Development Team Project: Project Report – Synputer

Team Name: MASH (Group 1)

Team Members: Anda Ziemele, Mario Butorac, OiLam Siu, Samuel Harrison

1. Introduction

English Digital Computers has contracted Synful Computing to develop a computer for personal and business use. Our main competitors are IBM and Microsoft.

2. Methodology

The Synputer will be developed using Scrum rather than Waterfall (Nagl, 2023) due to its adaptability through short sprints that allow for frequent feedback and changes, its high quality ensured by iterative development and continuous testing, customer satisfaction from regular feedback aligning the product with specifications (Layton et al., 2022), continuous improvement through sprint retrospectives, and streamlined communication via standups. To address hardware development constraints (Schmidt et al., 2017), the product will be developed in a VUCA environment (Atzberger & Paetzold, 2019), enhancing adaptability, innovation, risk mitigation, and collaboration (Weichbroth, 2022).

3. Statement of Work

Key deliverables:
- A PC
- A Compatible OS.

3.1 Gathered Requirements

Requirement ID	Requirement	Category
Hardware		
Req001	Built-in screen	Functional
Req002	CPU with forward compatibility	Non-Functional
Req003	Expansion slot for removable storage	Functional
Req004	Industry-standard storage	Functional
Req005	Ports for IO devices	Functional
Req006	Serial Ports for Networking	Functional
Req007	Weight <= 2KG	Non-Functional
Req008	> 2-hour battery life	Non-Functional
Req009	Support BASIC programming language	Functional
Req010	High-resolution display	Functional
Req011	External Keyboard connection	Functional
Software		
Req012	Industry-standard OS	Functional
Req013	Compatibility with EZ-Suite Applications	Functional
Req014	Networking compatibility	Functional
Req015	Gaming emulator for backward compatibility	Functional

Table 1. Gathered requirements.

3.2 Missing Requirements

A domain model was created to ensure the completeness of requirements (Appendix 1). This approach proved fruitful in requirements gathering (Arora et al., 2019). Table 2 lists the missing requirements identified.

Requirement ID	Requirement	Category
Req016	User management and authentication	Functional
Req017	Support for multiple user accounts	Non-Functional
Req018	Data encryption	Functional
Req019	On-going system updates	Functional
Req020	GUI and UX requirements	Functional
Req021	Data integrity mechanisms	Functional
Req022	Audio output capability	Functional

Table 2. Missing requirements.

3.3 Components List

The components list includes the elements for the proposed design, tailored to transform the gathered requirements into a practical solution (Appendix 2).

4. Gherkin Specifications

Figure 1 shows the Gherkin requirements for some of the critical functionality of our system (Rice et al., n.d.):

```
Scenario: Translation of Edubasic (EB) to HB
Given   EB programs need to run in HB
When    a converter for EB to HB translation is created
Then    EB programs should run in HB without any manual intervention required

Scenario: Old games connection for existing users
Given   new machine is faster than older machines
When    application that emulates older machines is created
Then    the emulator should run all old games at the same speed as on existing
        machines
And     allow users to run a word processor at the same time

Scenario: EZ-SUITE licensing and compatibility
Given   the machine comes bundled with EZ-SUITE business applications
And     the applications only run under HB/OS
When    £25 charge per machine is paid as per licensing deal
Then    the applications should run regardless of OS

Scenario: Default boot sequence
Given   attempts to load an operating system during startup
And     the boot loader is accessing drives
When    attempting to locate a boot sector
Then    it should first try and access drive/cartridge A
And if  unsuccessful, it should try drive/cartridge B
And if  neither drive has a boot sector, it should default to loading BASIC
        from ROM
And if  there is no ROM BASIC, the system will hang until rebooted

Scenario: Removable drives
Given   businesses exchange data regularly
When    compatibility is important
Then    removable storage must be compatible with industry standards
```

```
Scenario: Expansion Slots
Given   expandability is essential
When    the speed at which technology changes
Then    standardised expansion slots are mandatory

Scenario: 2 Serial Ports and SCSI
Given   businesses are accessing networks and edge devices such as printers
When    connecting networking devices
Then    Uuse one serial port
When    connecting non-networking devices
Then    use SCSI or the other serial port

Scenario: External Keyboard
Given   European countries have different layouts
When    they are not standardised
And     changing the integrated keyboard would be too challenging
Then    an external keyboard connector is required

Scenario: Portable
Given   machines need to be portable
When    need to run at least two hours powered by battery
Then    the weight must not be over 2kg

Scenario: Operating System
Given   OS industry compatibility is critical
When    MCC consultancy is already creating it
Then    MCC's Unix-based OS will be deployed
```

