Conference Paper

Estimating Software Development Costs for Agile Projects

Omar Mahmoud
Blaze Smallwood

New Orleans, LA
June, 2013

Table Of Contents

What is “Agile” software development?

- What is “Agile” Software Development?
  - A software development method based on iterative and incremental development, where requirements and solutions evolve through collaboration between self-organizing, cross-functional teams
  - Promotes adaptive planning, evolutionary development and delivery, a time-boxed iterative approach, and encourages rapid and flexible response to change
- Agile Principles
  - Customer satisfaction through early and continuous delivery of valuable software
  - Welcoming changing requirements, even late in development
  - Deliver working software frequently
  - Working software is the primary measure of progress

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

- **Individuals and interactions** over processes and tools
- **Working software** over comprehensive documentation
- **Customer collaboration** over contract negotiation
- **Responding to change** over following a plan

**Manifesto for Agile Software Development © 2001*

Traditional vs. Agile process overview

Traditional (Waterfall)

Sequential activity of one team

- Plan all of the requirements
- Design all of the requirements
- Develop all of the requirements
- Test all of the requirements
- Deploy all of the requirements

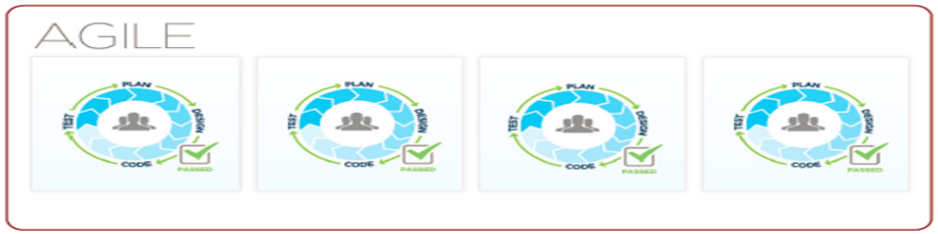
Users will receive end product once ALL requirements have been fully designed, developed, and tested

Agile

Iterative approach where constant user interaction is preferred and highest priority items are completed first

- Determine architectural requirements
- Take each Iteration:
  - Design it, Develop it, Test it, Deploy it
- Each requirement can be designed, developed, and tested simultaneously along with other requirements

Agile doesn't change the end product, only the way projects are scoped, managed and executed



Estimating Agile projects requires familiarity with a new set of terms and metrics

- **User Stories**: A high-level definition of a requirement, containing just enough information so that the developers can produce a reasonable estimate of the effort to implement
- **Story Points**: Quantification of a User story's scope or effort, a relative measure of complexity
- **Sprint / Iteration / Release**: *Sprint* - Fixed time-box in which development occurs (usually 2 - 4 weeks); *Iteration* - Minor subset of requirements designed to be released to the user community; *Release* - Multiple Iterations that fulfill a major subset of user requirements
- **Velocity**: Performance / productivity measure that indicates progress toward capability delivery (i.e., story points completed per sprint)
- **Project / Sprint Backlog**: A prioritized database that summarizes the User Stories / Requirements yet to be complete for the entire project
- **Burndown**: The concept, often shown as a graph over time, of working off or "earning" story points toward iteration or delivery completion

Table Of Contents

As with any software project, the first step in estimating an Agile project is defining its scope and size

- Step 1: Defining requirements in terms of user stories
- Step 2: Sizing requirements in terms of complexity vice physical size (i.e. story points vice SLOC)
- Step 3: Prioritize user stories (typically done in conjunction with product owner)
  - Top priority stories are executed first
- Step 4: Allocate user stories to iterations evenly by estimated story points
- Step 5: Layout sprint durations help determine schedule of iterations and releases
  - Sprint duration is typically two weeks or a month

Banking System - User Story	Story Point	Priority	Sprint	Iteration	Release
"As a user, I want a withdrawal feature so that I can withdraw money from my account in $20 increments"	4	1	1	1	1
"As a user, I want a deposit feature that accepts cash and check deposits so that I can deposit money into my account"	6	1	2	1	1
"As a user, I want to transfer money from one account to another so that I can complete the transfer and see the new balances in the relevant accounts"	8	2	1	2	1
. . .	. . .	. . .	. . .	. . .	. . .
"As a user, I want to be able to check my account balance so that I am aware of how much liquid cash I have available"	2	3	8	3	4
Cumulative Total:	1,728		96		

In conjunction with size and scope, a baseline cost estimate is determined using an initial release schedule and team size

- Initial cost estimate based on this schedule and a relatively constant SW development team size

Banking System - User Story	Story Point	Priority	Sprint	Iteration	Release
"As a user, I want a withdrawal feature so that I can withdraw money from my account in $20 increments"	4	1	1	1	1
"As a user, I want a deposit feature that accepts cash and check deposits so that I can deposit money into my account"	6	1	2	1	1
"As a user, I want to transfer money from one account to another so that I can complete the transfer and see the new balances in the relevant accounts"	8	2	1	2	1
.	.	.	.	.	.
"As a user, I want to be able to check my account balance so that I am aware of how much liquid cash I have available"	2	3	8	3	4
Cumulative Total:	**1,728**		**96**		

