Synful Computing – Project Update

Transcript - SEPM Assignment 2 - Group 1 - University of Essex

**Group 1**: Nassar Al-Naimi, Charles Kuyayama, Abdulahi Alihu Ngamjeh, Trevor Woodman

**Presenter**: Trevor Woodman

# Preface

Hello, and welcome to this presentation by Group 1 of the second assignment for the Software Engineering Project Management course, part of the University of Essex Online computer science program. The following will be read from the perspective of Synful Computing, our fictitious computing company creating the “Synputer” – a next generation computer built sometime in the mid eighties.

# Introduction

Thank you for joining our Synful Computing team today for this presentation. We have a short but informative one planned for you, and we hope that it will assuage EDC’s concerns related to specification 1, previously sent to EDC.

We will begin this presentation by outlining our primary project milestones, walking through the concerns and issues brought up by EDC and how we have remediated or exceeded these issues, we’ll take a look at our new fully costed project plan, and we’ll briefly go over and demo a program that we built to assist with our cost and timeline estimations.

# Agenda

As I mentioned in the introduction, we have 5 points to touch on today:

1. Milestones
2. Requirements Table
3. Costed Project Plan
4. Cost Estimation Program
5. Summary

# Milestones

Here we’ll go over our major milestones and go into some detail on each.

## Specification 1

This specification was created as close to the requirements as possible, but as we will talk about, ran into some issues due to challenges that came up in development.

## Project Report

Our project report was sent to EDC, and we received a very understandable response: our specification came under the requirements that were set forth for the purchase agreement between Synful and EDC.

## Specification 2

This specification is based on the feedback that EDC provided to Synful in response to specification 1 and the project report. The list of hard requirements that we received from EDC have been met or exceeded in some. This will be spoken about in more detail in the next slide, the requirements table.

## Project Update

This presentation serves as Synful’s final update prior to the mass production of the Synputer, assuming that there is no feedback that would require additional changes that would delay production. This being said, we are confident that specification 2 meets or exceeds the expectations of EDC.

## EDC Order Fulfillment

Specification 2, which meets or exceeds the requirements sent by EDC, is produced until completion. Assuming no major complications arise, this will be approximately 129 to 134 weeks (approx.. 32 to 34 months) from today.

## Consumer Market Release

The Synputer will be placed into full release, and marketing will ramp up for consumer market.

# Requirements Table

The first requirement provided by EDC was an industry standard operating system. This requirement was met, as specification 2 contains Synful Computing’s already popular HyperBasic operating system. Previous consumers will be familiar with this OS and be able to pick up new features quite easily.

The second requirement was an external keyboard connection. This requirement was met, with specification 2 containing two individual IOP-J SC150 2-channel ports that each support joysticks, mice, and keyboards. Having two IOP-J SC150 ports will support up to four different devices connected to the Synputer at the same time.

The third requirement was that the Synputer have 512KB of RAM at minimum. We are proud to say that specification 2 contains a whopping 2MB of RAM. Four 512KB RAM modules are installed on the socketed board. Having an abundance of RAM will assist in multitasking in HB/OS, accelerating user productivity and usability of the system, and pushing the boundaries of the size and complexity of programs that can be developed for the Synputer.

The fourth requirement was that the Synputer comes with at least one industry-standard removable drive. Specification 2 contains two storage cartridges, both removable, that will allow users to take advantage of our included emulation software and have plenty of space left for other programs or upgrading later.

The fifth requirement was SCSI expansion capability. While specification 2 does not contain SCSI capability, we believe the next requirement will make up for it. This feature was cut due to manufacturing cost and time constraint.

The sixth requirement was the CPU had to be a Motorola 68K at a minimum, and preferably upgradeable. We are very pleased to announce that specification 2 contains a socketed board, which allows for upgrading and swapping out of many components by users, for upgrades or personal preference. For this sixth requirement, we’ve gone above and beyond by including a EP7500FE, an undeniably top-of-the-line processing unit that far surpasses the 68K’s performance. The EP is an extremely powerful 50MHz CPU and contains a 4KB cache. This CPU was chosen to be a better fit for a next-generation system, and because of the socketed board, can be swapped out for a 68K by the consumer if they prefer.

The seventh requirement was that the Synputer have a minimum of two serial ports that support RS 422/485 standard. This second iteration of the Synputer has two IOP-S 16550 UART serial ports. Both are 1-channel and support the aforementioned industry standards.

The eights and final requirement was that the board be ready to support a GUI system and a mouse if required by the user. As we have mentioned, specification 2 contains 2-2channel IOP-J ports that conforms to this requirement. The board does support a GUI, and we have included our HyperBasic GUI as part of the unit.

# Costed Project Plan

The budget that Synful Computing is working with is purely based on the agreed on order amount for EDC, which is £500,000. In line with our strategic objectives, this fully costed project plan presents a succinct, but detailed, roadmap for the successful execution of the Synputer project. This planning process involved a thorough analysis of available components, hard requirements set forth by EDC and Synful, labour market cost, past and future Synful consumer requirements, and timelines. The project plan shows a breakdown of anticipated costs across unit costs, labour costs, profit, and completion timeline. With this plan, we aim to exceed expectations, delivering value and achieving our goals in a cost-effective and efficient manner.

Using the 3-point cost estimation technique and PERT estimation method, we have created the following tables. The first table is the 3-point cost estimation, showing pessimistic, most likely, and optimistic outcomes of the projected plan.

