Having worked within the Single Record department of Digital Health and Care Wales for a number of years, I have experienced the migration from using the traditional Waterfall methodology to a more robust Agile framework, while maintaining active testing projects for applications such as the:

* Welsh Clinical Portal (WCP)
* Welsh Admin Portal (WAP)
* Welsh Clinical Worklist Manager (WCWM)
* Other supported utilities, including Health related web forms for the recording of clinical data and many connected web services.

Due to the large scope of the individual projects we maintain, and their complexities, the transition to an Agile framework raised many challenges and solutions to our current testing practices, which included the need for modernisation due to increased delivery demand and limited available resource and time within the project. By adopting the Agile process, including the use of daily stand-ups, sprint reviews, retrospectives and visual management through Kanban boards in Azure DevOps, while developing and promoting a Gherkin driven, automation first testing framework and strategy, our team has become far more adaptable to change which has helped to alleviate cultural resistance and testing bottlenecks, and help create a more collaborative team approach to ensure knowledge and skill sharing. Through the course of this essay, I will look to describe the Agile processes we utilise within the Welsh Clinical Portal project, as one of our more mature products, detailing key challenges raised during the implementation and the subsequent solutions used to help resolve them, with a final reflection on experiences gained while transitioning to the adopted Agile framework and the lessons learned along the way.

As a brief history, the Welsh Clinical Portal is a mature and highly complex application, that has been used to record clinical data for patients within NHS Wales for over fifteen years. The data recorded can range between:

* Patient demographical data, including personal details such as NHS number, name, date of birth, address, next of kin, etc.
* Clinical recorded data, including pathology test requesting and results, and patient clinical documents, such as medical reports and record documents.
* Patient administration tasks, including the ability to admit and transfer patients into hospital wards, and generate follow up tasks for document records by communicating with the Welsh Clinical Worklist Manager application and others.

Prior to adopting the Agile framework, the project followed a Waterfall development model, which according to *https://en.wikipedia.org/wiki/Waterfall\_model,* is a structured approach to software development that consists of linear, sequential phases. The model provides easily identifiable milestones and is easy to understand. It tends to be less iterative and flexible compared to other approaches.

The key characteristics of the waterfall model:

* Linear progression: Each phase depends on the deliverables of the previous one, flowing in a linear direction.
* Sequential phases: Phases include conception, initiation, analysis, design, implementation, testing, and deployment.
* Documentation emphasis: The model places importance on documentation, such as requirements documents, design documents, and source code
* Testing at the end: Testing typically occurs only at the end of the process, which can be "risky" and invite failure, as noted in 1970 by Winston W. Royce.

The waterfall model is often used in areas where it is typical to follow a linear progression, such as engineering design. However, it has limitations in terms of flexibility and adaptability compared to other approaches which can cause bottlenecks within the development, and delays to delivery timescales, which aided in the decision of transitioning to a more agile framework, especially with an increased delivery schedule and expanding project portfolio.

The adoption of an Agile framework aligns with the organisational approach within Digital Health and Care Wales (DHCW), who play a pivotal role in modernising digital infrastructure to make health and care services more accessible, sustainable, and data informed. The organisation adopted agile software development as a response to the evolving demands of healthcare delivery, favouring an iterative and collaborative model over the rigidity of traditional linear methodology.

The Agile Manifesto was developed in Utah 2001, which according to *https://agilemanifesto.org/* was in response to the rigid development processes being used at the time, and is based on the ideology of delivering small, functional increments of a system iteratively. Agile software development encourages adaptability, close stakeholder engagement, continuous improvement, and values the following twelve principles:

* Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
* Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
* Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
* Business teams, developers and testers must work closely throughout the project.
* Build projects around motivated individuals, and trust them to do their job, while providing the required environments and support.
* The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
* Working software is the primary measure of progress. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
* Continuous attention to technical excellence and good design enhances agility.
* Simplicity is the art of maximizing the amount of work not done and is essential.
* The best architectures, requirements, and designs emerge from self-organizing teams.
* At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behaviour accordingly.

