Project Proposal: PHP-SRePS

(Marcus Rakkhit ⬛⬛⬛⬛⬛

Mon 2:30 EN402⬛⬛⬛⬛⬛

ACTUAL ESTIMATION

Back in task 61C, I've predicted that we'll complete 6 tasks for the first backlog item (Add a record to the database); The total time for all tasks would take around 5 hours to complete (the evidence in figure 1 was used in task 61C).

TASK	ESTIMATED TIME	JUSTIFICATION
Clever-Cloud	30 minutes	Tested Clever-Cloud in my free time
Write Connection statement C#	30 minutes	Statements was already researched in my free time
Create UML diagram	1 hour	We have 3 classes (management, façade and entity class) to plan out constructor variables and methods
Setup appClient	30 minutes	All we do is create objects to test the business tier
Setup business tier	2 hours	Requires to build the entity and façade class
Write add function	30 minutes	Statements was already researched in my free time

FIGURE 1

In our actual estimate, I would say that this sprint backlog item is within range of the original estimation (I would it has 55% accuracy).

Based on day 2-4 of our burndown chart, the completion time was reduced from 5 hours to 4 hours; The main reason I overestimated the timeframe was because I had previous experience for the following: setting up clever-cloud and a similar application for a different unit. This experience gave me a head start for executing all necessary tasks for this item. As a result, I think more experience in any future task can improve my efficiency for any other group project.

During the sprint the UML diagram task was replaced; When I looked at my Doubtfire tasks, I realised that the diagrams would be completed in task 14P; What we did instead was write entity boxes (this can be found in day 3 of the meeting). Even though it took around an hour to complete, I was multitasking this task with the application; Almost all the members were UML diagram, meaning it allowed me to focus more easily on the other task. The benefit of multitasking in this project (and if I have many other projects) is that it can burndown the time frame of multiple tasks at a time (rather than doing 1 task at a time)

The business class took about 2 hours as expected from my original prediction; Before I started the sprint, I'd tested the database connection/statements in C# to ensure everything was functional (for future success, pre-testing can help me familiarize with the content and give me a head start in the project). During the sprint, I needed to collect the entity boxes from the meeting to write out all variable for each entity class. Afterwards, the façade class required a connector and every add operation. One thing I underestimated was that I needed to write the add operation for EACH ENTITY CLASS (originally, I though there was only 1 add operation); Fortunately, all I needed to do (to address the problem) was create an add operation for each entity class.

Originally, I created an appClient class to test the add operations (This process took around less than 15 minutes); However, this was only written to test the database on the console. When we wrote the user interface ▓▓▓▓▓▓▓ wrote this code), the classes were different. Instead of using the appClient class, we used XAML (which is like an XML file) file to layout the client user interface and use XAML.CS class to write the code for the main class. Currently, I don't have much experience interacting with XAML, but I still found this component easy to pick up (I currently learning XML files in the Mobile Applications unit).

Overall, I believe that half of my predictions met my actual estimate; In the future, I believe that good communication/teamwork and work preparation would be my key goals that'll improve the efficiency/estimation of my projects.