The most likely outcome is based on internal costs and quotes received from manufacturers and labour agencies. It assumes that there are an average number of delays, no significant challenges arise, and average work efficiency.

The pessimistic outcome is based on the worst possible outcome, taking the most likely outcome and multiplying by 2 for each cost point. This would be the worst case scenario, and would involve basically all aspects of the projects hitting roadblocks, challenges posing significant time sink, and low work efficiency.

The optimistic outcome is based on the best case scenario, and essentially halves all cost points from the most likely outcome. This would assume that the project hit very minor to no roadblocks, minimal challenges arose, and work efficiency was high.

The PERT, or Program Evaluation and Review Technique, estimation table shows a “time expected” outcome of project completion, i.e., the order fulfillment for EDC. This is currently 134 weeks from today, or 34 months. This table also shows the PERT formula applied to unit and labour cost, and the expected profit from this estimate.

The bill of materials for specification 2 was heavily considered, and optimized for quickest possible production based on specification requirements. To this end, we suspect that the project will fall somewhere between most likely and optimistic outcomes, providing that no major external challenges arise in the production cycle.

# Cost Estimation Program

As part of this project update, we’ve developed a cost estimation program, and the figures presented in the preceding slide were derived from using this tool. The program employs the 3-point cost estimation technique, coupled with the PERT formula. For reference, we have included short summaries of the 3-point cost estimations below for reference while we run a short demonstration of the program.

If you’d like to pull up the repository in the meantime, please visit the GitHub URL included in this slide.

Usually, this would involve a very short demo of the program and the output it produces. Instead, we’d like to walk you through the source code itself, explain a bit about the internals of it and how the calculations are being made, and review the output from a more informed perspective.

We would also like to note that this was created very quickly, as budget did not allow for a large effort on this. Also, a second disclaimer, you would not generally comment code this extensively and is considered bad practice but for readability and the demonstration we have included copious comments to describe what the code is doing at any given time.

The program primarily pulls from a couple of csv files, for hardware and software specifications, pulls information from them, and compiles a series of numbers to then use in the 3-point and PERT estimations. This program was written in Python 3.10, although any newer version than 3.9 should work as expected. There are detailed install and running instructions contained in the repository README should you need them.

# Cost Estimation Program – Demo

Stepping into the code here, we import the csv module here at the top of the file, this is so we can work with the hardware and software csv specification files. We’ve also hardcoded the sales price of the Synputer here for later use in calculations.

The initial menu section here prints out a quick blurb on authors and where you can access the code if you would like to review it later.

Stepping into the hardware section here, we are declaring several variables that will later be used to compile some numbers for later calculations.

This “open” statement here is opening the file and storing the file object in an anonymous variable that we’ve aliased here as “hwSpecFile” for easy access for use just below.

This statement passes in the alias for our file object into the csv.reader method call. This stores the file as a csv reader object, which makes it possible to access each row and column in just a moment.

After this, we skip the header, as we don’t want to include the header in our loop over each row. We then iterate over every item, or row, in the csv reader variable, and assign local temporary variables for each column that we want.

After storing the local item variables, we go through some more statements that are all working towards compiling the final totals for each hardware variable we looked at before. For example, we need to compile the total unit cost for hardware, and here we are saying that if the item quantity is greater than 1, then we should multiply this item’s unit cost by its quantity. If there’s only one of the item in the Synputer spec, then we simply add the cost of the item to the total unit cost.

We do some more of the same, for manufacturing cost, design weeks, design labour cost, and redesign weeks and labour costs.

There is a near identical section for the software specification file, which we won’t walk through, but it is here and available in the repository. It has very similar software variables up top here, and uses the same method to open the file and iterate through the items in it.

This three point generation section focuses on compiling all of the hardware and software cost points per each concern. This is a little messy, but it’s primarily getting the values for three different things, three times for hardware and software each. In the middle here, we’re getting the values for unit cost, design cost, and design weeks. We do this twice more for pessimistic and optimistic points. We then do the same thing for software below this section.

Under the hardware and software three-point sections, we have a totals section that takes both values from hardware and software and creates totals for each outcome. This is used in the final output which we’ll go over in a moment.

Just below the totals estimations, we have a small section here for program evaluation and review technique, or PERT, estimations. These three cost estimates, unit, labour cost, and labour weeks, are all based on the PERT formula, which is shown here. This means we’re taking the optimistic estimate, adding it to four times the most likely estimate, adding that to the pessimistic estimate, and then dividing the resulting number by six, resulting in our PERT estimate.

This final output section looks noisy, but it is simply printing some explanation and the estimations in an easy to read format.

Now that we’ve walked through the code, let’s run the program. Here we can see that the estimations get printed out as we’ve coded them in the output section that we just looked at.

This sums up the code demonstration. If you are interested in knowing more, please visit the repository and have a look a the source code and the README file for significantly more information including tables and totals.

# Summary

In this presentation, we’ve covered several key points:

1. We’ve detailed our project milestones and the anticipated time for completion
2. Explored the project requirements, offering insights into the decisions outlined in specification 2
3. Examined our comprehensive project plan, including detailed cost considerations
4. Provided a brief overview and demonstration of our cost estimation program, showcasing how it has played a crucial role in shaping our revised project plan

We are confident that this updated plan not only meets, but exceeds, the requirements set by EDC. We anticipate that the Synputer will surpass expectations.

Thank you for investing your time with us, and we wish you a great day.