

Investigation of NoSQL Solutions to Welsh Care Record Service Problem

STEPHEN PROTHEROE-HEWITT

DIGITAL HEALTH AND CARE WALES



Research Context and Problem

- Digital Health and Care Wales (DHCW) - Special Health Authority (SHA) offering frontline staff systems to access patient information [1]
- Welsh Care Record Service (WCRS) - Central data repository service storing clinical documentation. Stores binary (like PDF and Word) and XML formats.
- Relational databases – Have storage and service integration [2], but minimal scalability and rigid schemas offer little flexibility [3]. Cost increases due to continuous storage requirements, maintenance, and administration difficulties.



Research Aim and Objectives

Aim

- To analyse and evaluate different NoSQL database solutions for document repository and retrieval

Objectives

- To critically analyse current literature to identify NoSQL solutions
- To assess effectiveness of all available and logical NoSQL solutions
- To compare and evaluate NoSQL solutions
- To create recommendations on proposed solutions



Literature Review

- Relational database management systems (RDBMS) - Primary sources of data storage and retrieval, using SQL as standard [4].
- Organisations use NoSQL databases for storage and retrieval of unstructured and variable data [5]
- **Key-value-based** – Stored by key-value pairs, using strings to represent keys [5]
- **Document-orientated** – Stores and retrieves data via hierarchies [5]
- **Graph-based** – Uses graph structures to query with nodes, edges and edge properties [5]
- **Column-based** – Uses columns to manage aspects like data warehouses [5]



Literature Review Cont.

- **MongoDB** – Open-sourced, document-oriented, and schema-free database, allowing for automatic scalability and high performance [6]
- **Couchbase** – Open-sourced, document-oriented database designed for web and mobile applications for create, read, update and delete (CRUD) operations [7]
- **Neo4j** – Location-type applications use it due to using one spatial geometry type (Point) [8]
- **Redis (REmote DIctionary Server)** – Key-value database providing the highest performance [9]



Literature Review Cont.

- **Data analysis** – Involves developers creating query code, making it more effective [10]
- **Scalability** – Replicas receive requests should nodes fail, increasing robustness [10]
- **Schema flexibility** – RDBMSs have tables with fixed schemas. NoSQL databases do not [10]
- **Deployment speed** – Deployed more effectively in clusters. Automatic replications and sharding configurations improve quickness [10]
- **Data location** – Databases cache remote data, allowing queries in nodes located closer improve efficiency [10]



Literature Review Cont.

- Concerns regarding data security and according to [11], NoSQL solution investigated patient storage solution, but minimal security presented significant issues
- **Script injections** – Document-oriented databases vulnerable to script injections [11]
- **Authorisation** – Key-value-based databases have little authorisation controls [11]
- **Authentication** – Graph-based databases have little to no authentication controls [11]
- **Data encryption** – Column-based databases have little to no data encryption [11]



Research Philosophy and Approach

- **Positivistic philosophy** - Investigates quantitative data based on real-time, first-hand data [12]
- Phenomenological philosophies have better validity but focuses on subjective social data [13]
- Data collated by primary researcher, focusing internally within DHCW
- **Deductive approach** - Focuses on theories and hypotheses [13]
- Theophilus [14] evidenced both approaches achieving quality qualitative data but leading to subjectivity
- For mitigation, interviews and data were cross-checked with documentation for verification



Research Strategies

- Bolinger, Josefy, Stevenson and Hitt [15] mention experimentation an undervalued research strategy
- Di Stefano and Gutierrez [16] agree, as experimentations establish causal relationships and improve validity
- Comparing and evaluating document retrieval times between both systems, testing performances by validating document storage speed and query times.
- Require agreed plans before creation and adapting to minimise threats [17]
- Allows investigation and mitigation of threats and challenges to create value [17]



Research Strategies Cont.

- Involves planning to establish communication and retrieve correct information between systems and testing intensively