Agile software development allows for early and frequent delivery of working features, enabling real-time testing and stakeholder feedback. Collaboration between all areas of the development team, including product owners, business analysts, developers, testers, clinicians, and end users, ensures full engagement throughout the process, to develop systems and services which fully align with clinical requirements and patient outcomes. Moreover, the ability to incorporate changes at later stages of development reduces risk, supports continuous development, to deliver a finalised product within strict delivery schedules, whilst also maintaining a high level of quality assurance. This provides a better alignment with user requirements and highlights and why it is particularly well suited within the delivery of healthcare services.

However, the agile approach also introduces challenges within the context of health and care, as continuous delivery can make it difficult to predict costs, resource allocation and timelines, which can run the risk of projects becoming open ended without careful scope management, and technical debt may accumulate across sprints especially where legacy systems and practices are used. These risks can also cause a level of resistance from legacy systems and/or teams unfamiliar with agile practices, which can result in poor collaboration and make testing more complicated, due to evolving requirements and a lack of stable test environments.

With the adoption of the Agile framework, the role of our test team has undergone significant transformation, where we are fully integrated within the development process and collaborate closely with product owners, business analysts and developers, to deliver a high level of quality assurance to all aspects of the development process. The following responsibilities have since been fully adopted within our team to ensure the Agile principles are upheld:

* Contribution to the definition of acceptance criteria within the project, which can involve collaboration to ensure the project requirements are fully met, while maintaining any DHCW UI/Coding standards
* Attending Sprint backlog, planning, retrospective, and scrum sessions, to ensure a constant dialogue with all development parties is maintained.
* Testing is carried out at the earliest opportunity, ensuring all manual and technical aspects of testing can be frequently run and easily maintained, allowing for immediate, real-time feedback on the project status.

This requires a tester to adopt a similar agile mindset and being more aligned with the idea of quality improvement and focus on providing safe, controlled, and repeatable ways to ensure quality of a developed application, ensuring it also conforms to the requirements of the product design and organisational strategy. Testers play a supportive role by assisting with the clarification of requirements, ensuring testability, and promoting effective communication, collaboration and skill sharing among team members, which is especially critical in healthcare settings where patient safety, regulatory compliance, and interoperability are paramount.

Within the Single Record test team, two major challenges faced while adopting the agile framework within our projects were:

1. Maintaining a high quality of assurance while adopting a more frequent delivery schedule, including other Single Record products, has been a massive challenge due to the size and complexity of the services provided, where manual testing of each application can be within the following ranges:
   1. The Welsh Clinical Portal can take between four to six weeks testing.
   2. The Welsh Admin Portal can take between one to two weeks testing.
   3. The Welsh Clinical Worklist Manager can take between two to three weeks testing.
   4. Each webform can take between one to two weeks testing, with over forty forms actively supported.

The revised delivery schedule has seen deployment of each version of the Welsh Clinical Portal application increased to be on a quarterly basis, as opposed to the previous bi-annual schedule. This creates further challenges on how we can maintain other projects within our portfolio that require further testing.

1. Increasing efficiency within our testing processes, by developing and implementing tools and frameworks that can be used to assist with the adoption of an increased development schedule, proved to a be another major challenge due to legacy practices and lack of testing resource, technical skills, and knowledge within the team.

