

Let's try...



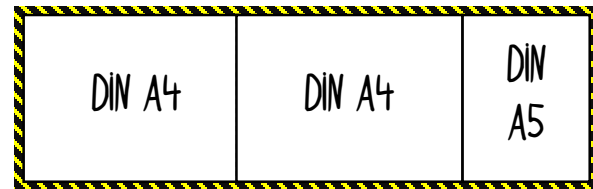


- The goal is to deliver approx. 100 Cards on the table in ascending order in the working area.
Doublettes are not allowed.
- We will play several Sprints (a 60sec), then debrief
- We will play two different rounds with a different rules of the game



Setup

- Playing with „The Mind“ cards, you need a set for 1 groups plus another set to add double cards:
 - Add 20 cards taken from your additional deck to your playing deck
- Shuffle the deck
- Place protocol-sheet on each table.
- Setup Working area



- Group of at least 4 active player, better 6 active player, rest observer



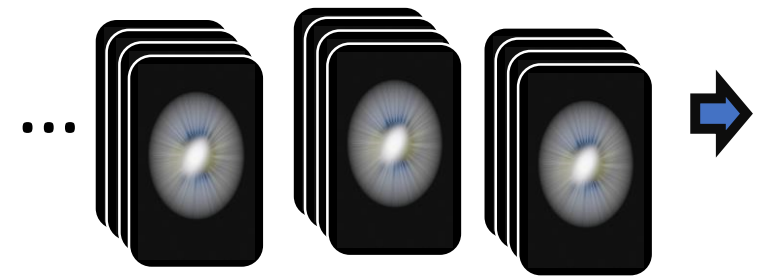
Facilitator Hints

- Setup each Table (One Deck of Cards, Protocol Sheet, Refactoring Cards, CR-Cards)
- Play a demonstration sprint with all participants explaining: Goal, Roles, Rules, CRs, Bugs, Timekeeping and the fact, that we will play 2 different rounds. One round ends after 6 Sprints or when all cards are played.
- Before the 1st sprint starts give the teams the order to discuss a placing strategy ~30sec
- Then start the first sprint
 - For each group there are different modes possible
 - No refactoring
 - Refactoring
 - Start with different modes each group and switch between Refactoring / No refactoring after the 1st round
- Debrief and compare the 2 rounds

Roles



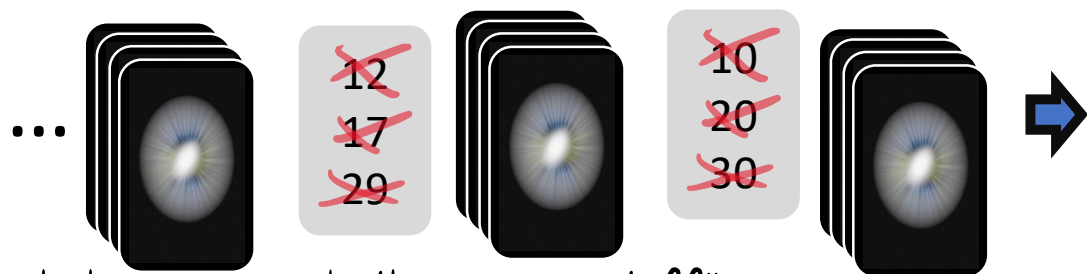
- 3 **Developer**: Take cards on his hands. Play cards into working area.
- 1 **Quality Assurance (QA)**:
 - Has to notify bugs
 - Can talk anytime and point everywhere
 - Can lift cards to check the number but must not move **any** card
- 1 **Product Owner**:
 - Arrange Features (4 Cards on a stack per feature, for the developer-team)
 - Count & Protocols # of features pulled by the dev-team each sprint
 - You can treat all these features as committed in the planning, the work from the PO here is just helping to keep the game table clear of too much cards
- 1 **Timekeeper**:
 - Checks time for Sprint 60sec, Doublecheck the number of features pulled by the development team
 - Gives a hint 15 sec before Sprint ends
 - Takes time after sprint, how long it takes to check for bugs
- n **Observer**:
 - Have a look a speed over time, decisions, discussions, cheating, strategies etc.