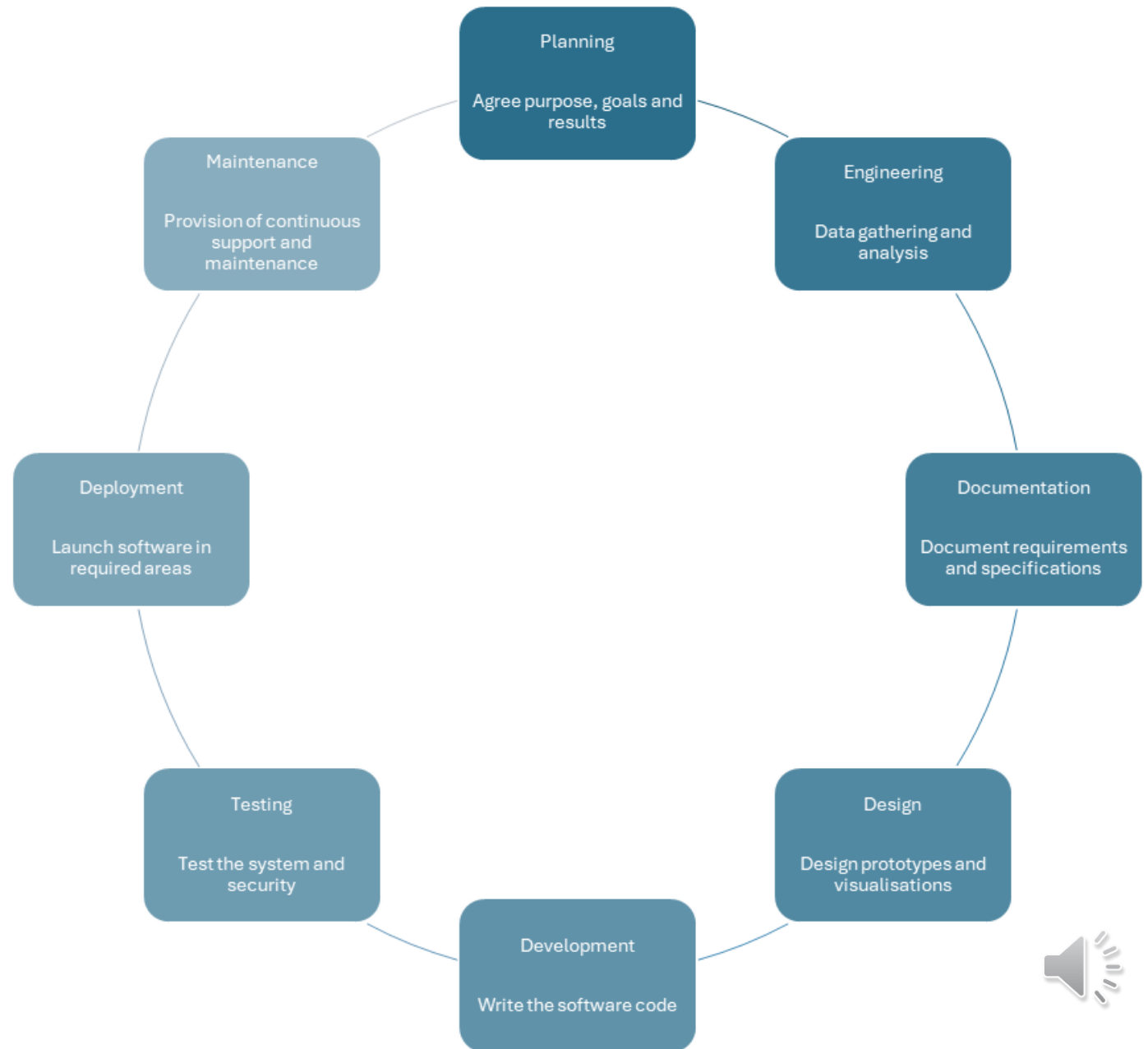


Fig. 2 Software Development Lifecycle [17]

Limitations

- **Tight deadlines** – Teams must collaborate to ensure task completion within required deadlines.
- **Senior management restrictions** – Management may restrict resources but mitigated via persuasion of project plan importance, in line with DHCW's strategic objectives [18].
- **Rigid NHS structure** – Rigid structures and change reluctance could cause inflexibility. According to [19] and [20], inflexibility can cause serious patient harm.
- **Software issues** – Unexpected bugs and errors could breach confidentiality and creating patient data risks but mitigate via key bug and error resolution.



Risks

Risk	Consequences	Current Mitigation	Current Risk			Further Mitigation	Future Risk		
			C	L	R		C	L	R
If the deadline cannot be met	The project will go over budget and risk collapsing	Ensure regular project meetings and collaboration between required teams	3	3	9	Ensure specific individual/team tasks meet deadlines	2	2	4
If senior managers/directors do not provide authorisation	The project cannot proceed	Provide a benefits realisation and justification for the project to proceed	4	3	12	Provide regular project updates and assurances	3	2	6
If hardware and software issues occur	The deadline will not be met	Ensure required hardware specifications and software are met	3	3	9	Plan and agree the recommended specifications required beforehand	1	3	3
If insufficient team resources are available	The project cannot proceed	Agree dedicated individual/team resources for the project	4	2	8	Continuously check and agree dedicated resource availability	2	1	2

Fig. 3 Project Risk Table

Ethical Issues

- **Patient confidentiality and security** – Patient data required for information flow confirmation between systems are successful
- Development and test versions have test patient data, but production version requires real patient data
- If confidentiality breaches occur or security vulnerabilities exploited, reputational harm could occur to researcher and DHCW
- Mitigations include closing vulnerabilities being closed and authorisation measures like access controls in place [21]



