



# **Network Traffic Profiler Dashboard**

Client - Plymouth City Council [Tomek Bergier]

## **Group 15**

Team Leader - Amelia Lee Sze Yai

(write your full name & role pls:) )

Scrum Master - Timothy Birtles

Tomiris

Sofia

# CONTENT PAGE

<b>Network Traffic Profiler Dashboard</b>	<b>1</b>
Client - Plymouth City Council [Tomek Bergier]	1
Group 15	1
CONTENT PAGE	2
IMPORTANT NOTES	2
1) Introduction (Sofia , Amelia)	3
2) Project Management (All members)	3
Project planning material (content from semester 1) (Tomi)	3
Sprint planning and execution (Tim)	3
Risk assessment and mitigation (Sofia)	3
Project schedule (Tomi)	3
Client management (Amelia)	3
3) Detailed System Design (Tim & Tomi)	4
4) Quality Assurance (Sofia & Amelia)	4
5) Conclusion (Tim, Tomi)	4
6) References (3 References Each)	4
7) Appendix	4

## IMPORTANT NOTES

- Each section/subsection should have the names of the member who wrote it.
- Try to distribute according to team contributions (equally if possible, but if not, the ones who did less technical work should contribute more here).
- Recommended page number is around 80 pages. Upper limit is 100 pages (penalty if crossing this). There is no minimum page limit. The key point is to be concise.

## **1) Introduction (Sofia , Amelia)**

– introduce client, problem statement, brief system requirements and any other necessary details on the project

### Client Introduction

The Network Traffic Profiler Dashboard project is being developed for the client, Plymouth City College, with Tomek Bergier as the client representative

### Problem statement & Part of system requirements & Target Users

Modern network security tools are often complex, difficult to configure, and aimed at technical experts. Smaller organisations and non-specialist users may struggle to interpret raw PCAP data or understand network behaviour using existing tools.

This project addresses this gap by creating a lightweight, user-friendly dashboard that allows users to upload PCAP files and receive understandable insights into their network traffic. The system is designed to extract and process meaningful network flow features, apply machine learning to classify traffic activity and detect abnormal behaviour, and present results in an intuitive visual format.

The target users are non-technical staff from small and medium-sized enterprises (SMEs). By combining data processing, machine learning, and visualisation into a single workflow, the dashboard provides actionable insights without requiring advanced networking expertise.

## **2) Project Management (All members)**

Project planning material (content from semester 1) (Tomi)

Sprint planning and execution (Tim)

To ensure development remained aligned with both the initial design requirements and our client's requirements, the Agile methodology was adopted. This approach was selected for its emphasis on adaptive planning and capacity for rapid response to change (Kumar and Bhati, 2012).

Project management began with the creation of a comprehensive product backlog, which was split into Semester One and Semester Two milestones. We operated in biweekly sprints which provided an excellent balance between development and evaluation. During planning sessions, tasks were subdivided based on technical prerequisites and estimated completion time, ensuring the workload assigned to each cycle was realistic.

We utilised Trello as our primary Kanban style project management tool, providing a single location to track the project status. The project's Trello board was organised into columns, 'Backlog', 'In progress', 'Completed', in addition to individual columns for each weekly Scrum.

At the start of each sprint, we met to evaluate and refine the backlog, ensure tasks were still relevant, and delegate responsibilities based on expertise and time availability. Twice-weekly standup meetings were held which focused on progress and pitfalls, ensuring progress was not stalled by technical hurdles nor lack of communication.

