatrazers feel as if the Lannisters are pushing too hard on teams
- Compliance finding a showstopper defect in prod validation, actually revealed the same defect existed in lower environments and had passed initial tests, and was accepted as a risk.
- Sprint Teams are not respecting code freeze dates, and putting releases at risk, due to compressed time to run regression tests.

Discuss these observations, and determine which observation should be escalated to the scaled retrospective. You may ONLY select one.

Categorize the observation, and try to determine how to measure, or make the observation actionable for Executives.

Program Team 1: Team Gossamer (Consists of a Program Manager, 3 Project Managers, 1 Program BA, and 2 Business Executives)



Background: After the April Release, the Gossamers conducted a Release Review, and subsequent retrospective to identify how well the Enterprise is delivering on program milestones. The following observations were made:

- The Compliance team discovered 3 showstopper defects after the production deployment, which required a roll back
- Program milestones are 20% behind schedule.
- There is the possibility that full program delivery will eventually be between 1-2 sprints behind schedule.
- There is an impending regulatory compliance deadline that if not delivered could result in fines for the Enterprise.
- Teams are reporting they need support and training for Hadoop and Angular js. Lack of knowledge is contributing to late delivery, and possible quality issues.
- Business teams and Product Owners are reprioritizing new stories a few days before the start of every sprint, for the last 7 sprints in a row.
- Decisions are being made on a string of lengthy emails, and its not always clear to sprint team members which decisions have been made, overruled, or confirmed.
- Compliance wants an audit trail of conversation in anticipation of an audit.
- Tracking scope changes has become difficult and cumbersome.
- Changes to priority are necessary to take advantage of market conditions. As much as the business team would like to follow a plan, the benefits of flexibility are putting the Enterprise at a competitive advantage.

Discuss these observations, and determine which observation should be escalated to the scaled retrospective. You may ONLY select one.

Categorize the observation, and try to determine how to measure, or make the observation actionable for Executives.

Program Team 2: The Pigeons (Consists of 1 Program Manager, 2 Project Managers, 2 Program BSAs, 1 Program BA, and 1 Business Executive)



Background: After the April Release, the Pigeon's conducted a Release Review, and subsequent retrospective to identify how well the Enterprise is delivering on program milestones. The following observations were made:

- One of the mobile teams assigned to a specific epic delivery within the release plan, was pulled midsprint to work on a "Secret" project that required everyone involved to sign a Non-Disclosure Agreement (NDA).
- Current program spend is beyond forecast, and over by 1.3 with 2 releases still left to deliver.
- A vendor commitment was not met, which delayed delivery by an anticipated 2 sprints.
- Most teams are cross functional, except for the mobile and web teams, which have not adopted the leadership structure required to promote cross functional teams or poly-skilling. Leaders do not want to give up their teams, and have agreed they will not evaluate changes until Q3.
- Dependency management has not gone well, especially with the highly distributed nature of the teams.
- Distributed teams are struggling to fix defects within the same day. Fixes are taking between 2-3 days to turnaround to promotion in QA environments.

Discuss these observations, and determine which observation should be escalated to the scaled retrospective. You may ONLY select one.

Categorize the observation, and try to determine how to measure, or make the observation actionable for Executives.

Sprint Team 1: Team Cobra Kai (consists of 4 DEV, 3 QA, 1 BA, 1 Test Automation specialist)



Background: After the April Release, team Cobra Kai conducted a sprint retrospective to identify opportunities within their sphere of control to change, and outside their sphere of control to change. The following observations were made:

- A code check-in broke the build, and rendered the QA environment “down” for over a full day (sprints are 10 days long).
- Teams have no way of receiving any communication about the build breaking.
- Teams must rely on SMs and other tech leads to help organize and communicate issues.
- Compliance did not attend the demo, and was not available for showcases throughout the last sprint.
- The team does not make an effort to burn down time
- The business team made a configuration change without telling the sprint team they checked-in new changes.
- Team morale is low, everyone is working late and on weekends for multiple sprints.
- The Business leaders are asking why the sprint team can’t just work harder

Discuss these observations, and determine which observation should be escalated to the scaled retrospective. You may ONLY select one.

Categorize the observation, and try to determine how to measure, or make the observation actionable for Executives.

Sprint Team 2: Team Scrumbags (consists of 4 DEV, 2 QA)



Background: After the April Release, the Scrumbags conducted a sprint retrospective to identify opportunities within their sphere of control to change, and outside their sphere of control to change. The following observations were made:

- Team met all sprint and release commitments despite environment challenges
- Code deploys really need to be more automated
- Team building event was FUN!
- The Release being rolled back was annoying, but we did manage to get code into production for our users – so....Success!
- Team needs to focus on building our runway. We had some slack time but weren't sure what to pull in for the remainder of the sprint.
- Environments were a disaster that definitely impeded our ability to deliver.

Discuss these observations, and determine which observation should be escalated to the scaled retrospective. You may ONLY select one.

Categorize the observation, and try to determine how to measure, or make the observation actionable for Executives.

### Sprint Team 3: Team Ghengis Khanban (consists of 5 DEV, 2 QA)



Background: After the April Release, team Ghengis Khanban conducted a sprint retrospective to identify opportunities within their sphere of control to change, and outside their sphere of control to change. The following observations were made:

- Builds to QA are taking 10 hrs which is limiting code check-ins
- Dev builds are cloud based, and QA builds are on premise
- Team took on 3 new stories as emergency requests and had to drop 2 other stories from sprint
- Team worked over 2 weekends during the sprint, but team members in Mumbai did not come in or login for support.
- Decision to move to angular was made by the Architecture team without providing any training on how to use it.
- Broken builds cost teams 3 days of testing
- Cycle time on stories actually trended very well, with most stories taking approx. 2 days from start to finish.

Discuss these observations, and determine which observation should be escalated to the scaled retrospective. You may ONLY select one.

Categorize the observation, and try to determine how to measure, or make the observation actionable for Executives.

Sprint Team 4: Team Falcor (consists of 9 DEV, 2 QA)



Background: After the April Release, team Falcor conducted a sprint retrospective to identify opportunities within their sphere of control to change, and outside their sphere of control to change. The following observations were made:

- Environments were down for 2 solid days
- Compliance was not engaged because they were too busy to attend showcases and demos
- Team composition is not balanced. And its hurting our ability to deliver
- Team was asked to take on 2 new prod support stories that were not quick fixes.
- Prod support fixes came 2 days before code freeze and forced us to check in code after code freeze
- PO was on vacation for 2-weeks with no one who could help make decisions on what stories to drop.