Figure 1: Gherkin statements

5. Assumptions and Dependencies

Category	Assumption/Dependency
Hardware	• Hardware components that will be externally procured with a maximum lead time of one week include the CPU, 512K RAM, mono sound card, IOP-J SC150 port component, GDISP component, and cartridge storage.
Software	• Software that will be externally procured and be available immediately with all required licencing, such as EZ-Suite Business Applications. • Ensure forward compatibility of hardware and software. • Develop a TB to HB code converter. • Provide I/O connectors for compatibility.
Cost and Resourcing	• Full availability of in-house resources; external resources if needed. • Sharing administrative resources with other projects at Synful Computers, with overhead allocated based on utilisation. • Some roles at Synful Computers allow flexible labour cost allocation based on utilisation without requiring a headcount. • The teams work five-day weeks and are productive during working hours. • Design and manufacturing costs are included in development costs. • The end-user bears the license for EZ-SYS applications.
General	• All units will be delivered in complete working order. • The user will interact with the system via Keyboard.

Table 3. Project Assumptions and Dependencies.

6. Development Plan
- Budget: £500,000 provided by EDC
- Timeline: 13 months
- Deliverables: 2,000 Synputers purchased by EDC for £250 per machine

6.1 Estimates and Project Schedule
The project will be delivered in 13 months. This timeline uses Brooks (1995) calculations to estimate the duration of the project's major stages (Appendix 3):
- Planning Time: T_p = 13/3 = 4.3 months
- Development Time: T_d = 13/6 = 2.2 months
- System Test Time: T_s = 13/4 = 3.3 months
- Testing Time: T_t = 13/6 = 2.2 months (3.2 months, see explanation below)

Since the sum of the above values is less than 13, and studies show that inadequate quality control is one of the major contributors to project failures (Jones, 2004), the additional time will be devoted to testing, bringing it up to 3.2 months.

6.2 Cost Plan
The material cost is £157 per machine (Appendix 4). Assuming that the initial design, development, manufacturing, and testing costs are included (Appendix 5), the cost of the first 2,000 machines can be controlled at **£250 per unit.**
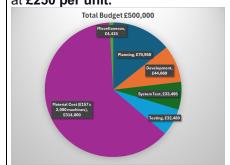

Figure 2: Total budget allocation for the project
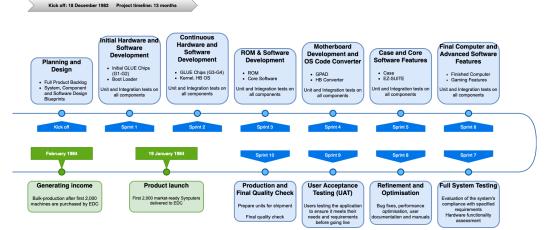
6.3 Milestones and Deliverables

Figure 3: Key deliverables for each milestone (Negi, 2019; Schwaber, 1997)

7. Testing Strategy
- Prototype Testing: To obtain early feedback, we'll perform prototype testing (Camburn et al., 2017) at various stages during the system's development.
- Software Testing: Before release, all software development will go through rigorous testing by developers, QA Engineers, and QE Engineers (Beizer, 1984)
- Hardware Testing: To meet customer expectations, hardware will go through reliability, functional, user acceptance, and certification testing (Hartman, 2005).

8. Pricing Strategy
To determine the pricing strategy for the Synputer, we considered the market price for PCs in the 1980s (Appendix 6), the production costs, and the industry average Gross Profit Margin (GP%) (Polymer, 2024).
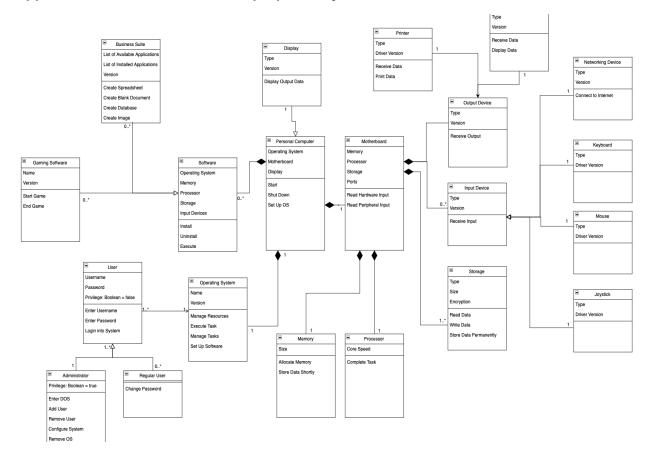Calculation:
- Unit Cost: £250
- GP%: 50% for a competitive edge and healthy profit.
- Sales Price: £250 / (1-50%) = **£500 per unit**