To overcome these challenges, we implemented the following practices and processes, which in turn have allowed us to maintain a high level of quality assurance and test efficiency with the increased delivery schedule:

* We adopted Gherkin/Cucumber syntax within all our test scripts to ensure efficiency in both writing, running, and maintaining our test cases. This has allowed a custom level of granularity within our test case steps which has seen a reduction in the step count of our test cases aiding to the efficiency improvement, by spending more time testing applications efficiently as opposed to forever writing and/or editing test cases.
* We have implemented an automation first strategy where possible, with an automation framework that can understand the gherkin syntax developed within our test cases. This also included ensuring there is sufficient support in place to train and advise testers where possible on any technical elements of testing.
* The automation strategy allowed us to also implement a more focussed manual testing approach to cater for the increased demand, where automation is used to test the UI and flow of an application and manual efforts are focussed towards more technical testing of back-end services and APIs.
* We implemented a shared documentation process, allowing the whole development team access to our test cases, reports, and automation code to promote further collaboration and aid continuous improvement within our own team’s practices, processes, and frameworks.

By introducing these processes and practices within the smaller projects at first, we were able to decrease the time taken to test the Welsh Admin Portal, Welsh Clinical Worklist Manager and the majority of our supported web forms to within 7 days, which helped drastically improve the overall efficiency within each of these projects by fifty percent. By further implementing these strategies within our legacy projects, we have also manged to decrease the time taken to complete testing against the Welsh Clinical Portal application to within three weeks, ensuring we are able to complete testing and deliver the project according to the quarterly schedule.

When adopting the agile framework within our project, the requirement for all development teams responsible for a specific project to attend daily stand ups, sprint planning, retrospective, and backlog refinement meetings assisted in promoting and harnessing a fully collaborative approach within the entire department, and this by far has had the most significant impact overall on our entire project by ensuring the test team have access to all knowledge and/or documentation from the project outset. A very good example of this in action, would be the inclusion of testers within the backlog refinement meetings, as this has given our team the opportunity to analyse user requirements prior to code development, and ensure the acceptance criteria is clearly defined and aligns with any organisational standards. This analysis allows a far clearer understanding of the acceptance criteria across the entire project and has the possibility of resolving future defects before they arise.

With the new responsibilities and development of automation projects and/or tools to aid testing quality, accuracy, and efficiency, this has contributed to the technical experience and skills gained within the test team. Which has empowered the test team to garner a greater understanding of the overall project development life cycle and provide further support during sprint planning, retrospective, and backlog refinement meetings, to ensure quality assurance is maintained throughout the entire project. Further to this, with a technical support process now in place, we are also able to provide far greater detail when raising defects by defining any technical aspects of the issue, which in turn can help direct developers towards any possible defect resolution.

In conclusion, the transition from a traditional Waterfall model to a fully integrated Agile framework within the Single Record department of DHCW, has been transformative in many ways including procedurally and culturally, by embedding testers directly into daily stand-ups, sprint planning, backlog refinement, and retrospectives, we have not only accelerated feedback loops but also fostered a genuine spirit of collaboration across multidisciplinary teams.

With the introduction of more agile testing strategies such as:

* The automation first principle and usage of gherkin syntax within our test cases, has seen a vast increase to efficiency within our testing, with a reduction in test time scales within all our projects, allowing us to maintain this efficiency level with the increased delivery schedule of existing projects and those yet to come.
* Shared documentation has been pivotal in ensuring all team members have access to the information and knowledge they require to fulfil testing responsibilities and has also allowed team members to be more adaptable to switching between projects where necessary.

Challenges do still remain, such as the management of evolving requirements, technical debt in legacy environments and projects, and balancing the unpredictability of continuous delivery with fixed resource constraints, however, the lessons learned underpin the importance of clear acceptance criteria, scope governance, and ongoing skills development, and with the practices we have implemented, we are far better placed to adept to the rigours of testing within an Agile development project.

Our experiences also demonstrate that the agile framework is not merely a set of rules to govern how you work, but more a mindset geared towards being adaptable to change, while promoting a culture of collaboration, continuous development, and a collective ownership of quality assurance. This collaborative culture also positions our development teams to adapt to emerging clinical needs, regulatory changes and technological advances when required, and illustrates how applying Agile principles within a software development process can reconcile the rigors of healthcare compliance and patient safety with the imperative for rapid, high quality software delivery, creating a resilient foundation for future innovation within Health and Care services.