PO: How to play

- If you have Refactoring Budget available you hand the amount of budget you like, over to the development team
- You will provide your team with features (4 Cards on a stack=1 Feature), CR's
- You will put all these things in a queue where the team can pull it.

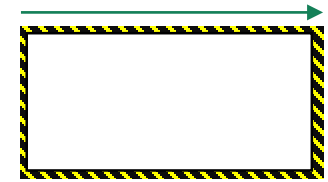


- The order (priority) in which you provide them your „stuff“ is mandatory.
- Count and protocol the number of features, the development team pulls from your queue
- CR's count as one feature. Of course it is only done when the all the conditions are met.
 - Start providing CR's from sprint 3.
 - 2 or 3 CR's per Sprint



Developer: How to play

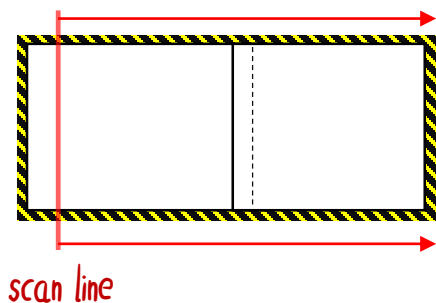
- Each feature card you get, you have to keep in your hand and play from hand into the gaming area card by card
- All cards in the gaming area has to be in an ascending order (from left to right)
See page „Bugs“ for further details
- Normally for the Developers it is not allowed to touch or move cards placed on the table (=refactoring) beforehand. Only your own last played card is movable. (Perhaps if QA gives a hint)
- You have to play your feature first (4 cards) before you take another feature





Gameplay each sprint

- If Sprint (60 sec) is over NO card must be played/moved
- After each sprint:
 - Count the fetures not played (on the developers hand) subtract them from features pulled by team
After counting the cards on each developers hand go back to the PO
 - Timekeeper stop the time for the next activity. QA figure out how many bugs are in the working area using the scan line method. If necessary you can lift cards to check the card number but never move any card!
 - Protocol the # of bugs and the time it consumed to check.
What is a bug: See bugs-rule.



Bugs



- Cards the frontiers of the working area
- Doublettes: Cards with the same number, Developers has to play those cards completely outside the area
- Cards not in the right ascending order (Scan the working area with a vertical "scan line" from left to right. Every time you found a card with a lower card value then the last scanned card, it is a bug. Every card will be scanned only once.



- 34 ok because reached by the scan line first
- 11 Bug because lower than 56
- 11 a second bug because out of area
- 76 ok, but...
- 2nd 76 is a bug because it is a doublette

- Every not fulfilled change request for this sprint (Change request ar only valid for that sprint, not for a later one)



Change request

- From a later sprint the PO places some change request cards between the stack of features
- Each card is valid for this specific sprint. This means all cards on the board that has the requested value to change, has to be removed from the board. Place them outside the working area on the table
- For a change request for the developers it is allowed to touch (but not move) the other older cards and remove the requested cards (without using refactoring)
- Every new card with the mentioned value coming this sprint has also be removed/placed outside the area.
- At the end of this sprint every card in the gaming area with a mentioned value is counted as one bug.
- After the sprint you remove the change request card from the table, it is no longer in effect, the removed cards stay out of the area.

Refactoring mode: Feature, Feature, Feature! No Refactoring



In this group/round:

- No refactoring is allowed, we have to produce features, no time to clean up.
- Bugs stay in this constellation and will be counted every sprint again

Refactoring mode: Unlimited refactoring



In this group/round:

- For the Developers it is allowed and suggested to touch cards placed on the table beforehand anytime.
- Refactoring can of course be used to remove bugs identified at the end of the sprint



Sprint	# Features			Time for bug search	# Bugs
	pulled	Not done	total		
1					
2					
3					
4					
5					
6					