

DEVELOPING A MONITORING SOFTWARE SOLUTION FOR USE IN HEALTHCARE RECORDS SYSTEMS

ADAM VIMPANY

1810613@student.uwtsd.ac.uk

Course:
Computing BSc (Hons)
(Software engineering)

University of Wales Trinity St David

ABSTRACT

Healthcare records systems are a complex chain of interconnected systems and applications, the management and maintenance of which can pose a major problem for healthcare organisations. The aim of this project was to plan a software solution that could be used to improve the monitoring of webservices and XML-dripfeeds for NHS Wales WelshPAS integration, and then implement it.



BACKGROUND

It is essential for modern healthcare records systems to be interoperable with a huge range of other software applications and systems. In NHS Wales, the Welsh Patient Administration System (WelshPAS) is a centralised, cloud-based healthcare records system that runs in all seven Welsh health boards and administers millions of patient records systems. It uses interfaces to send and receive over 100,000 messages per day, allowing other applications and systems across Wales to access up-to-date patient data.

PROBLEM

The WelshPAS uses SOAP interfaces to handle both inbound and outbound requests for data from external applications and services. There was limited monitoring of these interfaces. A more effective monitoring system was needed to avoid being reliant on user reporting of problems or downtime.

AIM

This project aimed to design, develop and implement a web application to track the connectivity of the SOAP webservices and XML drip-feed for all servers and health boards across Wales. The application needed to be accessed by integration department staff and must use a dashboard view to display relevant metrics that determine the status of each service.



LITERATURE REVIEW

A literature review found that monitoring architecture could be remote or local; agentless, agent-based passive or agent-based active; and internal (IMA) or external (EMA) [1]. A sophisticated monitoring architecture referred to as "Extended and Adaptive IMA" was identified as the most effective and best performing architecture [1], but would be too time-consuming and complex to complete something similar within the timeframe for the project. The literature review found that the most appropriate architecture would be an internal, remote, agentless IMA, as they had good performance, but were less intrusive than some agent-based approaches [2].

DESIGN

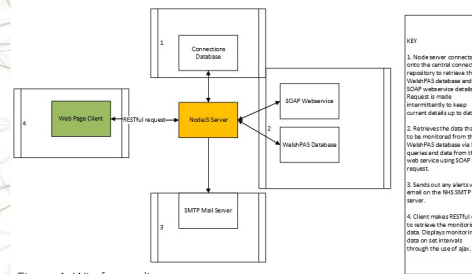


Figure 1: Wireframe diagram

Rapid Application Development (RAD) methodology was used, and requirements for the software were drawn up. It was decided that the application would monitor the WelshPAS database. NodeJS and JavaScript were used to develop the application.

IMPLEMENTATION

The software was developed using NodeJS, as this had the advantage of being able to use an asynchronous approach which prevented blocking of connections during data retrieval.

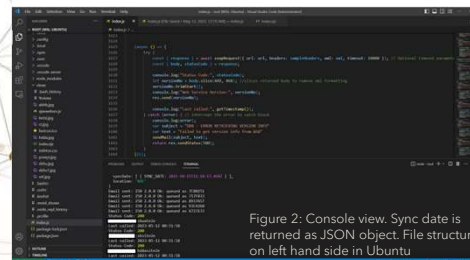


Figure 2: Console view. Sync date is returned as JSON object. File structure on left hand side in Ubuntu

During development, some areas for improvement were identified from user feedback, including a request for email alerting. This functionality was introduced using the NHS SMTP. It was also identified that the creation of a centralised database of connection strings would be highly desirable for use in this project and beyond, and the code was refactored to use the database rather than config files.

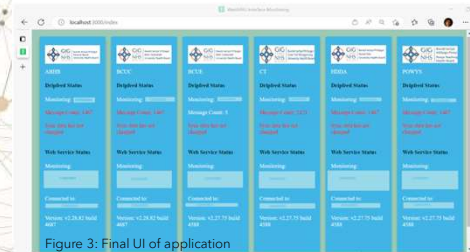


Figure 3: Final UI of application

EVALUATION

The application met the requirements and objectives of this project. RAD methodology was used, therefore evaluation took place concurrently with the user design and construction phase. This resulted in an improved application with additional functionality that was not originally envisioned in the initial development plans.

User evaluation recommended that environmental variables be used to control timings, and this may be implemented in future iterations.

FURTHER WORK

- Investigation of agent-based approaches that could be used to expand functionality of the application
- Identify other metrics that could be used to monitor the interfaces
- Use a frontend framework such as Vue or React
- Develop a dynamic graphical display of the data
- Investigate an appropriate test schedule

REFERENCES

- [1] J. G. A. Jose M. Alcaraz Calero, "Comparative analysis of architectures for monitoring cloud computing infrastructures," Future Generation Computer Systems, vol. 47, pp. 16-30, 2015.
- [2] S. K. T. W. F. D. T. & B. V. Jerico Moeyersons, "Towards cloud-based unobtrusive monitoring in remote multi-vendor environments," Software: Practice and Experience, vol. 52, pp. 427-442, 2022.