Planned Velocity → **18 story points / sprint**

Cost / Story Point: **$100** (CBY13)
Note: Can be derived from analogous programs, industry standards, etc…but may be a challenge to obtain

PMO SW Development Schedule Assumptions by Release

Release	Schedule Outputs			
Software Release by Iteration	Start Month of Dev Effort	End Month of Dev Effort	First FY of Develop-ment	Last FY of Develop-ment
R1-I1	May 13	Aug 13	FY2013	FY2013
R1-I2	Sep 13	Dec 13	FY2013	FY2014
.	.	.	.	.
.	.	.	.	.
R4-I3	Sep 17	Dec 17	FY2017	FY2018

Time-Phased Cost Estimate by Release (CBY FY2013 $K)

Software Releases	FY2013	FY2014	FY2015	FY2016	FY2017	FY2018	Total
Release 1 - Iteration 1	$ 842	$ -	$ -	$ -	$ -	$ -	$ 842
Release 1 - Iteration 2	$ 394	$ 394	$ -	$ -	$ -	$ -	$ 788
Release 1 - Iteration 3	$ -	$ 778	$ -	$ -	$ -	$ -	$ 778
Release 2 - Iteration 1	$ -	$ 756	$ -	$ -	$ -	$ -	$ 756
.	.	.	.	.	.	.	.
Release 4 - Iteration 1	$ -	$ -	$ -	$ -	$ 650	$ -	$ 650
Release 4 - Iteration 2	$ -	$ -	$ -	$ -	$ 469	$ 469	$ 938
Release 4 - Iteration 3	$ -	$ -	$ -	$ -	$ -	$ 629	$ 629

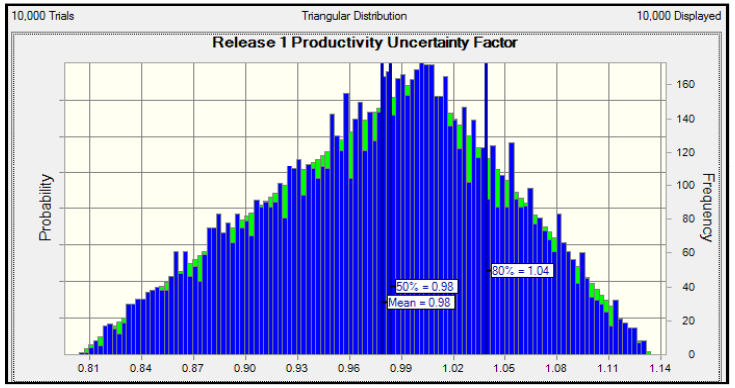
Table Of Contents

Model uncertainty around performance parameters after establishing a sound point estimate

- Uncertainty can be modeled using various risk / uncertainty tools as Argo and Crystal Ball

Uncertainty Variable	Factor	Distribution Form	Skew	Lower Bound	Expected Value	Upper Bound
Velocity	1.0000	*Traingular*	*Left*	0.8038	1.0000	1.1340

- A typical way to model uncertainty around Velocity is by triangular distribution
  - **Lower Bound:** Lowest normalized productivity *for a completed Sprint over an Iteration or through analogous data*
  - **Expected Value:** Average normalized productivity *for a completed Sprint over an Iteration or through analogous data*
  - **Upper Bound:** Highest normalized productivity *for a completed Sprint over an Iteration or through analogous data*



- Uncertainty in projected velocity and productivity and other key input parameters is modeled using Monte Carlo simulations

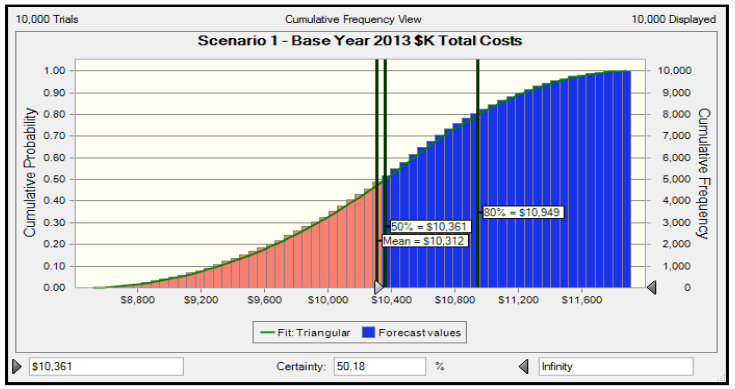


Table Of Contents

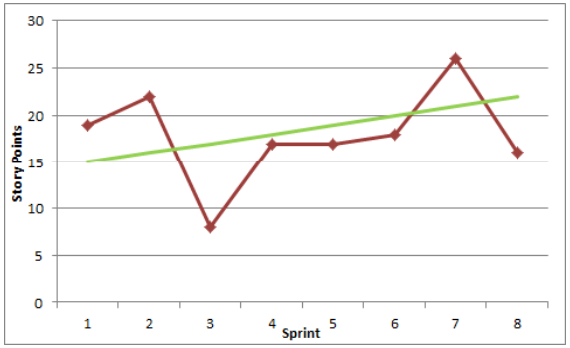
Utilizing this methodology of estimating SW development costs for "Agile" projects has several apparent benefits