Validity, Reliability and Generalisability

- Low validity due to positivistic research philosophy
- According to [22] and [23], the Danish Fracture Database (DFDB) had high and consistent validity between 82-100% in 2013 and 2023
- High reliability due to primary data retrieval by researcher
- If different researcher investigated same problem, they would get comparable results
- Suitable for similar problems in similar organisations, benefitting other public sector organisations
- Driver and Vehicle Licensing Agency (DVLA) use MongoDB to store and retrieve documentation



Data Collection Methods

- **Documentation** – Check current processes for documentation storage and retrieval in WCRS
- **Surveys** – Collected within DHCW due to knowledge base, focusing on respondents using SQL
- **Statistical Collation** – Collected on current and new processes, comparing, analysing and evaluating findings
 - Provides minimum datasets and comparative baselines between SQL and NoSQL databases



Data Evaluation Methods

- **Descriptive statistical analysis** – Core data from SQL and NoSQL databases, with data summarised using central tendencies
- **Exploratory statistical analysis** – Uses experiment research strategy to create hypotheses, with 0.05 statistical significance level due to 95% accuracy generalisation for researchers [24]
 - H_0 – No significant differences in performance between SQL and NoSQL databases
 - H_1 – Significant differences in performance between SQL and NoSQL databases
- Presented via data visualisation techniques with plain English used, avoiding technical jargon [25]



Development Methodologies

- Feature-driven development (FDD) agile methodology is easy-to-use for larger projects [26]
- DHCW teams help with continuous database design, development, testing and release
- Helps with design and implementation via feature-based approach



Development Methodology Cont.

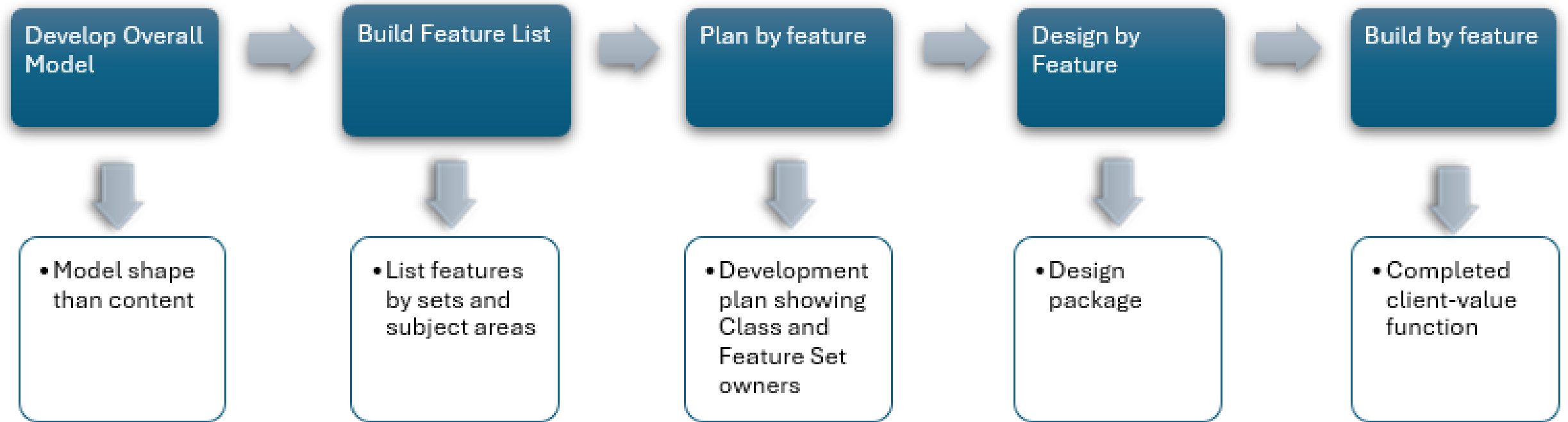


Fig. 4 FDD Process Model [27]



Development Methodology Cont.

Method	Key Points	Strengths	Weaknesses
ASD	Adaptive culture, collaboration, mission driven component based iterative development	Organisations seen as adaptive systems, creating emergent order out of web of interconnected individuals	ASD more about concepts and culture than software practices
DSDM	Application of controls to RAD, use of time boxing and empowered DSDM teams	First agile software method, using prototyping, has several user roles like 'ambassador' and 'visionary'	Only consortium members have access to white papers dealing with actual use of the method
XP	Customer drive development, small teams, daily builds	Refactoring – ongoing system redesign to improve performance and responsiveness to change	While individual practices suitable for many situations, overall views and management practices are given less attention
SCRUM	Independent, small, self-organising development teams, 30-day release cycles	Enforces paradigm shift from 'defined and repeatable' to 'new product development view of Scrum'	While Scrum details specifically how to manage 30-day release cycles, integration and acceptance tests are not detailed
FDD	Five-step process, object-orientated component (i.e. feature) based development	Method simplicity, design and implement the system by features, object modelling	Focuses only on design and implementation, needing other supporting approaches

Fig. 5 Agile Model Features and Comparisons [26]

Development Methodology Cont.