9. Conclusion
The Synputer aims to deliver a competitive, market-ready product within budget and timeline constraints, leveraging Agile methodologies to ensure flexibility and stakeholder engagement throughout the project lifecycle. This proposal outlines the strategic approach, requirements, and comprehensive plan we will follow to ensure the project's success.
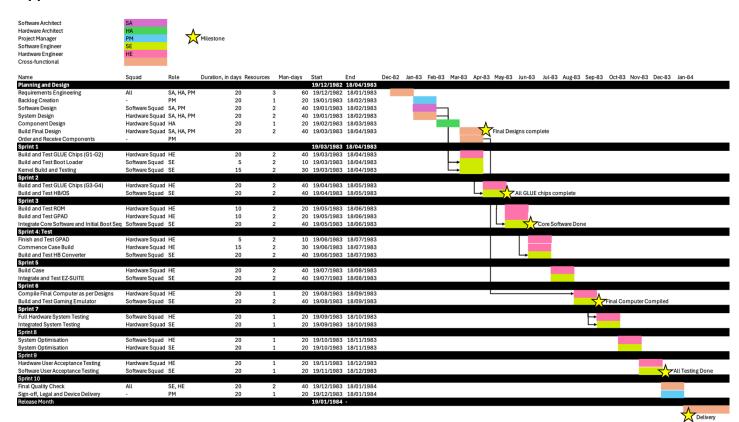
10. Appendix

Appendix 1: Domain model of the proposed system

Appendix 2: Components list

Component ID	Component	Qty	Category	Corresponding Requirement ID
Hardware				
Com001	Motorola 68000 series CPU (68k8)	1	Processor	Req002
Com002	128KB of RAM	4	Memory	General performance
Com003	32KB ROM Chips	2	Memory	Req009, Req012
Com004	Glue Chips	4	Logic	Integration
Com005	SC150 Chip	1	I/O	Req005, Req011
Com006	IO Chips	1	I/O	Req005
Com007	GDISP Chip	1	Graphics	Req010
Com008	Built-in Keyboard	1	Keyboard	Req001
Com009	Cartridge	2	Storage Drive	Req001, Req004
Com010	GPAD-SLDR (G-96)	1	Board	Req002, Req003
Com011	Desktop Case	1	Case	Req001
Com012	INTSND (i8042)	1	Sound	Req022
Software				
Com013	HyberBasic (HB) OS	1	OS	Req012
Com014	Telebasic (TB) to HB converter	1	Utility	Req009
Com015	EZ-Suite Business Applications	1	Application	Req013
Com016	OGRE96	1	Emulator	Req015

Appendix 3: Extracted Gantt Chart

Software Architect — SA
Hardware Architect — HA
Project Manager — PM
Software Engineer — SE
Hardware Engineer — HE
Cross-functional

