

# The Agile Modules CSC3045 & CSC3052

## Story Estimation

**School of Electronics, Electrical Engineering &  
Computer Science**

**Queen's University, Belfast**

**Dr Darryl Stewart**

# DILBERT

I PUT TOGETHER A  
TIME LINE FOR YOUR  
PROJECT



U.S. Address: SCOTTADAVIS@GMAIL.COM

I STARTED BY REASONING  
THAT ANYTHING I DON'T  
UNDERSTAND IS EASY  
TO DO.



PHASE ONE: DESIGN A  
CLIENT-SERVER ARCHITEC-  
TURE FOR OUR WORLD-  
WIDE OPERATIONS  
TIME: SIX MINUTES



© 1998 United Feature Syndicate, Inc.

# Story Estimation

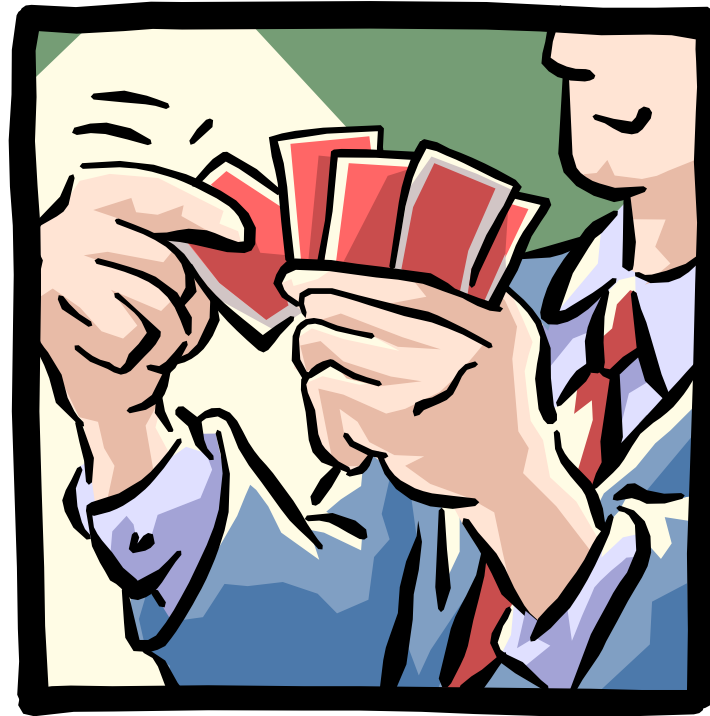
---

- ▶ At the start of a project the initial set of user stories need to be estimated
- ▶ Estimates are expressed in story 'points'
- ▶ Some practitioners say not to equate story points to a measure of time
- ▶ Others say think of them in terms of 'ideal programming days'
- ▶ The estimates should be *relative* as opposed to *absolute*

# How are the estimates made?

---

- ▶ The team should make these estimates together using all expertise available
- ▶ This can be done using a process called **Planning Poker**



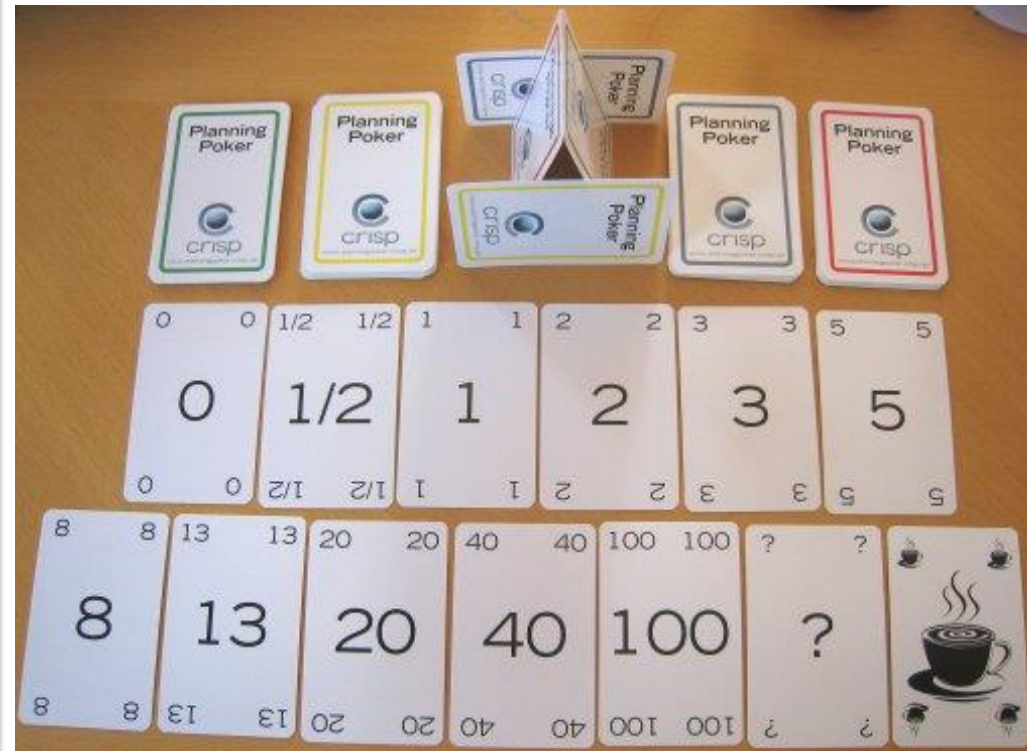
# Planning Poker

- ▶ Participants include **all** of the developers in the team
- ▶ Each estimator is given a deck of planning cards
- ▶ Each card has a valid estimation value on it
- ▶ Some practitioners suggest that these values should follow the Fibonacci-like sequence below:

