

Expert vs Algorithmic Estimation

2018-03-23

Software Effort Estimation

Effort estimation is an activity that is a part of software planning. Traditionally effort estimation was done by the project manager, but nowadays the whole development team participates. Uncertainties make effort estimation difficult in the early stages of development, but the estimate becomes more accurate as the project advances. The effort is usually underestimated compared to the real effort measured at the end of the project. Effort estimation plays several roles in software development, for example it is used to calculate the estimated cost and budget of the project's activities and communicate with customers and developers that how much time the project will take (Sommerville 2010, chap. 23).

The next sections discuss about the two models that can be used for effort estimation: **Experience based estimation** and **algorithmic estimation**, and then makes a comparison between the models.

Experience Based Estimation

Experience based estimation is based on the previous experience of the project's developers (experts) and it will lead to informed judgement of the effort. The estimation activity should involve a group (development team) and each of the member should explain their estimates. Involving a group reveals factors to others that they may not have considered helping to iterate towards agreed group estimate (Sommerville 2010, chap. 23). One such approach that involves a group is **planning poker** (mountaingoatsoftware) ...

The difficulties in experience based estimation are estimating the productivity of the developers which de-

pends on attributes such as the developers skill level, their familiarity of the problem and discipline. The lack of commonalities compared to previous projects can also make the estimation process difficult.

Algorithmic Estimation

- formulaic approach to effort estimation based on estimates of product attributes (size,...) and process characteristics (staff experience,...)
- models often based on simple formula

$$\text{effort} = A \cdot \text{size}^B \cdot M,$$

where

- A is constant factor depending on the local organizational practices and the software type
- B is the value of the exponent, usually between 1.0 and 1.5, the effort does not usually increase linearly with the project size (increase complexity -> communication overhead, ...)
- M is multiplier made by combining process, product and development attributes (dependability requirements, experience of the team)
- size may be assessment of code size or functionality estimate
- Shortcomings: Size is hard to estimate, estimates of B and M are subjective, models are difficult and hard to use, models require historical data for accurately estimating parameters, if using published values it is hard to know if they relate to their organizations values
- **COCOMO 2**

Comparison

Bibliography

mountaingoatsoftware, *Planning poker estimating in detail* [online]. Available from: <https://store.mountaingoatsoftware.com/pages/planning-poker-in-detail> [Accessed 23 Mar 2018].

Sommerville, I., 2010. *Software Engineering: 9th Edition*. 9th ed. Addison-Wesley Publishing Company.