⭐ Milestone

Name	Squad	Role	Duration, in days	Resources	Man-days	Start	End
Planning and Design						**19/12/1982**	**18/04/1983**
Requirements Engineering	All	SA, HA, PM	20	3	60	19/12/1982	18/01/1983
Backlog Creation	-	PM	20	1	20	19/01/1983	18/02/1983
Software Design	Software Squad	SA, PM	20	2	40	19/01/1983	18/02/1983
System Design	Hardware Squad	SA, HA, PM	20	2	40	19/01/1983	18/02/1983
Component Design	Hardware Squad	HA	20	1	20	19/02/1983	18/03/1983
Build Final Design	Hardware Squad	SA, HA, PM	20	2	40	19/03/1983	18/04/1983
Order and Receive Components	-	PM					
Sprint 1						**19/03/1983**	**18/04/1983**
Build and Test GLUE Chips (G1-G2)	Hardware Squad	HE	20	2	40	19/03/1983	18/04/1983
Build and Test Boot Loader	Software Squad	SE	5	2	10	19/03/1983	18/04/1983
Kernel Build and Testing	Software Squad	SE	15	2	30	19/03/1983	18/04/1983
Sprint 2							
Build and Test GLUE Chips (G3-G4)	Hardware Squad	HE	20	2	40	19/04/1983	18/05/1983
Build and Test HB/OS	Software Squad	SE	20	2	40	19/04/1983	18/05/1983
Sprint 3							
Build and Test ROM	Hardware Squad	HE	10	2	20	19/05/1983	18/06/1983
Build and Test GPAD	Hardware Squad	HE	10	2	20	19/05/1983	18/06/1983
Integrate Core Software and Initial Boot Seq	Software Squad	SE	20	2	40	19/05/1983	18/06/1983
Sprint 4: Test							
Finish and Test GPAD	Hardware Squad	HE	5	2	10	19/06/1983	18/07/1983
Commence Case Build	Hardware Squad	HE	15	2	30	19/06/1983	18/07/1983
Build and Test HB Converter	Software Squad	SE	20	2	40	19/06/1983	18/07/1983
Sprint 5							
Build Case	Hardware Squad	HE	20	2	40	19/07/1983	18/08/1983
Integrate and Test EZ-SUITE	Software Squad	SE	20	2	40	19/07/1983	18/08/1983
Sprint 6							
Compile Final Computer as per Designs	Hardware Squad	HE	20	1	20	19/08/1983	18/09/1983
Build and Test Gaming Emulator	Software Squad	SE	20	2	40	19/08/1983	18/09/1983
Sprint 7							
Full Hardware System Testing	Software Squad	HE	20	1	20	19/09/1983	18/10/1983
Integrated System Testing	Hardware Squad	SE	20	1	20	19/09/1983	18/10/1983
Sprint 8							
System Optimisation	Software Squad	HE	20	1	20	19/10/1983	18/11/1983
System Optimisation	Hardware Squad	SE	20	1	20	19/10/1983	18/11/1983
Sprint 9							
Hardware User Acceptance Testing	Hardware Squad	HE	20	1	20	19/11/1983	18/12/1983
Software User Acceptance Testing	Software Squad	SE	20	1	20	19/11/1983	18/12/1983
Sprint 10							
Final Quality Check	All	SE, HE	20	2	40	19/12/1983	18/01/1984
Sign-off, Legal and Device Delivery	-	PM	20	1	20	19/12/1983	18/01/1984
Release Month						**19/01/1984**	**-**

Milestones: Final Designs complete; All GLUE chips complete; Core Software Done; Final Computer Compiled; All Testing Done; Delivery

Timeline columns: Dec-82, Jan-83, Feb-83, Mar-83, Apr-83, May-83, Jun-83, Jul-83, Aug-83, Sep-83, Oct-83, Nov-83, Dec-83, Jan-84

Appendix 4: Breakdown of Material cost per machine

Material cost

Component ID	Type	Component	Category	Quantity per board	Unit Price	Total Material cost
Com001	Hardware	Motorola 68000 series CPU - 68k8	Processor	1	£5.5	£5.5
Com002	Hardware	128KB of RAM	Memory	4	£2.5	£10.0
Com003	Hardware	1 or more 32KB ROM Chips	Memory	2	£4.0	£8.0
Com004	Hardware	4 Glue Chips	Logic	4	£5.0	£20.0
Com005	Hardware	SC150 chip for supporting external devices	I/O	1	£15.0	£15.0
Com006	Hardware	IO Chips	I/O	1	£12.0	£12.0
Com007	Hardware	GDISP chip to support a high-resolution display	Graphics	1	£25.0	£25.0
Com008	Hardware	Internal Keyboard for Case	Keyboard	1	£5.0	£5.0
Com009	Hardware	Cartridge	Storage Drive	2	£5.0	£10.0
Com010	Hardware	GPAD-SLDR (G-96)	Board	1	£20.0	£20.0
Com011	Hardware	Desktop Case	Case	1	£25.0	£25.0
Com012	Hardware	INTSND (i8042)	Sound	1	£1.5	£1.5
Material cost per machine						**£157**

Appendix 5: Breakdown of Role allocation and personnel cost calculation

Total budget	£500,000
Agreed delivery	13 months
Assuming 20 paid working days per month	260.00

Stage	Months	Working days
Planning Time	4.3	86
Development Time	2.2	44
System Test Time	3.3	66
Testing Time (additional 1 months)	3.2	64
	13	**260**

Role Allocation and Cost Calculation (Working days x Manpower x Daily rate)