- **Scrums** – Considered most successful framework [28], focuses on smaller tasks, ensuring task completion within specified periods
- **Sprints** – Assists with smaller task completion for individual members and teams by specified deadlines
- **Daily Stand-ups** – Helps understand previous day's work and what will happen on current day



Resources and Materials

- **Integrated databases** – Must have seamless integration without affecting performance or efficiency
- **Dedicated server(s)** – Optimises database performance based on minimum requirements
- **Microsoft SQL Server Management Studio** – Used within WCRS managing all databases
- **Chosen NoSQL database** – Some NoSQL databases open-sourced with free download
- **Tableau** – Used to present data, though a cost involved if required for longer than 14 days



Resources and Materials Cont.

- Freedom of Information (FOI) requests made to DVLA, South Wales Police and Welsh Government
- Information can establish what other public organisations use, draw conclusions and recommendations for project progression
- Requires NOD team and other DHCW teams for development and support
- Senior management team to provide required permissions and approvals
- Test team available to perform internal testing
- End-to-end user testing performed externally by users in health boards



Project Plan

- Planned outcome – A NoSQL database designed, built and implemented for WCRS requirements, including contingency planning if task do not meet deadlines
- Expected duration – Six months, though planning and scoping can occur before design and implementation
- Lessons learnt document created upon completion to document what went well, what went wrong, and lessons learnt to mitigate risks, limitations and/or issues that occurred



