

CMPS115 EXAM 1 Q&A

- Process Drivers question:

Question: What software process should Darlene select as the best (choosing from the McConnell definitions)? Justify your answer as Darlene might have to!

Suggestion: list Darlene's needs according to the capability segments in Table 7-1.

Let's use a classic weighted score technique to rank the models as a rough cut, then decide between the top scorers. Our ranking starts with McConnell's Table 7-1 augmented with a "WC importance" weights column, and develops a score for each model by summing weighted scores for each model capability. To develop the weights, consider the description her boss gave Darlene to generate weights (0 being low weight or value and 5 being high weight or value.):

<i>Capability</i>	<i>Relevant Statements</i>	<i>Result</i>
Handles Poor requirements	For the first few years, let's do apps that are modeled after existing kitchen capabilities in their functions and, wherever possible, the user interfaces. The entire line of WC KitchenApps is to have a similar style of user interface, displayed on small monochrome touchscreens or using voice I/O – so think ahead on the UI stuff. User interfaces are to be simple but must be foolproof – the liability possibilities are huge – and will have to work for users all across the country.	2
Handles Poor architecture	We're looking at applications with limited lifecycle implications: documentation delivered to customers is just the back-of-the-package stuff, there is no need for licensing, update/patch management, all that stuff you've bothered me with in the IT department.	1
Produces Reliable System	Once we ship we have a devil of a time getting anything back, so take care that you know what works before it gets to my mother's kitchen ...these apps have to be as reliable as the existing products we sell: turkey basters that last for years without failure	5
Produces Growable system	simple and intuitive ...computerized interfaces will be stable for a long, long time	2
Manages risk	WC has developed a culture based on fine-grained control of their costs ... the liability possibilities are huge	5
Provides Definite schedule	Time to market is not critical, culture based on fine-grained control of their costs	3
Has Low overhead	[culture of] razor-thin margins on low prices	4
Allows Midcourse corrections	[culture of] reasonable adaptability to changing conditions fierce competition for shelf-space and customer notice	4

<i>Capability</i>	<i>Relevant Statements</i>	<i>Result</i>
Provides Customer Visibility	NO REQUIRMENT	0
Management Visibilty	WC has developed a culture based on fine-grained control of their costs, and reasonable adaptability to changing conditions	4
Little sophistication	brand new 4-person software development effort	4

Poor=1, Fair = 3, Excellent = 5

Capability	WC	P.W.	C&F	Spiral	M.W.	E.P.	S.D.	E.D.	DTS	DTT	COTS
Poor req	2	1	1	5	4	5	1	4	2	3	5
Poor arch	1	1	1	5	4	2	1	1	1	3	3
Rel Sys	5	5	1	5	5	3	5	4	3	3	3
Growable	2	5	2	5	5	5	5	5	4	1	0
Mng risk	5	1	1	5	5	5	3	3	4	2	0
Def sched	3	3	1	3	3	1	3	3	5	5	5
Low overhead	4	1	5	3	5	3	3	3	3	4	5
Midcourse	4	1	3	3	3	5	1	4	2	5	1
Customer vis	0	1	3	5	3	5	3	5	3	5	0
Manager vis	4	3	1	5	4	3	5	5	5	5	0
Little sophis	4	3	5	1	2	1	3	3	1	3	3
WEIGHTED SCORE	84	76	132	137	113	110	123	107	119	79	

There are 3 standouts: spiral, modified waterfall, and evolutionary delivery.

ED is lower-scoring and fairly easy to discard, because we have no need for a lot of customer redirection after the GUI is settled – the functions can be judged by their similarity to existing implementations.

Deciding between spiral and M.W. is easy when we go from this general discussion to realizing the Darlene is very unsophisticated (new to the job, never done a risk analysis before) and is building a series of **small apps** – probably won't actually get to go around the spiral enough times to get practiced in risk reduction ... the spiral model is too heavyweight to use for building a thermometer display!

And, the modified waterfall will be easy to use (GUI prototyping to settle the only risk issue then strict requirements available for the computer interfaces or the emulated kitchen function) and will fit the low-overhead goals of the company while not requiring the brand-new software team come up with sophisticated metrics and analyses of risk.

So, M.W. It is!