Role	Tasks/Skill	Daily Rate (£)	Role Allocation (person)	Planning	Development	System Test	Testing
Hardware Architect	Design, Layout, Fault finding	250	1	£21,500			
Software Architect	Design, Coding, Fault finding	300	1	£25,800			
Hardware Engineer	Build, test, troubleshoot	175	1-2		£15,400	£11,550	£11,200
Software Engineer	Code, test, troubleshoot	195	1-2		£17,160	£12,870	£12,480
Project Manager	Plan, manage, reporting, costing	275	0.5-1	£23,650	£12,100	£9,075	£8,800
Project Analyst	Update, replan, resourcing, costs	175	0				
Sub-total per stage				**£70,950**	**£44,660**	**£33,495**	**£32,480**
Total Development Cost				**£181,585**			

Appendix 6: Table of Home Micro costs and specifications in the 1980s given in the Case Study document

Model	Manufacturer	Category	Cost	Release Date	Comments
ZX81	Sinclair	Home Micro	£49.95	March 1981	Kit, membrane keybd, 1kB RAM, ROM with BASIC
Proton/ Micro B	Acorn/ BBC	Business/ Edu	£399	Dec 1981	Base unit with KB no FD, no mouse or screen, 32kB RAM
5150	IBM	Business	$1565	Aug 1981	Open Architecture; 16kB, CGA, no FD base spec.
Macintosh	Apple	Business	$2495	Jan 1984	Mono screen, keybd, FD, mouse & ROM, 128kB RAM
520ST	Atari	Home Micro	$799.99	June 1985	Mono screen, keybd, FD, Mouse, ROM, 512KB RAM
Amiga 1000	Commodore	Home Micro	$1295	July 1985/86	No screen, 256kB RAM, ROM, FD, KB, Mouse

References

Alshamrani, A. & Bahattab, A. (2015) A Comparison Between Three SDLC Models Waterfall Model, Spiral Model, and Incremental/Iterative Model. *2015 International Journal of Computer Science Issues* 12(1): 106-111. Available from: https://www.academia.edu/10793943/A_Comparison_Between_Three_SDLC_Models_Waterfall_Model_Spiral_Model_and_Incremental_Iterative_Model?auto=download [Accessed 12 August 2024]

Arora, C., Sabetzadeh, M. & Briand, L.C. (2019) An empirical study on the potential usefulness of domain models for completeness checking of requirements. *Empirical Software Engineering, 24*: 2509-2539.

Atzberger, A. & Paetzold, K. (2019) Current challenges of agile hardware development: What are still the pain points nowadays?. In *Proceedings of the design society: international conference on engineering design.* 1(1): 2209-2218. Cambridge University Press.

Beizer, B. (1984) Software system testing and quality assurance. London: Van Nostrand Reinhold.

Brooks, F.P. (1995) *The Mythical Man-Month*. Anniversary Edition. Philippines: Addison-Wesley.

Camburn, B., Viswanathan, V., Linsey, J., Anderson, D., Jensen, D., Crawford, R., Otto, K. & Wood, K. (2017) Design prototyping methods: state of the art in strategies, techniques, and guidelines. *Design Science*, (*3*): 1-13.

Hartman, A. (2005) Software and hardware testing using combinatorial covering suites. In *Graph Theory, Combinatorics and Algorithms: Interdisciplinary Applications* (237-266). Boston, MA: Springer US.

Jones, C. (2004) Software project management practices: Failure versus success. *CrossTalk: The Journal of Defense Software Engineering*, *17*(10): 5-9 [Accessed 19 August 2024].

Layton, M.C., Ostermiller, S.J. & Kynaston, D.J. (2022). *Scrum For Dummies*. John Wiley & Sons. [Accessed 18 August 2024].

Nagl, S. (2023). 5 Benefits of Agile Project Management. *School of Professional Studies at Wake Forest University*. Available at: https://sps.wfu.edu/articles/benefits-agile-project-management [Accessed 18 August 2024].

Negi, K. (2019) The Scrum Software Development Process. Available from: https://medium.com/@kunalnegi0003/the-scrum-software-development-process-6e48bb021d9f [Accessed 17 August 2024]

Polymer. (2024) What is a Good Gross Profit Margin?. Available from: https://www.polymersearch.com/blog/profit-margin-by-industry [Accessed 22 August 2024]

Rice, B., Jones, R., & Engel, J. (n.d.) Behavior Driven Development. Available from: https://behave.readthedocs.io/en/stable/philosophy.html#the-gherkin-language [Accessed 13 August 2024]

Schmidt, T.S., Chahin, A., Kößler, J. & Paetzold, K. (2017) Agile development and the constraints of physicality: a network theory-based cause-and-effect analysis. In *DS 87-4 Proceedings of the 21st Int*

ernational Conference on Engineering Design (ICED 17) Vol 4: Design Methods and Tools, Vancouver, Canada, 21-25.08. 2017 (199-208).