Project Gantt Chart

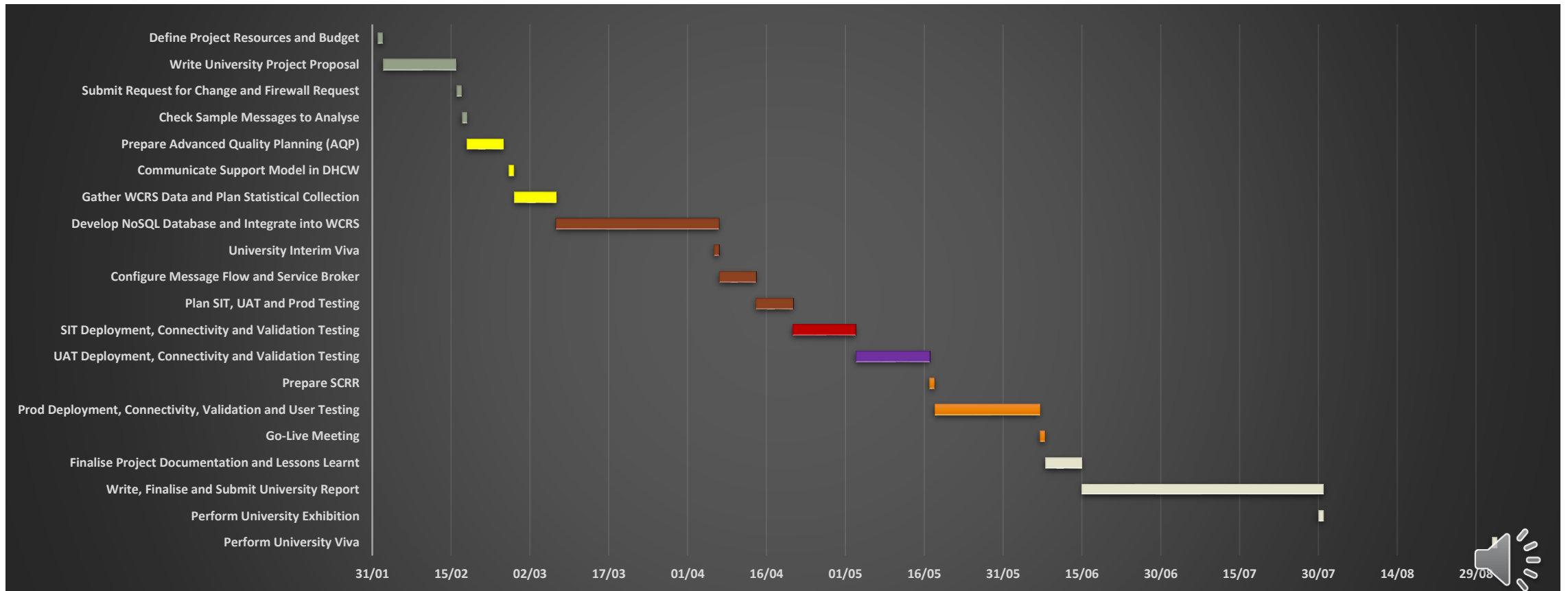


Fig. 6 Project Gantt Chart

Task Lists

Task Item	Task Name	Allocation	Predecessors	Resource Initials
1	Investigate and Implement No SQL Solution in WCRS	1,379 hrs (184 day equivalent)		
1.1	Initiation & Prerequisites	119.5 hrs		
1.1.1	Project resources defined and engaged [INSE, NOD, WCP, DCS, other DHCW staff]	2 hrs		SPH
1.1.2	Agree project budget	0.5 hrs		SPH, NOD
1.1.3	<i>Meeting: Project Initiation call</i>	1 hr	4, 5	SPH
1.1.4	Write university project proposal	105 hrs		SPH
1.1.5	Communicate project scope, justification, deliverables, roles and responsibilities	1 hr	4, 5	SPH
1.1.6	Submit Request for Change	1 hr	4, 5	SPH
1.1.7	Submit DHCW firewall request (SIT)	1 hr		NOD
1.1.8	Check sample messages provided to NOD	0.5 hrs	10	NOD
1.1.9	Perform preliminary analysis of sample message	7.5 hrs	10, 11	NOD
1.1.10	<i>All pre-requisites completed</i>	0 hrs	4, 5, 6, 7, 8, 9, 10, 11, 12	

Fig. 7 Initiation and Prerequisite Task List



Task Lists Cont.

Task Item	Task Name	Allocation	Predecessors	Resource Initials
1	Investigate and Implement No SQL Solution in WCRS	1,379 hrs (184 day equivalent)		
1.2	Planning	137 hrs		
1.2.1	Miscellaneous project management (PM) activity	3.75 hrs	13	SPH
1.2.2	Prepare Advanced Quality Planning (AQP)	52.5 hrs	13	SPH
1.2.3	Agree and formalise proposed support model with Service Management	3.75 hrs	16	SPH
1.2.4	Communicate support model to DHCW	7.5 hrs	17	SPH
1.2.5	<i>Meeting: Present AQP to WIAG</i>	<i>1 hr</i>	<i>15</i>	<i>SPH</i>
1.2.6	Gather data in current WCRS for analysis	30 hrs	15, 19	
1.2.6.1	Prepare survey questions	15 hrs		SPH
1.2.6.2	Finalise survey questions	3.75 hrs	21	SPH
1.2.6.3	Send survey	3.75 hrs	21	SPH
1.2.6.4	Analyse survey responses	7.5 hrs	21	SPH
1.2.6.5	Plan statistical collection of WCRS data	30 hrs	15, 19	
1.2.6.5.1	Confirm WCRS statistical requirements	7.5 hrs		SPH, NOD
1.2.6.5.2	Collate data and statistics	7.5 hrs	26	SPH
1.2.6.5.3	Analyse statistics	15 hrs	26	SPH
1.2.7	Raise ServicePoint calls for NOD/INSE at DEV stage	3.75 hrs	13	SPH
1.2.8	Confirm WCRS test data	3.75 hrs	29	SPH
1.2.9	Communicate SIT endpoint DNS URL, TCP/IP address & port to DHCW	1 hr	13	SPH
1.2.10	<i>Planning phase completed</i>	<i>0 hrs</i>	<i>15, 16, 17, 18, 19, 20, 29, 30, 31</i>	



Fig. 8 Planning Task List

Task Lists Cont.

Task Item	Task Name	Allocation	Predecessors	Resource Initials
1	Investigate and Implement No SQL Solution in WCRS	1,379 hrs (184 day equivalent)		
1.3	Development	351.75 hrs		
1.3.1	Miscellaneous PM activity	3.75 hrs	32	SPH
1.3.2	Create/Develop database with chosen NoSQL solution	116.25 hrs	32	SPH, NOD, DCS
1.3.3	Integrate into WCRS	116.25 hrs	32, 35	SPH, NOD, DCS
1.3.4	Configure Fiorano interface/message flow	26.25 hrs	32, 35	INSE
1.3.5	Configure Service Broker shredding process	26.25 hrs	32, 35	NOD
1.3.6	Test planning	52.5 hrs	32, 35	TEST
1.3.7	Request firewall rule changes for SIT and testing	3 hrs	32, 36	INSE, NOD
1.3.8	University interim viva	7.5 hrs	32, 35, 36, 37, 38, 39, 40	SPH
1.3.9	Development phase completed	0 hrs	34, 35, 36, 37, 38, 39, 40, 41	

Fig. 9 Development Task List



Task Lists Cont.