0, 1/2, 1, 2, 3, 5, 8, 13, 20, 40, 100

- ▶ Speeds up estimation because only a few choices
- ▶ Avoids false sense of accurate estimates

- ▶ Indicates the stories that need to be split up (>20 points)



# Steps in Planning Poker

---

## STEP I

- ▶ Someone reads out the next user story to be estimated
- ▶ The Product Owner answers any questions that the estimators might have about it
  - ▶ Although these discussions should not go on for too long
- ▶ The estimates obtained are **not** considered definite and final so it is not worthwhile to invest too much time trying to make them completely accurate. It is more important that the relative estimates are good

# Steps in Planning Poker

## STEP 2

- ▶ Each estimator privately selects a card representing his/her estimate
- ▶ **Cards are not shown until each person has made a selection**
- ▶ At that time **all cards are simultaneously turned over** and shown to the rest of the team





# Planning Poker

---

## STEP 3

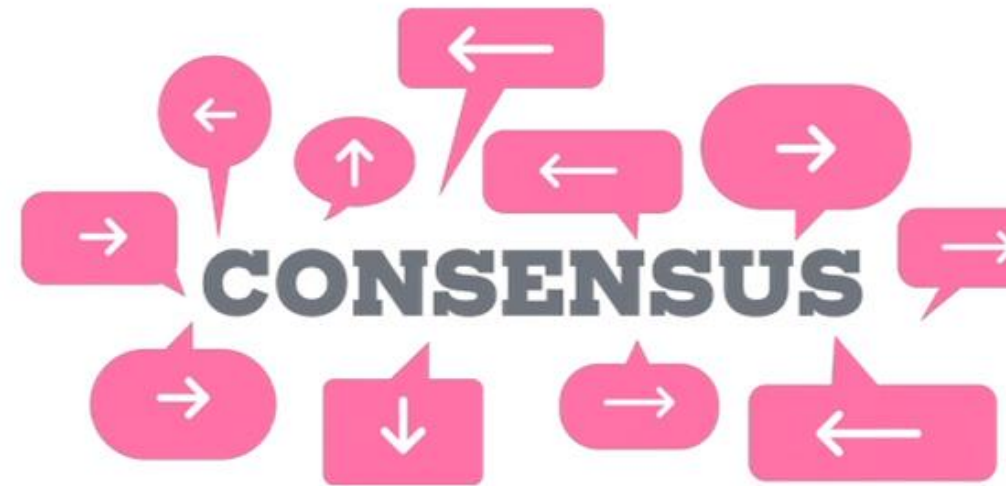
- ▶ It is likely that the estimates will not be the same
  - ▶ This is good!
- ▶ The high and low estimators for this story should explain their estimates
- ▶ This should **not** be about **arguing your case**
- ▶ This is just about learning what everyone was thinking
- ▶ A low estimator may not see some complexity in a story that the higher estimators can see
- ▶ Or alternatively the low estimators might see a straightforward solution to something that the high estimators cannot see
- ▶ The discussion should take **no more than about 2 minutes** before a second round of cards is performed in the same way as before



# Planning Poker

---

- ▶ In many cases this will be enough for the team to come to a consensus on an estimate
- ▶ However more rounds may be necessary
- ▶ If the team has the following estimates for a story : 5, 5, 5, 3, 5, 5
- ▶ Then the low estimator will be asked if they would be OK with a 5
  - ▶ Remember this process is not about precision – it is about *reasonableness*



# Planning Poker

---

- ▶ When beginning for the first time:
  - ▶ Try to select what everyone feels is the smallest/least time consuming story on the Backlog
  - ▶ How many story points is that story?
    - ▶ Perhaps it will be a 1 or 2 point story?
  - ▶ From then on you can **use this as a basis upon which to compare all others**

# Planning Poker

## Considerations:

- ▶ If User stories are **not completely independent** then you may need to provide more than one estimate and indicate what is assumed in each
- ▶ Each team will perhaps have different values for each user story. That's fine but there will probably be similar trends
- ▶ If an estimate is too high  $\geq 20$  then:
  - ▶ Is it because you don't understand the User Story?, or
    - ▶ Ask more questions until you do understand then refine your estimates
  - ▶ Is it because you do understand but it is actually **big** or **complex**?
    - ▶ **Complex**: Have a product backlog item which involves doing research for this
    - ▶ **Big**: Break up into smaller sub stories

# Take home messages

---

- ▶ Everyone in team estimates everything
- ▶ Groups past experience used to get best estimate
- ▶ Poker process forces everyone to estimate honestly
- ▶ Not a competitive process – there is no winner!
- ▶ Once you know what a “one pointer” looks like use this as the base against which all other values are estimated
- ▶ All scores are relative not absolute
- ▶ Values are not precise
- ▶ Estimation tends to improve with experience

---

Try this now with your draft backlog items

