Audience

- Agile (Scrum) teams
  - Pigs (Testers, Developers etc)
  - Chickens (Scrum Master, Product Owners, other stakeholders)

Objective

This artifact offers a template and guidelines **which can be used by agile teams to commit to stories for a sprint** using the proven top-down “Poker Planning” technique (*for both Developers and Testing*) and the capacity in hand.

Poker planning is a ‘consensus based estimation’ technique which benefits from multiple viewpoints, instant peer review of estimates, team ownership and encourages openness & broader understanding. It strongly discourages bullying by so-called-experts, any biasness and involves the entire team in the estimation process.

1. Poker Planning (Recap):
Planning Poker is a consensus-based approach to agile estimating. To start an estimating session, the product owner or customer reads a user story or describes a feature to the estimators, who should include everyone on the team. Each estimator is holding a deck of cards with values like 0, 1, 2, 3, 5, 8, 13, 20, 40, and 100, which is the sequence we recommend. The values represent the number of story points, ideal days, or other unit in which the team estimates.

Poker planning involves the members of the team independently developing quick effort estimates and then comparing their estimate, discussing the differences and arriving at a consensus based on the discussions. Their independent assessments are all shared concurrently with the group by holding up cards with the numeric values of their respective assessments. The process relates to the game of poker only in the sense that the participants make their assessments discreetly, concealing them from the other participants until they all show their hands together. It is important to the process that the participants conceal their assessments from each other and all participants reveal together.

2. Poker Planning – Six easy steps
3. **Poker Planning – Ground Rules**

- Only “pigs” can estimate using Poker planning.
- Points establish relative size, not duration.
- Planning Poker cards and are based on a Fibonacci number sequence.
- These points are also referred as Story points: 1, 2, 3, 5, 8, 13, 20, 40, Infinity
- At the end of the first round, one high and one low voter are selected to discuss and justify their vote to the group. If consensus is not reached, go for Round 2
• At the end of the second round, team can take the median value or the pessimist wins 😊

4. Committing to stories: Reusable Generic Template

*Post poker planning exercise, the given generic template is created to help the teams in determining the team’s capacity to commit to certain user stories for the current sprint based on the team’s velocity and availability of effort. You can know the stories you can commit to by just tweaking few parameters.*

**Download the template:**

**Six easy Steps:**

1. Open and Save the template (Read the comments for each field)

2. Enter metadata – like velocity of your team based on past sprints, week per sprint, productive hours per day etc

3. Enter your backlog for the sprint and agreed upon complexity points for each story for Development and Testing (*outcome of Poker Planning exercise*)

4. Look at Projected Effort, Available effort and Total Buffer which is calculated by the template

5. If Total Buffer calculated is beyond +/- 10 % then tweak parameters

• **Zero** % means required and available effort matches and team should be able to deliver on committed stories(*+/ - 10 % deviation is considered acceptable as the estimations are not very accurate and precise*)

• **Negative** % means that we are over-committing and available effort is not enough. In that case you can decide
to shed some user stories that have least priority. Update sheet and see the total buffer again (Keep repeating till buffer comes down)

- **Positive %** means that we can take more user stories / story points.

In that case, we have more bandwidth available and we can possible take more user stories. Add complexity points for new user stories from the backlog and have a relook at the total buffer

- If the above doesn’t help, consider tweaking the resource count by adding/removing resources if it can help.

6. Commit to stories based after tweaking the parameters

Author's Biography

Raj Kamal is a Senior Test consultant with 9 yrs. of experience, specializing in different types of testing techniques, test automation and testability in different domains like Manufacturing, Healthcare and Higher Education. He holds an APICS certification in Supply Chain Management. Familiar with Rational, Mercury & Microsoft testing tools, he has helped teams develop test automation strategies and architectures for such companies as Cognizant Technology Solutions, Oracle Corporation & Microsoft. He also provides training in automated testing architectures and design. He is QAI(CSTE) & ISTQB Certified. He has a master's degree in Computer Applications. He is currently working as a Test Lead at Microsoft, India, Business Intelligence domain. His passion: [http://geektester.blogspot.com/](http://geektester.blogspot.com/)
Past presentations:

- Presented a webcast at Microsoft worldwide customers for [Data quality Testing](http://www.qaiworldwide.org/qai.html)


- Represented Microsoft as a speaker at [Test 2008 International Conference](http://www.qaiworldwide.org/qai.html)

- Automation framework designed is accepted for Presentation and Publishing at [International Conference on Information Technology: New Generations](http://www.qaiworldwide.org/qai.html), Las Vegas & IEEE Digital Library

Publications:

- Published a paper on [Resurrecting the Prodigal Son--Data Quality](http://www.qaiworldwide.org/qai.html) ([Stickyminds](http://www.qaiworldwide.org/qai.html))

- Published a paper on “Adventure of BI/DW Testing” ([Stickyminds](http://www.qaiworldwide.org/qai.html))