Task Item	Task Name	Allocation	Predecessors	Resource Initials
1	Investigate and Implement No SQL Solution in WCRS	1,379 hrs (184 day equivalent)		
1.4	SIT Development & Testing	108.75 hrs		
1.4.1	Miscellaneous PM activity	3.75 hrs	42	SPH
1.4.2	Implement new WCRS NoSQL database	52.5 hrs	42	SPH, NOD, DCS
1.4.3	Deploy interface/message flow	3.75 hrs	42, 45	INSE
1.4.4	Configure Service Broker shredding process	15 hrs	42	NOD
1.4.5	Testing of new NoSQL database	22.5 hrs	42, 45, 46, 47	
1.4.5.1	Testing - Fiorano testing	7.5 hrs	42, 45, 46, 47	TEST
1.4.5.2	Testing - Service Broker testing	7.5 hrs	42, 45, 46, 47	TEST
1.4.5.3	Testing - WCRS messaging and data flow	7.5 hrs	42, 45, 46, 47	TEST, NOD
1.4.6	Connectivity testing	7.5 hrs	42, 45	SPH, INSE, NOD
1.4.7	Request firewall rule changes for UAT	3.75 hrs		INSE, NOD
1.4.8	SIT phase completed	0 hrs	44, 45, 46, 47, 48, 52, 53	

Fig. 10 SIT Development and Testing Task List



Task Lists Cont.

Task Item	Task Name	Allocation	Predecessors	Resource Initials
1	Investigate and Implement No SQL Solution in WCRS	1,379 hrs (184 day equivalent)		
1.5	UAT Development & Validation	105 hrs		
1.5.1	Miscellaneous PM activity	7.5 hrs	54	SPH
1.5.2	Fiorano deployment	3.75 hrs	54	INSE
1.5.3	Service Broker deployment	14 hrs	54	NOD
1.5.4	WCRS - messaging and data flow	6 hrs	54	NOD
1.5.5	Connectivity testing	30 hrs	56, 57, 58	SPH, INSE, NOD
1.5.6	Validation Testing	40 hrs	56, 57, 58	VAL
1.5.7	Request firewall rule changes for PROD	3.75 hrs		INSE, NOD
1.5.8	Validation testing signed off and certificate obtained	0 hrs	61	
1.5.9	UAT phase completed	0 hrs	56, 57, 58, 59, 60, 61, 62	

Fig. 11 UAT Development and Testing Task List



Task Lists Cont.

Task Item	Task Name	Allocation	Predecessors	Resource Initials
1	Investigate and Implement No SQL Solution in WCRS	1,379 hrs (184 day equivalent)		
1.6	PROD Development & Validation	144.5 hrs		
1.6.1	Miscellaneous PM activity	3.75 hrs	64	SPH
1.6.2	Agree Go-Live date	3.75 hrs	64	SPH, NOD, INSE
1.6.3	Prepare SCRR (What is this?) and submit to WIAG (Check WIAG acronym)	7.5 hrs	64	SPH
1.6.4	Meeting: Present SCRR to WIAG	2 hr	68	SPH
1.6.5	Deploy interface/message flow	7.5 hrs	64	INSE
1.6.6	Configure Service Broker shredding process	37.5 hrs	64	NOD
1.6.7	WCRS - messaging and data flow	37.5 hrs	64	NOD
1.6.8	Validation Testing	37.5 hrs	64, 70, 71, 72	VAL
1.6.9	End-to-end test to view result in WCP	7.5 hrs	64, 70, 71, 72, 73	WCP, INSE, NOD, SPH
1.6.10	SCRR sign-off	0 hrs	68, 69	
1.6.11	Ready for Go-Live	0 hrs	68, 69, 70, 71, 72, 73, 74, 75	
1.7	Go-Live	7.5 hrs		
1.7.1	Meeting: Real-time go-live	7.5 hrs	75, 76	WCP, INSE, NOD, SPH

Fig. 12 Production Development and Validation Task List



Task Lists Cont.

Task Item	Task Name	Allocation	Predecessors	Resource Initials
1	Investigate and Implement No SQL Solution in WCRS	1,379 hrs (184 day equivalent)		
1.8	Project Completion	405 hrs		
1.8.1	Finalise project documentation and lessons learnt	45 hrs	75, 76, 78	SPH
1.8.2	Write and finalise university report	344.5 hrs	75, 76, 78, 80	SPH
1.8.3	Submit university report	0.5 hrs	81	SPH
1.8.4	University Exhibition	7.5 hrs	81	SPH
1.8.5	University Viva	7.5 hrs	81, 83	SPH
1.8.6	Project completion	0 hrs	13, 32, 42, 54, 64, 75, 76, 78, 80	