- Utilizes metrics relevant to the development environment that are most likely those being reported in performer reports / CDRLs

CONTRACT DATA REQUIREMENTS LIST (CDRL)			Page 1 of 2
A. Contract line Item No.	B. Exhibit	C. Category *(Check appropriate one)* TDP ☐ TM ☐ Other X	
D. System/Item	E. Contract/PR No.	F. Contractor	
1. Data Item No. A004	2. Title of Data Item	3. Subtitle	
4. Authority SOW Paragraph 1.5	5. Contact Reference	6. Requiring Office AJW-	
7. DD 250 Req'd / 8. APP Code	9. Distribution Statement Required	10. Frequency Once	11. As of Date (AOD)
12. Date of First 20 days after contract award	13. Date of Subsequent Submission 20 days after STC Approval	15. Distribution	

- Delineates a method and structure to initially organize and prioritize requirements that promotes a clearer understanding of the project's scope for all stakeholders

- Produces in-progress metrics that makes it easier to gauge productivity during the project

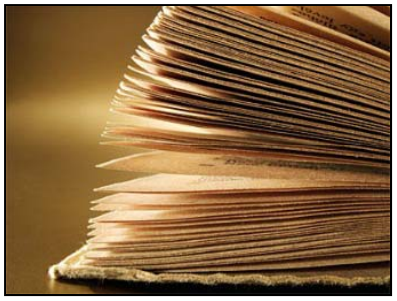


- Relates complexity to **effort** and not size, which seems to be more intuitive to engineers that help scope the effort

Complexity = 1, 2, 3, 5, 8, 13, 21 ≠ $PM = A \cdot Size^{E} \cdot \prod_{i=1}^{n} EM_i$

There are also several challenges in using this approach

- Lack of historical data or studies of Agile productivity metrics for Government software projects



- Defining the size of upfront Planning and Design effort that occurs prior to the development period
- Initially defining, prioritizing, sizing requirements is a large effort requiring attention from various project stakeholders

- Subjectivity in the initial rating of complexities (i.e. initial story point estimates)
- Continuous refinement of sizing/complexity metrics adds uncertainty to the estimate and requires more estimate maintenance over time

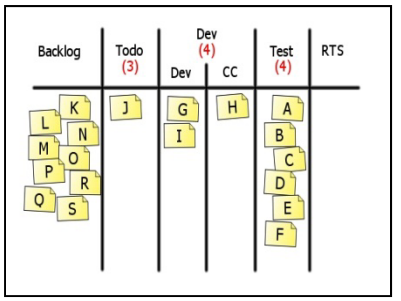
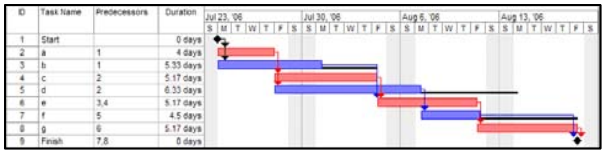


Table Of Contents

As “Agile” software development becomes increasingly prevalent in government software projects, this method is a viable option to estimate their costs

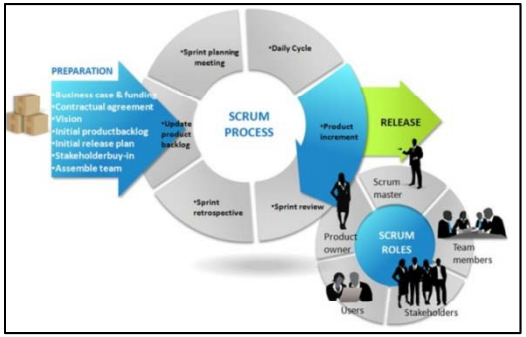
- Dynamic updates to outputs, both cost and schedule, as project experience/data is incurred



- Cost estimates that align to how the development is actually being executed

BY13 $K	FY2013	FY2014	FY2015	Total
CSCI A-1	$2,159	$1,158	$2,505	$5,822
CSCI A-2	$4,531	$3,423	$2,351	$10,305
Total	$6,690	$4,581	$4,856	$16,127

- Model will be more defendable early in the project life-cycle as more analogous data is collected and incorporated



Questions?

For further information . . .

Omar Mahmoud
Associate

Booz | Allen | Hamilton

Booz Allen Hamilton Inc.
1615 Murray Canyon Rd
Suite 140
San Diego, CA 92108
Tel 714.421.1231
mahmoud_omar@bah.com

Blaze Smallwood
Associate

Booz | Allen | Hamilton

Booz Allen Hamilton Inc.
225 West Wacker Drive
Suite 2270
Chicago, IL 60606
Tel 309.359.3160
smallwood_blaze@bah.com

Back-up Slides