



Software Developers Story

Jack Bray is the managing director of a software company that builds bespoke software for its clients. The company suffers from the three classic development issues that kill the most talented of teams:

1. 'It's ready when it is ready': a universal approach to software builds that is no respecter of cashflows or client deadlines.
2. 'Not really sure how much it will cost so we will say X and hope that's what happens'. This means the customer gets charge incorrectly and no-one dares look at the project budget to see what state it is in.
3. 'I don't remember anyone telling me that', when the investigation starts as to why key features or changes have gone missing.

We hear these ones loud and clear, we have done it ourselves in the past. Which is why we designed Ava to help you change the way you work and put an end to these killer issues.

Here's how:

It's ready when it's ready: er, no it isn't. It isn't even necessarily ready when the customer demands it to be. Your development team needs to iterate the project into the smallest detail possible – we favour tasks that take no more than 8 working hours, including alpha testing. Those tasks are then written into Ava's project workflow screen in those iterations, allocated and tracked

Task and Actions List

These are the things you want to remember to do

Go To Project Management Workflow

Build a detailed project management plan

Action	SM	Est Time.	Depends On	Due Date	Date Done	Set Alert?	Priority
Four							
Specification							
Copy shipping report needed from Matt	CMG	120 mins		20/07/2008	20/07/2008	<input type="checkbox"/>	
Analyse calculations from shipping report	CMG	180 mins	Four Specification Copy shipping	21/07/2008	21/07/2008	<input type="checkbox"/>	
Specify business management report	DA	60 mins	Four Specification Analyse	22/07/2008	24/07/2008	<input type="checkbox"/>	
Meeting with Matt	CMG	120 mins		24/07/2008		<input type="checkbox"/>	
Design							
Design layout of shipping report screen	DA	60 mins		28/07/2008		<input type="checkbox"/>	
Code							
Code shipping report screen	RG	75 mins	Four Design Design layout of	30/07/2008		<input type="checkbox"/>	
Code management report screen	RG	80 mins		04/08/2008		<input type="checkbox"/>	
Design							
Design layout of management report screen	DA	90 mins		02/08/2008		<input type="checkbox"/>	
Test							
First Test shipping report screen	DA	100 mins	Four Code Code shipping report	06/08/2008		<input type="checkbox"/>	
First Test management report screen	DA	120 mins	Four Code Code management report	07/08/2008		<input type="checkbox"/>	
Second Test shipping report screen	GH	125 mins	Four Code Code management report	08/08/2008		<input type="checkbox"/>	
Second Test management report screen	GH	130 mins	Four Code Code management report	09/08/2008		<input type="checkbox"/>	

You also ask all of your development team to record their time against the iterated tasks. In a great piece of simple design, each iterated task you specify in the *Workflow* will appear in the *Time Records* screen, making it child's play for someone to record time against it:

Time Details

Staff Initials: AVA (Default AVA Login)

Charge Date: 17/09/2008

Related Task:

Private Rate:

Stage	Group	Action
Four	Specification	Copy shipping report needed from Matt
Four	Specification	Analyse calculations from shipping report
Four	Specification	Specify business management report
Four	Specification	Meeting with Matt
Four	Design	Design layout of shipping report screen

Charge Code:

Rates etc

Chargeable Item	Amount	Reason	Rate	Start Time
Specification	00:00		£20.00	
Design	00:00		£20.00	
Coding	00:00		£20.00	
Other	00:00		£20.00	
Mail Out	0		£20.00	
Mail In	0		£20.00	
Telephone Call	0		£20.00	
Expense	£0.00		-	
Expense (vat)	£0.00		-	

Add Record

And here is the really good bit: by doing this, you will know

- a. How the project is going time wise
- b. What the deadline dates are in a clear, communicated to everyone style that brooks no argument
- c. When your team are becoming reliable on their time estimates

Point c is a particularly interesting one as it takes time to move developers into a commercial time constraint that feels comfortable. How well it happens depends entirely on how well you all learn to be more realistic about time estimates. A good time estimate is based on experience of how long something actually takes, which is a lot different to how long someone would like it to take.