Schwaber, K. (1997) Scrum development process. Business object design and implementation. In *10th Annual Conference on Object Oriented Programming Systems, Languages, and Applications Addendum to the Proceedings. ACM/SIGPLAN October* (117-134).

Srivastava, A., Bhardwaj, S. & Saraswat, S. (2017) SCRUM model for agile methodology. In *2017 International Conference on Computing, Communication and Automation (ICCCA)* (864-869). IEEE.

Weichbroth, P. (2022) A case study on implementing agile techniques and practices: Rationale, benefits, barriers and business implications for hardware development. *Applied Sciences*, *12*(17): 8457.

Bibliography

Agile Manifesto (2010) Manifesto for Agile Software Development. *Agile Manifesto*. Available from: https://agilemanifesto.org [Accessed 13 August 2024]

Alshamrani, A. & Bahattab, A. (2015) A Comparison Between Three SDLC Models Waterfall Model, Spiral Model, and Incremental/Iterative Model. *2015 International Journal of Computer Science Issues,* 12(1): 106-111. Available from: https://www.academia.edu/10793943/A_Comparison_Between_Three_SDLC_Models_Waterfall_Model_Spiral_Model_and_Incremental_Iterative_Model?auto=download [Accessed 12 August 2024]

Axelos (2017). Managing Successful Projects with PRINCE2. 6th ed. London: Tso [Accessed 17 August 2024].

Balaji, S. & Murugaiyan, M.S. (2012) Waterfall vs V-Model vs Agile: A Comparative Study on SDLC. *International Journal of Information Technology and Business Management*, 2(1): 26-30. Available from: https://www.studocu.com/row/document/comsats-university-islamabad/software-engineering-concepts/wateerfallvs-v-model-vs-agile-a-comparative-study-on-sdlc/42694547 [Accessed 13 August 2024]

Burek, J. (2024) How to Build a PC: The Ultimate Beginner's Guide. *PCMag.* Available from: https://www.pcmag.com/how-to/how-to-build-a-pc-the-ultimate-beginners-guide [Accessed 17 August 2024]

Cohen, D., Lindvall, M. & Costa, P. (2004) An introduction to agile methods. *Advances in Computers*, 62(3): 1-66.

Figueroa-Lorenzo, S., Añorga, J. & Arrizabalaga, S. (2020) A survey of IIoT protocols: A measure of vulnerability risk analysis based on CVSS. *ACM Computing Surveys (CSUR)*, 53(2): 1-53.

First (2013) CVSS v3.0 Specification Document. *FIRST — Forum of Incident Response and Security Teams*. Available at: https://www.first.org/cvss/specification-document [Accessed 19 August 2024].

Khan, S. & Mahadik, S. (2022) A Comparative Study of Agile and Waterfall Software Development Methodologies. *International Journal of Advanced Research in Science, Communication and Technology*, 2(1): 399-402.

Leung, H.K. & White, L. (1990) A study of integration testing and software regression at the integration level. In *Proceedings. Conference on Software Maintenance 1990,* (290-301). IEEE.

Luna, J., Suri, N. & Krontiris, I., 2012, October. Privacy-by-design based on quantitative threat modeling. In *2012 7th International Conference on Risks and Security of Internet and Systems (CRiSIS),* (1-8). IEEE.

MCS (2023). Quantitative vs. Qualitative: Choosing the Right Approach for Cybersecurity Risk Management and Assessment. *Microminder Cybersecurity*. Available at: https://www.micromindercs.com/blog/quantitative-vs-qualitative-choosing-the-right-approach-for-cyber-security-risk-management-and-assessment [Accessed 19 August 2024].

Micron Technology. (N.D.) How to build your own PC. *Crucial.* Available from: https://www.crucial.com/articles/pc-builders/how-to-build-a-computer [Accessed 17 August 2024]

Visual Paradigm. (N.D.) What is Agile? What is Scrum? *Visual Paradigm.* Available from: https://www.visual-paradigm.com/scrum/what-is-agile-and-scrum/ [Accessed 13 August 2024]