ARPG

The Agile Release Planning Game

Extended Version: The lost art of communication

Facilitator’s guide – version 1.3 August 2018

A game where participants run an agile project. Invented by James King ([www.kingsinsight.com](http://www.kingsinsight.com)) and extended by Eduardo Meira Peres (eduardo.peres@pucrs.br).

Contents

[The game 3](#_Toc522875532)

[The story so far 3](#_Toc522875533)

[What needs to be done 3](#_Toc522875534)

[The strategy 3](#_Toc522875535)

[Getting started 3](#_Toc522875536)

[Break the group into teams 3](#_Toc522875537)

[Explain the story so far 3](#_Toc522875538)

[Set up the board 3](#_Toc522875539)

[Explain the sprint goals 3](#_Toc522875540)

[Explain velocity and story points 4](#_Toc522875541)

[Prepare an initial release plan 4](#_Toc522875542)

[Running the first iteration 4](#_Toc522875543)

[Confirm the plan for the iteration 4](#_Toc522875544)

[Determine the actual velocity 4](#_Toc522875545)

[Impacts of the actual velocity 4](#_Toc522875546)

[Update the status reporting sheet 5](#_Toc522875547)

[Running remaining iterations 5](#_Toc522875548)

[Embrace Change 5](#_Toc522875549)

[Ending the game 5](#_Toc522875550)

[Debrief 5](#_Toc522875551)

[Copyright 5](#_Toc522875552)

# The game

The game is designed to help participants understand the concepts involved in release and iteration planning.

Participants must plan their work across a series of iterations, which we will call sprints, and then adapt their plans as the project evolves.

In order to be successful, participants must meet critical deadlines, minimise defects, monitor their velocity and replan as they gain further insight into the evolving project.

## The story so far

“NE 1 L8 – CU L8R”. The process of language simplification that was accelerated by the advent of the internet has reached its climax. In this scenario, the people of a certain city have completely forgotten the meaning of the letters, and therefore can’t communicate through written or spoken language.

## What needs to be done

In order to recover the ability to communicate in this community each participant was designated a team whose mission was to rebuild the maximum number of letters of the alphabet as possible. This will allow the gradual return of the lost art of communication.

At the end, the mission should reach the following criteria:

- It should be possible to construct the phrase: “It’s difficult to put it into words”. Notice that this phrase doesn’t contain the vowels A and E. This is important for the game.

## The strategy

The team will recreate some of the English language over 4 (four) cycles of short duration called sprints.

The amount of story points the team can complete each sprint (or iteration) is known as the team’s velocity.

The team’s velocity is estimated to be approximately 25 story points per sprint. This number is expected to change over time as the mission evolves.

The team will use this velocity to by letters of the alphabet so that they can recreate the English language.

# Getting started

The game can be introduced by first presenting an explanation of the concepts of release planning or it can be run prior to the explanation of the theory.

## Break the group into teams

To introduce the game, break the group into teams. Each team will need:

1. Team members. The best team size is between 2 and 8 people.
2. 1 copy of the iteration instructions
3. You can distribute a set of story cards or you can ask people to create post-it notes with stories and their cost as they go
4. A pair of dice.
5. There is a status report in the iteration instructions, or you can print this separately.

## Explain the story so far

Explain the background story to the group.

## Set up the board

Have each team lay the status out in front of them.

## Explain the sprint goals

At each sprint the team should complete stories to meet the sprint goal.

Each sprint must have the following goals:

|  |  |
| --- | --- |
| Sprint | Goal |
| Sprint 1 | Construction of a minimum of two words with two or more letters. |
| Sprint 2 | Construction of a minimum of two words with four or more letters. |
| Sprint 3 | Construction of two complete grammatically correct sentences. |
| Sprint 4 | Construction of the sentence which is the object of this mission |

Players are not permitted to use a Word more than once in the game but they can use letters as often as they like once the letter has been constructed.

## Explain velocity and story points

Each letter is represented by a story card with the letter and a story point rating on it.

* The story point rating is the cost to complete the story.
* The velocity is the number of story points the team can expend to complete stories each sprint.

## Prepare an initial release plan

Give each team 10 minutes to:

* Review the project rules and objectives; and
* Come up with a release plan.

Note, at the end of the game the team will have executed 4 sprints. They can choose to allocate stories to all 4 sprints, but initially they are only required to allocate stories at least to the first sprint.

# Running the first iteration

## Confirm the plan for the iteration

The team confirms the stories to be attempted before the start of the sprint.

In the subsequent iterations, the team needs to decide whether to reduce outstanding defects and technical debts. Fixing a defect requires 1 story point per defect. Each technical debt also requires 1 story point per technical debt.

However, in the first iteration there will be neither defects to fix nor technical debts to solve. So the team should place the “zero” defects and technical debt at start on the status report.

## Determine the actual velocity

We predict that the velocity for the first iteration will be 25. This is a good indicator of how much work the team will get done, but it is not a perfect prediction.

So we roll dice to represent the impact of unpredicted occurrences, as well as the ongoing improvements made by the team.

Have each team roll two dice to evaluate the actual velocity asbelow**.**

The first die represents the gains of improving the team performance and the result must be added to the planned velocity.

The second die represents distractions and impediments that reduce the performance and must then be subtracted from the planned velocity.

So, complete the status report with the following formula:

Actual velocity = Expected Velocity + die (first) – die (second)

## Impacts of the actual velocity

If the actual velocity is greater than planned, the team can insert new items into the sprint (new stories, correction of defects or reduction of the technical debt) until the total is complete.

But, if the actual velocity is unfortunately less than planned, the team should remove as many items as necessary. The items to be removed are those of less priority and should be postponed until the following sprint.

This actual velocity will also be the expected velocity for the next sprint.

Complete the status report with the following formula for the next sprint:

Expected velocity (sprint x+1) = actual velocity (sprint x)

## Update the status reporting sheet

The team update the status report to reflect what has occurred during the iteration.

# Running remaining iterations

If the team has reached the goal of the sprint they now move onto the next sprint.

The process is essentially the same as running the first sprint, except that defects and technical debt may now exist.

# Embrace Change

At the end of sprint 2 or 3 communicate to the team that the goal of the project needs to be changed to construct the phrase “It’s **really** difficult to put it into words” (adding the letters E and A to the scope).

Thus the team will experience how changes are incorporated into the project

# Ending the game

For a single team, the game ends after the 4th sprint.

If more than one team is playing then other teams can continue, or the game can end for the whole group if preferred.

# Debrief

Once the teams have completed the game, debrief as a group.

One way to do this is to hold a retrospective:

* Have each team answer 3 questions:
  + What worked well?
  + What caused problems?
  + What can we learn from this?
* If there are multiple teams, then have the teams discuss their findings as a group.

# Copyright

This work (The Agile Release Planning Game and the materials associated with it) is licensed under the Creative Commons Attribution 3.0 Unported License.

You may reproduce and use the game subject to the terms of that license, subject to attributing the game’s creation to James King and by including my website or contact details on the game board.

To view a copy of this license, visit [http://creativecommons.org/licenses/by/3.0/](http://creativecommons.org/licenses/by/3.0/%20) or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.

The game was updated slightly in 2018 to remove the concepts of defects and technical debt. The scoring sheets were also merged into one document.

However this version of the game is essentially the creation of Eduardo Peres.

It would also be appreciated if you could send feedback to [Eduardo.peres@pucrs.br](mailto:Eduardo.peres@pucrs.br) if you enjoy the game.