When your development team record their time against tasks, Ava can then tell you whether or not that time matched the original estimate in the Workflow in the *Timeshift Report*. At the end of each project plan, you look at this information and people can learn that, for example if Task A was scheduled for 8 hours and actually need 24, that becomes the benchmark the next time that task is listed.

‘Not really sure how much it will cost so we will say X and hope that’s what happens’.

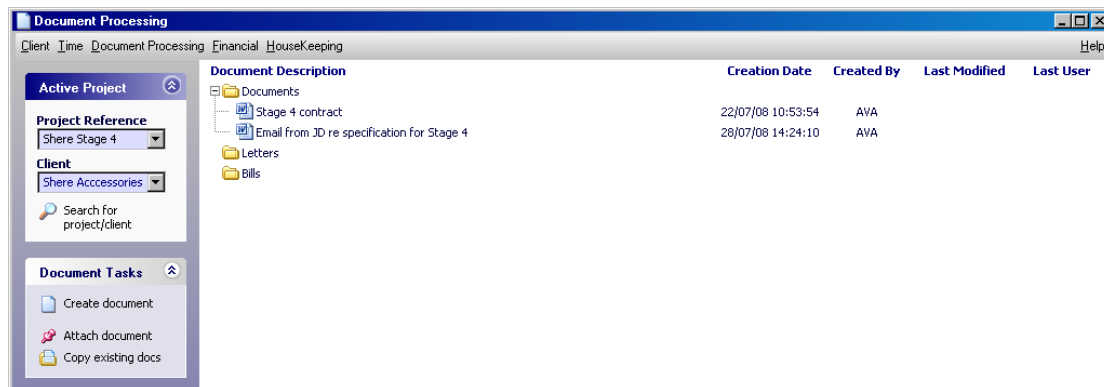
Probably no again! This continues the theme above because project costs are all about the ability to iterate and estimate accurately. Let us take an obvious example: if I tell a client that a task will take 8 hours because that is what my team are telling me, then I will cost the project at, say, £400. If it ends-up taking 24 hours, I have two choices, neither of them very comfortable: either I call the client and tell them that costs have now rocketed to £1200 without any warning or I write off £800.

So again we come back to using the tools in Ava to help you and your team learn and practice good iteration skills. This means you will cost projects properly and use the time records to confirm that the project is staying within the estimated time limits. If it isn’t, it enables you to take remedial action much earlier and before the effects become a problem.

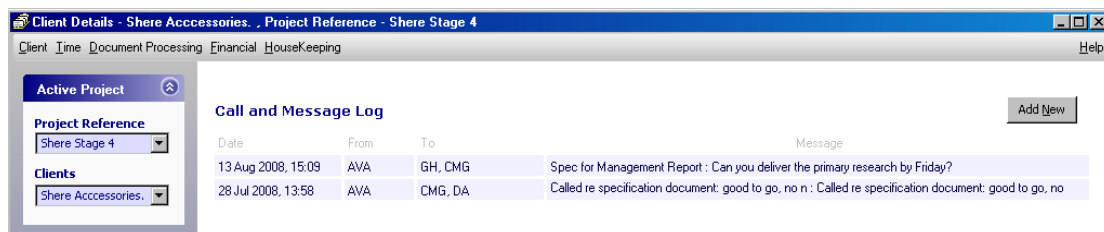
‘I don’t remember anyone telling me that’

There are two great Ava ways of communicating with your team that support and supplement the meetings you should be having with them throughout the project cycle. The first is the client *Document* archive, from which you can create all the specification and project documentation and into which you can attach electronic copies of anything related to the project, including Word documents, graphics and sound files. Anyone can access these

documents and remind themselves of the detail around which the *Workflow* is built – you can even attach document templates to the Workflow itself.



The second way of communicating with your team is the *Calls and Messages* service within Ava. This is an instant broadcast internal messaging system that stores everything against the Client. It is noisy, it alerts all named recipients of the need to read the message and if they are at their desks, it does so instantly. And you can even check if they have read it. In turn, they can read it and never lose it because it will be stored in the Client screens.



Our users have often made this the internal communication tool of choice over MS Messenger because it cannot be used externally and makes project communication much more immediate and long lasting.