Fig. 13 Project Completion Task List



References

- [1] Digital Health and Care Wales, “About Digital Health and Care Wales”, Digital Health and Care Wales [Online] Available: <https://dhcw.nhs.wales/about-us/> [Accessed Jun. 30, 2024].
- [2] Microsoft, “Compare Options for Storing Blobs (SQL Server)”, Microsoft, Feb. 28, 2023 [Online] Available: <https://learn.microsoft.com/en-us/sql/relational-databases/blob/compare-options-for-storing-blobs-sql-server?view=sql-server-ver16#Expectations> [Accessed Aug. 20, 2024].
- [3] R. Sheldon, “How to choose between SQL and NoSQL databases,” Redgate, Apr. 13, 2021 [Online] Available: <https://www.red-gate.com/simple-talk/featured/how-to-choose-between-sql-and-nosql-databases/> [Accessed Aug. 20, 2024].
- [4] A. Bajaj and W. Bick, “The Rise of NoSQL Systems: Research and Pedagogy”, Journal of database management, vol. 31, no. 3, pp. 67-82, Jul. 2020 [Online] Available: [10.4018/JDM.2020070104](https://doi.org/10.4018/JDM.2020070104) [Accessed Jul. 2, 2024].
- [5] H. Wang and S. Wang, “Teaching NoSQL Databases in a Database Course for Business Students”, Journal of information systems education, vol. 34, no. 1, pp. 32-40, 2023 [Online] Available: <https://www.proquest.com/scholarly-journals/teaching-nosql-databases-database-course-business/docview/2774977620/se-2> [Accessed Jul. 2, 2024].
- [6] C. A. Győrödi, D. V. Dumșe-Burescu, D. R. Zmaranda and R. S. Győrödi, “A Comparative Study of MongoDB and Document-Based MySQL for Big Data Application Data Management”, vol. 6, no. 2, p. 49, Jun. 2022 [Online] Available: [10.3390/bdcc6020049](https://doi.org/10.3390/bdcc6020049) [Accessed Jul. 9, 2024].



References Cont.

- [7] I. Carvalho, F. Sá and J. Bernardino, “Performance Evaluation of NoSQL Document Databases: Couchbase, CouchDB, and MongoDB”, Algorithms, vol. 16, no. 2, p. 78, Feb. 2023 [Online] Available: [10.3390/a16020078](https://doi.org/10.3390/a16020078) [Accessed Jul. 9, 2024].
- [8] D. Guo and E. Onstein, “State-of-the-Art Geospatial Information Processing in NoSQL Databases”, ISPRS international journal of geo-information, vol. 9, no. 5, p. 331, May 2020 [Online] Available: [10.3390/ijgi9050331](https://doi.org/10.3390/ijgi9050331) [Accessed Jul. 9, 2024].
- [9] G. Belalem, H. Matallah and K. Bouamrane, “Evaluation of NoSQL Databases: MongoDB, Cassandra, HBase, Redis, Couchbase, OrientDB”, International journal of software science and computational intelligence, vol. 12, no. 4, pp. 71-91, Oct. 2020 [Online] Available: [10.4018/IJSSCI.2020100105](https://doi.org/10.4018/IJSSCI.2020100105) [Accessed Jul. 9, 2024].
- [10] A. Corbellini, C. Mateos, A. Zunino, D. Godoy and S. Schiaffino, “Persisting big-data: The NoSQL landscape”, Information systems (Oxford), vol. 63, pp. 1-23, Jan. 2017. [Online] Available: [10.1016/j.is.2016.07.009](https://doi.org/10.1016/j.is.2016.07.009) [Accessed Jul. 9, 2024].
- [11] K. A. ElDahshan, A. A. AlHabshy and G. E. Abutaleb, “Data in the time of COVID-19: a general methodology to select and secure a NoSQL DBMS for medical data”, PeerJ. Computer science, vol. 6, p. 297, Sept. 10, 2020 [Online] Available: [10.7717/peerj-cs.297](https://doi.org/10.7717/peerj-cs.297) [Accessed Jul. 9, 2024].
- [12] G. Ryan, “Introduction to positivism, interpretivism and critical theory”, Nurse researcher, vol. 25, no. 4, pp. 14-20, Mar. 2018 [Online] Available: <https://doi.org/10.7748/nr.2018.e1466> [Accessed Jul. 16, 2024].



References Cont.