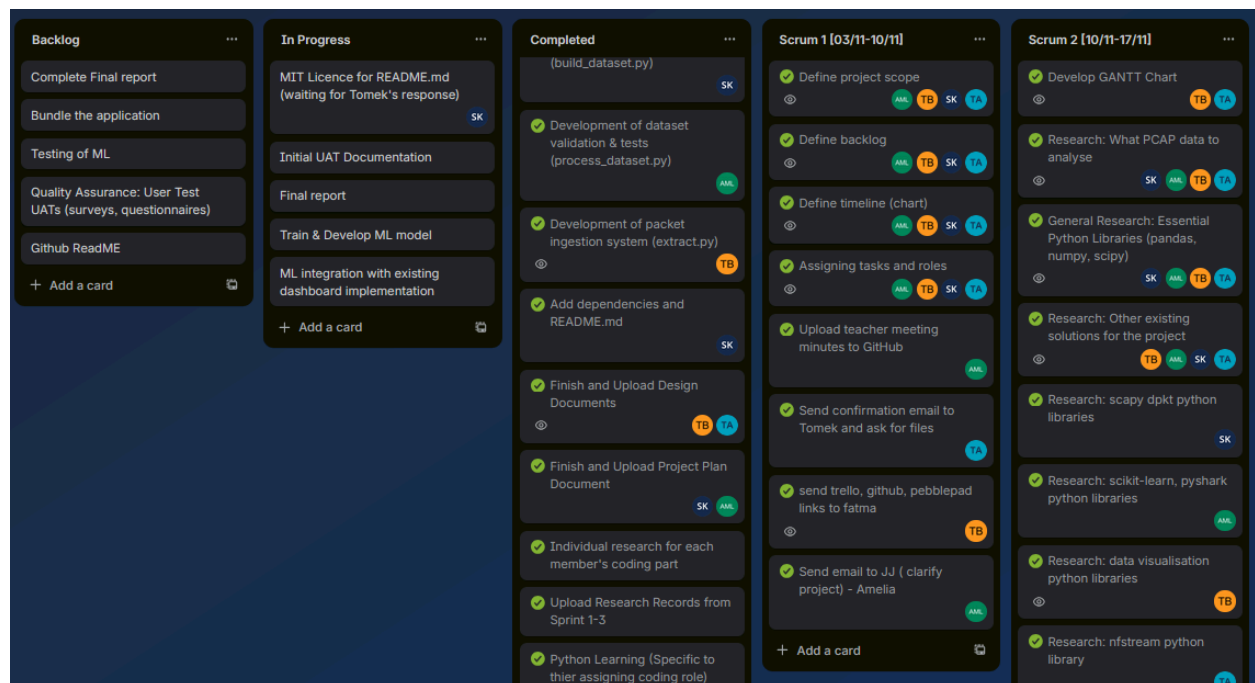


Figure 1 - Trello board snapshot

Risk assessment and mitigation (Sofia)

Project schedule (Tomi)

Client management (Amelia)

Effective client management was a key priority and critical to the success of this project, ensuring continuous alignment with the client's requirements, expectations, and timelines. A strong, collaborative relationship was maintained throughout the project lifecycle through the following methods:

### **1. Communication and Collaboration**

Regular communication channels were established to maintain transparency and collaboration:

- Bi weekly meetings were held every Thursday during the first semester and every Wednesday during the second semester, either in person at Portland Square, Room A03, or via Microsoft Teams. These meetings focused on project timeline and progress, upcoming tasks, short and long term plans, client feedback and product demonstration. All requirements, comments, and feedback were recorded in client meeting minutes.
- Consistent email updates were used to provide documentation, approvals, and follow ups on key decisions.
- Interim dashboard demonstrations were presented to the client for review to ensure alignment with expectations.

### **2. Feedback and Change Management**

Client feedback was actively incorporated into development. This included:

- Adjusting dashboard visualizations based on client feedback
- Refining AI model output presentation according to client preferences
- Prioritising requested features or modifications within sprint cycles

Any suggested changes were formally recorded, evaluated for their impact on timeline and scope, and either approved or deferred based on mutual agreement. Continuous monitoring of deliverables against initial requirements helped manage expectations.

### **3. Documentation and Approval**

All major milestones, including project requirements, design choices and the final product, required client approval. Approvals were documented through emails, signed forms, or recorded meeting minutes. The client's formal sign off also ensured that the delivered dashboard met agreed specifications and expectations.

## **3) Detailed System Design (Tim & Tomi)**

– user story, ERD, prototype screenshots, HIPO chart, state diagrams, DFDs, sequence diagrams flowcharts, UML etc. Screenshots of snippets that you think are important can be placed here as well, but no need for full source code, just link to git repo.

Potential code snippets: Extract pcap data, ML training?

## **4) Quality Assurance (Sofia & Amelia)**

– (post test: surveys & questionnaires), documentation on unit test ,UAT on methodology, results (Sofia)

- post-project support and any other quality assurance measures taken (Amelia)

Ensuring high quality was a priority throughout the development of the Network Traffic Profiler Dashboard. A combination of rigorous testing, structured reviews, and post-project support measures was implemented to maintain reliability, usability, and alignment with the client's requirements.

### Post- project Support

After project completion, the following support measures were planned to ensure smooth adoption and continued functionality to ensure that the client could successfully use and maintain the dashboard:

**Client Handover:** A comprehensive walkthrough of the dashboard was conducted, demonstrating installation, configuration, operation, and troubleshooting procedures.

**Documentation Provision:** Provided the client with up to date README files, user manuals, design diagrams, annotated source code, and testing reports. This ensures that both current users and future developers can understand and extend the system if needed.

**Knowledge Transfer:** Detailed technical explanations and diagrams were shared to enable future developers or IT teams to integrate the dashboard into larger systems or perform further maintenance.

### Other quality assurance measures taken

Several quality assurance practices were applied throughout the project lifecycle to maintain high standards:

Version Control and Peer Code Reviews: GitHub was used to track all code changes. Team members reviewed each other's code to prevent errors and ensure consistent code quality. Clear commit messages and structured branching improved traceability.

Continuous Testing: Beyond unit tests, manual testing of dashboard functions, including uploads, interactive charts, tables, and ML model outputs, was conducted to validate real-world performance and ensure accuracy and robustness.

- Client check in General group review (Every sprint)
- UI/UX Evaluation
- Performance and Data Quality Monitoring

## **5) Conclusion (Tim, Tomi)**

– summarize your project achievements and merits.

## **6) References (3 References Each)**

– any referenced sites/papers you cited within your report. Recommended 10-15 References.

Kumar, G. and Bhati, P.K. (2012) 'Impact of Agile Methodology on Software Development Process,' International Journal of Computer Technology and Electronics Engineeri, 2(4).  
[https://www.researchgate.net/profile/Gaurav-Kumar-175/publication/255707851\\_Impact\\_of\\_Agile\\_Methodology\\_on\\_Software\\_Development\\_Process/links/00b49520489442e12d000000/Impact-of-Agile-Methodology-on-Software-Development-Process.pdf](https://www.researchgate.net/profile/Gaurav-Kumar-175/publication/255707851_Impact_of_Agile_Methodology_on_Software_Development_Process/links/00b49520489442e12d000000/Impact-of-Agile-Methodology-on-Software-Development-Process.pdf).

## **7) Appendix**

– attach the 1000 word CW Submission Evidence and Evaluation file for each member as appendix. This will not count towards the page limit.