- [13] S. Joseph, Research Philosophies, Strategies & Methodologies, Swansea: University of Wales Trinity St. Davids, 2024.
- [14] T. Azungah, "Qualitative research: deductive and inductive approaches to data analysis", Qualitative research journal, vol. 18, no. 4, pp. 383-400, Nov. 2018 [Online] Available: [10.1108/QRJ-D-18-00035](https://doi.org/10.1108/QRJ-D-18-00035) [Accessed Jul. 16, 2024].
- [15] M. T. Bolinger, M. A. Josefy, R. Stevenson and M. A. Hitt, "Experiments in Strategy Research: A Critical Review and Future Research Opportunities", vol. 48, no. 1, pp. 77-113, Jan. 2022 [Online] Available: [10.1177/01492063211044416](https://doi.org/10.1177/01492063211044416) [Accessed Jul. 16, 2024].
- [16] G. Di Stefano and C. Gutierrez, "Under a magnifying glass: On the use of experiments in strategy research", Strategic organization, vol. 17, no. 4, pp. 405-507, Nov. 2019 [Online] Available: [10.1177/1476127018803840](https://doi.org/10.1177/1476127018803840) [Accessed Jul. 16, 2024].
- [17] T. Sire, "The Ultimate Software Development Strategy for Successful Development", Pulsion, Jul. 10, 2024 [Online] Available: <https://www.pulsion.co.uk/blog/software-development-strategy/> [Accessed Jul. 16, 2024].
- [18] Digital Health and Care Wales, "Our Strategic Objectives", Digital Health and Care Wales [Online] Available: <https://dhcw.nhs.wales/about-us/key-documents/strategies/organisational-strategy-2024-2030/our-strategic-objectives/> [Accessed Sept. 2, 2024].



References Cont.

- [19] N. Edwards and S. Black, “Targets: unintended and unanticipated effects”, BMJ quality & safety, vol. 32, no. 12, pp. 697-699, Sept. 2023 [Online] Available: [10.1136/bmjqs-2023-016247](https://doi.org/10.1136/bmjqs-2023-016247) [Accessed Jul. 23, 2024].
- [20] C. Kanaris, “Mind the power gap: How hierarchical leadership in healthcare is a risk to patient safety”, Journal of child health care, vol. 27, no. 3, pp. 319-322, Sept. 2023 [Online] Available: [10.1177/13674935231196197](https://doi.org/10.1177/13674935231196197) [Accessed Jul. 23, 2024].
- [21] A. Mohammad, “Distributed Authentication and Authorization Models in Cloud Computing Systems: A Literature Review”, Journal of cybersecurity and privacy, vol. 2, no. 1, pp. 107-123, Mar. 2022 [Online] Available: [10.3390/jcp2010008](https://doi.org/10.3390/jcp2010008) [Accessed Aug. 20, 2024].
- [22] K. Gromov, J. V. Fristed, M. Brix and A. Troelsen, “Completeness and Data Validity for the Danish Fracture Database”, Danish Medical Journal, vol. 60, no. 10, p. A4712, Oct. 2013 [Online] Available: https://content.ugeskriftet.dk/sites/default/files/scientific_article_files/2018-12/a4712.pdf [Accessed Jul. 23, 2024].
- [23] A. B. Roennegaard, P. H. Gundtoft, P. T. Tengberg and B. Viberg, “Completeness and validity of the Danish fracture database”, Injury, vol. 54, no. 10, p. 110769, Oct. 2023 [Online] Available: [10.1016/j.injury.2023.05.001](https://doi.org/10.1016/j.injury.2023.05.001) [Accessed Jul. 23, 2024].
- [24] S. Tenny and I. Abdelgawad, “Statistical Significance”, National Library of Medicine, Nov. 23, 2023 [Online] Available: <https://www.ncbi.nlm.nih.gov/books/NBK459346/> [Accessed Jul. 30, 2024].



References Cont.

- [25] B. Shander, “Learning Data Visualization”, Linkedin Learning, Carpenteria, 2022 [Online] Available: <https://www.linkedin.com/learning/learning-data-visualization-15572314/why-visualize-data?resume=false&u=77966930> [Accessed Jul. 30, 2024].
- [26] C. M. Budoya, M. M. Kissaka and J. S. Mtebe, “Instructional Design Enabled Agile Method Using ADDIE Model and Feature Driven Development Process”, International journal of education and development using information and communication technology, vol. 15, no. 1, pp. 35-54, 2019 [Online] Available: <https://research-ebsco-com.ezproxy.uwtsd.ac.uk/c/onvpsy/viewer/pdf/r5kzcfjgz> [Accessed Aug. 6, 2024].
- [27] S. Aftab, Z. Nawaz, F. Anwer, M. Ahmad, A. Iqbal, A. Ahmad and M. Salman, “Using FDD for Small Project: An Empirical Case Study”, International journal of advanced computer science & applications, vol. 10, no. 3, 2019 [Online] Available: [10.14569/IJACSA.2019.0100319](https://doi.org/10.14569/IJACSA.2019.0100319) [Accessed Aug. 6, 2024].
- [28] P. Ashun and T. R. Brown, “Agile project management,” PACKT Publishing, Birmingham, 2022 [Online] Available: <https://video.alexanderstreet.com/watch/agile-project-management> [Accessed Aug. 30, 2024].



YouTube Link

The YouTube link for this presentation is as follows:

<https://youtu.be/SEN2XO3HFWk>